UNIVERSITY MEDICAL CENTER UTRECHT
IMAGE SCIENCES INSTITUTE

Automated Generation of Pelvic CT-based DRR's

Thesis

Alexander H.I.P.R. Hustinx - 1604608
10/18/2015, Utrecht

UNIVERSITY MEDICAL CENTER UTRECHT
IMAGE SCIENCES INSTITUTE

Automated Generation of Pelvic CT-based DRR's

Thesis

Cover image:
Digital radiographic reconstruction of the hips, pelvis and lower lumbar vertebrae

Author:
Alexander H.I.P.R. Hustinx

Student number:
1604608

Version:
V1.1

Date:
10/18/2015

Mentors:
Dr. Ir. Koen Vincken
Associate Professor, Image Sciences Institute
University Medical Center Utrecht
K.Vincken@umcutrecht.nl

Dino Colo, MD
Researcher, Department of Orthopedic surgery
University Medical Center Utrecht
D.Colo-2@umcutrecht.nl

Examiners:
Dr. Leo van Moergestel
First Examiner
University of Applied Sciences Utrecht
Leo.vanMoergestel@hu.nl

Dr. Esther van der Stappen
Second Examiner
University of Applied Sciences Utrecht
Esther.vanderStappen@hu.nl

I. Foreword

This thesis is the result of four years of studying Computer Engineering at the University of Applied Sciences Utrecht. It is the last test I need to pass in order to be awarded my Bachelor of Science.

Over the last few years I have met a lot of different people, some class mates, some professors and even some future business partners. I would like to thank these people for making these four years extra enjoyable. I would also like to thank Erik Puik and the *Excellentie commissie* for accepting my application for the study trip to the *National Collegiate Honors Council*-conference in Denver, CO, USA.
I would also like to thank Rob Brink, Dino Colo and Koen Vincken for providing me with this interesting and challenging graduation project and counseling me where necessary. Next, I would like to thank Lonette Barge, Octave Hustinx and Tom Oosterwijk for proof reading the thesis and helping me clarify some of the confusing sentences.
Finally, most of all, I would like to thank Esther van der Stappen for counseling me over the last full year; first with my project on face recognition in teaching and now with this project.

When I started Computer Engineering in 2011 I had no idea what I wanted to do after finishing it. Now however, I do know; I got very interested in the field of computer vision and image processing, I also rather enjoyed researching and writing about my findings. These two fields of interest have helped me pick my next challenge; obtaining a Master degree in Artificial Intelligence. The Master degree program of Artificial Intelligence at the University of Amsterdam is a broad 120 ETCs, 24 month program. It also allows the student to choose a specialization in, e.g. computer vision.

II. Management summary

In this thesis the research on the automatic constructing of intelligible digitally reconstructed radiographs (DRR's) of the hips is described. This project was initialized when the Department of Orthopedic surgery, within UMC Utrecht, decided to research the relation between spinopelvic parameters and osteoarthritis (OA) of the hip.
The best way to measure the pelvic parameters is in a three-dimensional way, e.g. using computed tomography (CT) scans (Vrtovec, Janssen, Pernus, Castelein, & Viergever, 2012). Despite the high quality and increasing easy availability of current CT scans, the conventional imaging method, and gold standard for diagnosing OA, also because of radiation concerns (Chan, et al., 1991). For that reason, CT scans can be used for diagnosing and mainly classifying hip OA, however, it is preferable to obtain a reconstruction to a DRR's, approaching similarity to X-rays.

Currently Dino Colo and Rob Brink use software suite OsiriX to get these reconstructions, but this requires significant manual actions. Manually creating DRR's has two serious disadvantages:

- When manually creating DRR's, user input makes it a *reader dependent* process, as different users/readers have different interpretations of what they see on a DRR;
- Creating DRR's is a time consuming task, every DRR takes roughly 5-10 minutes to construct.

In order to successfully address the disadvantages mentioned above, the UMC will need to find a method to automatically generate readable DRR's of the hip. By doing so the UMC will not only speed up the process, but more importantly the DRR's will be reproducible. Being able to replicate results is one of the main principles of scientific research.

The main goal of the project is that, by the end, readable DRR's of the hip can be automatically constructed from CT scans, speeding up the process of constructing DRR's from CT scans and being able to replicate these constructions in a robust way.
During the execution of the project assignment the following main research question has been answered:
"In what way is it possible to develop software for UMC Utrecht to automatically construct intelligible DRR's of the hip, pelvis and lower lumbar vertebrae from CT scans, to facilitate reproducible scoring of hip osteoarthritis?"

In order to answer the main research question, the following sub research questions have been formulated:

1. What are the functional and non-functional requirements, set by the users, that the program or plugin has to meet?
2. What are the most prominent methods of constructing DRR's from CT scans?
    - How do these methods work?
    - Which of these methods is most suitable for this project?
3. How can intelligible DRR's be constructed without user input?
    - When is a DRR intelligible enough to be accepted as proper output?
4. Which solutions have been presented for the automated localization of the hip, pelvis and lower lumbar vertebrae in CT scans?
    - Which alterations need to be made to which one of these solutions for it to be acceptable for this project?

Research was conducted to accumulate information about some of the most well-known methods for the construction of DRR’s. Not all of these methods are equally capable of constructing DRR’s of high enough quality to meet the clients’ demands. Of these methods, only ray casting and splat rendering are currently still fairly popular.

After presenting several DRR’s created by splat rendering and by ray casting to the client, the conclusion was reached that splat rendering was not suitable for the research on OA. Ray casting however resulted in DRR’s of high enough quality to meet the clients’ demands and these results can even be enhanced further through trilinear interpolation (although at the cost of output speed).

Afterwards, research was conducted on when the resulting DRR’s could be viewed as *intelligible* and how such DRR’s could be achieved. After discussions with the clients a number of demands were made regarding the quality requirements of the DRR’s for them to be acceptable for their research and thus been viewed as intelligible:

- The results are reliable; all the DRR’s have to be created using the same method(s);
- The rotation of the hips/pelvis has to be compensated for, when possible;
- The region of interest (ROI) needs to contain the entire pelvis and the femoral heads;

Finally, research on how to locate the hips, pelvis and femoral heads (the ROI) in CT scans was conducted. Recent literature showed that a very popular way of locating certain regions in CT scans was done using random regression forests. However, most of the test sets used for the algorithms described in the literature used different inputs when locating the femoral heads, e.g. CT scans containing only the pelvic region. This is not ideal for this project, as the only thing the available CT scans for the research on OA of the hips have in common is that they all contain at least the pelvis. Instead, the graduate proposed a feature-based technique, tailor-made for finding the pelvic region in CT scans of ‘normal’ human beings, by creating maximum intensity projections (MIP’s) and searching in 2D. The technique abuses the pelvic girdle’s high amount of bone tissue and high density, as well as the rough circular shape of the femoral heads to find the ROI and the centers of the femoral heads, in order to correct the axes in case the patient was not laying straight.

So, by creating a program that allowed the creation of MIP’s and DRR’s using ray casting, the ROI can be located using simple 2D feature-based techniques. Afterwards, locating the femoral heads and correcting the patient’s position in the ROI and shooting a DRR, will result in intelligible DRR’s of the hips, pelvis and femoral heads.

Based on the proof of concept’s results and the research done in order to create the proof of concept, some recommendations can be made. These recommendations regard the technique used to determine the ROI, the program’s ability to read DICOM-files, the values used for OpenCV’s[1] circle detection algorithm and its robustness.

[1] OpenCV is a popular open-source computer vision library.

III. Term definitions

Computed Tomography (CT) scan	CT scans are created by combining multiple X-ray images from different angles to produce tomographic images.
Digitally reconstructed radiograph (DRR)	A digitally reconstructed radiograph, or DRR, is a two-dimensional image representing an X-ray. It is usually created from a CT scan or any other three-dimensional source.
DICOM	DICOM stands for Digital Imaging and Communications in Medicine. It is a standard for the handling, storing and communicating information in medical imaging. It also defines its own image file format.
HU	Hogeschool Utrecht, also known as University of Applied Sciences Utrecht is the institute the graduate is from.
Hounsfield units (HUs)	Hounsfield units are a unit of measurement for radiodensity.
OsiriX	OsiriX is a popular (Mac-only) DICOM-viewer. It is also UMC Utrecht's current choice for manual conversion to DRR's.
Osteoarthritis (OA)	Osteoarthritis is a degenerative joint disease, caused by multiple factors, leading to a thinned cartilage. In this document osteoarthritis, or OA, will refer to osteoarthritis of the hip, unless specifically mentioned. *e.g. OA of the knee.*
Pelvic incidence (PI)	The pelvic incidence (PI) describes the sagittal pelvic alignment and is position-independent, i.e. it is an anatomical parameter (Jentzsch, Geiger, Bouaiacha, Slankamenac, Nguyen-Kim, & Werner, 2013)
Region of interest (ROI)	The region of interest in a CT scan refers to the location (usually illustrated as a box) containing the relevant information.
University Medical Center (UMC)	In this document UMC will refer to University Medical Center *Utrecht*, unless specifically mentioned. *e.g. UMC Groningen*.
Window level (WL)	The window level, or WL, is the Hounsfield unit value in the center of the *window width.*
Window width (WW)	The window width, or WW, describes the range of Hounsfield units displayed in a CT scan.

Table of Contents

1. Introduction

The University Medical Center (UMC) Utrecht, more specifically the Department of Orthopedic surgery, is currently researching the relevance of spinopelvic parameters to the development osteoarthritis (OA) of the hip. Roughly 400 computed tomography (CT) scans of patients, aged between 50 and 90 years, are available for this research project. A part of these CT scans has already been used by (Schlösser, et al., Evolution of the ischio-iliac lordosis during natural growth and its relation with the pelvic incidence, 2014).This database has been expanded to include all patients who had undergone CT examination for reasons unrelated to spinal pathology from 2008 and onwards. Most had undergone CT examination of the abdomen for acute abdominal pathology or trauma screening at the UMC Utrecht. If there was any radiological evidence for trauma of the spine or pelvis, or any pathology or previous surgery of the pelvis, spine or hips, or syndromes associated with disorders of growth were excluded (Schlosser et al.). CT scans without complete visualization of the pelvis, including the most distal parts of both ischia, femoral heads and L5, or severe artifacts also led to exclusion. The scans were acquired with Philips Brilliance 16 and 64 scanners (Philips Medical Systems Nederland B.V. Best, The Netherlands). These scans had then to be reconstructed for this research into a two-dimensional projection of the CT scan data.. This resembles similar to an X-ray.

It is possible to convert CT scan data into a digitally reconstructed radiograph (DRR) (Alakuijala, Jäske, Sallinen, Helminen, & Laitinen, 1996), which resembles an X-ray. However possible, converting the data manually, using tools like OsiriX[2], is not reliable for research purposes. This is because relying on user input makes the conversion irreproducible. Therefore, software needs to be used that enables the automated and reproducible generation of intelligible CT-based DRR's.

In this document, the reader will find information about the project context, including the theoretical framework, followed by the project definition, explaining the assignment, the goals and the research questions and methods. Afterwards, the research conducted to answer the research questions will be presented, followed by the design for the proof of concept, based on the results of the conducted research. In the chapter after that, the practical implementation is presented, including important classes and functions used by the proof of concept. This is followed by the conclusion of this research, which answers the main research question, and the recommendations made by the graduate based on the results of the project and conducted research. Finally, an evaluation of the process, from the start of this project till the end, is described.

[2] OsiriX is a Mac only program used for viewing and manipulating DICOM-files. UMC Utrecht currently uses it for the generation of DRR's from CT's.

2. Project context

In this chapter, information about the organization where the project is going to be executed can be found, followed by some background information about the ongoing research on the relation between spinopelvic parameters and OA and some background information on medical imaging. Finally, the reader will be presented with the motivation for the research.

2.1. University Medical Center and Image Sciences Institute

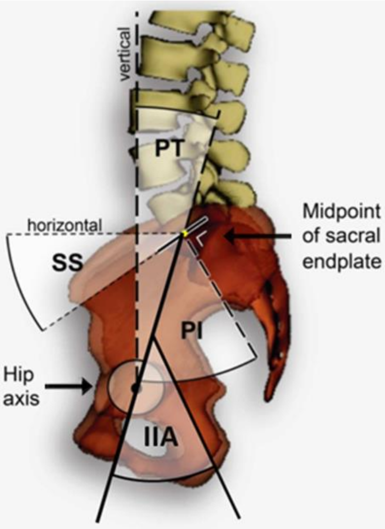
The graduation assignment will be executed at the University Medical Center (UMC) Utrecht at the Imaging Sciences Institute (ISI), in collaboration with the Department of Orthopedic surgery.

UMC Utrecht has about 12,000 employees, making it one of the largest public health care institutions in the Netherlands. UMC Utrecht can be described with three key words: *care, research* and *education* (UMC Utrecht). Its main goal is of course to *care* for their patients. But it also aims to conduct exceptional *research* on the international stage, which can then be translated into patient care. And high-quality *education* guarantees the arrival of new talent, allowing UMC to maintain and strengthen its position.

ISI, housed within the Department of Radiography in UMC Utrecht, employs around 50 researchers. This makes it one of the largest medical image research groups in Europe. Because of its location a close collaboration between scientists, clinicians and radiologists is enabled (ISI Utrecht, 2006).

2.2. Spinopelvic parameters and osteoarthritis of the hip project

The Department of Orthopedic surgery, within UMC Utrecht, is currently researching the relation between spinopelvic parameters and the development of hip osteoarthritis. Osteoarthritis (OA) is a degenerative joint disease caused by multiple factors leading to thinned cartilage. OA can result in painful and swollen joints. To complete the diagnosis for OA, X-rays of the joints is the most used imaging method (Altman, et al., 1991). In a research conducted by (Wittenauer, Smith, & Aden, 2013) it is projected that in 2050 over 15% of the elders aged 60 and over will suffer from OA, being calculated at roughly 130 million people who will suffer from OA. Costs associated with OA include costs for adaptive aids and devices, medicines, surgery, and time off at work and are thus a significant burden for society.



The hips, pelvis and spine are anatomically closely related to each other. In this complex there are a lot of parameters that could be measured to explain the relationship between these structures. The pelvic incidence (PI) is one of the most investigated parameters (Figure 1) (Boulay, et al., 2006). Another parameter is the ischio-iliac lordosis (IIL), recently described and studied by the Department of Orthopedic surgery in (Schlösser, et al., 2014) (Figure 2).

Figure 1. (left) Pelvic parameters in the sagittal plane are described. The pelvic incidence (PI) is the angle between the perpendicular of the sacral endplate and the line connecting the sacral endplate to the hip axis. (Schlösser, et al., 2014)

It is suggested by other studies, like (Yoshimoto, et al., 2005), that the pelvic parameters in patients with OA differ from patients without OA, which could have a predicted value to OA. However, there is no study that directly compares the pelvic parameters measured on CT scans, which is the most reliable method, between patients with OA of the hip and patients without OA.

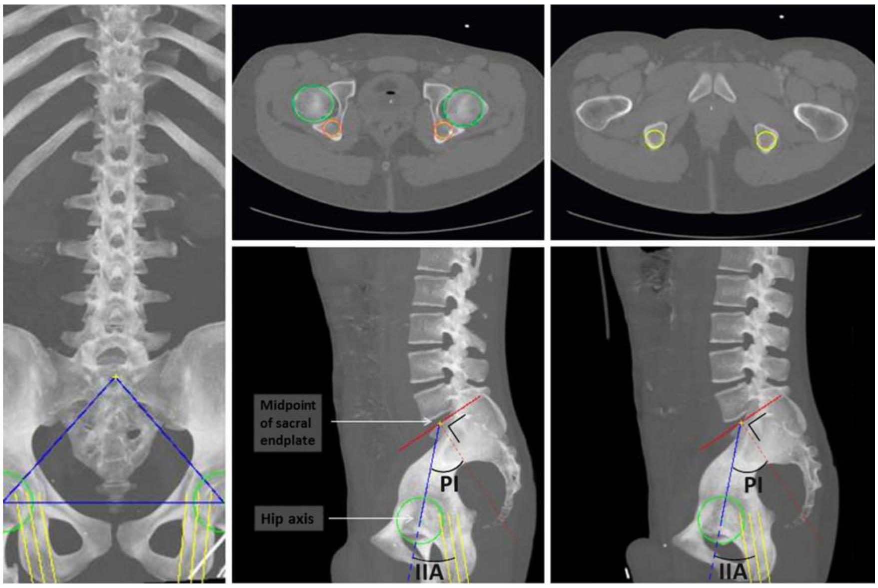


Figure 2. Computerized measurements of the ischio-ilias angle (IIA) and PI on CT scans, using in-house development software. Image courtesy of (Schlösser, et al., 2014)

The aim of a current study is to compare the PI and IIL between people with OA and people without OA. The parameters will be evaluated using CT scans of patients aged between 55 and 60 years.

The best way to measure the pelvic parameters is in a three-dimensional way, e.g., using CT scans (Vrtovec, Janssen, Pernus, Castelein, & Viergever, 2012). Despite the high quality of current CT scans, the conventional imaging method, and gold standard, to diagnose OA is still the two-dimensional X-ray (Chan, et al., 1991). For that reason, the CT scans should also be reconstructed to X-rays. Such a reconstruction is called a digitally reconstructed radiograph DRR, see Figure 3 for an example. Dino Colo and Rob Brink currently use OsiriX for this purpose.

Figure 3. Example of a DRR, constructed from CT-data, of the hips.

2.3. Medical imaging and image processing

Medical imaging is the process of creating visual representation of the interior body for medical or clinical analysis. In this document only medical imaging and image processing regarding X-rays and CT scans is relevant. Medical image processing is the process of retrieving information from the images created by medical imaging, e.g., MRI, CT, X-ray, etc.

During this project the graduate will mainly work with CT scan data. A CT scan is a helical tomography technique which creates two-dimensional data of the hard- and soft tissue inside of the human body. In order to create the two-dimensional data, a beam of X-rays will spin around the object. The X-rays will pass through the object from multiple angles and will be picked up by radiation detectors. Afterwards a computer analyzes the information picked up by the detectors and processes it; this is done slice-by-slice.
During a CT scan the object is enclosed in a ring, with on one side the X-ray source and the opposite side a series of sensitive radiation detectors. This ring will spin around the object in order to obtain information from multiple angles.

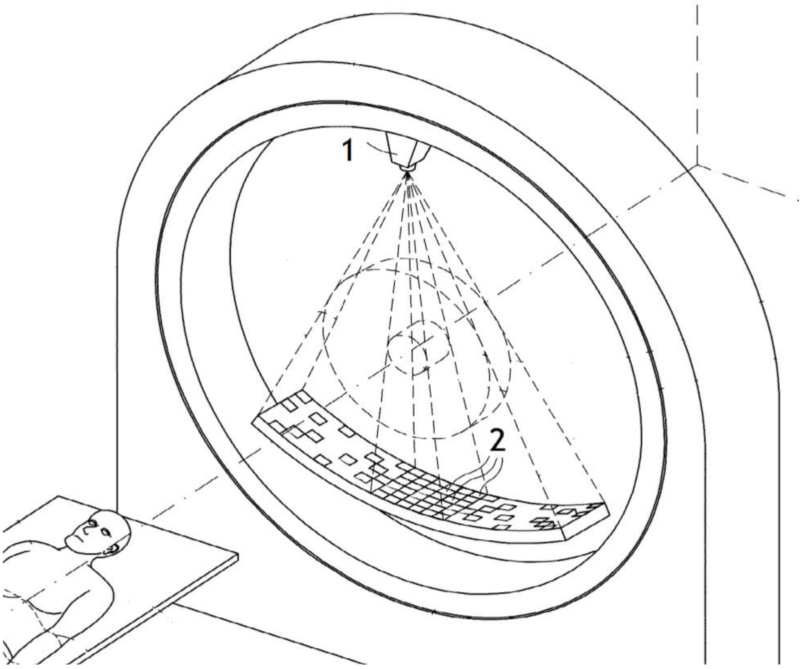


Figure 4. Schematics for a CT scanner where (1) is the source of the beam of X-rays and (2) are the sensitive radiation detectors (Dafni, 2006).

A CT scan basically measures the radiodensity of the object that the rays pass through. Soft tissue has a lower radiodensity than hard tissue. The radiodensity is measured in Hounsfield units (HUs). Hounsfield chose a scale that defines the following four basic densities (Introduction to CT physics, 2004):

- Air = -1000
- Fat = -60 to -120
- Water = 0
- Compact bone = +1000

CT scans usually consist of 512x512 pixels per slice and 12-bits per pixel. These pixels can store values varying from -1024 to +3072. In reality, as compact bone is roughly +1000, the values usually do not get over +1500. This value could however be surpassed in case of certain dense metals.

In order to display these values on a computer[3] they have to be processed into greyscale images. Afterwards, in order to be able to perceive the values, two variables have been introduced:

- Window level (WL);
- Window width (WW).

The WW describes the range of Hounsfield units displayed, the WL is the Hounsfield unit value in the center of the WW. See Figure 5 for a schematic representation.

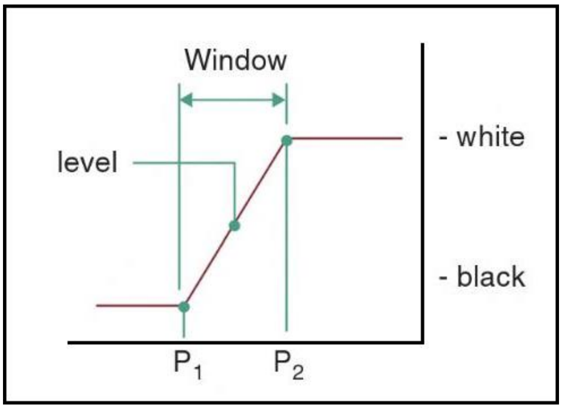


Figure 5. The purple line indicates the contrast in the displayed within the window, where window level is in the exact middle of the window and where window width is the difference between P1 and P2 (Bushberg & Boone, 2011)

The WW and WL settings greatly impact the shown data when viewing a CT scan. See Figure 6 for a visual example. The process of constructing DRR's is highly *reader dependent* because the user/reader selects the window width and window level.

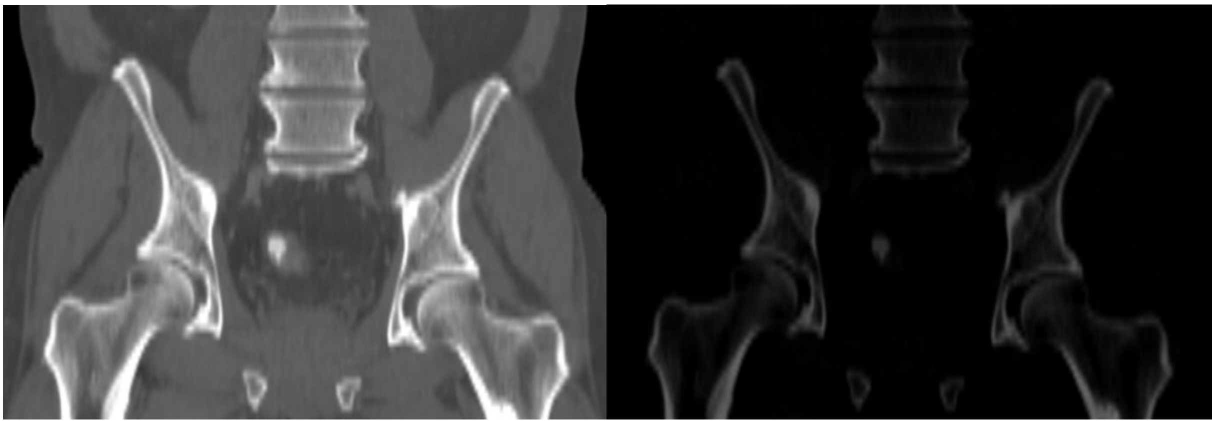


Figure 6. (left) CT slice displayed where WL = 250 and WW = 1450. (right) CT slice displayed where WL = 1800 and WW = 3500.

A digitally reconstructed radiograph, or DRR, is basically an X-ray constructed from a CT scan. This process generally works by placing all the slices of the CT scan on top of each other and 'shooting' rays, originating from a virtual X-ray source, through the slices onto the DRR's canvas (Russakoff, Rohlfing, & Maurer, 2003). See Figure 7 for a visual representation of the construction of a DRR.

[3] Most modern computer screens are only capable of displaying 8-bits, so 256 shades of grey.

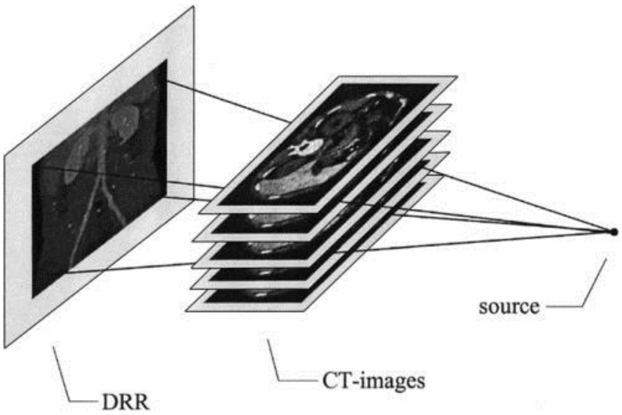


Figure 7. Visual representation of the construction of a DRR. (Bahner, Debus, Zabel, Levegrun, & van Kaick, 1999)

The value of each pixel in the DRR depends on the technique used for the projection. Several well-known techniques are:

- **Maximum intensity projection** (MIP), a technique of forming projection images by casting rays through a volume dataset and selecting the maximum pixel value along each ray (Bushberg & Boone, 2011). This technique is popular when the DRR is used to display bones.
- **Minimum intensity projection** (mIP), a technique of forming projection images by casting rays through a volume dataset and selecting the minimum pixel value along each ray. This technique is popular when the DRR is used to display soft tissue.
- **Weighted values ray casting**, technique of forming projection images by casting rays through a volume dataset and weighing the voxel values. These weighted values usually depend on distance from the source to the canvas and distance traveled in each voxel. Several variations have been proposed to speed up this process (Alakuijala, Jäske, Sallinen, Helminen, & Laitinen, 1996).
- **Alpha blended texture mapping**, a technique that basically draws the slices on top of each other starting from the bottom slice and ending at the top slice where the alpha-value of each resulting pixel is determined by the height of the slice of the pixel it passes. This method works very quickly, yet it only works correctly on very small datasets. (Alakuijala, Jäske, Sallinen, Helminen, & Laitinen, 1996)

A two-dimensional example of *weighted values ray casting* can be found in Figure 8 where the ray travels through four pixels. The ray only just pierces some pixels while going straight through the middle of others. The voxels barely getting pierced will have a smaller weight than the pixels getting pierced straight through the middle. It is important to note that the actual ray casting for the creation of DRR's is three-dimensional.

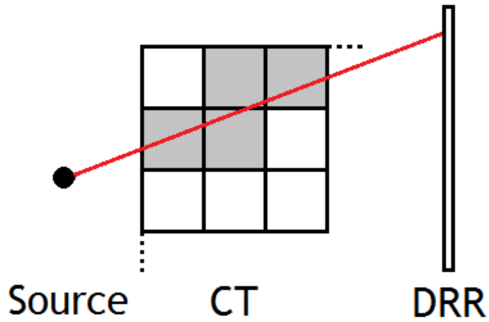


Figure 8. Two-dimensional visual representation of the X-ray (red line) travelling through four pixels, illustrating the weight of the values.

2.4. Motivation for the current research

As explained in paragraph 2.2, the available CT scans have to be reconstructed to images that are similar to X-ray images. Currently Dino Colo and Rob Brink use OsiriX to do so, but this requires manual actions. Manually creating DRR's has two serious disadvantages:

- When manually creating DRR's, user input makes it a *reader dependent* process, as different users/readers have different interpretations of what they see on a DRR;
- Creating DRR's this way is a time consuming task; every DRR takes roughly 5-10 minutes to make.

In order to successfully address the disadvantages mentioned above, the UMC will need to find out how to automatically generate readable DRR's of the hip. By doing so the UMC will not only speed up the process, but more importantly the DRR's will be reproducible. Being able to replicate results is one of the main principles of scientific research.

This research will also benefit the UMC in the future, as this will most likely not be the last time they will convert CT scans to DRR's. As the UMC does a lot of research it is important that, when they use DRR's in their research, the created DRR's will also be reproducible. These reproducible DRR's could be constructed using the program or algorithm created during this project.

3. Project definition

This chapter first describes the details about the project assignment, followed by the goals that are set for the project. Thereafter the main and sub research questions are described. After that the project deliverables are listed, followed by how the research will be approached, which research methods will be applied and which tools will be used. Finally, the project scope will be described.

3.1. Project assignment

Product assignment

The assignment that needs to be realized is to automate the construction of DRR's from three-dimensional CT scans by using OsiriX or a different DRR program and to ensure all the reconstructions are done the same way.

- In order to automate the process, the user input needs to be minimized. This means the hips and the lower lumbar vertebrae need to be automatically detected. As shown in Figure 3, at least the two lowest vertebrae of the lumbar spine need to be visible on the DRR and of course the pelvis, hip sockets and femoral heads. The localization method needs to be reproducible.

- Another important part of constructing DRR's is the location of the virtual X-ray source. In order for the results to be reproducible, the DRR's need the virtual X-ray source to be perpendicular to the center of the pelvis on the CT scans. This differs per scan as patients usually tend not to lay on the table perfectly straight.

- To ensure that the resulting DRR is reproducible, the window width (WW) and window level (WL) need to be either fixed or automatically selected using an algorithm with reproducible results. Since having an overall fixed WW and WL will likely cause readability issues on parts of the DRR's, using an algorithm to decide the WW and WL will more likely be the best choice. However, this will need to be researched.

- Finally, the application will need to be robust enough that when executed on two different CT scans of the same patient, the resulting DRR will be similar. This will be proven by inputting the resulting DRR into the program discussed in (Schlösser, et al., 2014).

The software that needs to be developed for the project assignment has to be either a plugin for either OsiriX or a different suitable DRR program, or it has to be a (standalone) program. Optionally, a library or API can be created so the software can be reused for different purposes. In any case, the heart of the application should be a separate functional collection of code.
Preferably it will be a plugin for an existing DRR program, there is no need to reinvent this. In case a plugin cannot be created for a suitable DRR program a good alternative might be to use MeVisLab[4] to create the program.

[4] MeVisLab is a software framework for medical image processing and is currently used for most in-house software at the UMC.

3.2. Project goals

The main goal of this project is that, by the end, intelligible DRR's of the hip can be automatically constructed from CT scans and thereby speeding up the process of constructing DRR's from CT scans and being able to replicate these constructions in a robust way.

3.3. Research questions

During the execution of the project assignment the following main research question will be answered:
"In what way is it possible to develop software for UMC Utrecht to automatically construct intelligible DRR's of the hip, pelvis and lower lumbar from CT scans, to facilitate reproducible scoring of hip osteoarthritis?"

In order to answer the main research question, the following sub research questions have been formulated:

1. What are the functional and non-functional requirements, set by the users, that the program or plugin has to meet?

2. What are the most prominent methods of constructing DRR's from CT scans?
    - How do these methods work?
    - Which one of these methods is most suitable for this project?

3. How can intelligible DRR's be constructed without user input?
    - When is a DRR intelligible enough to be accepted as proper output?

4. Which solutions have been presented for the automated localization of the hip, pelvis and lower lumbar in CT scans?
    - Which alterations need to be made to which one of these solutions for it to be acceptable for this project?

3.4. Quality demands and assessments

This paragraph describes the methods that will be used to answer each research question and afterwards the tools and development methods used for the project management and development of the software.

3.4.1. Research methods

The main research question will be answered by answering each of the sub research questions. The research methods and techniques used to answer each sub research questions are shown in Table 1.

Table 1. Research methods and techniques per sub research question

#	Research question	Question type	Research method(s)	Method and technique
1	What are the functional and non-functional requirements, set by the users, that the program or plugin has to meet?	Descriptive	Interviews	Through interviews with the clients and Koen Vincken a requirement and MoSCoW analysis will be executed.
2	What are the most prominent methods of constructing DRR's from CT scans?	Descriptive	Desk research	A list of methods from literature will be drafted. These methods will be compared based on several factors to form a short list of methods, which should give enough insight.
2.1	How do these methods work?	Descriptive	Desk research	By analyzing existing material the methods will be described.
2.2	Which one of these methods is most suitable for this project?	Comparative	Desk and field research	Through factor scoring a table will be drafted, containing several scores (e.g. speed, accuracy, ease of implementation, etc) for each method.
3	How can intelligible DRR's be constructed without user input?	Descriptive	Desk and field research	By analyzing the existing methods and weighing their relevance and applicability.
3.1	When is a DRR intelligible enough to be accepted as proper output?	Designing	Interviews	A requirement analysis will be realized through an interview with the clients. This will also result in the method of testing.
4	Which solutions have been presented for the automatic localization of the hip, pelvis and lower lumbar in CT scans?	Descriptive	Desk research	By analyzing existing material, a short list of relevant and automated localization methods will be drafted.
4.1	Which alterations need to be made to which one of these solutions for it to be acceptable for this project?	Descriptive & Designing	Desk and field research	By comparing the situation of the existing localization methods to this projects situation and adjusting where needed.

3.4.2. Tools and development methods

During the research and software design multiple tools and development methods will be used.

Document sharing and versioning

Documents that might be modified during the graduation period will not only be provided with the date it was last modified, but will also be provided a version number. The document versioning will be done according to (National Institute of Dental and Craniofacial Research, 2014).

The graduate will also use cloud services to share documents. This will allow his mentors to check the graduate's documents for progress and correctness at any time.
These cloud services can also be used to keep track of documents' versioning.

Mendeley
The graduate will use Mendeley to easily keep track of the papers used and their importance. Mendeley is a free reference manager and PDF viewer in which the user can easily browser through the papers and make notes in them (Mendeley).

Waterfall model
As development and project management method a variation of the waterfall model was chosen. The original waterfall model is a sequential design process. It progresses from requirements through design, implementation, verification and finally maintenance (Royce, 1970).
In the variation on the waterfall model used for this project, the waterfall progresses from requirements through research, design, coding, validation and finally the finalization. A visualization of the model is shown in Figure 9.

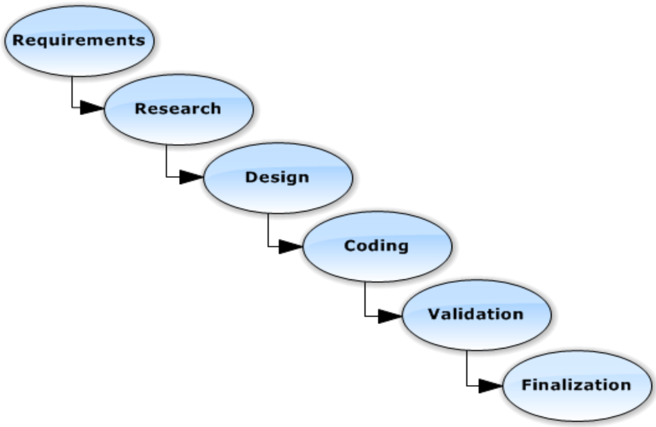


Figure 9. Variation on the waterfall model, used for this project

The reason behind using this version of the waterfall model is that its phases fit the project deliverables. First the requirements for the project are set, followed by doing the necessary research to gather information required for the next phase: design, in which the technical-function design will be created. After completing the design phase, the project can be realized in code, of course followed by validating the code through testing. When all these phases have been completed, the project enters the phase "finalization", where the thesis and end presentation are finished and turned in.

A downside of the waterfall model is its disability to go back more than one phase, e.g., if during the testing of the code one finds out that the design is flawed, it is best to start over entirely. One way to prevent this issue is to ensure that everything is well-thought out. This could be troublesome because it is unclear what will be realized by the end of the project. As the result of research question #1 (see Chapter 4.1) will be a MoSCoW analysis, the prioritization of the functionalities was not fully known until the question was answered. Also, because some of the testing of the product will be done in collaboration with doctors from the Department of Orthopedic surgery, it will not be easy to say which functionalities have successfully been implemented before the end of the project.

3.4.3. Quality demands

To ensure the code for the program will meet all the requirements, tests will be designed and conducted and a report will be written containing the test cases, the test results and the raw test data. The tests, the program is subjected to, will be explained in paragraph “Quality demands and testing” of Chapter 4.2.

In Chapter 4.1, the MoSCoW prioritization is described and explained. MoSCoW prioritization is a technique used to quickly and easily group requirements into one of four categories (Hatton, 2007).

4. Research

This chapter describes all the research done to answer each of the research questions, followed by a brief recap.

4.1. MoSCoW analysis

What are the functional and non-functional requirements, set by the users, that the program or plugin has to meet?

In order to get a clear view of all the product requirements and their importance a method called the *MoSCoW analysis* is applied. The MoSCoW analysis is a prioritization method used to reach a common understanding between the client and service provider about the importance of certain functionalities or requirements (DSDM Consortium, 2015). MoSCoW is short for “**M**ust have, **S**hould have, **C**ould have, **W**on’t have”.
Over the course of several meetings with Dino Colo, Rob Brink and Koen Vincken the MoSCoW analysis has been executed, based on discussions about the clients’ demands for the program. In Table 2 the results of the MoSCoW analysis are presented. The iterations leading to the current MoSCoW prioritization can be found in APPENDIX C. Iterations MoSCoW prioritization.
Note that the process of scoring for OA is partly subjective, as each doctor might have a different opinion about each X-ray/DRR. So there is not just one golden answer regarding the scoring of OA.

Table 2. MoSCoW prioritization for this project

Priority	Functionalities/Requirements
Must have	Ability to construct DRR from CT
	In the resulting DRR the bones are visible enough for medical professionals to score OA
	The DRR’s are of high enough quality for medical professionals to score them for OA
	The proof of concept is a standalone program
	The results are reproducible
	The results are reliable; all the DRR’s have to be created using the same method(s)
Should have	The construction of DRR’s requires as little user input as possible
	Ability to automatically locate the ROI
	Ability to automatically place the X-ray source perpendicular to the hips
	The code is well-documented, so its function can be easily grasped by other programmers of equal or greater experience as the graduate.
	The proof of concept is provided with a GUI
Could have	The proof of concept and library are reusable for different purposes and/or expandable with additional code
	The proof of concept has configurable settings, regarding e.g., alternating location of the X-ray source, like an offset.
	Ability to alter selected WW/WL after processing
	Resulting DRR’s have their quality enhanced by trilinear interpolation, for a higher resolution
Won´t have	Automated scoring for OA

When looking at the requirements, a simplified work flow can be created:
Locate femoral heads → Correct CT rotation → Create DRR → Post processing

4.2. Construction of intelligible DRR's

How can intelligible DRR's be constructed without user input? When is a DRR intelligible enough?

In order to construct intelligible DRR's of the desired ROI without any user input, two problems need to be taken into account:

- All CT scans are different and so is the location of the ROI; patients might be laying slightly tilted as the CT scan is created; the CT scan might be initially created for a different purpose than scoring OA, so it does not only contain the relevant region;
- When creating a DRR and saving it as a simple image, the pixel values will be 0-255 instead of 0-4095. This means that the user can no longer play with the WW and WL, so an initial WW and WL will need to be selected, resulting in a clear image.

Tackling the first problem is the hardest. In order to create a DRR of the ROI in which the patient is no longer tilted, the CT scan will need to be rotated over the x-, y- and z-axis. The amount in which the CT scan will need to be rotated and over which axes depends on the position of both femoral heads. In Figure 10 a visual representation of the front and top view are depicted, when the angles have been corrected the hips will be straight from the focal point.

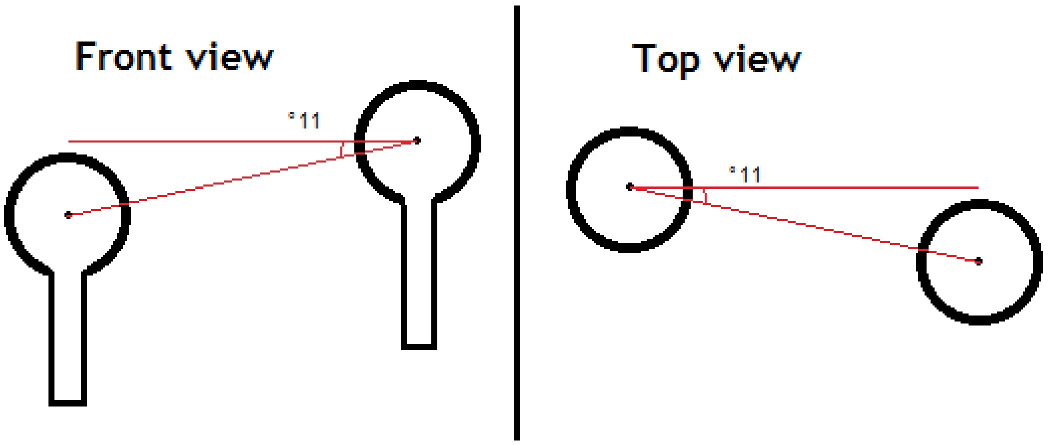


Figure 10. Schematically represented femoral heads, view from the front (left) and from above (right), including the angle that needs to be corrected. The front view will need a rotation of °11 over the z-axis and the top view will need a rotation of °11 over the y-axis.

Ideally, the line between the focal point and the line between the two femoral heads is perpendicular. In order to achieve this, first the centers of both femoral heads needs to be located in 3D. Once both centers have been found, it is possible to calculate either the desired position of the focal point or the degrees and axes in which the CT scan needs to be rotated. Afterwards a ROI can be calculated in order to have the DRR contain only the important part of the CT.

Secondly, selecting a WW and WL which allows the user to clearly see what he/she is looking for, in this case the acetabulum and femoral heads, is very important. This can be accomplished by having a high amount of contrast between bone and cartilage, allowing the user to see the degree of wear of the cartilage. As there is a set range for the Hounsfield units of bone and a suggested WW/WL setting for bone in CT scans, this could be used as base. Another way to improve the clarity of the acetabulum and femoral head is through post-processing, e.g., histogram equalization and edge enhancement, as these algorithms can be used to increase the contrast in an image (Kaur, Kaur, & Kaur, 2011) (Shanmugavadivu & Balasubramanian, 2011).

Quality demands and testing

After several discussions with medical professionals; Dino Colo and Rob Brink, about when a DRR is intelligible enough to be accepted as proper output, it has been concluded that only the following demands have to be met for medical professionals to accurately score for OA:

- The results are reliable; all the DRR's have to be created using the same method(s);
- The rotation of the hips/pelvis has to be compensated for, when possible;
- The ROI needs to contain the entire pelvis and the femoral heads;

Most of these demands were already explained in the MoSCoW analysis (paragraph 4.1) or project description.

Each CT scan is different; each patient lays differently, each patient is different, etc. So in order to test if the list of demands has been met, the resulting DRR's will be subjected to a test. This test will be executed by a group of medical professionals, referred to as the examiners. The examiners will each receive ±10 DRR's created by the proof of concept and will be asked to score the patients for OA, based on their DRR. After roughly a week, each examiner will be asked to score the patients based on the same DRR's as the week before. The resulting scores will be compared and will state the reliability (John & Soto, 2007).

4.3. Automated localization in CT's

Which solutions have been presented for the automated localization of the hip, pelvis and lower lumbar in CT scans? Which alterations need to be made for one of these methods to be accepted for this project?

In literature, techniques regarding regression forests are very popular for automated localization in CT scans, mainly for the localization and segmentation of the spine. Forest regression is a technique used for machine learning and data mining to predict the relation between independent variables and a dependent variable, and to explore these relations (Breiman, 2001). Regression forests are built from one or more decision trees. A decision tree builds regression or classification models in the form of a tree, breaking down a dataset into smaller subsets. Using multiple trees and feeding them with slightly different data leads to more robust results, after averaging all the outcomes of all trees. These small alterations in the data can be as simple as e.g., instead of using the mean value, use the mean plus or minus the standard deviation. See Figure 11 for a visual example of a regression forest.

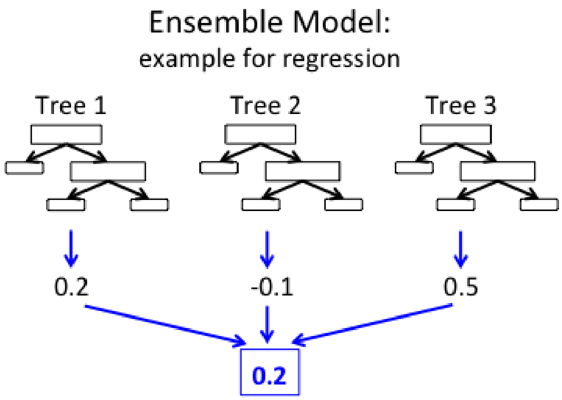


Figure 11. Example showing three trees each resulting in slightly different data. In order to get a more robust result, the average is used. Image courtesy of **(Bradley & Amde, 2015)**.

A few authors have also proposed using regression forests specifically for the localization of femoral heads and the pelvis (Chen & Zheng, 2013). Regression forests are usually not the only part contributing to the localization of organs/bones, they are mainly used to train the program which aspects to consider. In most cases a training set of ~100 CT scans is used, where the important parts have been manually selected. Afterwards, combining what it learned through supervised training, with another technique, will result in the (precise) location. A few of these other techniques are:

- Hidden Markov model (Glocker, Feulner, Criminisi, Haynor, & Konukoglu, 2012);
- Atlas-based registration (Forsberg, 2015);
- Superpixels (Hutt, Everson, & Meakin, 2015).

(Korez, Ibragimov, Likar, Pernuš, & Vrtovec, 2015) do not use regression forests, instead they use an interpolation-based approach combined with a mean shape model for their localization. Note that these techniques are all used on a very specific dataset of CTs, usually using only CTs of a certain region of the body, e.g., only the pelvic region, only the upper spine, etc.

Localization for this project

The above named approaches are not ideal for this project, since the only thing the CT scans we use have in common, is that they contain at least the pelvic region, while those algorithms require standardized input. Instead, I propose a feature-based approach, looking for specific local features in our images to locate the desired objects.

A smaller region in which the program has to search for the femoral heads can be obtained by first constructing a MIP from the standard position of the CT scan and then locating the pelvic region, based on what we know about its intensity. The pelvis is made out of thick bone and thus has high HUs. In fact, the pelvis is one of the largest bones in the human body (Schünke, Ross, Schulte, Schumacher, & Lamperti, 2006).

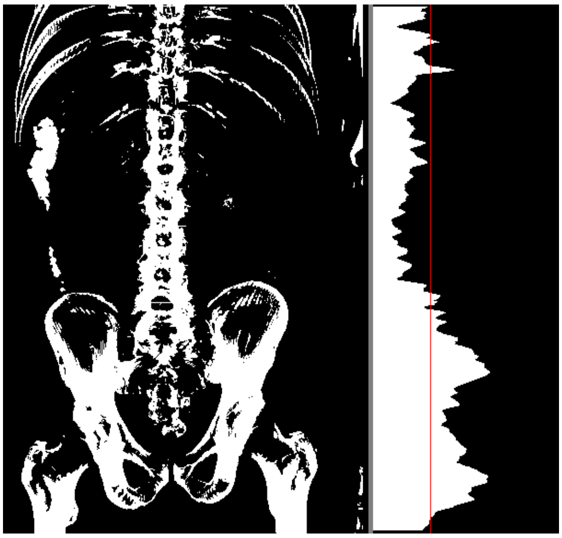


Figure 12. Thresholded (20% P-tile) image of the MIP of a CT-scan, with on the right the histogram counting the amount of set pixels per line (after smoothing). The vertical red line indicates the average of the histogram.

This would mean that, after the MIP has been subjected to thresholding, the region with the most white pixels would be the pelvic region. As depicted in Figure 12, there is an obvious increase in intensities in the pelvic region. Eventually, the region will be selected based on the average intensity.
The same as what has been done in Figure 12 can also be done over the x-axis to obtain the ROI, see Figure 13 for an example of such a ROI.

Figure 13. Example of the calculated ROI using histograms, the red lines indicate the starts and ends of the x- and y-axis.

Now, having created a ROI, it is easier to locate the femoral heads. As the shape of the pelvis is roughly the same for every person, some assumptions can be made regarding the approximate location of the femoral heads. The first assumption, based on the height and width of the pelvis and the located ROI in the tests, is that the femoral heads can be found in the bottom half of the ROI and that one can be found in the first third and the second in the last third of the ROI (over the x-axis), see Figure 14 for a visual example. If the CT is too tilted, the femoral heads will no longer be in these regions. Instead, the margins can be adjusted if, e.g., no femoral head could be found at first.

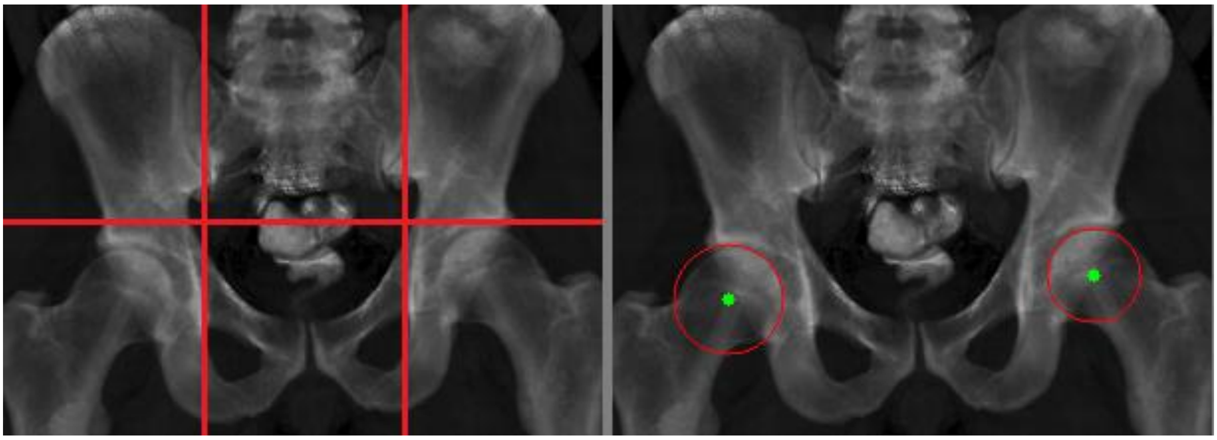


Figure 14. (left) the DRR of the ROI split into 6 smaller ROI's. (right) The resulting circle and center of the femoral heads, based solely on the circular shape.

A fact about the femoral heads is that they roughly have a spherical shape, as they are *ball and socket joints* (Hammond & Charnley, 1967). This means that searching for a circular shape of bone tissue within the smaller ROIs should result in finding the center x- and y-coordinates of the femoral heads (See Figure 14). In 2D image processing this could for instance be done using the Hough circle transform algorithm (Atherton & Kerbyson, 1999).

Once the x- and y-coordinates of the femoral heads have been located, they can be converted into approximate x- and y-positions of the center of the femoral heads on the CT scan. Afterwards, by selecting the slice over the known x-axis of the center, the z-axis of the femoral head can be located per head.

Figure 15. Slice over the x-axis of ROI of the CT-scan, where the center of a femur head was located.

As can be seen in Figure 15 the same rough circular shape of the femoral head can be used to find the z-coordinate of the center. After their approximate location has been found, the 3D center of each femoral head is obtained by combining the x-, y- and z-coordinate.

4.4. Methods for DRR construction

What are the most prominent methods of constructing DRR's from CT scans? How do these methods work? Which one of these methods is most suitable for this project?

According to (Alakuijala, Jäske, Sallinen, Helminen, & Laitinen, 1996) and (Meißner, Huang, Bartz, Mueller, & Crawfis, 2000) there are four prominent methods for the construction of DRR's. These are:

1. Ray casting
2. Splat rendering
3. Shear-warp
4. Texture mapping

Some of these methods have variations to increase image quality or speed of the process. As texture mapping and shear-warping are nowhere near as detailed as the other two, this document will only describe ray casting and splat rendering. (Alakuijala, Jäske, Sallinen, Helminen, & Laitinen, 1996) also refers to *splat rendering* as *anti-aliased splatting*. As anti-aliased splatting has been revised, this document will discuss *splat rendering* instead.

4.4.1. Ray casting

Ray casting is the most prominent method for the construction of DRR's. There are a lot of variations on how to determine the pixel value on the DRR, as discussed in chapter 2.3. Ray casting is an easy to grasp technique for creating DRR's and also very easy to write in code. The earlier discussed forms of ray casting describe how a pixel value is calculated. The three methods below are used to select the voxel values while traversing trough the dataset:

- Trilinear interpolation
- Mono-linear interpolation
- Weighted sampling

The first two variations are quite similar in execution, except that mono-linear interpolation considers less data than trilinear interpolation and thus is faster in execution time. The third method, weighted sampling, is very basic; the resulting pixel value is determined by the amount of time spent in each voxel (as explained in chapter 2.3). This technique is easy to grasp, quick to code and yields very good results. Sadly this method is one of the most computationally intensive ones (Meißner, Huang, Bartz, Mueller, & Crawfis, 2000), however this does not affect the decision much, as performance is not an important criterion here. Note that weighted sampling can be combined with linear interpolation to improve the results. Weighted sampling will not be discussed further in this document.

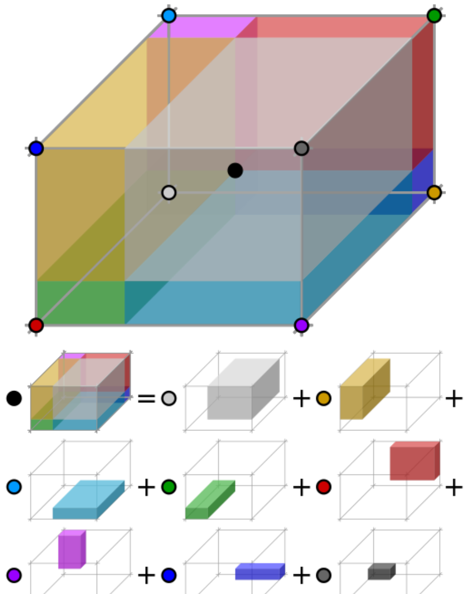


Figure 16. Visualization of trilinear interpolation. (Wikipedia, 2015).

- **Trilinearly interpolated ray casting**

Trilinear interpolation computes values located between voxels, by linearly weighting the eight closest neighbours. Because this method looks at the eight closest voxels, it is a fairly computationally intensive process. As shown in Figure 16, the product of the value at the desired point and the entire volume is equal to the sum of the products of the value at each corner and the partial volume diagonally opposite the corner (Wikipedia, 2015). Once the value of a voxel has been calculated, a more accurate pixel value can be calculated based on these more accurate values.
For instance, when piercing just the top left corner of a voxel, the voxel value should be closer to the value of its left neighbour than to the value of its right neighbour. When calculating an average, using voxel values calculated by trilinear interpolation, the resulting pixel value is more realistic.

- **Mono-linearly interpolated ray casting**

Because trilinear interpolation is fairly computationally intensive an alternative is mono-linear interpolation. Mono-linear interpolation only interpolates over the z-axis, instead of x, y and z. This means that only the two closest voxels, over the z-axis, will be taken into account. The resulting voxel values will be less accurate than those calculated by trilinear interpolation, but sometimes the difference is trivial, e.g., when all the surrounding voxels have the same value.

4.4.2. Splat rendering

Splat rendering, or splatting, is an object space based technique. This technique is the result of optimizing early stages of volume rendering. During splatting all voxels will be projected onto an empty canvas and every one of these pixels will influence their neighbours as they are 'splatted'. The projected position is used as the center of a *footprint* that determines how much each of the neighbours is effected by the splatted voxel (see Figure 17). All footprints are rendered like flat disks, their properties (color and transparency) varying diametrically.

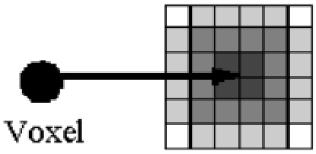


Figure 17. The voxel will be splatted over a footprint. When further away from the center the impact from the voxel on the footprint decreases. Image courtesy of **(Representación y Rendering de Volúmenes, 2005)**

Once all the footprints have been created, the resulting DRR will depend on the rendering method used. These methods are quite similar to some of the ray casting ones. E.g.: maximum intensity projection can be created by discarding all voxel masks, except for the highest (Spörk, 2010). What is different however, is how the splats are composed together, in order to construct a DRR.

Splat rendering is quick and yields decent quality images. Compared to ray casting, it produces images containing discretization artifacts, though this can be partially addressed by *wobbled splatting*, resulting in images with less artifacts but more blur. *Wobbled splatting* is a method where either the dataset or the projection source is 'wobbled' around, this reduces regular line artifacts, but causes more noise. This noise can partly be filtered by using some basic filters (Birkfellner, et al., 2005).

(Hansen & Johnson, 2005) distinguishes and describes three different types of splat rendering:

- **Composite-only splatting**: Every voxel is splatted onto all other splatted voxels to create the final image (See Figure 18). This is the most basic form of splat rendering also making it the fastest way, however it also yields the most flawed results, as the interpolation and composition cannot be separated, causing color bleeding.

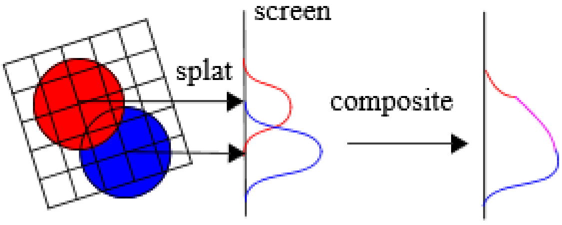


Figure 18. composite-only splatting implemented as footprint mapping (Kaufman & Meuller, 2005)

- **Axis-aligned sheet-buffered splatting**: This technique attempts to result in as little artifacts as possible by composing all voxels separately by creating sheets of data, normal to axis most aligned to the point of view. By composing these data sheets the DRR can be created (See Figure 19a). Changing the point of view might cause artifacts, as it has been composed for a certain point of view.

- **Image-aligned sheet-buffered splatting**: This technique works similar to axis-aligned sheet-buffered splatting, but sheets and voxels are combined to be orthogonal to the viewing direction (See Figure 19b). This is a more complex operation, but does not lead to as much artifacts and thus creates higher quality DRR's.

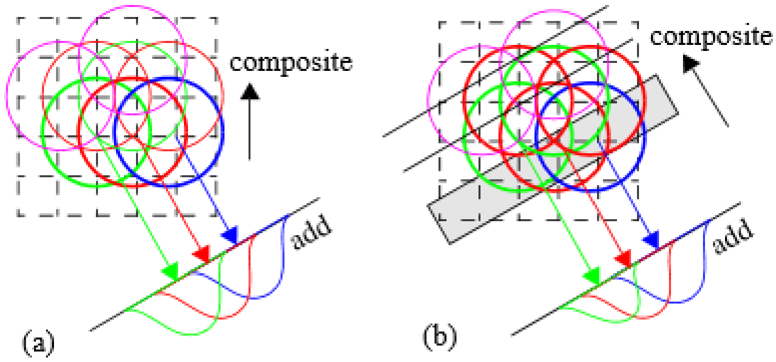


Figure 19. Axis-aligned sheet-buffered splatting (a): the entire kernel within the sheet is added. Image-aligned sheet-buffered splatting (b): only slices of the kernels intersected by the current sheet-slab are added (Kaufman & Meuller, 2005)

Lots of variations have been designed to increase quality of the DRR's and complexity of the algorithm as well as runtime. The most popular one of these is *wobbled splatting*, as mentioned before. Wobbled splatting, or wobbling, has proven to be a very fast technique that yields decent results, when paired with a low pass filter. These results however cannot be used when looking for a single high quality DRR. See Figure 20 for an example of a simplified splat rendering algorithm combined with wobbling and a low pass filter (Birkfellner, et al., 2005).

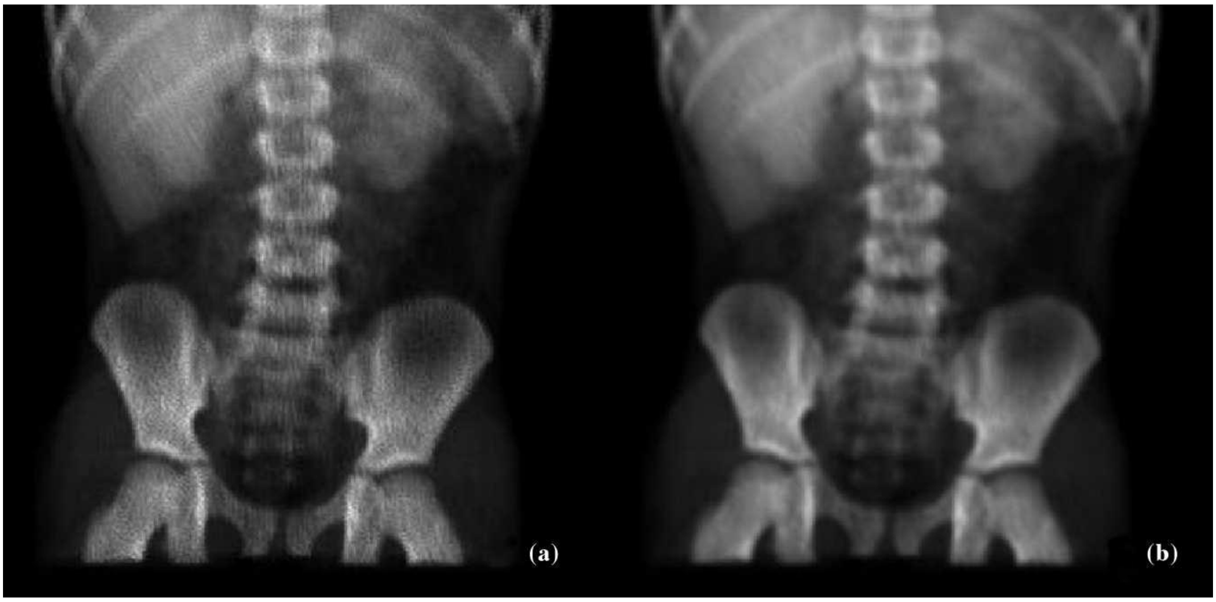


Figure 20. The image on the left (a) shows the result of a simplified splat rendering algorithm with voxel wobbling. The image on the right (b) shows (a) after being subjected to a low pass filter.

Splat rendering has several disadvantages, one of them being that it is not possible to choose which splats you will use before all the voxels have been splatted. It is also impossible to exclude parts of the dataset or to use a subset to create DRR's because of a flaw in the design (Hansen & Johnson, 2005). This could mean that the construction of DRR's is slightly delayed as only part of the DRR is relevant to the medical professional. Compared to ray casting, this technique is more complex and results in DRR's of slightly lower quality, but it is faster to construct DRR's using this technique.

4.4.3. Method for this project

In order to decide which of the previously mentioned methods are most suitable for this project, a short list of advantages and disadvantages for each method has been created. Table 3 contains the advantages and disadvantages of both ray casting and splat rendering:

Table 3. Advantages and disadvantages of ray casting and splat rendering.

	Advantage	Disadvantage
Ray Casting	High quality results	Computationally intensive
	Easy to grasp algorithm	
	Easy to program	
	Easily reusable for different types of DRR/MIP creation	
Splat rendering	Fast construction of DRR's	Hard to grasp algorithm
	Decent resolution	Blurry results

Judging by the requirements set in paragraph 4.1, only a few of these requirements actually apply as criteria to select the DRR method. This is because most of the requirements apply to pre- or post-processing of the CT or DRR, e.g., "reliable and reproducible results" in this case mainly apply to the chosen WW and WL; automated localization of the ROI is pre-processing; the ability to alter the WW and WL is post-processing, etc. Based on the relevant requirements a small factor scoring table can be created, see Table 4. In the factor scoring table ray casting and splat rendering, when scored by medical professionals, are scored with either a '+' or a '-'. A '+' has been given if most medical professionals agrees that the method's results meet the requirement and a '-' when most medical professionals disagrees.

Table 4. Factor scoring of DRR methods to the relevant requirements

Requirement	Ray casting	Splat rendering
DRR's are not blurry	+	-
Reusable for different purposes	Yes	Yes
Quality enhanceable through trilinear interpolation	Yes	No
DRR's scorable for OA by medical professional	+	-

When looking at the factor scoring table, one obvious disadvantage of splat rendering sticks out; blurry results. Compared to ray casting with high quality results the only viable choice is to go with ray casting. Not only is the ray casting algorithm easily reusable for MIP and mIP, its quality can be improved in several ways through trilinear interpolation; once by shooting more rays per resulting pixel and once by having better weighted values when penetrating a voxel. Currently only two medical professionals have been asked for their opinion, judging by their opinion about splat rendering not resulting in DRR's of high enough quality no more medical professionals were asked as ray casting was deemed fit.

4.5. Suitable DRR programs

Which programs are capable of constructing DRR's that can be extended with a plugin? Which of these programs are suitable for this project?

This research question is no longer relevant as Koen Vincken is no longer interested in simply using an existing program and extending that with a plugin. He would rather have a (reusable) library. This could partly be achieved by creating a module[5] in MeVisLab, still enabling the use of existing code.

After some research and testing, it quickly became clear that MeVisLab is currently not suitable for the creation of intelligible DRR's. Below two DRR's can be found, one created by MeVisLab and one created by a draft of the proof of concept. Note that the WL and WW are different in the two images, but the main problem with the DRR created with MeVisLab is its quality. The femoral head and the acetabulum are not detailed enough to score for OA.

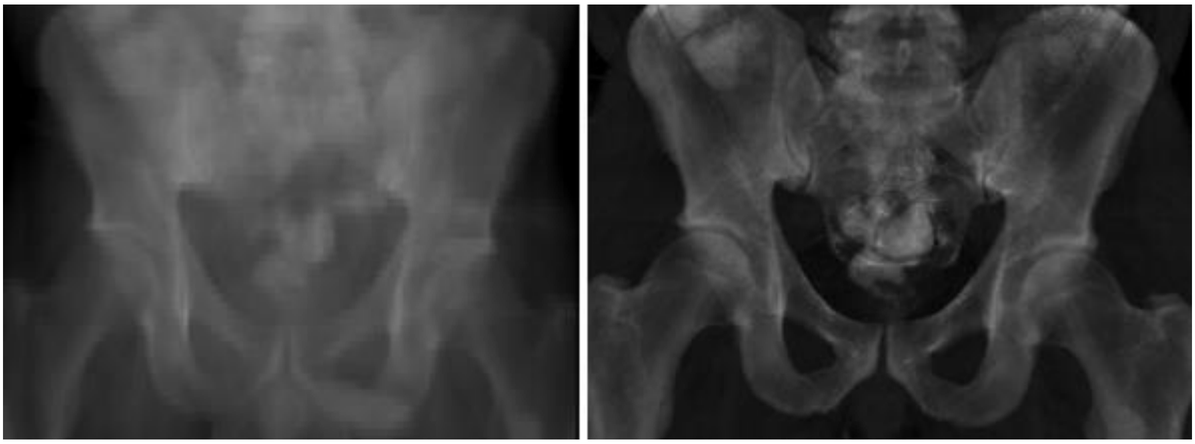


Figure 21. (left) DRR created with MeVisLab with the same ROI as (right) the DRR created with the proof of concept.

As the UMC no longer wants a plugin for an existing program, the products delivered at the end of this project are a library and a proof of concept, which will be an extension of the draft version used to create the image in Figure 21 (right).

[5] A module is either a network of other modules or a small program with certain in- and/or output(s) that can be dragged and dropped into the MeVisLab working environment.

4.6. Recap

Now, all the sub research questions have been answered and several choices have been made regarding the functionalities of the program. This chapter gives a brief recapitulation on each one of the research questions and choices.

At first a MoSCoW prioritization analysis has been created to get a clear view of the product requirements and their priority, followed by a simplified workflow was also presented:
Locate femoral heads → Correct CT rotation → Create DRR → Post processing
followed by a section describing the criteria for when a DRR is considered intelligible, including some extra quality demands.

Then, a feature-based method for the localization of the femoral heads was proposed. This method first selects the ROI, based on the known intensities of the pelvic girdle. Afterwards, it locates the femoral heads in this ROI by looking for the roughly circular shape of the femoral heads.

Next, we describe several popular methods for the creation of DRR's . Eventually ray casting was chosen as the most suitable method for this project, because ray casting delivers the highest quality images of the described methods and in addition is very easy to grasp and program.

Finally, the graduate had a look at MeVisLab to see if it is suitable to be used for the DRR creation in this project. The DRR's resulting from MeVisLab were off to low quality to be scored for OA. The last research question was not pursued further as it was no longer relevant; the clients preferred a library and proof of concept.

The following Chapters (5 and 6) will describe respectively the program flow of the product and the actual implementation, as well as present the explanation of some practical choices.

5. Design

In order to test if the created library works as intended, a proof of concept will be created. This chapter first describes the proof of concept's program flow. Afterwards, the GUI design will be illustrated and explained.

5.1. Program flow

In order to create an intelligible DRR of the pelvis and hips from a point perpendicular to the hips, and thus fulfill the requirements from Chapter 4.1 and 4.2, the program will take several steps. This means, when not encountering any problems, the application will have the program flow as depicted in Figure 22. Meaning the program will flow through the following steps:

A1. First step is to load a DICOM-file, so a CT scan, into the program.

A2. After the CT scan has successfully been stored into the program's memory, a ROI will be automatically determined.

A3. Then, the focal point, or X-ray source, is automatically set to the point perpendicular to the line connecting the centers of the femoral heads.

A4. Once these steps have been completed, the actual DRR can be created. This DRR will only show the important parts and the hips will be presented in a straight position, which means that there will be as little interference, from other bones, as possible.

A5. Afterwards, the ideal WW and WL will be selected, in order to show as much contrast between the cartilages and bones as is desired, for the DRR to be intelligible when presented.

A6. Finally, the actual presentation of the DRR is shown to the user.

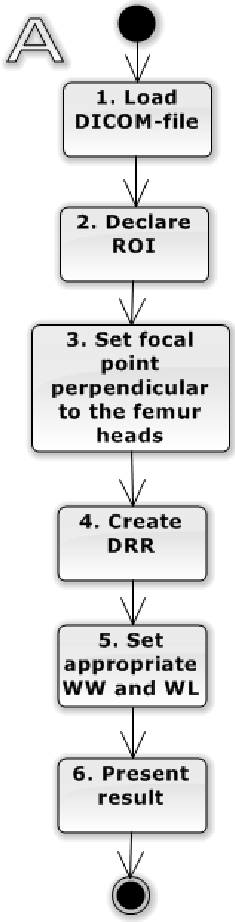


Figure 22. Initial program flow

Now the initial program has been explained briefly, some of the steps will be explained in more detail, starting with how to declare the ROI, based on the research of Chapter 4.2. As depicted in Figure 23, declaring the ROI can be done with just a few feature-based steps.

B1. First an initial MIP needs to be constructed from the base position. This is basically a normal MIP of the entire CT scan, but depending on the CT, is slightly tilted.

B2-3. This MIP needs to be subjected to thresholding. Afterwards, the most densely populated region of the MIP will be located.

B4. Once the region has been found, the only thing that still needs to be determined is where it really starts and ends. This is done by looking at where the intensity gets over the average intensity of the thresholded MIP (beginning) and where it drops below average (ending).

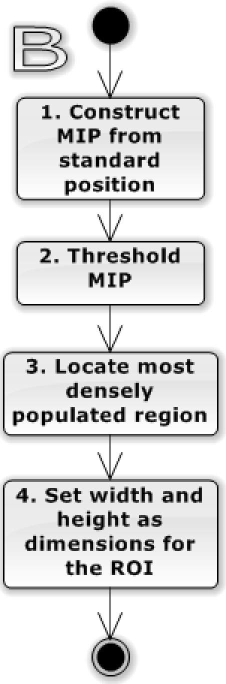


Figure 23. Program flow for declaring the ROI

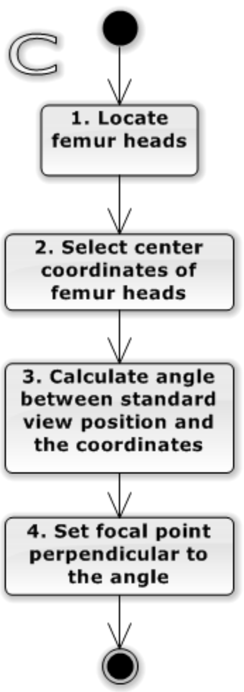


Figure 24. Program flow setting the focal point

Finally, below we describe how to set the focal point perpendicular to the femoral heads.

C1. As depicted in the flow diagram in Figure 24, first the femoral heads need to be located. This can be done in a number of ways; sphere localization in the CT scan and Hough circle transform on the DRR are two of these ways. In Chapter 6.1 one of these methods will be chosen.

C2. Once the femoral heads have been located their center coordinates need to be determined.

C3. These center coordinates will be used to calculate the angles in which they differ from each other per axis, this will be done in 3D.

C4. This is automatically also the angle in which the original focal point will need to be rotated, originating from in between the two femoral heads.

5.2. GUI design

The proof of concept will be fitted with a GUI to allow the clients to easily use it. After several discussions with the clients (Dino Colo and Rob Brink), we came to a small list of wishes regarding the functionalities and design of the GUI (See Table 5 for a small MoSCoW analysis). Implementing as much of these wishes as possible will make the GUI more user friendly and will allow the user to complete his/her task quicker, as they will not need to e.g., open a second program to view the CT scan.

Table 5. MoSCoW analysis GUI

Priority	Functionalities/Requirements
Must have	Ability to show and save the resulting DRR
	Ability to work down a file list of CT scans and convert them all into DRR's
Should have	Ability to show some patient data, as saved in the DICOM header
	Ability to stop the process of constructing the DRR
	Ability to display feedback regarding the steps
Could have	Ability to alter the WW and WL after construction of the DRR
	Ability to scroll through the CT scan
	Ability to construct the DRR step-by-step and showing the results
Won´t have	n/a

In order to meet these wishes the GUI design depicted in Figure 25 has been created. Note that the GUI will not help achieve any requirements from Chapter 4.1, except of course for the requirements that the program can be provided with a GUI.

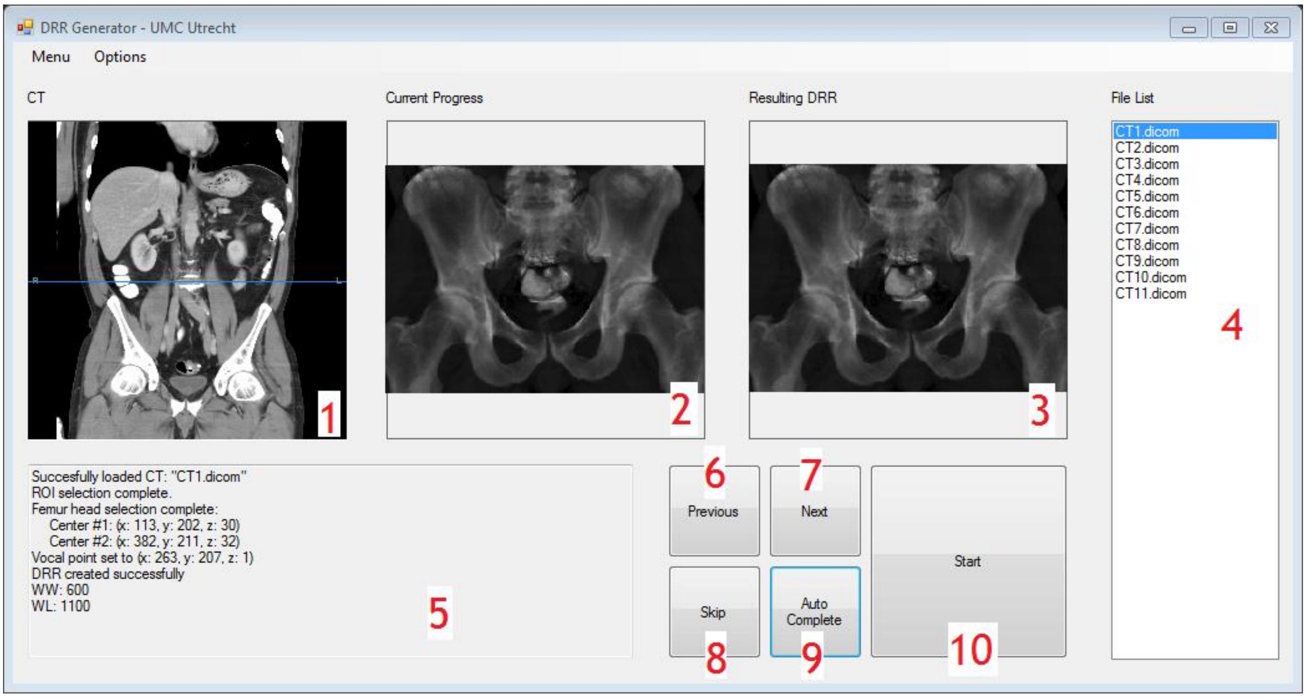


Figure 25. GUI design with numbers for further explanation

As can be seen in Figure 25, the design has been provided with numbers (1-10), in Table 6 each of these numbers will describe the field or button they are fitted to.

Table 6. GUI field and button descriptions

#	Name	Description
1	CT panel	This panel shows the current CT scan.
2	Progress panel	This panel shows the step of the program flow, where the application is currently at.
3	DRR panel	This panel shows the resulting DRR, once available.
4	File list	This is a list of CT scans which will be converted into DRR's. The CT the application is currently working on, is highlighted in the list.
5	Feedback textbox	This is a textbox containing different types of feedback, e.g., which steps have been completed and which values have been selected.
6	Previous button	This button allows the user to go back to the previous step when e.g., something went wrong and the user wants to redo, or simply wants the see what happened between the steps.
7	Next button	This button makes the application execute the next step of the program flow.
8	Skip button	This button allows the user to skip the conversion of a certain CT, e.g., when that CT scan has already been converted in the past.
9	Complete button	This button sets the application to Auto-complete mode, which automatically completes the conversions from now on, or until the user presses the Previous button (6), Next button (7), Skip button (8), Complete button again (9) or the Start button (10) which is labeled "Pause" when the program is running.
10	Start button	This button starts the program and automatically loads the first CT scan in the File list (4) and starts in Auto-complete mode (as explained in Complete button (9)). This button is labeled "Pause" when the program is running.

6. Implementation

This chapter describes the practical implementation of the design (see Chapter 0) by explaining certain choices made while programming the library/proof of concept. Afterwards, the important classes and functions of some classes will be explained and presented in simplified UML.

6.1. Practical implementation

During the implementation of the program flow (explained in Chapter 5.1) no alterations were made to the design. As the design was rather abstract for certain parts, like which method to use for localization of the femoral heads, these choices had to be made during the implementation. Table 7 describes the choices made and the reason for these choices.

Table 7. Choice made for each relevant step op the program flow

Step *(e.g., B3)*	**Choice**	**Description**
B2 – Threshold MIP	20% Percentile thresholding	The desired result after the thresholding is that only the thicker/denser bone tissue is set (to white). As no two CT scanners will share the exact same HUs for the same bone tissue, retaining the highest 20% of the HUs is what is basically achieved using percentile thresholding. The reason for choosing 20% as the percentage is because the bone tissue in the pelvic girdle has high HUs, higher than that of the ribcage and collarbones. This means that those bone structures will fade before the pelvis would, making the pelvis the widest bone structure visible.
B3 – Locate dense region	Histograms used for counting set pixels	A very easy, quick and robust way to locate dense regions is by creating histograms that keep track of the amount of set pixels for both the vertical and horizontal axes. As thoroughly explained in paragraph *Localization for this project* of Chapter 4.2, by calculating the average amount of set pixels and keeping track where the histogram goes over and under this average, wider (over *y-axis*) and higher (over *x-axis*) regions can be located.
C1 – Locate femoral heads	Hough circle transform	As the femoral heads have a roughly circular shape it is possible to look for spheres (3D) or circles (2D) to find them. Even though sphere localization most likely gives very robust results, it is also a very computationally intensive process, as it is in 3D space. Since eventually this program is going to create a DRR of the ROI, we can abuse it to create a 2D image of the CT scan, allowing us to work in 2D space instead of 3D space. This significantly reduces the computation time required to find the femoral heads. A very quick, robust and well-known method for 2D circle localization is the Hough circle transform. This algorithm is used by some of the most prominent computer vision libraries (e.g., OpenCV and MatLab).

Step *(e.g., B3)*	Choice	Description
C2 – Select center locations of femoral heads	Hough circle transform	For the location of the femoral heads the Hough circle transform algorithm was chosen, this algorithm generally determines the location of a circle and the radius of that circle. So finding two circles per femoral head, which is required to obtain 3D-coordinates required for the later steps, the center locations are automatically found as well.
C4 – Set focal point perpendicular to angle	Rotation of the focal point	Setting the focal point perpendicular to the angle between the femoral heads and the standard position of the CT scan can be done two ways: (1) rotate the entire CT scan; (2) rotate the focal point. Each way would have its advantages and disadvantages regarding the creation of the DRR: (1) would not require the calculation of a unit vector to shoot each ray through the CT scan, but would also need to interpolate every single voxel making it very computationally intensive; while (2) is simple, accurate and quick as only a single point has to be interpolated. However, a unit vector will need to be determined and calculated for each step in the CT scan. Rotating the focal point was chosen as only the pixels pierced during the ray casting will have to be interpolated, this means the performance is better while the results are roughly the same.

Table 8. Choice made for each relevant step op the program flow

A choice was also made to use the popular open-source library OpenCV for the implementation, as OpenCV provides a lot of the algorithms required during the execution of the program and has lots of useful options, such as:

- Thresholding;
- Smoothing;
- Rotation (2D);
- Ability to store 16bits of data per pixel (which is useful because DICOM saves the pixels as 16bit values);
- Hough circle transform;
- Histogram creation;
- Edge detection, required for Hough circle transform;
- Etc.

On top of these useful options and provided algorithms, OpenCV has a very well-designed image (*cv::Mat*) class for fast pixel access. OpenCV can also be used for commercial use, it may be distributed, modified, etc. as long as their copyright is included to the project[6].

[6] OpenCV uses the revised BSD 3-Clause license: https://tldrlegal.com/license/bsd-3-clause-license-%28revised%29

6.2. Classes and functions

When trying to read a CT scan, first its file path is fed into ***Dicom***'s constructor. **Dicom** will create:

- A ***DicomHeader***, which will read the DICOM-header of the given file path, storing all the data sections in a dictionary and the most crucial ones, required for the rest of the DRR construction, in private attributes. These crucial attributes include *width, height, bits per pixel, index of the first pixel, etc.*
- A ***DicomBody***, responsible for reading the body of the DICOM-file, which contains all the pixels. In order for the DicomBody to be created, the DicomHeader will also need to be available, as it contains the crucial parameters regarding the amount of pixels and useful data in the bytes.

In order to easily use OpenCV in combination with 3D imaging a class named ***Image3D*** has been created (see Figure 26 for a simplified class representation in UML). This class basically contains an array of *Mat*[7]-objects, corresponding with the CT-slices, and functions for obtaining basic information, e.g., width, height and voxels, all based on 3D coordinates. A few important functions in Image3D are:

- *getTrilinearlyInterpolatedVoxel(…)* which, compared to *getVoxel(…)*, takes coordinates as doubles and calculates its return value based on the trilinear interpolation of the given coordinates;
- *createSubImage(…)* which allows the user to create an Image3D of a smaller portion of the current Image3D;
- *getSagittalSlice(…)* which, like *getSlice(…)*, returns one slice of the Image3D as a Mat, but over the sagittal plane instead of the coronal plane.

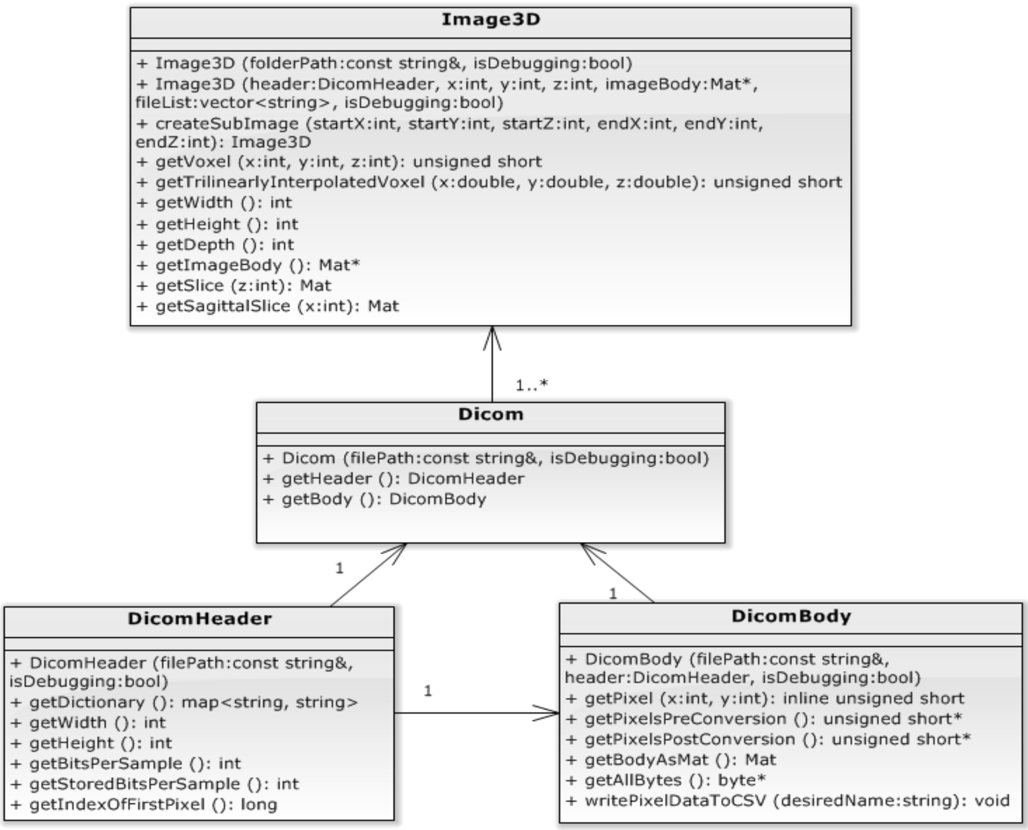


Figure 26. Simplified representation of the public functions of classes *Image3D*, *Dicom*, *DicomHeader* and *DicomBody* in UML

[7] *Mat* is the class used in the OpenCV library containing all the image data.

The project also uses three classes filled with only static attributes and functions (depicted in Figure 27). These classes are:

- ***Histogram***, which is a class with static functions that return an integer pointer containing data about the amount of set pixels. The class also has a function to simply calculate the minimum, maximum and average value of a histogram and a function to create an image (*Mat*) when provided with a histogram. These functions can be used both vertically as well as horizontally;
- ***Thresholding*** is a class with only one function. This function is *pTileThresholding(...)*, it takes two arguments; an input image and a desired percentage of the input image to be set after the thresholding is completed. After completion, an the thresholded image (Mat) is returned. Thresholding is simply an addition to OpenCV which lacked a percentile thresholding method.
- ***ROI***, is the class responsible for finding the ROI based on three histograms, which are the input for the static function *findROI(...)*. This function also takes the lengths of the histograms used as input; these should be the same as the dimensions of the original *Image3D*. Once executed, the function will set the static parameters *startX, startY, startZ, endX, endY* and *endZ* to the values associated to the rectangles that portrait the *Image3D*'s ROI.

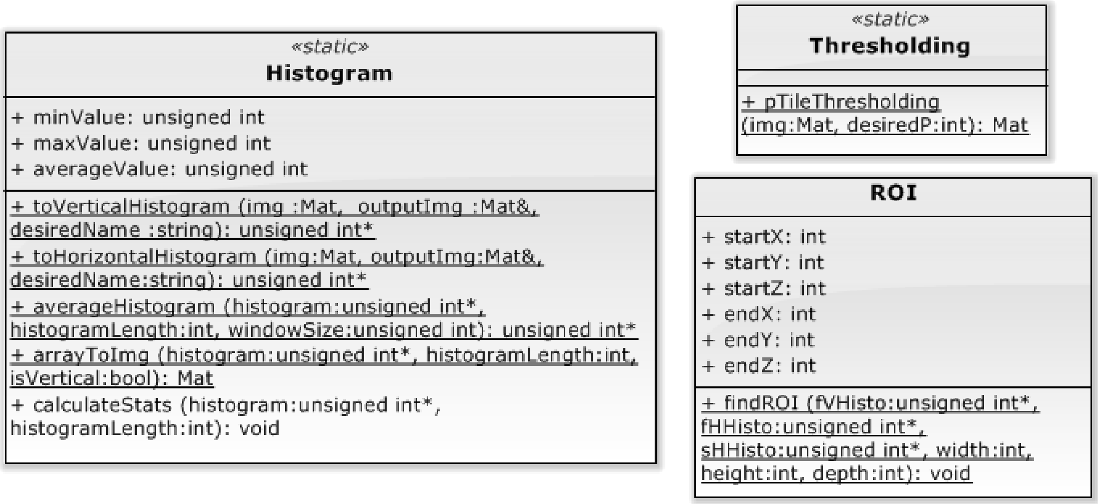


Figure 27. Simplified representation of the static functions and attributes of classes *Histogram* and *Thresholding* in UML.

The class responsible for the creation of DRR's is named ***DRR***. This class, as depicted in Figure 28, has an *enum* named *DrrType,* which can be set to *MIP* or *AVG* (=average)[8]. This enum is used for two functions:

- sideRayCasting(…), which creates a DRR or MIP from the side, so through the sagittal plane, of the Image3D used to construct the ***DRR***-object;
- rayCasting(…),which creates a DRR or MIP from the front of the Image3D used to construct the ***DRR***-object.

Both functions also take the *bool convertTo8bit*, which - when set - converts the (originally 12-bit) pixel values to 8-bit values, this can be useful when the user simply wants to display the DRR or when the user does not want to alter the WW/WL later.

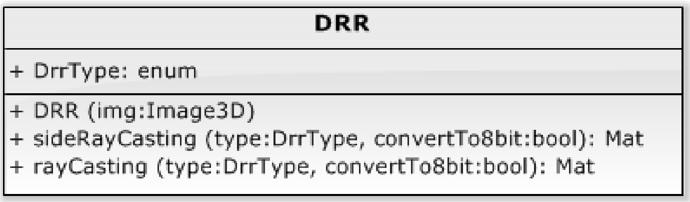


Figure 28. Simplified representation of the static functions of the class *DRR* in UML.

[8] When DrrType is AVG an actual DRR will be created.

In order to shoot the DRR from such an angle that the patient in the CT scan is no longer tilted, the femoral heads need to be located. The class ***FemoralHeadLocator***, as depicted in Figure 29, is responsible for this. It can be constructed either by inputting an ***Image3D***-object containing only the ROI (from ***ROI***), or by inputting the full CT as ***Image3D***-object and adding the start and end coordinates of all axes.

The class uses the public function named *locateFemoralHeads(…)* to locate the 3D coordinates of the femoral heads. This function returns a ***Point3D***-pointer, containing two ***Point3D***s. The function calls *locateFemoralHeadsCoronalPlane(…)* and *locateFemoralHeadsSagittalPlane(…)* respectively to locate the coordinates. These functions use the method proposed in Chapter 4.3 paragraph "Localization for this project", to each return two 2D-coordinates, one for each femoral head. Combining these two 2D-coordinates will result in one 3D-coordinate of a femoral head. Note that both these functions use OpenCV's *HoughCircleTransform(…)* to locate the roughly circular femoral heads.

The function *locateFemoralHeadsSagittalPlane(…)* actually takes the x-coordinate of both femoral heads located in *locateFemoralHeadsCoronalPlane(…)*. This has been done so the function can simply use the *getSagittalSlice(…)* function from class ***Image3D***, and search that slice for a femoral head.

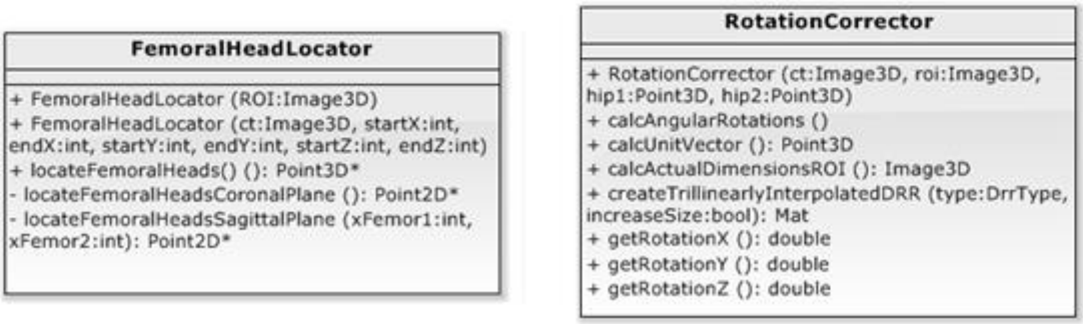


Figure 29. Simplified representation of classes *FemoralHeadLocator* and *RotationCorrector* in UML.

Finally, there is one class ultimately responsible for determining and correcting the rotation in the DRR, the ***RotationCorrector*** (depicted in Figure 29). The constructor for this class takes entire CT-scan as ***Image3D***-object, the ROI as ***Image3D-object*** and two ***Point3D***-objects each containing the location of a femoral head. A few important functions of this class include:

- *calcAngularRotations(…)* which calculates the angle of each axis in degrees and stores it in the private variables *rotX, rotY, rotZ*;
- *calcUnitVector(…)* which calculates the unit vector, which the rays will use for ray casting during the creation of a corrected DRR. Note that this unit vector will always have its *z-value* set to 0.1 and the other values scaled accordingly;
- *calcActualDimensionsROI(…)* which calculates the actual width of the pelvis on the DRR, as if the patient in tilted in the CT, the algorithm used to determine the width of the ROI is incorrect. The actual value can be obtained by using *Phythagorean theorem* in 3D. Also a new Image3D is returned which is big enough for the unit vector to travel the same distance as it could at first.
- *createTrilinearlyInterpolatedDRR(…)* which is a very important function, as it actually creates a DRR that has corrected the rotations of the patient. Note that this can take either a AVG or MIP *DrrType* and that the bool *increaseSize* can be used to trilinearly increase the DRR size (width * 2 and height * 2).

7. Conclusion and recommendations

In this chapter the main research question will be answered based on the information acquired from answering the sub research questions. Afterwards, some recommendations will be made regarding the project.

7.1. Conclusion

In order to create a program capable of automatically constructing intelligible DRR's of the hips, several topics needed to be researched.

First, a MoSCoW prioritization analysis was conducted based on multiple discussions with the clients, Table 9 presents the "Must haves" of this analysis.

Table 9. Resulting "*Must haves*" of the MoSCoW prioritization analysis

Priority	Functionalities/Requirements
Must have	Ability to construct DRR from CT
	In the resulting DRR the bones are visible enough for medical professionals to score OA
	The DRR's are of high enough quality for medical professionals to score them for OA
	The proof of concept is a standalone program
	The results are reproducible
	The results are reliable; all the DRR's have to be created using the same method(s)

Afterwards, research was conducted to accumulate information about some of the most well-known methods for the construction of DRR's. This research shows that of the available methods, only ray casting and splat rendering are currently still fairly popular.
After presenting several DRR's created by using splat rendering and by using ray casting to medical professionals, it was concluded that splat rendering was not suitable for the research Rob Brink and Dino Colo are conducting. Ray casting however, resulted in DRR's of high enough quality to meet the clients' demands and these results can even be enhanced further through trilinear interpolation (at the cost of speed).

Once it was clear which method would be used for the construction of the DRR's, research was conducted on when the resulting DRR's could be viewed as *intelligible* and how that could be achieved. The clients made a number of demands for when the DRR would be acceptable for their research and thus be viewed as intelligible:

- The results are reliable; all the DRR's have to be created using the same method(s);
- The rotation of the hips/pelvis has to be compensated for, when possible;
- The ROI needs to contain the entire pelvis and the femoral heads;

To prove these demands have been met, a simple test has been presented:
A group of medical professionals will judge a series of ±10 DRR's created by the proof of concept on the three demands mentioned above and the general intelligibility. After roughly a week, this test will be repeated using the same DRR's. Possible alternating results will deem the DRR's fit or unfit.

Research was also conducted to determine if MeVisLab, or another existing DRR program, is a suitable program for the construction of DRR's, to use as a basis for the proof of concept. After some brief testing in MeVisLab, it could be concluded that this tool would not be deemed useful for the scoring of OA and thus not useful for this research. Afterwards, the clients preferred a library with functions for the construction of intelligible DRR's of the hips, proved by a standalone proof of concept.

Finally, research on how to locate the hips, pelvis and femoral heads (the ROI) in CT scans was conducted. The graduate proposed a feature-based technique tailor-made for finding the pelvic region in CT scans of 'normal' human beings, by creating MIP's and searching in 2D. The technique uses the pelvic girdle's high amount of bone tissue and high HUs, as well as the rough circular shape of the femoral heads to find the ROI and afterwards the centers of the femoral heads, in order to correct the axes in which the patient was not laying straight.

So, by creating a program that allowed the creation of MIP's and DRR's using ray casting, the ROI can be located using simple 2D feature-based techniques. Afterwards, locating the femoral heads and correcting the patient's position in the ROI and post-processing the DRR, will result in intelligible DRR's of the hips, pelvis and femoral heads.

7.2. Recommendations

Based on the proof of concept's results and the research done in order to create the proof of concept, some recommendations can be made. These recommendations regard the technique used to determine the ROI, the program's ability to read DICOM-files, the *magic values* used for OpenCV's Hough circle transform and its robustness.

Determining the ROI
Currently the proof of concept uses a very basic algorithm for determining the ROI. Nearing the end of the project, one of the CT scans in the test set seemed to contain an artifact. The algorithm used to determine the ROI still found the correct region, if the artifact was slightly bigger however, it could have influenced the results. It is possible to improve the robustness of this part of the program by having a closer look at the measurements of the peaks found in the histograms over each axis, e.g., distance between peaks, height of peaks, etc.
Another way of improving the robustness might be by, when e.g., the height of the ROI has been determined, to only measure the peaks between the start and end of the MIP over the y-axis.

Reading DICOM-files
There are two formats in which DICOM files can be presented; firstly by storing the header information in the **.dicm*-file and the pixel data in a **.tiff*-file, secondly by storing all the data (header and body) in the **.dicm*-file. Currently the program only works with the last of these two. The reason the program only works with the last of the mentioned formats is because the DICOM files in the test set all used this format. The program could be expanded to include the first format as well as the second.

OpenCV's Hough circle transform
The function for OpenCV's Hough circle transform takes multiple parameters. These parameters include standard parameters like the input image, an array to store the data and some default values. However, it also contains some parameters that are not so trivial:

- *param1*: the upper threshold for the Canny edge detector, which is one of the steps required to find the circles (default value = 200);
- *param2*: the accumulator's threshold for center detection (default value = 100);

- *min_radius* and *max_radius*: obviously regarding the radius of the found circles. However, when left at *0* the radius is unknown and the function will try to figure it out.

For parameters *param1* and *param2*, the program currently uses values that are rather hard to explain. This is because changing these values can easily cause the program to no longer find the desired circles. In order to determine the ideal value for both parameters, a range was determined for each input image containing a femoral head. This part of the gathered data can be found in Table 10.

Table 10. Example ranges for *param1* and *param2*

Param1 (Canny threshold):	
Test image #	**Range**
1	71-95
2	88-139
3	82-123

Param2 (Accumulator threshold):	
Test image #	**Range**
1	11-83
2	1-92 (93-94)
3	131-135 (1-130)

Currently the parameter values are determined by taking the highest of the lower thresholds and the lowest of the upper thresholds to create a range suitable for all of the tested images and using the mean of that range as the parameter value. To improve the robustness of this part of the program, collecting more data regarding these parameters and narrowing down the range even more, could yield better results.
If OpenCV's Hough circle transform eventually somehow proves unfit for the program it could be considered to either rewrite the function or think about an alternative way of locating the center of the femoral heads, e.g., a variation of the algorithm used in (Harris, Reese, Peters, Weiss, & Anderson, 2013).

8. Process evaluation

While conducting the research, designing and implementing the proof of concept and writing the thesis some issues worth mentioning were encountered. This chapter describes those issues and how they were addressed.

Strict planning

At first following the schedule, as described in the thesis plan, was no issue. Almost everything took as much time as expected. Sadly at one point the graduate got delayed a few weeks, once behind it seemed almost impossible to get back on schedule. The schedule left little margin for error. After the issue had been discussed with Dino Colo, the issue was addressed by programming the GUI after the thesis was finished instead of before. This allowed the graduate to get back on track, until he got sick for a week in which case he had to request postponement.

Relevant medical literature

As the graduate's background is a technical one, finding literature related to the subject on a technical level was not a problem. However, finding medical literature, relevant, to the project proved not to be as easy. Luckily, Rob and Dino were very helpful in helping the graduate with finding the desired literature and explained how to find it more easily, which was basically by using the *correct terms* and search engines.

Low DRR resolution

After creating some DRR's using the proof of concept it seemed their resolution was very small, usually 300x210 pixels. Dino pointed out that in order for the medical professionals to be able to score for OA more easily they would require bigger images. At first the graduate was skeptical, the zooming in on the image would make it bigger. However, this would of course pixelate the image or blur it. Koen pointed out that the graduate already uses a method that allows an increase in resolution, trilinear interpolation. Shooting 4 rays per voxel e.g., *{(0;0), (0,5;0), (0;0,5), (0,5;0,5)} (See* Figure 30 *for a visual example)*, would increase the image's width and height both by twofold, creating an image that has roughly the size 600x420pixels that actually has even better results. This size is also big enough for medical professionals to score for OA more easily.

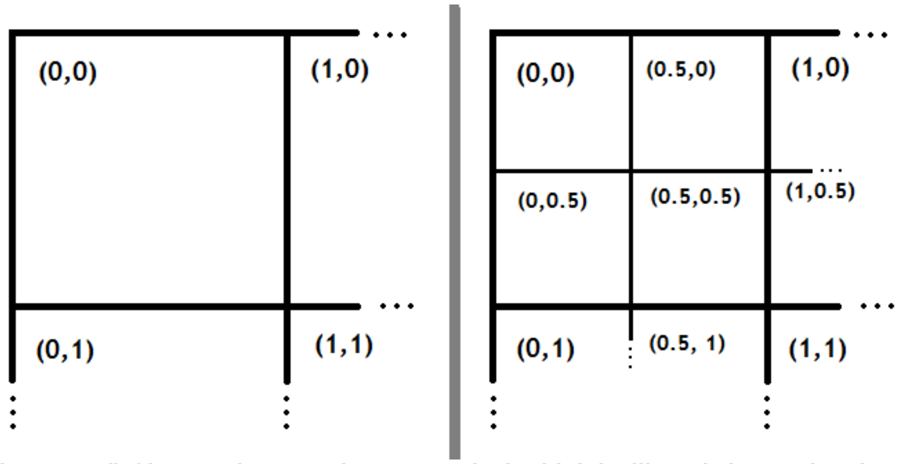


Figure 30. (left) normal ray casting target pixels. (right) trilinearly interpolated ray casting pixels.

9. References

Introduction to CT physics. (2004, July 5). Retrieved June 22, 2015, from Elsevier Health: http://web.archive.org/web/20070926231241/http://www.intl.elsevierhealth.com/e-books/pdf/940.pdf

Representación y Rendering de Volúmenes. (2005, May 12). Retrieved August 5, 2015, from VISUALIZACIÓN CIENTÍFICA: http://www.lip.uns.edu.ar/vcuba/vol.htm

Alakuijala, J., Jäske, U., Sallinen, S., Helminen, H., & Laitinen, J. (1996, October). Reconstruction of Digital Radiographs by Texture Mapping, Ray Casting and Splatting. *Engineering in Medicine and Biology Society, 1996. Bridging Disciplines for Biomedicine. Proceedings of the 18th Annual International Conference of the IEEE, 2*, 643-645.

Altman, R., Alarcón, G., Appelrouth, D., Bloch, D., Borenstein, D., Brandt, K., et al. (1991, May). The American College of Rheumatology criteria for the classification and reporting of osteoarthritis of the hip. *Arthritis & Rheumatism, 34*(5), 505-514.

Atherton, T. J., & Kerbyson, D. J. (1999, September). Size invarient circle detection. *Image and Vision Computing, 17*(11), 795-803.

Bahner, M., Debus, J., Zabel, A., Levegrun, S., & van Kaick, G. (1999, October). Digitally Reconstructed Radiographs From Abdominal CT Scans as a New Tool for Radiotherapy Planning. *Investigative Radiology, 34*(10), 643.

Birkfellner, W., Seemann, R., Figl, M., Hummel, J., Ede, C., Homolka, P., et al. (2005). Fast DRR Generation for 2D/3D Registration. In J. Duncan, & G. Gerig (Eds.), *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2005* (pp. 960-967). Palm Springs: Springer Berlin Heidelberg.

Boulay, C., Tardieu, C., Hecquet, J., Benaim, C., Mouilleseaux, B., Marty, C., et al. (2006, April). Sagittal alignment of spine and pelvis regulated by pelvic incidence: standard values and prediction of lordosis. *European Spine Journal, 15*(4), 415-422.

Bradley, J., & Amde, M. (2015, January 21). *Random Forests and Boosting in MLlib*. Retrieved Oktober 9, 2015, from databricks: https://databricks.com/blog/2015/01/21/random-forests-and-boosting-in-mllib.html

Breiman, L. (2001). Random Forests. (R. Schapire, Red.) *Machine Learning*, 5-32.

Bushberg, J. T., & Boone, J. M. (2011). *The Essential Physics of Medical Imaging* (3rd ed.). Philadelphia: Wolters Kluwer Health/Lippincott Williams & Wilkins.

Chan, W., Lang, P., Stevens, M., Sack, K., Majumdar, S., Stoller, D., et al. (1991). Osteoarthritis of the knee: comparison of radiography, CT, and MR imaging to assess extent and severity. *American Journal of Roentgenology, 157*(4), 799-806.

Chen, C., & Zheng, G. (2013). Fully Automatic Segmentation of AP Pelvis X-rays via Random Forest Regression and Hierarchical Sparse Shape Composition. In R. Wilson, E. Hancock, A. Bors, & W. Smith, *Computer Analysis of Images and Patterns* (pp. 335-343). York: Springer.

Dafni, E. (2006, May 31). *Patent No. US 8111804 B2.*

DSDM Consortium. (2015, September 11). *MoSCoW Prioritisation | DSDM CONSORTiUM*. Retrieved September 11, 2015, from DSDM CONSOTRiUM: http://www.dsdm.org/content/10-moscow-prioritisation

Forsberg, D. (2015). Atlas-Based Registration for Accurate Segmentation of Thoracic and Lumbar Vertebrae in CT Data. In J. Yoa, B. Glocker, T. Klinder, S. Li, J. Tavares, & R. Natal Jorge (Red.), *Recent Advances in Computational Methods and Clinical Applications for Spine Imaging* (Vol. 20, pp. 49-59). Springer.

Glocker, B., Feulner, J., Criminisi, A., Haynor, D., & Konukoglu, E. (2012). Automatic Localization and Identification of Vertebrae in Arbitrary Field-of-View CT Scans. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI* (pp. 590-598). Springer Berlin Heidelberg.

Hammond, B. T., & Charnley, J. (1967, September). The sphericity of the femoral head. *Medical & Biological Engineering & Computing - Journal of the International Federation for Medical and Biological Engineering, 5*(5), 445-453.

Hansen, C. D., & Johnson, C. R. (Eds.). (2005). *Visualization Handbook.* Elsevier.

Harris, M. D., Reese, S. P., Peters, C. L., Weiss, J. A., & Anderson, A. E. (2013, June). Three-dimensional Quantification of Femoral Head Shape in Controls and Patients with Cam-type Femoroacetabular Impingment. *Annals of Biomedical Engineering, 41*(6), 1162-1171.

Hatton, S. (2007). Early Prioritisation of Goals. In J.-L. Hainaut, E. A. Rundensteiner, M. Kirchberg, M. Bertolotto, Y.-P. P. Chen, S. S.-S. Cerfi, et al., *Advances in Conceptual Modeling - Foundations and Application* (Vol. 4802 2007, pp. 235-244). Auckland, New Zealand: Springer.

Hogeschool Utrecht, Faculteit Natuur en Techniek, Institute for ICT. (2014, Juli 7). *Afstudeerleidraad Instituut voor ICT - juli 2013 - Afstudeerleidraad Instituut voor ICT cursus 2014-2015.pdf.* Retrieved June 10, 2015, from Bedrijven - Hogeschool Utrecht: www.voorbedrijven.hu.nl/~/media/HU-BEDRIJVEN/docs/Praktijkbureau%20cluster%20ICT/Afstudeerleidraad%20Instituut%20voor%20ICT%20cursus%202014-2015.pdf?la=nl

Hutt, H., Everson, R., & Meakin, J. (2015). Segmentation of Lumbar Vertebrae Slices from CT Images. In J. Yao, B. Glocker, T. Klinder, & S. Li, *Recent Advances in Computational Methods and Clinical Application for Spine Imaging* (pp. 61-71). Springer.

ISI Utrecht. (2006, Januari 26). *Image Sciences Institute: About*. Retrieved May 25, 2015, from http://www.isi.uu.nl/About/

Jentzsch, T., Geiger, J., Bouaiacha, S., Slankamenac, K., Nguyen-Kim, T. D., & Werner, C. M. (2013, November 5). Increased pelvic incidence may lead to arthritis and sagittal orientation of the facet joints at the lower lumbar spine. *BMC Medical Imaging, 13*, 34-44.

John, O. P., & Soto, C. J. (2007). The importance of being valid: Reliability and the process of construct validation. In R. W. Robins, R. C. Fraley, & R. F. Krueger, *Handbook of research methods in personality psychology* (pp. 461-494). Guilford Press.

Kaufman, A., & Meuller, K. (2005). Overview of Volume Rendering. In C. D. Hansen, & C. R. Johnson, *Visualization Handbook* (pp. 127-174). Elsevier.

Kaur, M., Kaur, J., & Kaur, J. (2011). Survey of Contrast Enhancement Techniques based on Histogram Equalization. *International Journal of Advanced Computer Science and Applications, 2*(7), 137-141.

Korez, R., Ibragimov, B., Likar, B., Pernuš, F., & Vrtovec, T. (2015). Interpolation-Based Shape Constrained Deformable Model Approach for Segmentation of Vertebrae from CT Spine Images. In J. Yao, B. Glocker, T. Klinder, & S. Li, *Recent Advances in Computational Methods and Clinical Applications for Spine Imaging* (pp. 235-240). Springer.

Meißner, M., Huang, J., Bartz, D., Mueller, K., & Crawfis, R. (2000, October). A Practical Evaluation of Popular Volume Rendering Algorithms. *Proceedings of the 2000 IEEE symposium on Volume visualization*, 81-90.

Mendeley. (n.d.). *Overview | Mendeley*. Retrieved June 11, 2015, from www.mendeley.com: https://www.mendeley.com/features/

National Institute of Dental and Craniofacial Research. (2014, March 6). *Version Control Guidelines.* Retrieved June 17, 2015, from National Institute of Health (NIH): http://www.nidcr.nih.gov/Research/ToolsforResearchers/Toolkit/VersionControlGuidelines.htm

Royce, W. (1970, augustus). *Managing the Development of Large Software Systems.* Retrieved June 10, 2015, from http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf

Ruijters, D., ter Haar Romeny, B., & Suetens, P. (2008). GPU-ACCELERATED DIGITALLY RECONSTRUCTED RADIOGRAPHS. *BioMED*.

Russakoff, D. B., Rohlfing, T., & Maurer, C. R. (2003, October). Fast Intensity-based 2D-3D Image Registration of Clinical Data Using Light Fields. *International Conference on Computer Vision, 1*, 416-422.

Schlösser, T., Janssen, M., Vrtovec, T., Pernuš, F., Cumhur Öner, F., Viergever, M., et al. (2014, July). Evolution of the ischio-iliac lordosis during natural growth and its relation with the pelvic incidence. *European Spine Journal, 23*(7), 1433-1441.

Schünke, M., Ross, L. M., Schulte, E., Schumacher, U., & Lamperti, E. D. (2006). *Thieme Atlas of Anatomy: General Anatomy and Musculoskeletal System* (Vol. 1). Thieme.

Shanmugavadivu, P., & Balasubramanian, K. (2011). Image Edge and Contrast Enhancement Using Unsharp Masking and Constrained Histogram Equalization. *Communications in Computer and Information Science, 140*, 129-136.

Spörk, J. (2010). *High-performance GPU based Rendering for Real-Time, rigid 2D/3D-Image Registration in Radiation Oncology.* Graduation thesis, University of Vienna, Faculty of Computer Science, Vienna.

Turmezei, T., Fotiadou, A., Lomas, D., Hopper, M., & Poole, K. (2014, October). A new CT grading system for hip osteoarthritis. *Osteoarthritis and Cartilage, 22*(10), 1360-1366.

UMC Utrecht. (n.d.). *About us - UMC Utrecht*. Retrieved May 25, 2015, from http://www.umcutrecht.nl/en/About-us

Vrtovec, T., Janssen, M. M., Pernus, F., Castelein, R. M., & Viergever, M. A. (2012, April 15). Analysis of Pelvic Incidence From 3-Dimensional Images of a Normal Population. *Spine, 37*(8), 479-485.

Wikipedia. (2015, July 2). *Trilinear interpolation*. Retrieved August 5, 2015, from Wikipedia, the free encyclopedia: https://en.wikipedia.org/wiki/Trilinear_interpolation

Wittenauer, R., Smith, L., & Aden, K. (2013, January 28). *Background Paper 6.12 Osteoarthritis.* Retrieved October 13, 2015, from http://www.who.int/medicines/areas/priority_medicines/BP6_12Osteo.pdf

Yoshimoto, H., Sato, S., Masuda, T., Kanno, T., Shundo, M., Hyakumachi, T., et al. (2005, July 15). Spinopelvic Alignment in Patients With Osteoarthrosis of the Hip: A Radioographic Comparison to Patients with Low Back Pain. *Spine, 30*(14), 1650-1657.

APPENDIX A. Thesis plan

UNIVERSITY MEDICAL CENTER UTRECHT
IMAGE SCIENCES INSTITUTE

Automated Generation of Pelvic CT-based DRR's

Thesis Plan

Cover image:
DRR of the hips, pelvis and lower lumbar

Author:
Alexander H.I.P.R. Hustinx

Student number:
1604608

Version:
v1.5

Date:
08/18/2015

Mentors:
Dr. Ir. Koen Vincken
Associate Professor, Image Sciences Institute
University Medical Center Utrecht
K.Vincken@umcutrecht.nl

Dino Colo, MD
Researcher, Department of Orthopedic surgery
University Medical Center Utrecht
D.Colo-2@umcutrecht.nl

Examiners:
Dr. Leo van Moergestel
First Examiner
University of Applied Sciences Utrecht
Leo.vanMoergestel@hu.nl

Dr. Esther van der Stappen
Second Examiner
University of Applied Sciences Utrecht
Esther.vanderStappen@hu.nl

I. Term definitions

Computed Tomography (CT) scan	CT scans are created by combining multiple X-ray images from different angles to produce tomographic images.
Digitally reconstructed radiograph (DRR)	A digitally reconstructed radiograph, or DRR, is a two-dimensional image representing an X-ray. It is usually created from a CT scan or other three-dimensional source.
DICOM	DICOM stands for Digital Imaging and Communications in Medicine. It is a standard for the handling, storing and communicating information in medical imaging. It also defines its own image file format.
HU	Hogeschool Utrecht, aka. University of Applied Sciences Utrecht is the institute the graduate is from.
Hounsfield units (HUs)	Hounsfield units are a unit of measurement for radiodensity.
OsiriX	OsiriX is a popular (Mac-only) DICOM-viewer. It is also UMC Utrecht's current choice for manual conversion to DRR's.
Osteoarthritis (OA)	Osteoarthritis is a degenerative joint disease, caused by multiple factors, leading to a thinned cartilage. In this document osteoarthritis, or OA, will refer to osteoarthritis of the hip, unless specifically mentioned. *e.g. OA of the knee.*
Pelvic incidence (PI)	The pelvic incidence (PI) describes the sagittal pelvic alignment and is position-independent. (Jentzsch, et al., Increased pelvic incidence may lead to arthritis and sagittal orientation of the facet joints at the lower lumbar spine, 2013)
University Medical Center (UMC)	In this document UMC will refer to University Medical Center *Utrecht*, unless specifically mentioned. *e.g. UMC Groningen*.
Window level (WL)	the window level, or WL, is the Hounsfield unit value in the center of the *window width.*
Window width (WW)	The window width, or WW, describes the range of Hounsfield units displayed in a CT scan.

Table of Contents

1. Introduction

The University Medical Center (UMC) Utrecht, more specifically the Department of Orthopedic surgery, is currently researching the relevance of spinopelvic parameters to osteoarthritis (OA) of the hip. Roughly 350 computed tomography (CT) scans of patients aged between 50 and 90 years, are available for this research. For some parts of the research a two-dimensional projection of the CT scan data is sufficient. This is similar to an X-ray.

It is possible to convert CT scan data into a digitally reconstructed radiograph (DRR) (Ala(Alakuijala, Jäske, Sallinen, Helminen, & Laitinen, Reconstruction of Digital Radiographs by Texture Mapping, Ray Casting and Splatting, , which resembles an X-ray. However, converting the data manually, using tools like OsiriX[9], is not reliable for research purposes, because relying on user input makes the conversion irreproducible.

Therefore, software needs to be used that enables the automated and reproducible generation of intelligible CT-based DRR's.

In this document, the reader will find information about the project context, including the theoretical framework, followed by the project definition, explaining the assignment, the goals, the research questions and methods, the deliverables, the project scope and the approach. A planning is presented with a phasing of the project, the milestones and a representation of the full planning. Then, the project management will be described, listing the people involved during the project, agreement on communication between them and the graduate, relevant to the project, and finally the risk management.

[9] OsiriX is a Mac only program used for viewing and manipulating DICOM-files. UMC Utrecht currently uses it for the generation of DRR's from CT's.

2. Project context

In this chapter, information about the organization where the project is going to be executed can be found, followed by some background information about the ongoing research on the relation between spinopelvic parameters and OA and some background information on medical imaging. Finally, the reader will be presented with the motivation for the research.

2.1. University Medical Center and Image Sciences Institute

The graduation assignment will be executed at the University Medical Center (UMC) Utrecht at the Imaging Sciences Institute (ISI), in collaboration with the Department of Orthopedic surgery.

UMC Utrecht has about 12,000 employees, making it one of the largest public health care institutions in the Netherlands. UMC Utrecht can be described with three key words: *care, research* and *education* (UMC Utrecht). Its main goal is of course to *care* for their patients. But it also aims to conduct exceptional *research* on the international stage, which can then be translated into patient care. And high-quality *education* guarantees the arrival of new talent, allowing UMC to maintain and strengthen its position.

ISI, housed within the Department of Radiography in UMC Utrecht, employs around 50 researchers, making it one of the largest medical image research groups in Europe. Because of its location, a close collaboration between scientists, clinicians and radiologists is enabled (ISI Utrecht, 2006).

2.2. Spinopelvic parameters and osteoarthritis of the hip project

The Department of Orthopedic surgery, within UMC Utrecht, is currently researching the relation between spinopelvic parameters and osteoarthritis of the hip.
Osteoarthritis (OA) is a degenerative joint disease caused by multiple factors leading to thinned cartilage. OA can result in painful and swollen joints. To complete the diagnosis for OA, X-rays of the joints is the most used imaging method (Altman, et al., The American College of Rheumatology criteria for the classification and reporting of osteoarthritis of the hip, 1991).

The hips, pelvis and spine are closely related to each other. In this complex there are a lot of parameters that could be measured to explain the relationship between these structures. The pelvic incidence (PI) is one of the most investigated parameters (Figure 1a) (Boulay, et al., Sagittal alignment of spine and pelvis regulated by pelvic incidence: standard values and prediction of lordosis, 2006). Another parameter is the ischio-iliac lordosis (IIL), recently described and studied by the Department of Orthopedic surgery in (Schlösser, et al., Evolution of the ischio-iliac lordosis during natural growth and its relation with the pelvic incidence, 2014) (Figure 1b).

It is suggested by other studies that the pelvic parameters in patients with OA of the hip differ from patients without OA, which could have a predicted value to OA. However, there is no study that directly compares the pelvic parameters measured on CT scans between patients with OA of the hip and patients without OA.

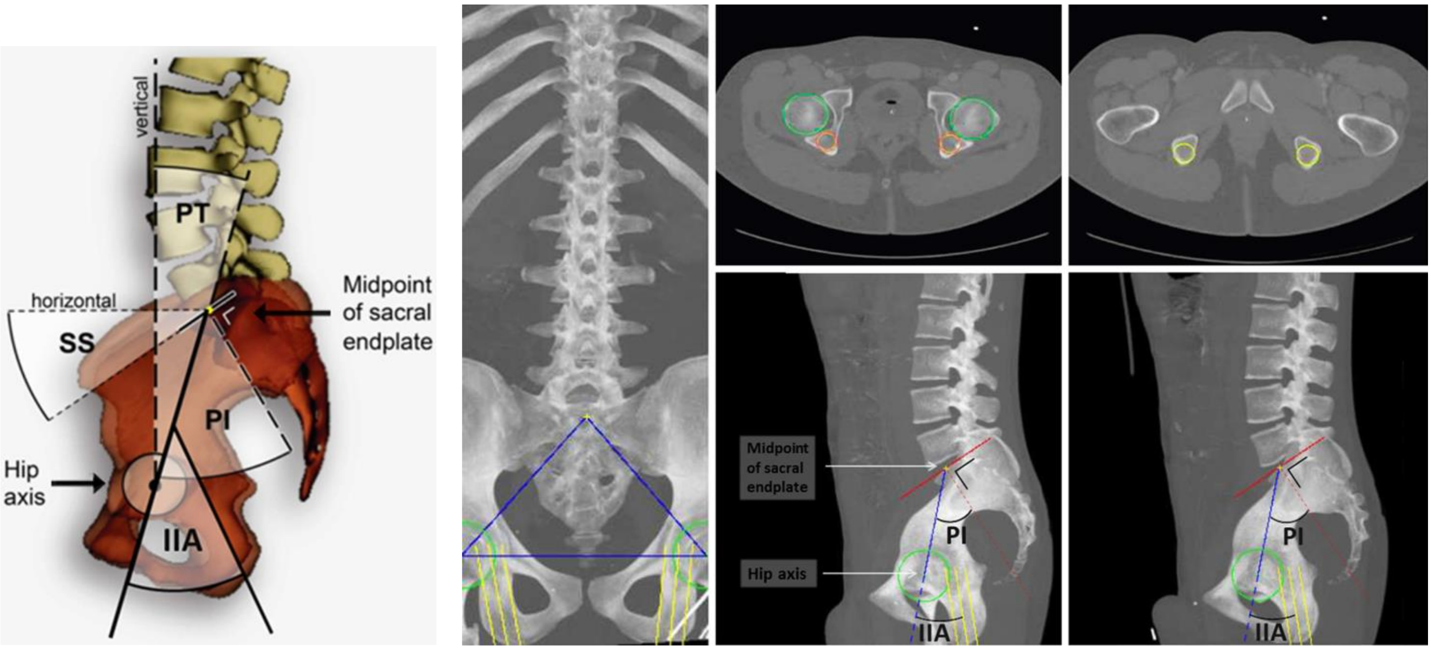


Figure 31a. (left) Pelvic parameters in the sagittal plane are described. The pelvic incidence (PI) is the angle between the perpendicular of the sacral endplate and the line connecting the sacral endplate to the hip axis. (Schlösser, et al., Evolution of the ischio-iliac lordosis during natural growth and its relation with the pelvic incidence, 2014)
Figure 1b. (right) Computerized measurements of the ischio-ilias angle (IIA) and PI on CT scans, using in-house developed software. (Schlösser, et al., Evolution of the ischio-iliac lordosis during natural growth and its relation with the pelvic incidence, 2014)

The aim of the current study is to compare the PI and IIL between people with OA of the hip and people without OA. The parameters will be evaluated in CT scans of the patients aged between 50 and 90 years.

The best way to measure the pelvic parameters is in a three-dimensional way, e.g., using CT scans. Despite the high quality of current CT scans, the conventional imaging method to diagnose OA is still the two-dimensional X-ray (Chan, et al., Osteoarthritis of the knee: comparison of radiography, CT, and MR imaging to assess extent and severity., 1991). For that reason the CT scans should also be reconstructed to X-rays. Such a reconstruction is called a digitally reconstructed radiograph (DRR), see Figure 3 for an example. Dino Colo and Rob Brink currently use OsiriX for this purpose.

Figure 32. Example of a DRR, constructed from CT-data, of the hips.

2.3. Medical imaging and image processing

Medical imaging is the process of creating visual representation of the interior body, for medical or clinical analysis. In this document only medical imaging and image processing regarding X-rays and CT scans is relevant. Medical image processing is the process of retrieving information from the images created by medical imaging, e.g., MRI, CT, X-ray, etc.

During this project the graduate will mainly work with CT scan data. A computed tomography, or CT, scan is a helical tomography technique which creates two-dimensional data of the hard- and soft tissue inside of the human body. In order to create the two-dimensional data, a beam of X-rays will spin around the object. The X-rays will pass through the object from multiple angles and will be picked up by radiation detectors. Afterwards a computer analyzes the information picked up by the detectors and processes it, this is done slice-by-slice. During a CT scan the object is enclosed in a ring, with on one side the X-ray source and the opposite side a series of sensitive radiation detectors. This ring will spin around the object in order to obtain information from multiple angles.

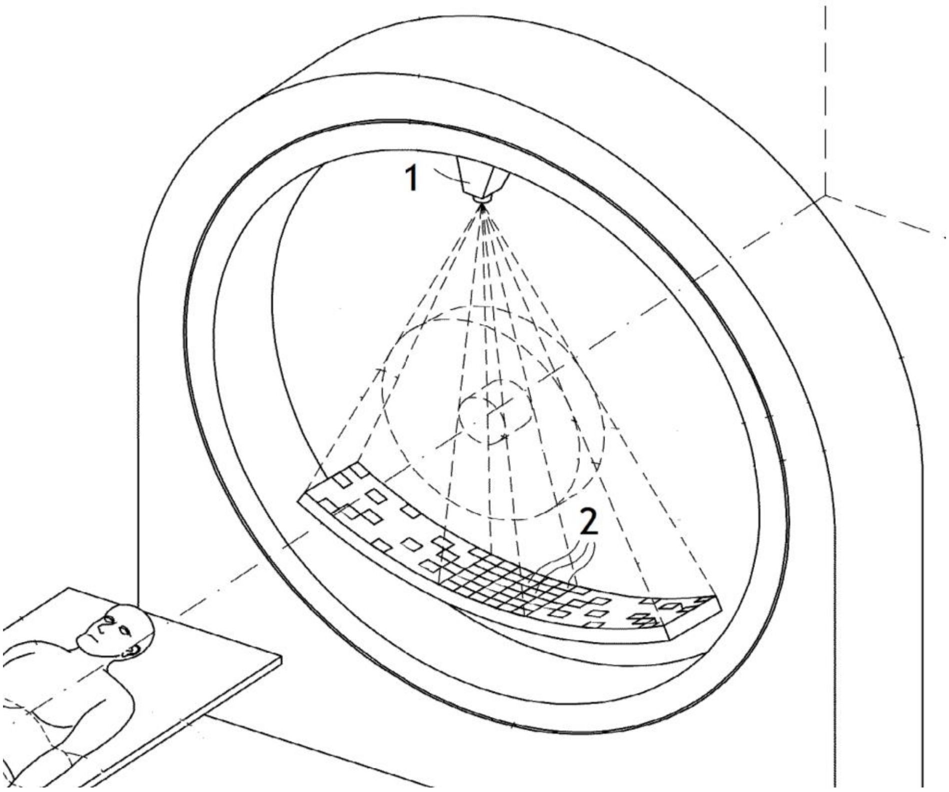


Figure 33. Schematics for a CT scanner where (1) is the source of the beam of X-rays and (2) are the sensitive radiation detectors (Dafni, Graded resolution field of view CT scanner, 2006).

A CT scan basically measures the radiodensity of the object the rays pass through; soft tissue has a lower radiodensity than hard tissue. The radiodensity is measured in Hounsfield units (HUs). Hounsfield chose a scale that affected the following four basic densities (Introduction to CT physics, 2004):

- Air = -1000
- Fat = -60 to -120
- Water = 0
- Compact bone = +1000

CT scans usually consist of 512x512 pixels per slice and 12-bits per pixel. These pixels can store values varying from -1024 to +3072, although in reality, as compact bone is roughly +1000, the values usually do not get over +1500, this could happens in case of certain metals.

In order to display these values on a computer[10] they have to be processed into images. Afterwards, in order to be able to perceive the values, two variables have been introduced:

- Window level (WL);
- Window width (WW).

The WW describes the range of Hounsfield units displayed, the WL is the Hounsfield unit value in the center of the WW. See Figure 5 for a schematic representation.

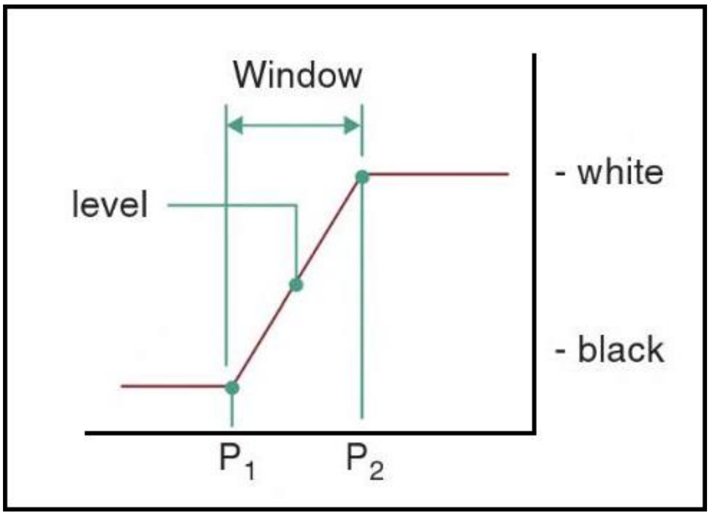


Figure 34. The purple line indicates the contrast in the displayed within the window, where window level is in the exact middle of the window and where window width is the difference between P1 and P2 (Bushberg & Boone, The Essential Physics of Medical Imaging, 2011)

The WW and WL settings greatly impact the shown data when viewing a CT scan. See Figure 6 for a visual example. The process of constructing DRR's is highly *reader dependent* because of the, by the user/reader selected, window width and window level.

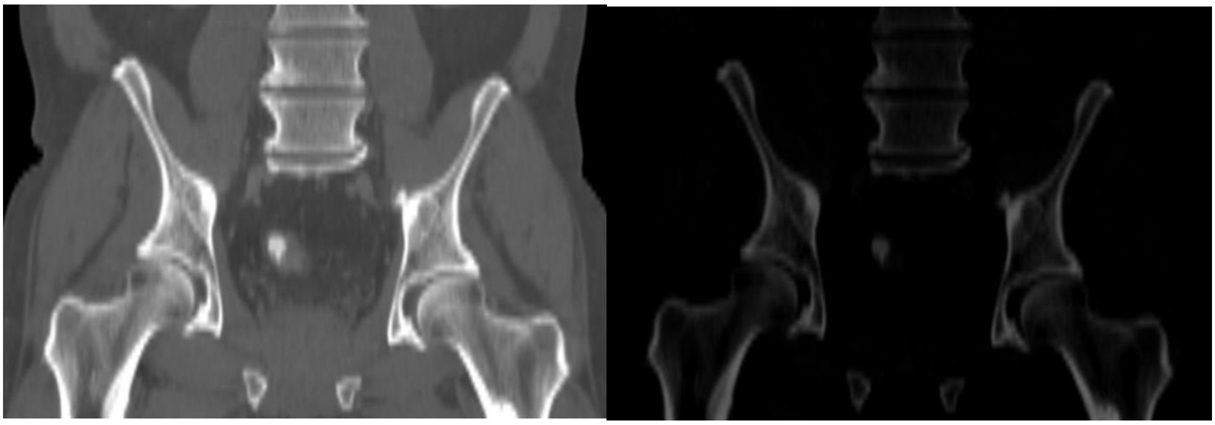


Figure 35. (left) CT slice displayed where WL = 250 and WW = 1450. (right) CT slice displayed where WL = 1800 and WW = 3500.

[10] Most modern computer screens are only capable of displaying 8-bits, so 256 shades of grey.

A digitally reconstructed radiograph, or DRR, is basically an X-ray constructed from a CT scan. This process generally works by placing all the slices of the CT scan on top of each other and ‘shooting’ rays, originating from a virtual X-ray source, through the slices onto the DRR’s canvas (Russakoff, Rohlfing, & Maurer, Fast Intensity-based 2D-3D Image Registration of Clinical Data Using Light Fields, 2003). See Figure 7 for a visual representation of the construction of a DRR.

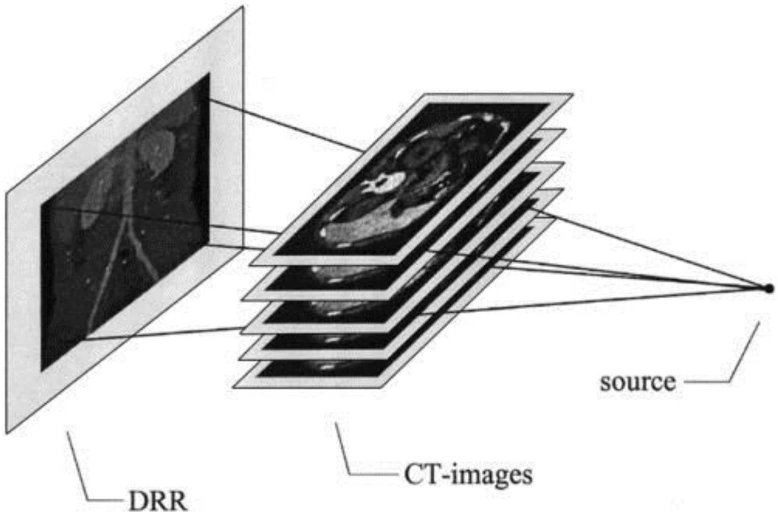


Figure 36. Visual representation of the construction of a DRR. (Bahner, Debus, Zabel, Levegrun, & van Kaick, Digitally Reconstructed Radiographs From Abdominal CT Scans as a New Tool for Radiotherapy Planning, 1999)

The value of each pixel in the DRR depends on the technique used for the projection. Several well-known techniques are:

- **Maximum intensity projection** (MIP), a technique of forming projection images by casting rays through a volume dataset and selecting the maximum pixel value along each ray (Bushberg & Boone, The Essential Physics of Medical Imaging, 2011). This technique is popular when the DRR should display bones.
- **Minimum intensity projection** (mIP), a technique of forming projection images by casting rays through a volume dataset and selecting the minimum pixel value along each ray. This technique is popular when the DRR should display soft tissue.
- **Weighted values ray casting**, techniques of forming projection images by casting rays through a volume dataset and weighing the voxel values. These weighted values usually depend on distance from the source to the canvas and distance traveled in each voxel. Several variations have been proposed to speed up this process (Alakuijala, Jäske, Sallinen, Helminen, & Laitinen, Reconstruction of Digital Radiographs by Texture Mapping, Ray Casting and Splatting, 1996).
- **Alpha blended texture mapping**, a technique that basically draws the slices on top of each other starting from the bottom slice and ending at the top slice where the alpha-value of each resulting pixel is determined by the height of the voxel it passes through. This method works very quickly, yet it only work correctly on very small datasets. (Alakuijala, Jäske, Sallinen, Helminen, & Laitinen, Reconstruction of Digital Radiographs by Texture Mapping, Ray Casting and Splatting, 1996)

A two-dimensional example of *weighted values ray casting* can be found in Figure 8 where the ray travels through four pixels. The ray only just pierces some pixels while going straight through the middle of others. The voxels barely getting pierced will have a smaller weight than the pixels getting pierced straight through the middle. It is important to note that the actual ray casting for the creation of DRR's is three-dimensional.

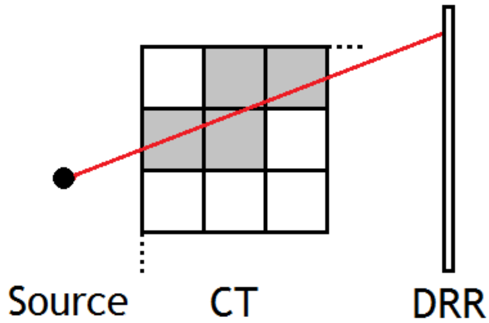


Figure 37. Visual representation of the X-ray (red line) travelling through four pixels, illustrating the weight of the values.

2.4. Motivation for the current research

As explained in paragraph 2.2, the available CT scans have to be reconstructed to images that are similar to X-ray images. Currently the UMC uses OsiriX to do so, but this requires manual actions. Manually creating DRR's has two serious disadvantages:

- When manually creating DRR's, user input makes it a *reader dependent* process, as different users/reader have different interpretations of what they see in a DRR;
- Creating DRR's is a time consuming task, every DRR takes roughly 5-10 minutes to make.

In order to successfully address the disadvantages mentioned above, the UMC will need to find out how to automatically generate readable DRR's of the hip. By doing so the UMC will not only speed up the process, but more importantly the DRR's will be reproducible. Being able to replicate results is one of the main principles of scientific research.

This research will also benefit the UMC in the future, as this will most likely not be the last time they will convert CT scans to DRR's. As the UMC does a lot a research it is important that, when they use DRR's in their research, the created DRR's will also be reproducible. These reproducible DRR's could be constructed using the program or algorithm created during this project.

3. Project definition

This chapter first describes the details about the project assignment, followed by the goals that are set for the project. Thereafter the main and sub research questions are described. After that the project deliverables are listed, followed by how the research will be approached, which research methods will be applied and which tools will be used. Finally the project scope will be described.

3.1. Project assignment

Product assignment

The assignment that needs to be realized is to automate the construction of DRR's from three-dimensional CT scans by using OsiriX or a different DRR program and to ensure that all the reconstructions are done the same way.

- In order to automate the process, the user input needs to be minimized. This means the hips and part of the lumbar need to be automatically detected. As shown in Figure 3, at least the two lowest vertebrae of the lumbar need to be visible on the DRR and of course the pelvis, hip sockets and femoral head. The localization method needs to be reproducible.

- Another important part of constructing DRR's is the location of the virtual X-ray source. In order for the results to be reproducible, the DRR's need the virtual X-ray source to be perpendicular to the center of the pelvis on the CT scans. This differs per scan as patients usually tend not to lay on the table perfectly straight.

- To ensure that the resulting DRR is reproducible, the window width (WW) and window level (WL) need to be either fixed or automatically selected using an algorithm with reproducible results. Since having an overall fixed WW and WL will likely cause readability issues on parts of the DRR's, using an algorithm to decide the WW and WL will more likely be the best choice. However, this will need to be researched.

- Finally the application will need to be robust enough that when executed on two different CT scans of the same patient, the resulting DRR will be similar. This will be proven by inputting the resulting DRR into the program discussed in (Schlösser, et al., Evolution of the ischio-iliac lordosis during natural growth and its relation with the pelvic incidence, 2014).

The software that needs to be developed for the project assignment has to be either a plugin for OsiriX, or a different suitable DRR program, or it has to be a (standalone) program. Optionally, a library or API can be created so the software can be reused for different purposes. In any case, the heart of the application should be a separate functional collection of code.

Preferably it will be a plugin for an existing DRR program, there is no need to reinvent the wheel. In case a plugin cannot be created for a suitable DRR program a good alternative would be to use MeVisLab[11] to create the program.

3.2. Project goals

The main project goal is that, by the end of this project, readable DRR's of the hip can be automatically constructed from CT scans, speeding up the process of constructing DRR's from CT scans and being able to replicate these constructions in a robust way.

3.3. Research questions

During the execution of the project assignment the following main research question will be answered:
"In what way is it possible to develop software for UMC Utrecht to automatically construct intelligible DRR's of the hip, pelvis and lower lumbar from CT scans, to facilitate reproducible scoring of osteoarthritis?"

In order to answer the main research question, the following sub research questions have been formulated:

5. What are the functional and non-functional requirements, set by the users, that the program or plugin has to meet?

6. What are the most prominent methods of constructing DRR's from CT scans?
   - How do these methods work?
   - Which one of these methods is most suitable for this project?

7. How can intelligible DRR's be constructed without user input?
   - When is a DRR intelligible enough to be accepted as proper output?

8. Which programs are capable of constructing DRR's that can be extended with a plugin?
   - Which one of these programs is most suitable for this project?

9. Which solutions have been presented for the automated localization of the hip, pelvis and lower lumbar in CT scans?
   - Which alterations need to be made to which one of these solutions for it to be acceptable for this project?

3.4. Approach

This paragraph describes the methods that will be used to answer each research question and afterwards the tools and development methods used for the project management and development of the software.

3.4.1. Research methods

The main research question will be answered by answering each of the sub research questions. The research methods and techniques used to answer the sub research questions are shown in Table 1.

[11] MeVisLab is a software framework for medical image processing and is currently used for most in-house software at the UMC.

Table 11. Research methods and techniques per sub research question

#	Research question	Question type	Research method(s)	Method and technique
1	What are the functional and non-functional requirements, set by the users, that the program or plugin has to meet?	Descriptive	Interviews	Through interviews with the clients and Koen Vincken a requirement and MoSCoW analysis will be executed.
2	What are the most prominent methods of constructing DRR's from CT scans?	Descriptive	Desk research	A list of methods from literature will be drafted. These methods will be compared based on several factors to form a short list of methods, which should give enough insight.
2.1	How do these methods work?	Descriptive	Desk research	By analyzing existing material the methods will be described.
2.2	Which one of these methods is most suitable for this project?	Comparative	Desk and field research	Through factor scoring a table will be drafted, containing several scores (e.g. speed, accuracy, ease of implementation, etc) for each method.
3	How can intelligible DRR's be constructed without user input?	Descriptive	Desk and field research	By analyzing the existing methods and weighing their relevance and applicability.
3.1	When is a DRR intelligible enough to be accepted as proper output?	Designing	Interviews	A requirement analysis will be realized through an interview with the clients. This will also result in the method of testing.
4	Which programs are capable of constructing DRR's and can be extended with a plugin?	Descriptive	Desk research and interviews	A long list of programs will be drafted by analyzing existing material and an interview with Koen Vincken.
4.1	Which one of these programs is most suitable for this project?	Comparative	Field research and group discussing	Through factor scoring a table will be drafted containing several scores (e.g. speed, accuracy, ease of implementation, etc) for each program.
5	Which solutions have been presented for the automatic localization of the hip, pelvis and lower lumbar in CT scans?	Descriptive	Desk research	By analyzing existing material a short list of relevant and automated localization methods will be drafted.
5.1	Which alterations need to be made to which one of these solutions for it to be acceptable for this project?	Descriptive, Designing	Desk and field research	By comparing the situation of the existing localization methods to this projects situation and adjusting where needed.

3.4.2. Tools and development methods

During the research and software design multiple tools and development methods will be used.

Document sharing and versioning

Documents that might be modified during the graduation period will not only be provided with the date it was last modified, but will also be provided a version number. The document versioning will be done according to (National Institute of Dental and Craniofacial Research, 2014).
The graduate will also use cloud services to share documents. This will allow his mentors to be able to check the graduates documents for progress and justness at any time.
These cloud services can also be used to keep track of documents' versioning.

Mendeley

The graduate will use Mendeley to easily keep track of the used papers and their importance. Mendeley is a free reference manager and PDF viewer, in which the user can easily browser through the papers and make notes in them (Mendeley).

Waterfall model

As development and project management method a variation on the waterfall model was chosen. The original waterfall model is a sequential design process. It progresses from requirements through design, implementation, verification and finally maintenance (Royce, 1970).
In the variation on the waterfall model, used for this project, the waterfall progresses from requirements through research, design, coding, validation and finally the finalization. A visualization of the model is shown in Figure 9.

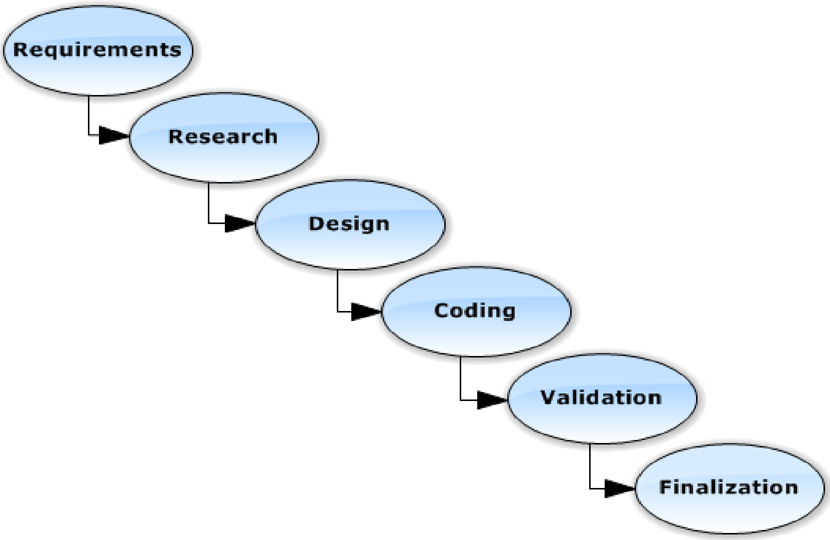


Figure 38. Variation on the waterfall model, used for this project

The reason for using this version of the waterfall model is because its phases fit the project deliverables. As first the requirements for the project are set, followed by doing the necessary research to gather information required for the next phase, design, in which the technical-function design will be created. After completing the design phase, the project can be realized in code of course followed by validating the code through testing. When all these phases have been completed, the project enters the phase “finalization”, where the thesis and end presentation are finished and turned in.

A downside of the waterfall model is its disability to go back more than one phase, e.g., if during the testing of the code one finds out that the design is flawed, it is best to start over entirely. This could be troublesome because it is unclear what will be realized by the end of the project. As the result of research question #1 (see paragraph 3.4) will be a MoSCoW analysis, the prioritization of the functionalities is not fully known at the time of writing this document. Also because some of the testing of the product will be done in collaboration with doctors from the Department of Orthopedic surgery, it will not be as easy to say which functionalities have successfully been implemented before the end.

3.4.3. Quality demands

To ensure the code for the program or plugin will meet all the requirements, tests will be designed and conducted and a report will be written containing the test cases, the test results and the raw test data. The tests the program or plugin is subjected to will be designed in phase 5, Validation. This is because at the moment of writing this document it is unclear if a program or a plugin will be developed.

During the execution of this project a MoSCoW prioritization will be drafted. MoSCoW prioritization is a technique used to quickly and easily group requirements into one of four categories (Hatton, Early Prioritisation of Goals, 2007). A provisional MoSCoW prioritization can be found in APPENDIX C. Iterations MoSCoW .

3.5. Project deliverables

During the execution of this project several different products will be delivered. These products, excluding the products required by the University of Applied Sciences Utrecht (Hogeschool Utrecht, Faculteit Natuur en Techniek, Institute for ICT, 2014) are listed below:

- A functional/technical design of the program or plugin for the automated generation of DRRs;
- A report on the current progress of the project, halfway the graduation period;
- The program or plugin for automated generation of DRR’s and it’s source code;
- Test report(s) to validate that the program or plugin works as required;
- The end presentation for ISI.

3.6. Project scope

This project is part of a research project on the parameters, PI and IIA, that describe the shape of the pelvis and lumbar, on patients with and without coxarthrosis to find the possible relation between the formation of coxarthrosis and these parameters. Dino Colo MD and Rob Brink BD lead this research.

This project is only responsible for delivering the program that enables the automated generation of readable DRR’s of the hips, pelvis and lower lumbar from CT scans.

During this project no work will be done regarding the classification of coxarthrosis or any medical research irrelevant to the completion of this project. In case there is time left in the graduation period, after achieving the main goal of this project, the graduate will contribute to a research on the automated 3D classification of coxarthrosis.

4. Project phasing and planning

This chapter describes the phases the project will go through, followed by the milestones of each phase and finally the project planning.

4.1. Phases

As the chosen project management method is a variation on the waterfall model (see paragraph 3.4.2 for more information), the phases this project will go through are:

- **Requirements**: In this phase the graduate will explore the company and the requirements will be drafted for the program or plugin.
- **Research**: In this phase all the required research will be done to be able to answer the sub research questions, to the extent in which this is possible.
- **Design**: In this phase the program or plugin will have all its functionalities designed.
- **Coding**: In this phase the program or plugin will have all its functionalities realized.
- **Validation**: In this phase the program or plugin will be tested to ensure all the requirements are realized.
- **Finalization**: In this phase the thesis will be finalized and the presentation will be created and delivered.

4.2. Milestones

The projects most important milestones have been listen in the table below.

Phase	Milestone	Description
Requirements	Thesis plan	Turn in thesis plan
	Program/plugin requirements	Draft the program or plugin requirements
Research	Literature reports	Finalize the reports resulting from the literature studies
	Choose program/framework	Advise a program or framework and continue working with the one chosen by the clients
Design	Design report	Finalize the designs and describe them in the design report
Coding	Code functionality	Finalize the source code for the functionality of the program or plugin
	Code GUI	Finalize the source code for the GUI
Validation	Test report(s)	Write test report(s) about the tests designed to validate the program or plugin to meet its requirements
Finalization	Thesis	Finalize and turn in the thesis
	End presentation	Create and prepare the end presentation for ISI and the HU

4.3. Planning

In the table below the planning for the graduation period is described. The milestones (paragraph 4.2) have been highlighted for better readability, note that the progress report has also been highlighted. For a Gantt chart of this planning see Thesis Plan - APPENDIX A. Gantt chart.

#	Description	**Start date** (mm/dd/yy)	**End date** (mm/dd/yy)	**Estimated workload** (in days)
1	**Thesis plan**	**06/02/15**	**07/13/15**	**21**
2	Concept	06/02/15	06/29/15	15
3	Feedback	06/29/15	07/06/15	n/a
4	Final version	07/06/15	07/08/15	3
	Extension of the schedule	07/08/15	07/13/15	3
5	**Program/plugin requirements**	**06/22/15**	**07/08/15**	**3**
6	Interview Dino and Rob	06/22/15	07/03/15	1
7	Draft program/plugin requirements	06/26/15	07/08/15	2
	Extension of the schedule	07/08/15	07/13/15	1
8	**Literature studies**	**07/06/15**	**08/04/15**	**24**
9	Literature study: Reconstruction of X-ray from CT	07/06/15	07/23/15	8
10	Literature study: Localization in CT scans	07/11/15	07/29/15	8
11	Literature study: Automated generation of DRR's	07/16/15	08/04/15	8
12	**Choose programs/frameworks**	**07/21/15**	**07/31/15**	**9**
13	List of capable programs/frameworks	07/21/15	07/30/15	7
14	Advise program/framework	07/30/15	07/31/15	2
15	**Design report**	**08/03/15**	**08/26/15**	**18**
16	Design activity diagram(s)	08/03/15	08/05/15	3
17	Design class diagram	08/05/15	08/07/15	3
18	Design state transition diagrams	08/07/15	08/13/15	5
19	Design GUI	08/13/15	08/18/15	4
20	Feedback	08/18/15	08/24/15	n/a
21	Finalize design	08/24/15	08/26/15	3
	Extension of the schedule	08/26/15	08/31/15	3
22	**Progress report**	**08/18/15**	**08/24/15**	**5**
23	Write progress report	08/18/15	08/24/15	5
24	**Code functionality**	**08/26/15**	**09/09/15**	**11**
25	Work out models in code	08/26/15	09/04/15	7
26	Work out remaining code	09/04/15	09/09/15	4
	Extension of the schedule	09/09/15	09/14/15	3
#	**Description**	**Start date** (mm/dd/yy)	**End date** (mm/dd/yy)	**Estimated workload** (in days)

27	**Code GUI**	**09/09/15**	**09/16/15**	**6**
28	Work out design in code	09/09/15	09/16/15	6
	Extension of the schedule	09/16/15	09/21/15	3
29	**Test report(s)**	**09/16/15**	**09/24/15**	**7**
30	Design tests	09/16/15	09/21/15	3
31	Conduct tests	09/21/15	09/23/15	2
32	Write report	09/23/15	09/24/15	2
	Extension of the schedule	09/24/15	09/28/15	2
33	**Thesis**	**08/01/15**	**10/13/15**	**27**
34	Concept #1	08/01/15	08/13/15	10
35	Feedback	08/13/15	08/19/15	n/a
36	Concept #2	08/19/15	09/29/15	10
37	Feedback	09/29/15	10/05/15	n/a
38	Turn in final version	10/05/15	10/13/15	7
39	**Finalize project**	**10/19/15**	**11/04/15**	**…**
40	Finalize program/plugin	10/19/15	…	…
41	Finalize testing	…	11/04/15	…
42	**End presentation**	**10/19/15**	**11/03/15**	**10**
43	Create PowerPoint presentation	10/19/15	10/23/15	3
44	Prepare presentation	10/21/15	11/03/15	5
45	Present project HU	11/??/15	11/??/15	1
46	Present project ISI	11/??/15	11/??/15	1

5. Project management

This chapter describes the people involved in this project, followed by the communication between the graduate and the other involved people and finally the potential risks the project might encounter.

5.1. Project organization

During the graduation period the graduate will work under supervision of Dino Colo, MD Medicine and MD PhD candidate. Dino is in charge of the research on the relation between coxarthrosis and hip-spine alignment together with Rob Brink, BD Medicine.
Dino and Rob will support the graduate mainly on the medical side of the project and will be the end users of the product.

Koen Vincken, associate professor of ISI, will guide the graduate on the technical side of the project. Koen got his Master Degree in the field of Computer Science and his PhD in the field of Medical Image Processing. He has also mentored a lot of graduates, Master students and PhD candidates.

Esther van der Stappen, associate lector and second examiner, will guide the graduate with the didactical side of his graduation project. She will mainly advise him on documentation and research strategies.

In the table below the names, contact information and role of each of the above mentioned people, and the graduate, will be listed.

Name	E-mail	Telephone	Role
	UMC Utrecht – Image Sciences Institute		
Koen Vincken	K.Vincken@umcutrecht.nl	06 48 34 52 34	Mentor
	UMC Utrecht – Department of Orthopedic surgery		
Dino Colo	D.Colo-2@umcutrecht.nl	06 53 20 66 88	Mentor
Rob Brink	R.C.Brink@students.uu.nl	06 13 75 51 93	Researcher
	HU University of Applied Sciences		
Esther van der Stappen	Esther.vanderStappen@hu.nl	06 12 72 30 62	Mentor, Second Examiner
Alexander Hustinx	alexanderhustinx@gmail.com	06 21 19 75 74	Student researcher, graduate

Beside the people mentioned above, some other medical specialists might be involved in the project. These specialists are involved in the research project mentioned in paragraph 2.2, which this project is part of. They are mainly involved through the Department of Orthopedic surgery. It particularly refers to staff members prof. Castelein, prof. Öner, prof. Weinans and prof. Kruyt all employed at the UMC. Dr. Vrtovec from Slovenia is also involved for analyzing of the scans.

5.2. Communication

During the execution of this project there will be a lot of communication between the graduate and several parties, mentioned in paragraph 5.1. The table below describes with whom the graduate will communicate on a regular basis and about what.

Name	Description	Involved parties
Weekly meeting	There will be a weekly meeting with Dino Colo, and occasionally Rob Brink, about the progress.	Graduate and Dino Colo, on occasion Rob Brink will also be involved.
Bimonthly meeting	There will be a bimonthly meeting with the mentors from the UMC about the progress and encountered problems.	Graduatue, Dino Colo and Koen Vincken, on occasion Rob Brink will also be involved.
Document revisal	In the weeks before a report is due the graduate will request of the mentors and Rob Brink to revise that report.	Mentors, incl. Rob Brink.
Advice regarding graduation	When something regarding my graduation is unclear, Esther van der Stappen can be contacted for advice. Esther will also give advice on how to improve the thesis quality.	Esther van der Stappen.
Timesheet[12]	The graduate will keep track of his working hours per day in an Excel-sheet. This sheet will also describe what the graduate has been doing each day.	Mentors.

Communication between the parties, mentioned in paragraph 5.1, will not be limited to the table above. Some communication between the parties can occur that is not expected at the time of writing this document.

5.3. Risk management

During the execution of the project some problems could be encountered. These problems, with the likelihood of them occurring, their severity and a possible measure that could be taken are listen in the table below.

Risk	Chance (%)	Severity (1-10)	Consequence	Measure
Too little information can be found about the generation of DRR's of the hips	40%	4	Too little information can be gathered from the literature study, causing the designs for the program to be flawed.	The search for information will be expanded. Not only hips will be searched but other body parts that might have similar approaches.
Too little information can be found about the automatic generation of readable DRR's	40%	3	Too little information can be gathered from the literature study, causing the designs for the program to be flawed.	Koen Vincken will be asked for advise and the graduate will design the program based on experience with computer vision.

[12] The latest version of the Timesheet can be found at: https://www.dropbox.com/s/zez80d99xd71nui/Timesheet.xlsx?dl=0

Risk	Chance (%)	Severity (1-10)	Consequence	Measure
The estimate time of some parts of the planning are inaccurate	20%	5	Deadlines could be missed and some parts of the project might be left unfinished at the end of the project.	The most important parts of the project would be prioritized and executed first.
Stakeholders conflict over the functionality of the to-be-designed program/plugin	25%	4	It becomes unclear to the graduate what exactly he needs to design.	Schedule another interview with the stakeholders to get a clear view on what the hard demands of the program should be.
Impossibility of creating a plugin	20%	4	If a plugin cannot be created, or the client does not want a plugin, more code will have to be written in order to have the same result.	Using a good and extensive framework can reduce the time spent working on most of the algorithms.
It is not possible to automatically construct DRR's with an application	40%	7	The application will not be able to automatically construct DRR's as it was intended to do.	The application can instead be used for the semi-automatic construction of DRR's, which will require as minimal user input as possible.
The CT scans are not stored as *.dcm or a different DICOM-format	60%	4	All the CT scans will not be usable unless they are first converted conform the DICOM-format.	All the files need to be converted. Which can be done by looking for a program online or by writing a simple program to convert them.

Risk	Description of the risk.
Change	The chance, expressed in percentage, of a risk occurring.
Severity	The severity of the risk where 1 is not severe and 10 is very severe.
Consequence	Description of the consequences the risk has on the project.
Measure	The measure(s) that can be taken to minimize the impact if the risk occurs.

6. References

Introduction to CT physics. (2004, July 5). Retrieved June 22, 2015, from Elsevier Health: http://web.archive.org/web/20070926231241/http://www.intl.elsevierhealth.com/e-books/pdf/940.pdf

Alakuijala, J., Jäske, U., Sallinen, S., Helminen, H., & Laitinen, J. (1996, October). Reconstruction of Digital Radiographs by Texture Mapping, Ray Casting and Splatting. *Engineering in Medicine and Biology Society, 1996. Bridging Disciplines for Biomedicine. Proceedings of the 18th Annual International Conference of the IEEE, 2*, 643-645.

Altman, R., Alarcón, G., Appelrouth, D., Bloch, D., Borenstein, D., Brandt, K., et al. (1991, May). The American College of Rheumatology criteria for the classification and reporting of osteoarthritis of the hip. *Arthritis & Rheumatism, 34*(5), 505-514.

Bahner, M., Debus, J., Zabel, A., Levegrun, S., & van Kaick, G. (1999, October). Digitally Reconstructed Radiographs From Abdominal CT Scans as a New Tool for Radiotherapy Planning. *Investigative Radiology, 34*(10), 643.

Boulay, C., Tardieu, C., Hecquet, J., Benaim, C., Mouilleseaux, B., Marty, C., et al. (2006, April). Sagittal alignment of spine and pelvis regulated by pelvic incidence: standard values and prediction of lordosis. *European Spine Journal, 15*(4), 415-422.

Bushberg, J. T., & Boone, J. M. (2011). *The Essential Physics of Medical Imaging* (3rd ed.). Philadelphia: Wolters Kluwer Health/Lippincott Williams & Wilkins.

Chan, W., Lang, P., Stevens, M., Sack, K., Majumdar, S., Stoller, D., et al. (1991). Osteoarthritis of the knee: comparison of radiography, CT, and MR imaging to assess extent and severity. *American Journal of Roentgenology, 157*(4), 799-806.

Dafni, E. (2006, May 31). *Patentnr. US 8111804 B2.*

Hatton, S. (2007). Early Prioritisation of Goals. In J.-L. Hainaut, E. A. Rundensteiner, M. Kirchberg, M. Bertolotto, Y.-P. P. Chen, S. S.-S. Cerfi, et al., *Advances in Conceptual Modeling - Foundations and Application* (Vol. 4802 2007, pp. 235-244). Auckland, New Zealand: Springer.

Hogeschool Utrecht, Faculteit Natuur en Techniek, Institute for ICT. (2014, Juli 7). *Afstudeerleidraad Instituut voor ICT - juli 2013 - Afstudeerleidraad Instituut voor ICT cursus 2014-2015.pdf.* Retrieved June 10, 2015, from Bedrijven - Hogeschool Utrecht: www.voorbedrijven.hu.nl/~/media/HU-BEDRIJVEN/docs/Praktijkbureau%20cluster%20ICT/Afstudeerleidraad%20Instituut%20voor%20ICT%20cursus%202014-2015.pdf?la=nl

ISI Utrecht. (2006, Januari 26). *Image Sciences Institute: About*. Retrieved May 25, 2015, from http://www.isi.uu.nl/About/

Jentzsch, T., Geiger, J., Bouaiacha, S., Slankamenac, K., Nguyen-Kim, T., Werner, C. M., et al. (2013, November 5). Increased pelvic incidence may lead to arthritis and sagittal orientation of the facet joints at the lower lumbar spine. *BMC Medical Imaging, 13*, 34-44.

Mendeley. (n.d.). *Overview | Mendeley*. Retrieved June 11, 2015, from www.mendeley.com: https://www.mendeley.com/features/

National Institute of Dental and Craniofacial Research. (2014, March 6). *Version Control Guidelines.* Retrieved June 17, 2015, from National Institute of Health (NIH): http://www.nidcr.nih.gov/Research/ToolsforResearchers/Toolkit/VersionControlGuidelines.htm

Royce, W. (1970, augustus). *Managing the Development of Large Software Systems.* Retrieved June 10, 2015, from http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf

Russakoff, D. B., Rohlfing, T., & Maurer, C. R. (2003, October). Fast Intensity-based 2D-3D Image Registration of Clinical Data Using Light Fields. *International Conference on Computer Vision, 1*, 416-422.

Schlösser, T., Janssen, M., Vrtovec, T., Pernuš, F., Cumhur Öner, F., Viergever, M., et al. (2014, July). Evolution of the ischio-iliac lordosis during natural growth and its relation with the pelvic incidence. *European Spine Journal, 23*(7), 1433-1441.

UMC Utrecht. (n.d.). *About us - UMC Utrecht*. Retrieved May 25, 2015, from http://www.umcutrecht.nl/en/About-us

Thesis Plan - APPENDIX A. Gantt chart

Thesis Plan - APPENDIX B. Provisional MoSCoW prioritization

Must have:

- Construction of DRR from CT;
- Bones on the DRR’s are visible enough for medical professionals;
- DRR’s are clear and intelligible enough for medical professionals, so they should not be blurry;
- Simple, stand-alone program or a plugin as proof of concept;
- The realized method has such a reproducibility that it is acceptable for users, like Dino Colo.

Should have:

- Automated construction, meaning as little user input as possible;
- Automated localization of the regions of interest (e.g., pelvis, hips, lower lumbar);
- X-ray source location will automatically be set, so the resulting DRR will be practically symmetric;
- Well-documented code, allowing ISI to easily reuse the software for similar purposes.

Could have:

- Reusable for different purposes;
- Configurable settings before execution of the program (e.g., altering X-ray source);
- Usable with, or in combination with, a GUI.

Won’t have:

- ...

APPENDIX B. Evaluation of personal functioning

While looking for a graduation project I knew I wanted the subject to be related to computer vision. After asking around I can in contact with Koen Vincken, associate professor at the Image Sciences Institute of UMC Utrecht. Who had two very intersection projects available in the field of medical imaging. Eager to start I wrote my project proposal for the University of Applied Sciences Utrecht (HU). While writing that proposal I started reading literature on the subject. After a few articles I felt somewhat insecure, my knowledge in the field of computer vision is OK, but I didn't have a lot of experience on the medical field. Having to understand all the relevant medical terms in English was also hard, at first. However, after a few days of reading papers that feeling of insecurity started to fade.

In the past year I chose to do the research semester at the HU, so I had some experience researching and writing documents about my findings. With the knowledge I attained during that semester I knew some of the things I had to pay more attention to, e.g., planning. I decided to start reading literature immediately after I had finish my project proposal in order to get ahead of schedule. By the time my project got approved and I had to start writing my thesis plan, I had done enough research to finished a big part of my theoretical framework. This allowed me to get ahead of schedule even more. Writing a good thesis plan will also save lots of time writing the actual thesis, so I tried to document everything as vividly as possible. Eventually I might have overdone it a bit, as I ended up describing a few trivial things, which is of course unnecessary. Esther's extensive feedback on my thesis plan helped making it a lot better.

While I was ahead of schedule during the start of the research I started experimenting in Java with the DICOM-files containing the slices of a CT scan. In a few days I create a simple program capable of reading these files. Afterwards, after some discussion with Koen, I had a look at MeVisLab, to see what it can do. The program is nice, but it is definitely not suitable for creating good quality DRR's. This scared me a little, as Rob and Dino require high quality DRR's for their research. I was afraid that creating the DRR's would be a lot harder than I thought at first, so I decided to just try it. I expanded my simple DICOM reader, allowing it to create DRR's. The resulting DRR's were actually quite good, so luckily simply creating a normal DRR was not as hard as MeVisLab made it out to be.

At this point I was still well ahead of schedule, I started answering some of the research questions and expanding the program further and further. Sadly, my girlfriend broke up with me and it all went downhill. After some time I was no longer ahead of schedule, I even started getting behind. Eventually I picked myself back up again and continued my work. At which point I found out that my schedule had very little room for these delays. It seemed very hard to be able to get back on track. I eventually decided to program the GUI after finishing my thesis. This allowed me to get back on track.

After further expanding my program I decided it was time to stop writing it in Java and to convert it to C++. It had been a while since I programmed in C++ so I was rusty at first. However, after a couple of days I converted all the code and included OpenCV in the project. I kept encountering new bugs and errors though, so it ended up taking quite some time to finish writing it, but in the end the performance was great. The C++ program was roughly 80x faster than the Java program.

Finally, it was time to write my thesis. I had already answered all the research questions so it was just a matter of copying them into the thesis. Of course some parts had to be rewritten to fit the document better, but that wasn't very time consuming. After writing the first concept of the thesis, Esther had lots of feedback. This feedback was very welcome, as for some parts of the document, my description wasn't clear enough. After a few days of rewriting everything looked a lot better than before, there was more structure and everything was more clear.

Overall I was happy with the results and the process that led to these results. There were however some minor inconveniences regarding the guidance. Koen Vincken was very busy most of the time and thus had little time left to provide me with feedback for the thesis. Had I received his feedback on my thesis, I would have been able to add some desired information as mentioned in his review. This is information regarding e.g. future work, more information about osteoarthritis treatment, costs, etc.

Nonetheless, I really enjoyed working on this project, as I learned a lot of new things. One of the things I was curious about was whether or not medical image processing was a topic I would like to get into. Although I'm still not 100% certain, I can say that there are some extremely interesting topics, this having been a very interesting one. However, there is one big downside to medical image processing, every patient is different. This means that there isn't one answer to a question, there are several. There isn't one method that will work for every single patient. Perhaps that's the charm that medical image processing has to so people.

APPENDIX C. Iterations MoSCoW prioritization

First iteration (25/06): After discussion with Dino Colo and Rob Brink

Priority	Functionalities/Requirements
Must have	Ability to construct DRR from CT
	In the resulting DRR the bones are visible enough for medical professionals to score OA
	The construction of DRR’s requires as little user input as possible
	The proof of concept is a standalone program
	The results are reproducible
Should have	The proof of concept is reusable for different purposes and/or expandable with additional code
	Ability to automatically place the X-ray source perpendicular to the hips
Could have	Usable in or in combination with MeVisLab
	The proof of concept has configurable settings, regarding e.g., alternating location of the X-ray source, like an offset.
Won´t have	n/a

Second iteration (2/07): After discussion with Koen Vincken and Dino Colo

Priority	Functionalities/Requirements
Must have	Ability to construct DRR from CT
	In the resulting DRR the bones are visible enough for medical professionals to score OA
	DRR’s are clear and intelligible enough for medical professionals, so they should not be blurry;
	The proof of concept is a standalone program
	The realized method has such a reproducibility that it is acceptable for users, like Dino Colo.
Should have	Automated localization of the regions of interest (e.g., pelvis, hips, lower lumbar)
	Ability to automatically place the X-ray source perpendicular to the hips
	The construction of DRR’s requires as little user input as possible
	Well-documented code, allowing ISI to easily reuse the software for similar purposes
Could have	Usable with, or in combination with, GUI
	The proof of concept has configurable settings, regarding e.g., alternating location of the X-ray source, like an offset.
	The proof of concept is reusable for different purposes and/or expandable with additional code
Won´t have	n/a

Final iteration (13/07): After revision and discussion with Koen Vincken, Dino Colo and Rob Brink

Priority	Functionalities/Requirements
Must have	Ability to construct DRR from CT
	In the resulting DRR the bones are visible enough for medical professionals to score OA
	The DRR's are of high enough quality for medical professionals to score them for OA
	The proof of concept is a standalone program
	The results are reproducible
	The results are reliable; all the DRR's have to be created using the same method(s)
Should have	The construction of DRR's requires as little user input as possible
	Ability to automatically locate the ROI
	Ability to automatically place the X-ray source perpendicular to the hips
	The code is well-documented, so its function can be easily grasped by other programmers of equal or greater experience as the graduate.
	The proof of concept is provided with a GUI
Could have	The proof of concept and library are reusable for different purposes and/or expandable with additional code
	The proof of concept has configurable settings, regarding e.g., alternating location of the X-ray source, like an offset.
	Ability to alter selected WW/WL after processing
	Resulting DRR's have their quality enhanced by trilinear interpolation, for a higher resolution
Won´t have	Automated scoring for OA