EDGE SMOOTHING IN DIGITAL HUMAN PHANTOMS IMAGES

A DISSERTATION SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF MASTER OF SCIENCE
IN THE FACULTY OF ENGINEERING AND PHYSICAL SCIENCES

2019

ZhaoYang Liu

Student ID: 10306261

School of Engineering

Contents

Word count：14332

List of Table

List of Figures

Abstract

With the increasing demand for a digital image of the human, medical imaging technology remains to be of a major significance to human medical service. But due to noise generated during image acquisition and image processing, make the image visual effects, image pretreatment process, smooth quality directly affects the subsequent processing and analysis, the traditional image smoothing algorithm and mean filtering, median filtering and Gaussian filter these methods at the same time of smoothing and denoising tend to damage the important characteristics of image edge and texture, therefore how to better keep the edge while removing noise is the first thing to consider. As noise and image edges and details are very important in high-frequency signals, it is difficult to identify them better, which leads to noise removal at the same time, image structure features are also damaged, and serious image quality loss. This paper discusses the development of an improved majority smoothing algorithm to achieve more accurate picture smoothing and reduce the impact of sharp edges of the digital mannequin. Unlike other smoothing techniques, the methods found in the article generates no new values that mean new organization. At the same time, the number of organizations per row and column does not decrease after smoothing.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institutes of learning.

Copyright

Acknowledgments

First of all, I would like to thank my supervisor, Dr. Fumie Costen, for providing me with excellent advice and guidance during the project implementation and thesis writing.

Secondly, I would like to thank my classmates and friends who have provided me with help and good advice.

Finally, I would like to thank my family for their support during the whole process.

1. Introduction

A digital image is a two-dimension analog picture made up of many pixels and is constructed using a digital circuit or a computer. For a digital image, a pixel is a crucial component which is obtained through the process of discretizing non-ending space when building a digitized image. All pixels have numeral row (which is the height) and column point coordinates (which is the width) and whole number greyscale or color values. A computer is usually used to store the pixels as a two-dimension pattern matrix image of numerals, which are typically stored and transferred through a compressed form.

The purpose of digital human phantoms is to show a human body's internal structure. They are created through magnetic fields which can then be scanned and segmented by MRI, and the data is then stored in a greyscale file. These gray integer values are a representation of different tissues human body tissues.

To improve the quality of digital images a technique called "digital image operation" on a computer is applied to detach noise, restore, improve, portion and obtain image elements. Three are major factors are contributing to the significant advancement of computer technology and digital image. First of all, computer technology has improved in the past decades. Second, it is due to the growth in the formation and development of discrete mathematic models. Third, it is due to rising demand from a wide range of industries (e.g. agriculture industry, industrial sector, environmental sector, military uses, etc.).

In image processing, smoothing operation (e.g. line smoothing, convolution, interpolation) is used to focus image that has a wide area or image that have low frequencies. The technology can also reduce image noise or control components with high frequency thus smoothing the image result. The type of processing method used depends on the image noise characteristic (e.g. salt-pepper noise), some example of smoothing methods include linear smoothing and nonlinear smoothing.

The prime focus and direction this project is to study digital human phantoms by using relevant smoothing algorithm. This smoothing algorithm is unique from other algorithms (e.g. median algorithm) because it will not produce additional greyscale values (e.g. in digital human image smoothing no new tissue is generated). The digital human phantom's gray number value denotes a distinct structure and has its particular meaning. For example, during the smoothing process if a new gray value arises this means there is an error (e.g. an image with 52 represents skin and 47 is fat. If after the image is smoothed, the number between 47 and 52 should not appear, such as 49, because it represents another body muscle tissue). Therefore, the algorithm checks that new tissues do not appear after image smoothing.

In this study, the smoothness of an image is calculated by using different smoothing parameter values. Using the obtained parameters values, the user can then select the appropriate smoothness.

The smoothing process involves scanning and smoothing of pixels by rows (width) and columns (height), this is to make sure there is no oversight of human tissue problems, e.g. fat and muscle are on a straight line. The algorithm ensures both fat and muscle tissues remain present by inserting the missing tissue after smoothing.

The smoothing algorithm used in this study is appropriate for digital human phantom smoothing. To meet the requirements of the simulation, the digital image is expanded many times before smoothing so that it does not bring or omit tissues (creating millions of pixels as the image could be expanded dozens of times before meeting the requirement of the simulation). As a result, there must be enough memory storage and the hardware device and CPU specs are high enough to run the data.

1.1 The Project's Objectives

The objective of this project is to create an algorithm program that can enlarge and smooth the images of digital human phantom without the need to produce new pixel values. Moreover, the pixels that disappear after smoothing are inserted into the original image by interpolation method to ensure that the pixels are not missing. Last, the user interface is used to provide help with users operating the system. Below are the results that can be obtained.

•Comprehend the data structures of PGM (which is different from other graphic files, e.g. JPEG, PNG, TIFF) and digital human phantoms.
•Learning Python.
•The PGM file is generated by the code and the original data is read from the PGM file.
•Smoothing 2D images with an improved majority smoothing algorithm.
•Complete smoothing rate calculation.
•Complete PGM file display platform, it can input any value (x(0:100), y(0:200), z(0:30)) to query the picture of 3d model in three directions.
•Design software to complete user interface.

1.2 Introduction to the thesis chapter

The second chapter is the introduction of background information, which is divided into three parts. The first part is about the development process of the digital human phantom, the second part is the introduction of typical medical pictures, and the third part is the introduction of PGM files.

The third chapter is the introduction of the algorithm, including image interpolation algorithm and smoothing algorithm, such as linear interpolation, bilinear interpolation, Gaussian algorithm, and average smoothing algorithm, etc. Through learning and understanding the principles of these algorithms, it is helpful to better understand the requirements of this project and complete image smoothing.

The fourth chapter is the introduction of the project method, first introduces the Python programming language, then describes the implementation steps of the improved majority smoothing algorithm, and finally analyzes the results of the smoothed image.

The fifth chapter is a summary of the project, describing the shortcomings of the improved majority algorithm, and the parts that need improvement in the future.

2. Background

The second chapter introduces the background knowledge related to the project. Part 2.1 introduces the development process of Digital Human Phantom, part 2.2 introduces three typical medical images, and part 2.3 introduces PGM files.

2.1 The Digital Human Phantom

The way to digitalize the structure of a human body is called the Digital Human Phantom. The anatomical structure of the body of a human being was displayed in three dimensions using computer language symbols, with the support of image processing technology, and computer technology. The shape, size, location and relationship and spaces between each tissue and organ are digitalized to express the digitalized information of a person’s anatomical structure.

The exploration of ionizing radiation dosimetry was started by the radiological community in the 1960s, and several models were obtained. With the skyrocketing computer technology, development of digitalized human simulation model was boosted. From the quadratic equation-based simplified model to a more complex voxel model utilizing the tangible medical pictures of a human body, the improvement was dramatic. The newest models are built on advanced numerical techniques, which includes polygon mesh technology and methods using Non-Uniform Rational B-Spline (NURBS).

The development of such technology moves beyond three dimensions and into exploring the four-dimensional space which includes the time dimension. More than a hundred and twenty digitalized simulation models are being used to calculate the simulation globally. They can be categorized into three groups–stylized phantom, voxel phantom, and deformable boundary representation (BREP) phantom using polygon mesh technology and NURBS. These models simulate human bodies of men and women, youth and children, even pregnant women. Radiation dose can then be simulated on the models to test general radiology and nuclear medication. The findings of the studies are used to generate standard adopted by the Global Commission on Radiological Protection (ICRP) [8].

2.1.1 Stylized phantom–the first generation

The Global Commission on Radiation Safety (ICRS) had adopted this first model until the 1950s. The very first artificial mannequin simulated the impact radioactive materials deposited on a worker's or a patient's body on their tissues or organs, in order to test radiation dose. In the model, organs of a person are replaced by spheres of a number of "effective radii". In a ball, radionuclides would be displayed in the center, surrounded by organs that should have various “effective absorption energies”. Improvement was made by scientists who tried to mimic a real person’s structure, even its appearance to make to more realistic. The stylized human models were close to human anatomy.

In the roughly 20 years since the 1960s, the stylized model was continuously being improved. A virtual version was created to illustrate a human body in a digital way. Monte Carlo radiation transport encryption was utilized to detect energy deposition and radiation relations in the body. Characteristics such as the position, size, and shape of the organs can be altered using equation parameters in various programming models.

The MIRD Model

At Oak Ridge National Laboratory (ORNL), the MIRD model was suggested by Fisher and Snyder, around the 1960s. The model has twenty-two organs inside and more than a hundred sub-areas, which consists of three major parts, a truncated elliptic cone representing leg and feet; an elliptic cylinder that is the arms, trunk, and bottoms; and an elliptic cone where the head and neck are. MIRD represents the first gender-neutral model of a human body used to assess radiation dosimetry. Development was done since the very first model, and several versions were proposed. They are the stylized “family” model suggested in the 1980s by Cristy and Eckerman, the "ADAM and EVA" model proposed by GSF (Germany), and the CAM model brought forward by NASA. The family version of the model is presented in figure 2.1.

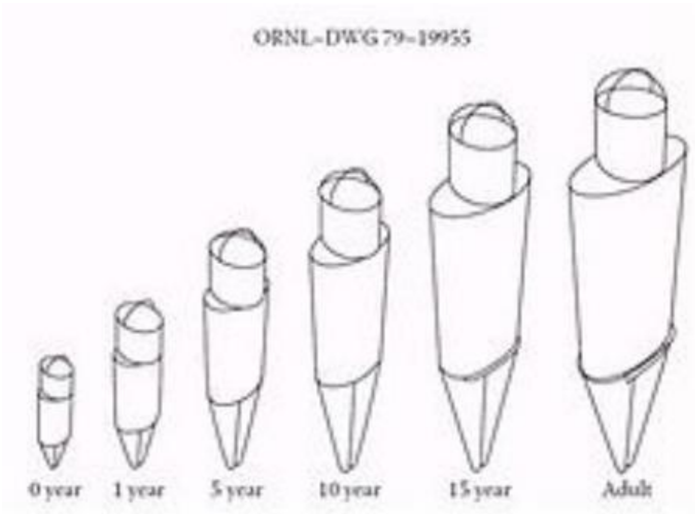


Figure 2.1: Family model series [13].

Limitations of the stylized models

Vastly applied in radiotherapy, radiation protection as well as medical imaging, and continuously improved, the stylized models has its limitations–defects that are difficult to fix. The stylized model is merely built depending on geometry – the locations of the organs inside a human body. However, as complicated as the anatomy of a human body, a general placement cannot entail all the anatomical details, causing the model to become outdated. With the fast advancement of computer technology and tomographic advancement in the 1980s, a more accurate voxel simulation human model was introduced.

2.1.2 Voxel model–the second generation

At the time, the mere representation of the geometry in a human body provided by the stylized models showed its limitation and it called for more detailed simulation. The advancement of computer technology offered a chance for further exploration. Based on MRI and computed tomography (CT) images of human bodies, new simulation models were developed. MRI and CT images capture accurate details of the gut in three dimensions, which are collected and converted to a voxel form, creating a digitalized structure of the human body in three dimensions. The internal anatomy was precisely captured by these three-dimensional images, but it costs extensive time to sort the organ segments and categories first to use in the Monte Carlo Radiation Transport calculations. Numerous voxels, which contains the codes of the organs and materials, are put together in the voxel models. Until now, there have been more than 38 voxel models developed for various purposes [24].

There are two main obstacles in the advancement of voxel models - obtaining crucial graphics and analyzing an enormous amount of data produced in these images. However, the process of CT generates ionizing radiation does in the body, which is precisely what simulated models are created to prevent in the first place. On the other hand, nuclear magnetic resonance does not produce ionizing radiation but requires a lot of time. In order to acquire data to work with, a whole human body needs to be scanned, while most of the time, the scanning process can only focus on one organ at a time. There is also a challenge in processing the data. The storage space required to store and process the images to produce an ideal voxel size is big, although new generations of computers tend to be equipped with hard disks with a big capacity [12].

The Advancement of the Voxel Model

Many different types of voxel models all have similar ways to develop a human model. To start with, CT scan, an MRI scan, or through direct anatomic photography firstly obtains the raw data. Then tissues and organs need to be separated and categorized. Thirdly, the composition of the parts and density need to be decided. At last, data is compiled into a structure in three dimensions for future reference.

In 1982, The voxel model was firstly explored simultaneously but separately by Dr.Gibbs of Vanderbilt University and Dr.Zankl of the Domestic Research enter for

Environment and Health (GSF) from Germany [25]. Particularly, Dr. Gibbs started with X-ray images, but not CT or MRI scans, and concentrated on reconstructing mannequins in order to simulate radiation doses in medical usage.

A team led by Dr. George Xu at the Rensselaer Polytechnic Institute (RPI) has used cutting-edge methods to develop a so-called VIP-man, which is an adult male simulation model, with the support of data extracted from the VHP scheme at the National Library of Medicine (NLM). The VIP-Man body model is the first high-precision radiation dosimetry voxel body model built globally, based on the image segmentation of Visible Human Project's colored cross-section photos [22]. The model is presented below.

Figure 2.2: The VIP - MAN model

2.1.3 Deformable Boundary Representation (BREP) model

The voxel models are capable to effectively reveal the anatomy of human bodies, but it is challenging for them to change the size of the organs or body structures for various physical characteristics and positions. Organ adjustments are essential in many cases, such as people with obesity, movements of the hearts and breathing motions. As a result, in order to address the different anatomic deformations in structures, researchers have looked into developing a new method to provide flexibility–the Boundary Representation (BREP) method. It includes the design of Non-Identical Rational Reference Spline (NURBS), and the polygon grid technology. Relative to the voxel models, BREP models are capable to adjust for deformation and alterations. It is easier to perform various operations on the anatomical structure in the BREP models, such as chamfering, stretching, dislocation, mixing, torsion and peeling. The most crucial advantage of BREP models is that it can be transitioned into an existent standard model and can also be transitioned into anatomical models of patients and workers in reality. It facilitates the easy calculation of personalized doses [17].

Models that apply the NURBS method

NURBS models use a series of NURBS equivalences produced from many control arguments. The structure and volume of the NURBS shell change with the matchup of the regulative points, which in general, are applied to design the human model in time-dependent four dimensions.

Polygonal mesh model

The product of numerous planes, lines, and dots used to illustrate a polyhedron in three dimensions is called a polygonal mesh model. Most of the surface of the model is a triangular mesh. There are three advantages of the polygonal mesh models. To start with, the grid surface that describes the anatomical features of a human body is easily acquired from images of human bodies. Secondly, the polygonal mesh model has excellent elasticity to entertain the adjustment in geometric configurations, which facilitates complicated anatomical simulations. Lastly, many computer software (e.g. Rhinoceros, AutoCAD, Visualization Toolkit) are able to change polygon meshes quickly to NURBS [17]. Below is an example of a model of a woman that is 9-month pregnant.

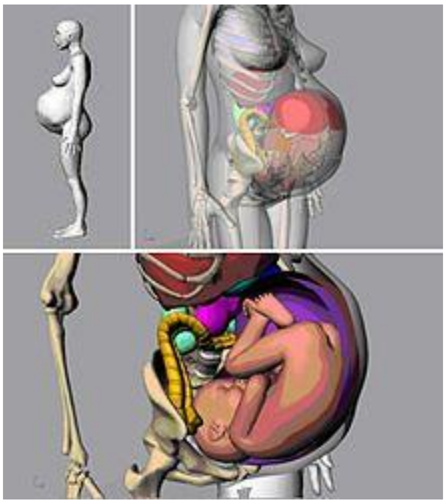


Figure 2.3: A 9-month pregnant woman model

2.1.4 Procedure of Digital Human Phantoms (DHP)

Different from many types of digital images, the process of producing DHPs together with their interior data structures is extremely hard. To complete this project, it is essential to understand how to produce DHPs. A good comprehension of this progression plays an important role to understand the internal information construction of DHPs and to lay down a foundation to produce enlarged and smoothed images in this project. There are several steps to take to produce a DHP that is good enough to simulate an accurate human structure:

1. To put in a nutshell, computed tomography or magnetic resonance imaging scans

can generate DHPs. For example, when magnetic resonance imaging performs a full scan of an entire human body, a sequence of scanned images is produced and the images are vertical to the course of the spine. Generally, the space between two consecutive MRI scans is usually about 1 mm or 2 mm.

2. Images scanned from the first step will then be divided by a medical professional or an experienced doctor, who can precisely differentiate various tissues or organs, such as muscles, lungs, and fats. Each tissue in a digital body image corresponds to a unique number. For example, in the DHPs, the number 48 is used to represent body fat and the number 45 for bones.

2.2 Types of medical images

The process of graphic representation of a human body's internal structures is called medical imaging. The procedure is adopted for clinical assessment, in order to improve efficient medical intervention and supervision. It is also used to generate clear pictures of organs to better diagnose, in case the problems are concealed by bones and skin layers. The past few years have seen the development of the acquisition technology and reconstruction algorithms, but it remains challenging to obtain images in the resolution needed. Many factors, the shortcomings of the imaging system, the imaging environment may cause blur and noise. A few kinds of medical imaging techniques are presented below.

2.2.1 X-ray computed tomography

X-ray computed tomography refers to a procedure that uses computer-generated X-rays on a part of the body to scan and to produce an image in 3 dimensions, which allows the radiologist to examine the inside of a body without necessarily opening up the body. It has become an important tool in medical imaging, complementary to ultrasound examination. Recently X-ray computed tomography is applied to precautionary medication or diagnosis for diseases, such as the whole-heart imaging for people prone to heart disorders, or colon imaging to diagnose a person's prevalence to colon cancer. Some agencies provide a full-body scan of the entire population, even though this exercise runs counter to with recommendations and formal positions of most proficient organizations in the domain, mainly due to the radiation doses [19].

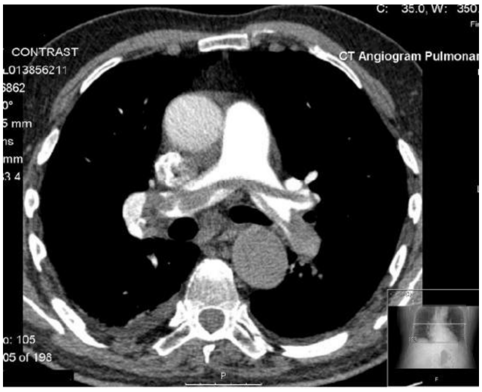


Figure 2.4: Computed tomography (CT) image

2.2.2 Positron emission tomography (PET)/Computed tomography (PET-CT)

Positron Emission Tomography (PET) and Computed Tomography (CT) are two commonly used medical instruments, which apply different scanning technologies to tumor diagnosis and treatment preparation of patients. The CT scanner works to capture a group of two-dimensional cross-sectional pictures, which are then processed in computer programs and constructed into three-dimensional pictures. Via efficient imaging, this technology largely promotes precise anatomical positioning, thus making a significant difference in the diagnostic field. Owing to the PET-CT, quick transformation, for example, has been witnessed in the imaging processes of tumor diagnosis, providing great help for therapists to precisely determine the stage of cancer, develop the most suitable radiation therapies and get comprehensive surgery preparation. As a result, PET-CT tools have now been widely used to replace traditional PET medical instruments in spite of their high costs, as the combined application of two scanning technologies makes great contribution to the effectiveness and independence of medical examination [19].

In addition to the equipment itself, the extremely high price and complex distribution of the necessary radioactive drugs required in PET imaging have also been hindering the popularization of PET-CT. These drugs not only require expensive manufacturing facilities but also need to be carefully stored due to their perishability. A picture of PET-CT equipment is presented below.

Figure 2.5: PET-CT machine

2.2.3 Magnetic resonance imaging (MRI)

Another high-tech medical examination instrument is Magnetic Resonance Imaging (MRI), which works to generate quality multi-planar or multi-directional images by using a sphere with strong magnetism. In comparison with a CT scanner, MRI may still be a more ideal option even though effective regulation of medical X-ray harm has been widely achieved. Patients undergoing MRI do not have to be worried about radioactive exposure, for which medical institutions have commonly introduced this scanning technology. Certainly, MRI also has its limitations and disadvantages. First of all, the scanning process may be very inconvenient as it requires the patient to stay in a small tight cylinder for a long time. Secondly, for patients who have been implanted with medical metal, MIR may not be a safe examination method [1].

The early MRI was referred to as NMRI (Nuclear Magnetic Resonance Imaging), which works in external magnetic domains to generate and absorb radiofrequency vigor. Hydrogen particles play an important role in the working mechanism of MIR, which is generally employed in the production of an evident indicator of radiofrequency vigor, and an antenna installed near the subject of the examination will then collect this indicator. Natural human bodies contain a large number of hydrogen particles, which, as mentioned above, play a significant role in the scanning mechanism of MRI. Therefore, MRI scanning basically works to mark where water and fat located in human bodies [1]. The figure below presents a MIR-captured image of the human brain.

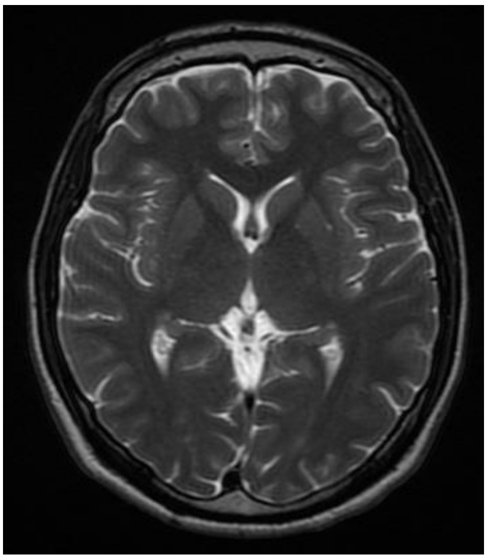


Figure 2.6: MRI image scanning

2.3 Portable Graymap Format

So far, the most commonly recognized and widely adopted grayscale imaging standard is still Portable Gray Map (PGM). PBM and PPM are other two similar imaging, which the former matching color aura and the latter matching white and black images. PGM images, tagged with ".pgm," as a suffix of their file names are generally used in white and black ultrasonic imaging. Furthermore, there are two subtypes of PGM images, which are P2 and P5, and PGM files of both subtypes are composed by two main sections, that is, the sections of header and data [9].

The information that can be extracted from the header section includes specific PGM file subtype (P2 or P5), the height of the image, the width of the image and the image's greatest grayscale value possible, among which the image's greatest grayscale value possible refers to the highest pixel grayscale cost possible in the data section of the file [4].

All the information distilled from the file header will be stored in ASCII code, as it makes sure that the highlight information can immediately capture the attention of users once a PGM folder is opened. A delimiter is used to clearly separate the four types of extracted information, and there are four kinds of legally accepted delimiters, including newline, carriage return, TAB, and space. The images below have demonstrated a standard PGM header, with all the four kinds of information clearly separated by legal delimiters.

Each pixel's grayscale worth is stored and recorded in a top-to-bottom and left-to-right manner in the data section of the file. Binary can be used to represent each pixel in P5 files. For example, if there is a P5 format image, the maximum possible gray value is 255, and the first column of the first row has a pixel value of 100. Then the image is represented by one byte per pixel, and the first row, the first column, is a binary one-byte representation of the value 100. If the possible maximum value of the gray value of this picture is 65535, then the first column of the first row is a binary two-byte representation with a value of 100 (because the representation to 65535 requires two Bytes). The interval of two sets of pixel data cannot be used to store the information continuously. When an information line of an image comes to an end, it continues to start from the following row's first column, where the content cannot be stored in two row's intervals till the full representation of all information. Messy code will appear when a P5 folder is opened due to the binary representation.

Different from P5 images, in P2 ones, it is a sequence that is used to represent each pixel. Hypothetically, the greatest grayscale value possible of a P2 image and the pixel value of the first column of its first row are 255 and 100, respectively. Three ASCII codes are then employed to represent each of this image's pixels, with the data of the first column in the first row represented by ASCII as 100. In this image, all pixels must be separated by space, with a new line added to the end of every image line. Additionally, it is also worth mentioning that when the information in a P2 image surpasses 70 bytes, its data section will be self-wrapped. Therefore, image wrapping is necessary for one when the number of stored bytes surpasses 70.

The figure big presents an example of a P2-design PGM file in Visual Studio, where, as shown in the figure, newlines are added to the end of each row of data, and thus the number of newlines mounts to 70 (the image's first row of data finishes at the end of 0, 1, 1, 1).

```
P2
508 444
255
2   1   1   1   1   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
0   1   1   1
1   255 255 254 254 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255 255
```

Figure 2.7: PGM file in P2 format

3. Introduction of image interpolation algorithm and smoothing algorithm

This chapter is the introduction of the algorithms, including image interpolation algorithms and the smoothing algorithms. such as linear interpolation, bilinear interpolation, Gaussian algorithm, and average smoothing algorithm, etc. Through learning and understanding the principles of these algorithms, it is helpful to better understand the requirements of this project and complete image smoothing.

3.1 Introduction of the image interpolation algorithm

3.1.1 Nearest-neighbor interpolation

The value of an adjacent pixel is used as the value for output in nearest-neighbor interpolation. With a given F (k), the following function can be used to represent interpolation function F's value on coordinate x:

$$F(x) = F([x + \frac{1}{2}])$$

Via the selection of the nearest sample points to x, with [x] being an integral function, the greatest value of which is equal to or smaller than x, the definition of F(x) is valid on the whole real axis. However, F(x) does not have differentiability or continuity (with jumps involved) as a function.

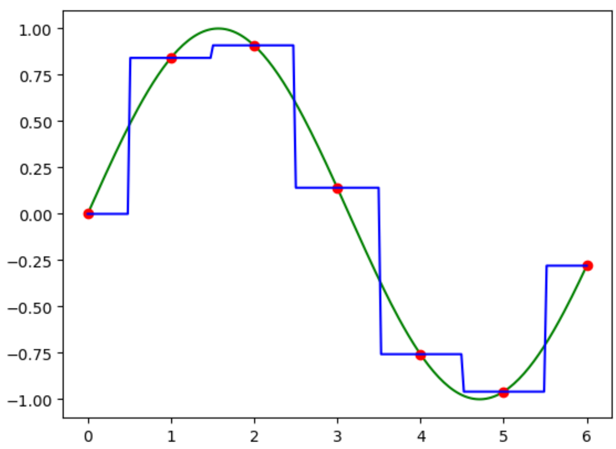


Figure 3.1: Nearest neighbor interpolation

Figure 3.1 has presented a green smooth line to represent function f with continuity (it is worth noting that it is just a sampled function with a small width, which is why it seems like a continuous function on the surface). Besides, the blue line and red line represent the interpolation function and sampling function, respectively.

A sample image with 3 pixels in width and height is listed to explain this algorithm. According to the different brightness of the pixel, its value ranges from 0 to 255, representing the darkest and the brightest, respectively.

234	38	22
67	44	12
89	65	63

234	38	22	22
67	44	12	12
89	65	63	63
89	65	63	63

Table 3.1: The left side and right side of Table 3.1 represents the original image with 3 pixels in both width and height, and the target image with 4 pixels in both width and height, respectively.

This algorithm is the most fundamental and simplest for image scaling, the effect of which is, of course, the least ideal, with the reduced image seriously distorted and serious Mosaic in the enlarged image. The way this algorithm is calculated determines its unsatisfactory effect. To be more specific, in the calculation of this algorithm, the value of the nearest pixel and the number of the floating-point are directed used when pushing the target figure to obtain the source map's coordinates, which can be seriously biased when, for instance, the value of the coordinate is 0.75 while the approximation of 1 is directly used. Therefore, images obtained via this algorithm can be largely reduced in quality. As a result, it is suggested that the calculation of the value of the target pixel be based on the four true points adjacent to the source graph's virtual point, by which a higher quality of sealing effect can be achieved [10].

3.1.2 Linear interpolation

The connection point makes the method of linear interpolation possible. Based on the assumption that there is a linear function between k and k+1, which results in the following equation:

$$f(x) = \big(1 - (x - k)\big)F(k) + (x - k)F(k + 1)$$

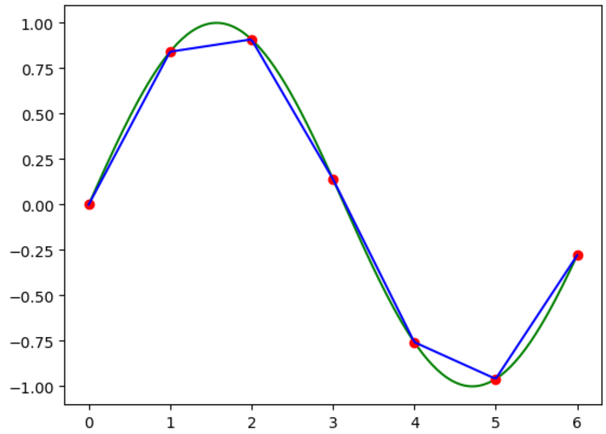


Figure 3.2: Linear interpolation

The majority of linear interpolation functions are characterized by continuity and differentiability (with no differentiability presented in sample points). Besides, with all derivatives equal to zero (except for the derivatives), linear interpolation functions are simple, yet commonly employed due to its easy and fast implementation [6].

3.1.3 Bilinear interpolation

Mathematically speaking, an interpolation function lined interpolation extension between two variables is referred to as bilinear interpolation. Bilinear interpolation is a numerical analytical algorithm widely adopted to dispense signals, process digital images and dispense videos.

Bilinear interpolation example

According to Figure 3.3, the known red points and intended interpolation green point are marked out to gain the unknown function f's significance at the point of P = (x, y), and the function f is at Q11 = (x1, y1), Q12 = (x1, y2), Q21 = (x2, y1) and Q22 = (x2, y2). Firstly, R1 and R2 are obtained via the achievement of the x-direction linear interpolation, and the subsequent achievement of y-direction linear interpolation helps obtain P. This is how to obtain f(x, y). With the coordinates of the four red points already known in the Figure, the linter interpolation is performed in the following steps: 1) inserting R1 into Q11 and Q21, and R2 into Q12 and Q22 (which is the x-direction interpolation); 2) performing the y-direction interpolation of the previously calculated R1 and R2 to obtain the desired P. These two steps can be reversed, which makes no difference to the final result. What is firstly obtained from the bilinear interpolation is the independence in the direction of interpolation [11]. These two steps can be expressed with the following two equations:

$$\mathrm{f}(\mathrm{x}, y_1) \approx \frac{x_2 - x}{x_2 - x_1}\mathrm{f}(Q_{11}) - \frac{x_1 - x}{x_2 - x_1}\mathrm{f}(Q_{21})$$
$$\mathrm{f}(\mathrm{x}, y_2) \approx \frac{x_2 - x}{x_2 - x_1}\mathrm{f}(Q_{12}) - \frac{x_1 - x}{x_2 - x_1}\mathrm{f}(Q_{22})$$

Assuming the coordinates of the red points Q11, Q12, Q21 and Q22 are (0,0), (0,1), (1,0) and (1,1), respectively, it can simplify the above equation as：

$$\mathrm{f}(\mathrm{x}, \mathrm{y}) \approx f(0,0)(1-x)(1-y) + f(1,0)x(1-y) + f(0,1)(1-x)y + f(1,1)xy,$$

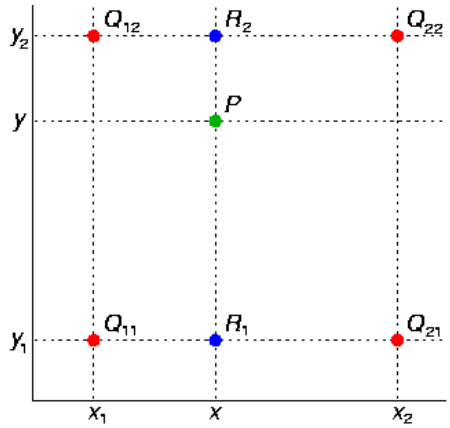


Figure 3.3: Bilinear interpolation

One of the problems of employing this algorithm is that when the four near pixels are interpolated, the slope is inconsistent, despite the consistent surface obtained in the adjacent area. Besides, image details, particularly those of enlarged images, are likely to be degraded due to bilinear interpolation's smoothing effect.

3.2 Introduction to image smoothing methods

3.2.1 The Average Filter

As a standard linear smoothing, the average filter points out an outline of the target pixels in an image, which refers to the environment of adjacent pixels (this outline comprises of about 8 pixels around the target pixel), which is then applied to the restoration of the pixels in their mean [7].

The average filter, also known as linear filtering, mainly employs the approach of average vicinity. The average filter is established based on one basic assumption that the values of all pixels in the original image is replaced by their mean, which is working with the current pixel (x, y). Then, an outline comprising of a number of adjacent pixels is chosen, endeavors for the average of all pixels in the outline, and then results to the mean of the current pixel point (x, y), as a grayscale picture at that position in g (x, y), namely $g(x, y) = 1 / m \sum m f(x, y)$ as the outline comprises the current pixel, the total number of pixels.

The average filter also has one obvious limitation that details of images cannot be well reserved via this approach. Specifically, image details are usually distorted during the process of image de-noising. As a result, the applicability of the average filter to the present project is low.

3.2.2 The Gaussian filter

The Gauss filter is a linear smooth filter for eliminating Gaussian noise. Gaussian smooth filters are very useful at suppressing noise that is different from a normal distribution. Widely used for noise denoising processing of image processing. Gaussian filtering is a weighted average for the entire picture. Each pixel value is obtained by its bias average value and other surrounding pixel values. The precise effect of Gauss filtering is to use a template (mask or convolution) to scan each pixel in the picture and utilize the offset average gray value of the surrounding pixels established by the contour instead of the center pixel of template value.

The Gauss filter is a signal filter that uses the appearance of a Gaussian to choose a weight that is used to cancel the noise and smooth the signal. It is recognized that in the future use of digital images, the main challenge remains noise, and most image processing textbooks have introduced Gaussian filters to obtain signal-to-noise ratio pictures (suggesting actual signals). Using Gaussian smoothing to eliminate image noise, then calculating the second derivative vector and using its zero crossings to determine the edge, a better image profile can be obtained.

A filter belongs to a mathematical-based concept that passes specific frequency components in a signal and greatly attenuates other frequency components. The noise is suitable for the largest part of the generation, so a smooth Gaussian filter can reduce the effects of noise. When using a perfect filter, a click will appear in the picture. When using the Gaussian filter, the system functions smoothly and ringing would be evitable.

A large amount of image noise is Gaussian noise and is therefore widely applicable to Gaussian filters. The Gauss filter is a linear smooth filter, which chooses weights based on the appearance of the Gaussian, and is very useful for denoising that obeys a common distribution, and is effective in eliminating image noise. Typically, Gaussian filters are based on a weighted average of the entire picture to eliminate noise. The value of per-pixel position is obtained by averaging the deviation of its value from other surrounding pixel values [23].

Gaussian fuzziness is a picture blur sieve, which employs Gaussian operation (that also represents normal distribution in statistics) to calculate the alteration applied to every pixel in the picture. The formula for the gaussian in 1D is

$$\mathrm{G(x)} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}$$

The expression of 2D Gaussian function is

$$\mathrm{G(x, y)} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

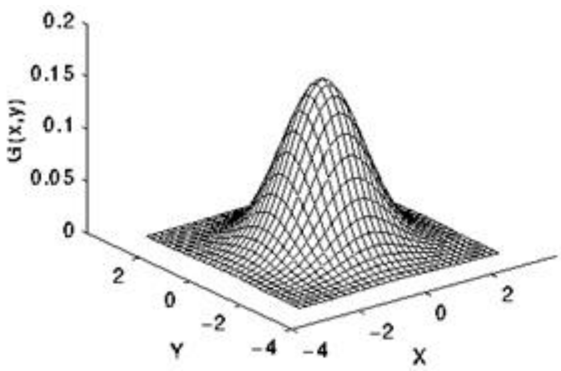


Figure 3.4: Gaussian filter layout

In principle, as shown in Figure 3.4, X represents the gap from the start of the horizontal axis, Y represents the gap from the start of the vertical axis, and sigma represents the standard deviation of the Gaussian Distribution. If used in 2D, the formula brings a surface whose shape represents a circle with a Gaussian Distribution from the center. The value in this distribution is utilized to construct the convolution matrix used in the initial picture. The contemporary value of each pixel is fixed to the offset average of the neighbors of the pixel, and the advantages of the initial pixel get the heaviest weight (with the highest value), and when they extend from the gap of the initial pixel, the surrounding pixels receive lesser weights that preserve the boundaries and edges better.

Theoretically, the Gaussian distribution has non-negative values in all domains, which requires an infinitely large convolution kernel. In fact, it just takes the value within 3 standard deviations of the mean, and just take the outside part. The following figure shows an integer value of Gaussian kernel with a standard deviation of 1.0.

$$\frac{1}{273}\begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

Figure 3.5: Gaussian filter with a standard deviation of 1.0.

Gaussian smoothing filters are very significant low-pass filters not only in the spatial domains but also in the frequency domains and have been widely utilized in the early procession of real images. There are five very significant properties in Gaussian functions as following:

(1) The rotational symmetry of a 2D Gaussian means that the filter smoothness in all directions is uniform. Generally speaking, the edge direction of a picture is uncertain, so it’s difficult to determine which direction requires more smoothing than the others before operating. Rotational symmetry indicates that the Gaussian smooth filter is not biased in either direction in subsequent image processing.

(2) Gaussian is single-valued functions which suggest that the Gaussian filter instead of the actual pixel value of the point with the weighted value of the surrounding pixel, and the weight of each surrounding pixel monotonously increases or decreases with the gap from the point to the center point. This property has a great impact since the edge belongs to one of the local features of the image. If the smoothing has a significant impact on pixels that are far from the center of the operator, the smoothing distorts the picture.

(3) The Gaussian Fourier transform spectrum is single-lobed. Thus, the Gaussian function Fourier transform can be considered to be equal to the Gaussian function itself. Actual pictures are usually affected by unwanted high-frequency signals (noise

and fine texture). Picture characteristics (eg. edge) include both low-frequency components as well as high-frequency components. The single lobes of the Gaussian function Fourier transform mean that the smoothing image is not affected by unexpected high-frequency signals, while still maintaining many expected signals to the maximum extent.

(4) The width of the Gauss filter (determining the smoothness) is evidenced by the sigma, and the association between the sigma and smoothness is very clear. The bigger the sigma, the wider the band of the Gauss filter, indicating that the smoothness is more positive. Therefore, by the parameter smoothness adjustment, it is possible to achieve a compromise between excessive blurring of the picture characteristics (excessive smoothing) and excessive unwanted abrupt amounts (under-smoothing) due to noise in the smoothed picture and fine texture.

(5) Since the Gaussian is separable, a large Gaussian filter can be effectively achieved. Convolution through a 2D Gaussian can be divided into two parts. First of all, the picture is convolved with a 1D Gaussian, and the convolved outcome is convolved with the identical 1D Gaussian perpendicular to the directions. Consequently, the amount of computation of the 2D Gaussian ascends linearly with the width of the filter template rather than squared [21].

3.2.3 The Median filter

Median filtering is a nonlinear image processing approach that determines the gray level of the center pixel based on the results of graying the pixels in the neighborhood. Median filtering is one of the normal steps in picture processing and is very useful in filtering out speckle and pulse noise. It is able to effectively protect edge invariant blur while suppressing random noise. The design of mid-value filtering is to check the sample in the input signal and determine whether it represents the signal. Implement an observation window consisting of an odd number of samples. Sort the values in the monitoring window and use the median in the monitoring window as the output. Next, discard the oldest value, collect new samples, and repeat the calculation process above.

This technique uses a given constructed 2D sliding model to process pixels in the panel based on the number of pixels and to generate a monotonically rising (or falling) 2D statistics pattern. The output of the 2D median filter is $g(x, y)= med \{f(x-k, y-l),(k, l \in W)\}$, where $f(x, y)$,$g(x, y)$ is the initial picture and processed images. W represents a 2D template, usually 3 * 3, 5 * 5 regions, or intricate appearances, for example, lines, circles, crosses, rings, etc. [15].

Median filtering is very helpful in blurred impulse noise, especially in terms of filtering noise. Meanwhile, it can avoid the blurred edges of the picture and select the right point to take the place of the value of the affected point, so the processing effect is positive. Linear filtering techniques do not have these perfect features. Additionally, the average filtering modes are intuitive and simple to achieve in hardware. Consequently, the average filtering technique has already been widely applied to the process of the digital signal.

Median filtering method: it refers to the neighborhood of a particular length or shape of a point like a window. When filtering the digital signal sequence X_j $(-\infty<j<\infty)$, first define an L-length window of odd length, $L = 2N + 1$, where N is a positive integer. At some point, the signal samples in the window are $x (iN),...,x (i),...,x (i + N)$, where $x(i)$ is the window of the center signal sample value. After the L signal sample values are sorted in ascending order, the value of the sample at i is defined as the output value of the median filter.

As shown in figure 3.2, the values of a certain part of pixels are as follows. After sorting the 9 values, they are 1, 3, 7, 12, 23, 31, 32, 43, 43. Therefore, after using median filtering, the middle pixel point will be replaced by 23.

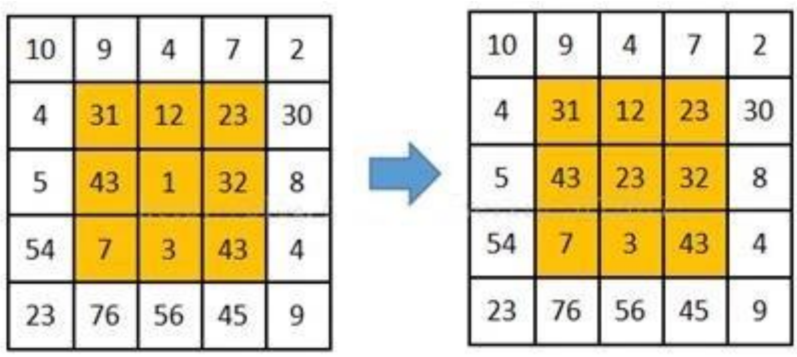


Figure 3.6：Median filter principle

In applications, as the length of the selected window increases, the amount of computation of the filter will increase rapidly. Therefore, the fast algorithm for seeking median filtering is a significant research of the median filtering theory. Three approaches are adopting in the fast algorithm of median filtering as following: (1) the correction approach for histogram data; (2) sample value binary representation logic judgment approach; (3) digital and analog selection network approach.

The study of mid-value filtering in theory also concentrates on statistical characterization and depiction of the root sequence. When a signal sequence is repeatedly processed by a median filter of an exact window length, it converges to a sequence that does not change. This is the so-called root sequence of the median filter.

The root sequence belongs to a significant concept for describing the mid-value filtering characteristics. Studying the structure of the root sequence is helpful to determine which components of the initial signal sequence can be maintained by mid-value filtering and which components will be inhibited. This deals with a significant basis for how to determine the window length of the mid-value filter. A mid-value filter chip implemented in VLSI is available for real-time processing [20].

4. Research Method

This chapter is the introduction of project methods. 4.1 introduces the Python programming language, 4.2 introduces the implementation steps of most improved smoothing algorithms, and 4.3 analyzes the results of smoothing images.

4.1 Programming Language Python

Python belongs to a programming language， which is high-level interpretive, compulsive, and object-oriented scripting. The design for Python is easy to read, it uses English keywords more frequently than other languages, and some punctuation marks in other languages, it has a unique distinctive grammatical structure than others. Python is an interpretive writing semantics and network extension that can be applied on the Internet, scientific computing, education, AI, education, desktop interface advancement, software creation, etc. [5].

Python's design style is simple and clear, making Python a versatile language easy to read, retain, and popular with many operators. Python uses a very strict syntax that makes bad programming habits (for example, the next line of an if statement not indented to the right) impossible to compile. The most significant of these is the indentation rules. One distinction from most programming languages (such as C++) (the boundary of a unit) is that it is completely determined by the position of the first character of each line in the line (C++ uses some curly braces {} to describe its personality status. However, this has caused controversy. Since the advent of C Language, the grammatical meaning has been separated from the arrangement of characters and is regarded as an advancement in programming terminology.

Nevertheless, Python ensures better enforcement of computer operator indentation (including if, for, etc., all modules need to be used, so that the program is clearer and more beautiful.

For scientific computing, MATLAB is the initial thing needs to be mentioned. Nevertheless, except for some highly versatile MATLAB toolboxes that cannot be substituted, the majority of the common features of MATLAB can be implemented in Python. Compared to MATLAB, the merits of scientific computing with Python are as follows:

First, MATLAB is a business mathematics software that is hard to afford. Python is still freely available and many libraries provide Python's calling interface. Operators can install Python and the majority of its extensions for free.

Second, Python is a programming language that is easy to learn. Users can code that is easy to understand, while you can make extensive changes to the algorithmic details of scientific computing.

Lastly, MATLAB concentrates on scientific computing and engineering. Nevertheless, even in the computing field, there exist various needs for network communication, file arrangement, and interface design. For another, Python has a rich extension library that allows programmers to get access to performing various high-level tasks easily, and users can utilize Python to complete all the instructions required for entire applications.

This project requires downloading Anaconda which is available software that can be utilized to install various forms of collections and their dependencies on similar machines and then change between surroundings. Anaconda contains Python Conda and a set of existed toolboxes, for example, pandas and other toolkits.

The development environment used in this project is PyCharm, and its main functions include:
Code Assist, with a full-featured editor, including code completion, code snippets, support for code closure and splitting Windows, helps operators implement coding quickly and easily.

Project code turns to this IDE to help developers wrap specific files in real-time to different files, from approaches to their declarations or usage, and even to class hierarchies. If the operators learn to utilize the shortcuts it provides, it will be faster.
Code analysis, operators are able to utilize their code alignment, highlighting errors, smart recognition, and one-click code to easily achieve recommendations and make the code more clearly.

Python refactoring allows operators to quickly rename, extract approaches/superclasses, input fields/variables/constants, move projects and push/rewind refactoring.

4.2 The execution process of modified majority smoothing algorithm

Section 4.2 elaborates on how to execute the whole program. With all sorts of significant details and processes, section 4.2.1 gives a thorough description of the procedure of learning and displaying information from PGM files, creating PGM files, and extending the data range. In terms of section 4.2.2, it centers on the key algorithm of two-dimensional digital picture smoothing, including its improved methods and developments. The section also elaborates on the idea of the smoothing system and the method to use it in Python semantics.

4.2.1 Preliminary preparatory work

At the primary part of the program, additional manipulation on PGM files appears to be exceedingly important. Ultimately, this lays the foundation for smoothing. The preliminary operational procedure is comprised of three essential parts.

• Access and present data from PGM files

• Create artificial PGM files

• Enhance picture resolution

Before smoothing, the issue of accessing documents from a PGM file should be solved. In consideration of the need for simulation, it is requisite to expand the PGM picture resolution prior to smoothing. In this way, in contrast with the original dimension, the smoothed features, primarily the borders of the studied picture, can exert better effects. Preprocessing must be done before developing smoothing algorithms. Only after the completion of preprocessing can it be available to test the results of smoothing algorithms. Thus, learning to create an artificial PGM file is crucial, because a smoothed picture is the basis for the judgment of how well the picture is smoothed.

On the purpose of ensuring that raw data can be read and saved in Python's PGM files, Numpy is needed for the program. As a Python library and an extension of Python, Numpy works as a multi-source instrument statistical computing for and is used to accumulate and process large numbers of matrices. When compared with Python's own nested directory structure (preferred in terms of matrices description), Numpy performs much better and more efficiently. The following four parts constitute the Numpy library. The first part is about n-dimensional array processing; The second is libraries with increasingly sophisticated functions; The third is a tool bag that contains C/ C++ and Fortran encryption; The fourth includes real-world linear algebra, the calculation of Fourier transformation as well as the function of generating a random number. Numpy and Scipy, the programming platform for sparse matrix, are comparatively easier for operation.

NumPy (Numeric Python) has some progressive tools for numerical coding, such as improved arithmetic libraries, vector processing, and matrix data types. These are designed for strict digital handling.

NumPy provides ndarray, a transformation of the n-dimensional group, to define groups of "objects" with a lot in common in nature. ndarray differ essentially with python native lists (see the following figure):

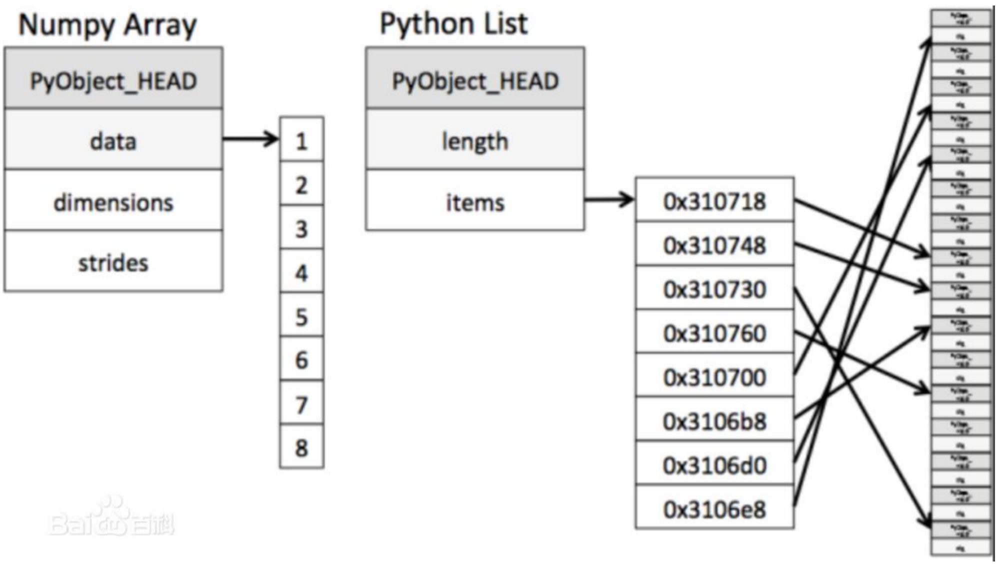


Figure 4.1: Differences between ndarray and native python lists

The figure demonstrates that the elements of an array operate more rapidly in ndarray owing to the continuity that the data address and data show when storing data in ndarray.

The reason is that all elements of ndarray belong to the uniform type while those of the Python list is at random. This difference determines that ndarray is able to store incoherent elements, whereas Python native list can only address to the next element.

With regard to general performance, the element causes Numpy's ndarray to be less extraordinary than Python native list; however, as for scientific calculating, Numpy's ndarray can store numerous looping statements and use much easier code than the Python list.

2.3 above mentions that two key sections form the fundamental content in a PGM layout file: a data part and the document header part. Besides, the document title area is distributed to four elements with the magic, comment, the picture size, as well as extreme value included. These four parts are independent of each other.

It should draw attention that there is no need to access and keep all information in a PGM picture. The comment section is basically not designed for supporting smoothing and the generation of original pictures, and such being the case, this section is supposed to be given up during data accessing.

The code in figure 4.2 is a Python implementation for reading and storing data to the PGM files.

```
#Read and store data from PGM images
import numpy as np
from skimage import morphology
import cv2
def readpgm(path):
    with open(path) as f:
        lines = f.readlines()

    for l in list(lines):
        if l[0] == '#':
            lines.remove(l)
    assert lines[0].strip() == 'P2'
    data = []
    for line in lines[1:]:
        data.extend([int(c) for c in line.split()])
    f.close()

    return (np.array(data[3:]),(data[1],data[0]),data[2])
```

Figure 4.2 shows the code that operates Python to access and store data into PGM files.

4.2.2 Generate an artificial PGM image

The first task is to generate the 3d model before generating the cross-section in three directions. This is displayed in figure 4.1. The following formula helps to understand how to generate 3d models:

$$V(i,j,k) = 4 \; \left|\mathrm{i} - \frac{ni}{2}\right| * \left|\mathrm{j} - \frac{nj}{2}\right| \le ni * nj * \frac{(0.6 * (1 - 2|k - \frac{nk}{2}|))^2}{4}$$

$$V(i,j,k) = 3 \; \left|\mathrm{i} - \frac{ni}{2}\right| * \left|\mathrm{j} - \frac{nj}{2}\right| \le ni * nj * \frac{(0.5)^2}{4}$$

$$V(i,j,k) = 2 \; \left|\mathrm{i} - \frac{ni}{2}\right| * \left|\mathrm{j} - \frac{nj}{2}\right| \le ni * nj * \frac{(0.4)^2}{4}$$

$$V(i,j,k) = 1 \; \left|\mathrm{i} - \frac{ni}{2}\right| * \left|\mathrm{j} - \frac{nj}{2}\right| \le ni * nj * \frac{(0.3)^2}{4}$$

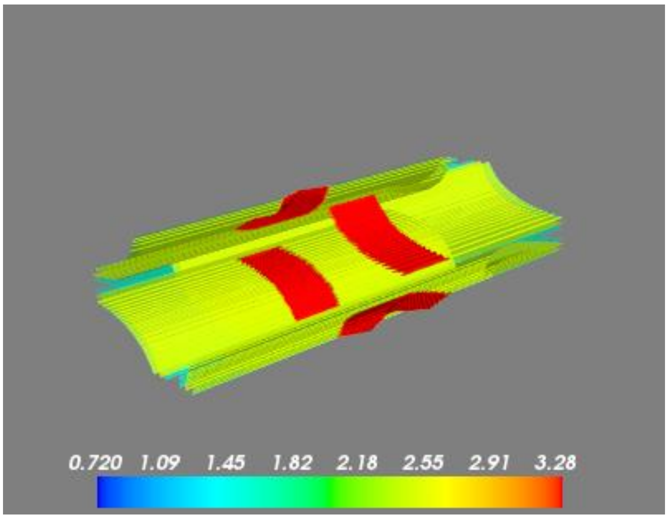
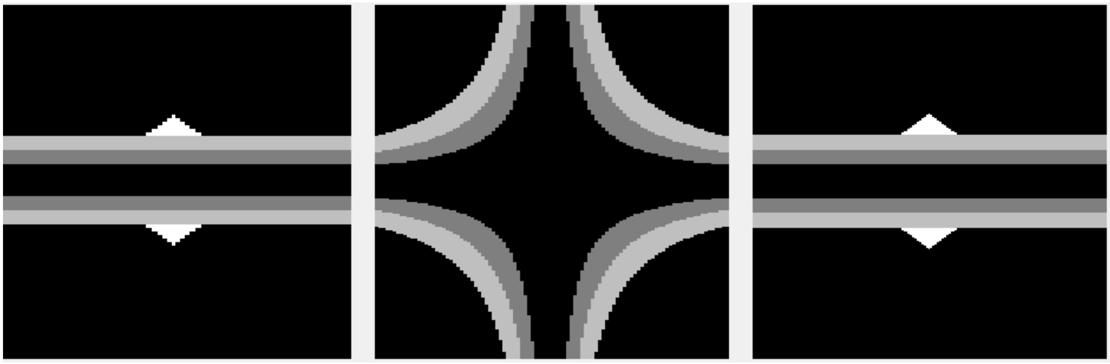


Figure 4.3: Generate 3d model diagram

The next step is to create a user interface diagram to facilitate the reading of cross-sections in three directions. The effect diagram is shown as follows:



X Y Z

Figure 4.4: The result of extracting pictures in three sides

4.2.3 Algorithm for enlarging the picture resolution

With the expansion of the picture, there exists an increase in the number of pixels and resolution. Prior to the usage of the smoothing algorithm, it is imperative to magnify the picture resolution because this enables the user to detect the picture's details better. The enlarged picture plays a pivotal role in perceiving the picture construction, bone configuration, organ edge as well as the form or edge of small foreign body objects. The larger scale of the picture makes it easier to observe the picture in details.

At present, many fields have witnessed the notable value of picture magnification, such as medicine, where picture magnification is conducive for doctors in seeking more useful information in pictures. Also, doubling the size of the picture means the enlargement of the space in the pixel without generating new data in the enlarged area, which is crucial for pre-processing the picture. As the project requires, the adjustment of the picture size can be made accordingly. The process of smoothing is beneficial for enhancing the visual quality of enlarged pictures. Furthermore, in the matter of smoothing effect, the expanded picture has a better smoothing effect than the initial one. Table 4.1 presents the process of enlarging images.

In table 4.1, the image on the left has a size of 2 x 2. and there are four different pixel values. The size of the image on the right is 4 * 4, and the height and width of the image on the left are both doubled. The size of the image on the right is 4*4. Compared with the image on the left, the height and width are both enlarged by twice, which indicates that the image has been enlarged by 2. However, from the result of image enlargement, there are still only four types of pixel values, and no new pixels are generated, so the image looks the same as the original image after zooming in, but the size has been increased.

A	B
C	D

2*2 (original figure)

A	A	B	B
A	A	B	B
C	C	D	D
C	C	D	D

2 times bigger(4*4)

Table 4.1: Principle of image enlargement resolution

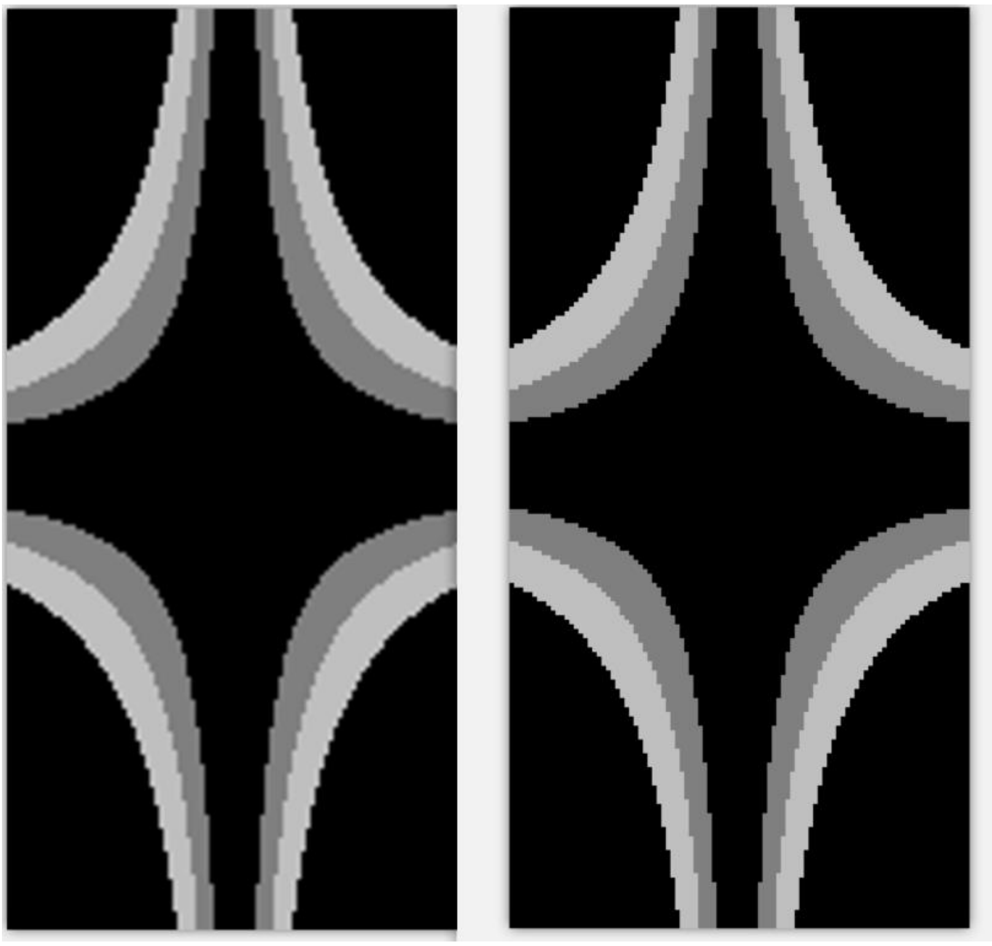


Figure 4.5: The initial picture is on the left, and the picture that has been magnified by three times is on the right.

In figure 4.5, It is important to note that when the size of the image increases, the image only changes the size, not shape. However, in comparison with the original one, the enlarged image has a jagged feature, and as a result, the smoothing algorithm is required for the expanded picture.

In short, increasing the resolution of an image makes it bigger. Since no new pixels are generated in the process of image enlargement, the human organs and tissues in the image are not changed. However, the size of the image after enlargement may cause sharp edges and other problems, which will be solved by the smoothing algorithm described below.

4.3 Improved Majority smoothing algorithm

For pictures with the enlarged resolution, smoothing arithmetic is in need for minimizing the influence of its sharp edges. The project uses the improved majority smoothing algorithm to process the sharp edges of the enlarged image by using a filter (contour or overlay) to scan the focus pixel and the additional adjacent pixels to smooth images. Generally speaking, the principle is simple, by calculating the frequency of each pixel and then replacing the number of focus pixels with the highest frequency.

If only the majority smoothing method is used to process the picture, the result is not perfect. Figure 4.6 shows that the image is doubled and then processed using a 3×3 filter. The results are shown below.

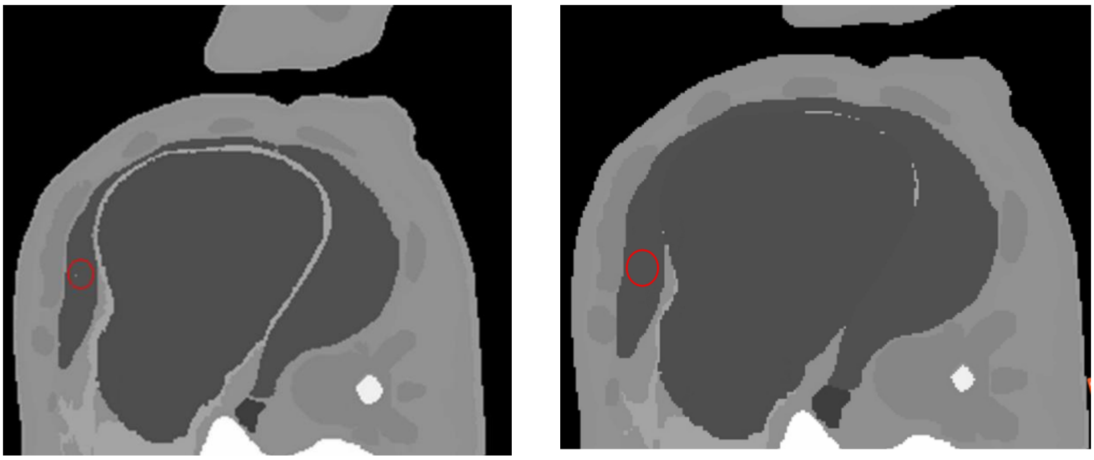


Figure 4.6: The initial picture and the smoothed picture.

Thus, the majority smoothing algorithm can lead to a deviation of the shape of the enlarged picture, which brings difficulties to medical staff in diagnosing patients. In addition, with some tiny details lost (see the red circle), the diagnosis of diseases can be affected. Therefore, for the sake of tackling the algorithm's disadvantages, the project proposes an improved majority smoothing algorithm which starts from Canny edge detection for pictures.

4.3.1 The introduction of Canny edge detection algorithm

In 1986, John F. Canny invented the Canny edge function, a system that implements a multi-level algorithm to recognize edges in pictures. Overall, the goal of recognizing edges is to greatly reduce the information dimension of the picture without leaving out its original features. There are all kinds of systems for edge recognition, although Canny system is so well-established with a long history that it is regarded as a norm for edge recognition and still has wide application in research.

Canny aims at perfecting the edge recognition system to be optimum. The value of generating optimal edge recognition lies in:

(1) Perfect identification - this algorithm can maximize the amount of the actual edge that can be detected. Also, it is useful in minimizing the likelihood of missing and mistaking real edges as well as the frequency of detecting non-edges improperly.

(2) Accurate locating standards - identify the edge points of the noise, or there is only a minimum deviation of the detected edge from the real edge under the influence of noise.

(3) The one-to-one correspondence between identified points and edge positions -- edge points should be associated with actual edge points. In order to realize these needs, Canny adopts a specific exponential function to achieve optimal detection, which is very similar to the first derivative of Gaussian function [16].

There can be five steps for the Canny edge recognition system:

1. Smooth the picture and eliminate noise by using Gaussian filtering.
2. Figure out the picture intensity gradient.
3. Remove the improper edge detection (which itself should not be detected, but is detected) by using non-maximum suppression technique.
4. Detect the questionable edges by applying the method of double thresholding.
5. Keep track of the edges by means of hysteresis thresholding.

The first step: Smoothing (devoicing)

No edge detection algorithm is able to process original data correctly; as a result, this step requires a Gaussian mask to convolve the unprocessed images, which results in a slightly blurred picture in comparison with the original one. In this way, the noise of a single-pixel almost makes no differences on an picture that has been smoothed by Gaussian filtering.

The second step: Finding the intensity gradient

Basically speaking, Canny system is to detect the location where the grayscale intensity of pictures changes most strongly. Fiercest changes, in other words, refer to the gradient direction. Sobel operator is a calculation for convolution and its encapsulated function in OpenCV helps work out the nth derivative of each pixel. It can assist the acquisition of the slope of each pixel in the smoothed picture. Here is the instruction. Firstly, use the following kernel functions to figure out G_X gradients in x directions and G_Y gradients in y directions. K_{GX} = [-1 0 1; -2 0 2; -1 0 1], K_{GY} = {-1 -2 -1; 0 0 0; 1 2 1}, secondly, obtain the pixel gradient amplitude one by one.

In some situations, for the simplicity of calculation, infinite norms of G_X and G_Y take the place of the two norms. After the replacement of G to replace each spot in smoothed pictures, the result is as the picture below. The figure below reveals that at the position with drastic change (the border), there is a larger measured value of gradient G with the corresponding white color. However, due to the border's characteristic of being thick, their actual locations are hard to pinpoint. To achieve accurate calibration of borders (see the section about non-maximum suppression), it is also a must to maintain the gradient direction. In other words, both the information of the intensity gradient and that of the gradient direction needs to be stored at this step.

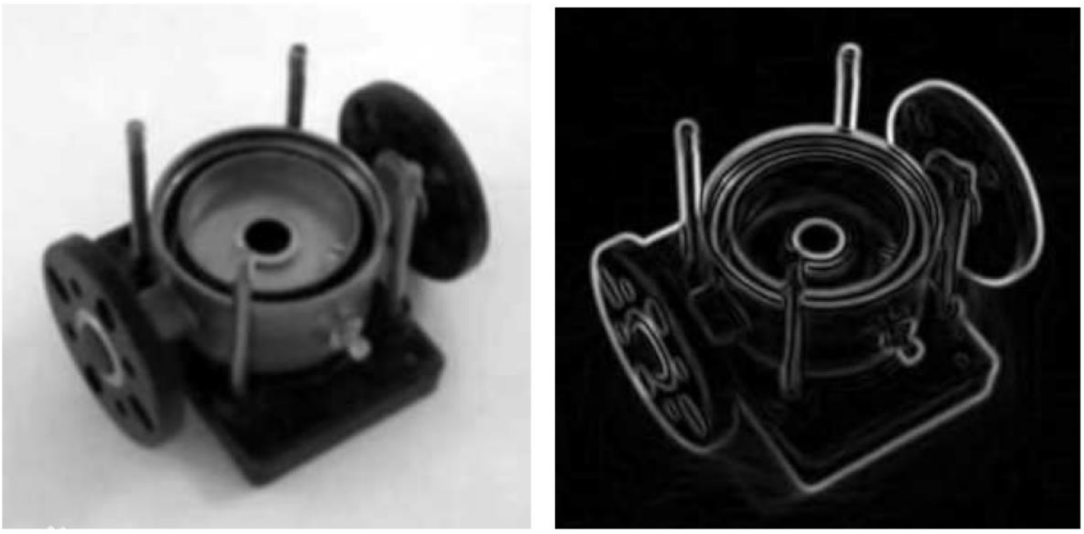


Figure 4.7: A display of the intensity gradient.

The third step: Non-maximum inhibition.

In the third step, the aim is to improve the quality of blurred borders， the highest value of the gradient strength at every pixel is conserved, and other values are removed. By examining each pixel, it is possible to:

i) Predict the gradient direction and pixel value.

ii) Compare the pixel's gradient strength and direction (e.g. if the gradient direction is positive or negative).

iii) The pixel gradient intensity with the biggest value is kept, if the value is not the largest then it is deleted and the value is set to 0.

Figure 4.8 below provides an explanation of this concept which will help with the understanding of this step.

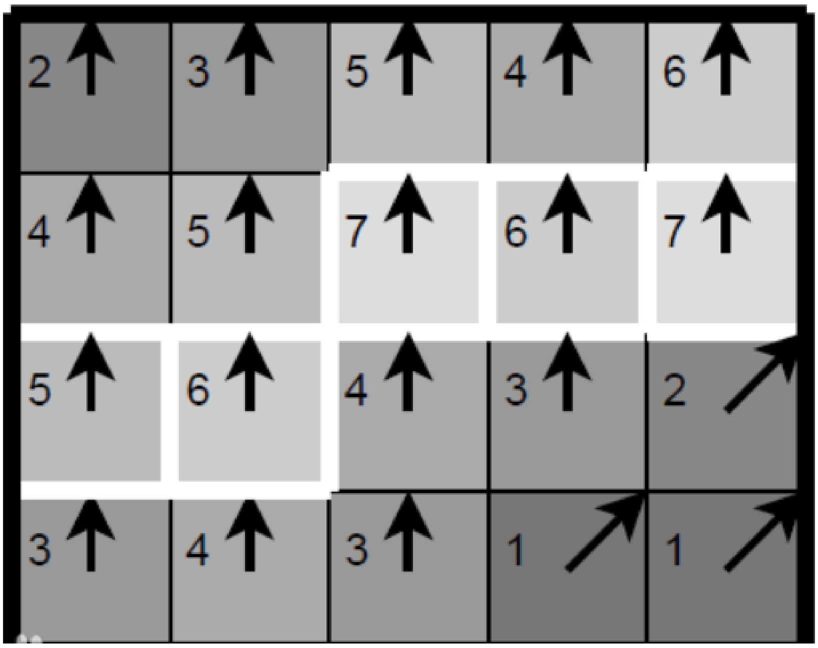


Figure 4.8: A simple example of the Non-maximum suppression principle.

The Pixel's gradient strength is defined by the numbers in Figure 4.8, and the gradient direction is represented by the arrows. For example, the third pixel on the second row shows that the gradient strength is 7 and it is compared with the pixels above and below (from looking at the arrow). The value is kept because the intensity is the largest in the figure. Through studying the figure, it can be observed that the arrow direction at the boundaries is always pointing in the way at right angles to the boundary (the thinnest line along the boundary).

The fourth step: Double Thresholding

After non-maximum suppression, noise points still exist thus "double threshold" technique is used by the Canny algorithm. The method sets a threshold for both upper and lower bounds. The bounds are typically set by a human in the OpenCV. If the image pixel is bigger than the upper threshold then the pixel reckoned as a boundary ("strong edge"). In addition, if the pixel is smaller than the lower threshold then it is not a boundary, and these two will need no further processing as they are reckoned as weak edges. Below is an image after double threshold processing. In figure 4.9, the white line on the right image represents a strong boundary, and the gray line on the left image represents a weak boundary.

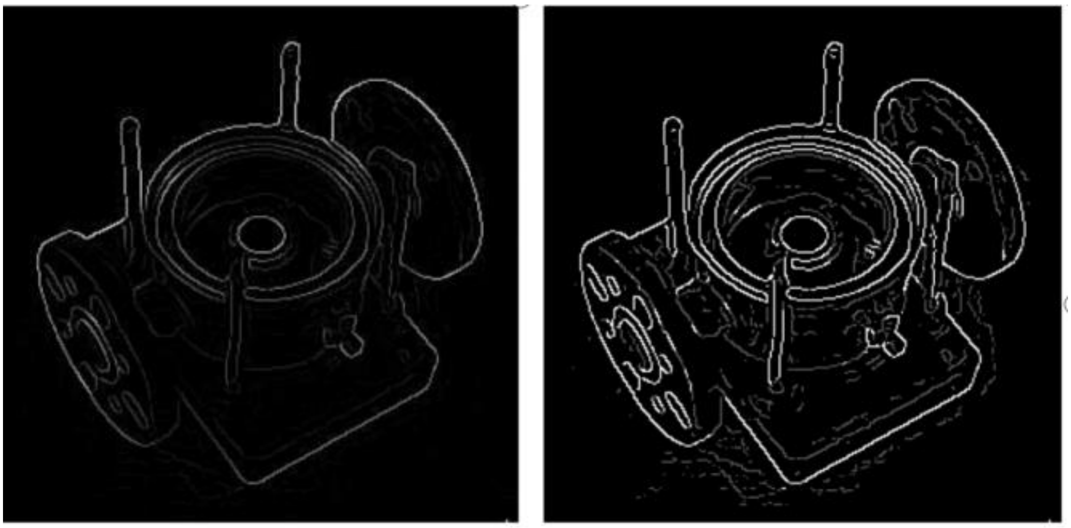


Figure 4.9: Left image-edges after non-maximum suppression. Right-image after double thresholding is processed.

The fifth step: Lagging boundary tracking.

At this point, a strong edge can be considered a real edge. Weak edges may be real edges, or they may be caused by noise or color changes. In order to get accurate results, the weak edge points caused by the latter should be removed. It is generally considered that the weak edge points caused by real edges are connected with the strong edge points, while the weak edge points caused by noise are not. The so-called lag boundary tracking algorithm checks the 8 connected domain pixels of a weak edge point. As long as there are strong edge points, the weak edge point is considered to be the real edge. This algorithm searches all connected weak edges. If any point of a connected weak edge is connected with a strong edge point, the weak edge is retained, otherwise, the weak edge is suppressed [14]. The image below shows the effect of Canny edge detection of this project.

Figure 4.10: DHP image result processed using Canny edge detection.

Before the smoothing operation, to prevent the small edge from smoothing out a suitable threshold value is chosen. In this stage, the method of the connected component is applied to make note of the edge images that are binarized. After this step, images that have an area below 10*n (n is the magnification number) are removed, the effect is shown below.

Figure 4.11: The left side is the original image, and the right side is the processed result

4.3.2 The principles of Connected Component labeling

When discussing “connected component,” it is usually referring to the image region containing foreground pixels that have the same pixel value and the adjacent position. The finding and marking of these connected areas are referred as connected component analysis.

Connected region analysis is a basic method that is typically used in CVPR and the processing and analysis of the image. For instance, it is used for segmenting and extracting character in OCR recognition (e.g. recognition for car license plate, movie substitutes, pure text, numbers), moving objects in visual tracking (e.g. intrusion detection of burglars, tracking of cars, automated car that use vision-based detection), medical image (e.g. finding objects in the area of interest) and many more. In general, the method can be used in situations whereby foreground targets require extraction for subsequent processing, and these are usually in the form of binarized images [18].

The connected region can be found in an image when the connected region is comprised of adjacent pixels with value. Unique labels are given to these connected regions found so they can be differentiated from other connected regions. The most frequently used two algorithms, two-pass method and seed-filling method (basic algorithm and improved algorithm), for analyzing connected regions are both applied in this study, this project chooses the former method.

Two-pass Scanning

During two-pass scanning, the images are scanned two times so that all the connected areas can be found and marked. In the first scan, each pixel position is assigned with a label. During scanning, pixels in the same connected region may be assigned to one or more labels. These labels having different values but are in the same connected area are combined, this is to document the equality relationship between the labels. In the second scan, pixels that are noted with equal labels of equality are grouped into a connected region, which is then appointed the same label (this is often the smallest number in equal labels).

Finally, find the points with white pixel value, that is, each edge position, and conduct mode filtering on the white points. The obtained smooth results are as follows:

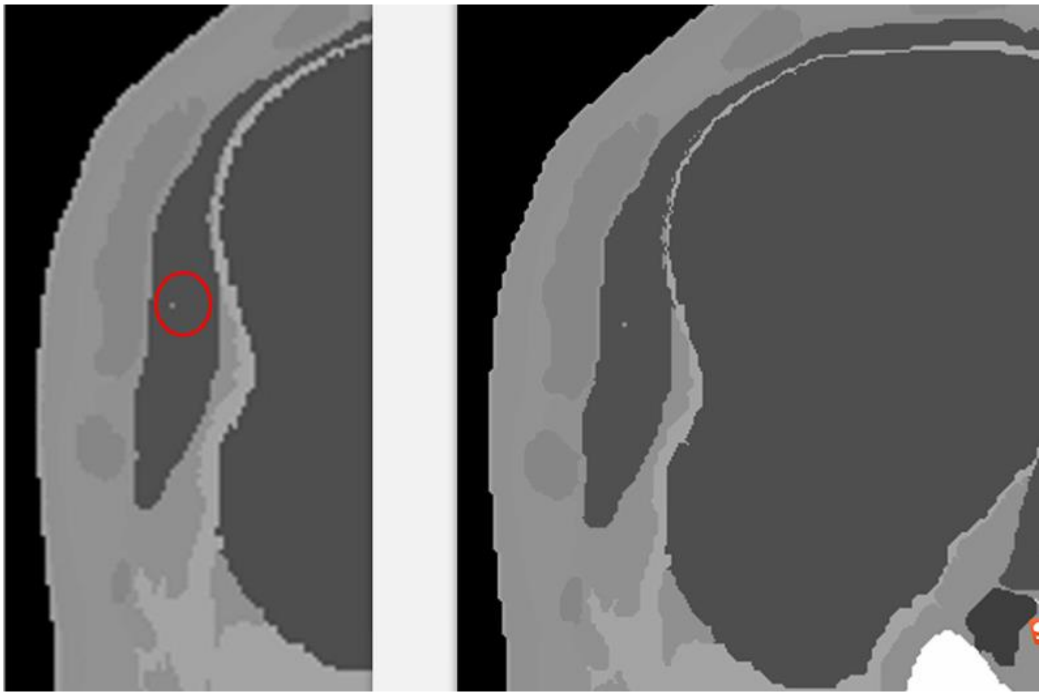


Figure 4.12: The left side is the original image, and the right side is the smooth effect.

Through contrasting the two images above, it can be found that the preprocessing method can effectively preserve the details of the image and prevent the loss of a small part of the original image by smoothing algorithm. The next step is to smooth the edges found.

4.3.3 The Majority filter

Majority filtering is a technique that can be used as a tool to substitute pixels according to the mode value in the nearby pixel. For the replacement to happen, two necessary requirements need to be met. The first condition is that the adjacent pixel number that equal value, it must either be big enough to be multivalued or that 1/2 of these pixels need to have equal value. For example, if the mode boundary is set as 3/4 or 5/8 of these connected pixels, then the pixels should have the equal value, if half is set, then 1/2 or 4/8 of these should have the equal value. The second condition follows the requirement that these pixels should be adjacent to the specified filter center (e.g. 3/4 of these pixels should be the same). These two conditions must be met for substitution to happen or else the pixel value will be the same.

Figure 4.13 shows an input grid being processed by majority filtering using the four adjacent pixels (the four pixels in orthogonal shaped). It requires that 3/4 of these pixels have the same value before these can be substituted. Therefore, substitution can only happen to the pixels if they are encircled by the 3 pixels (orthogonal shaped) that have equal or more value.

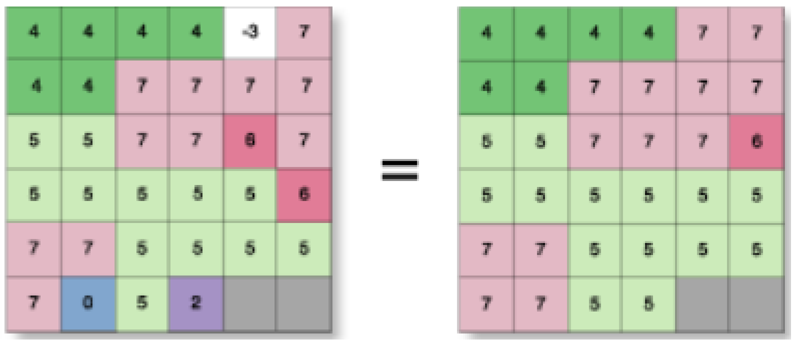


Figure 4.13: Majority smoothing algorithm using four pixels as a filter

In the figure 4.14, use the last eight pixels as a filter and require at least half of the values (four-eighths of the pixels) to be the same in order to change the values of the pixels, which results in a better smoothing effect.

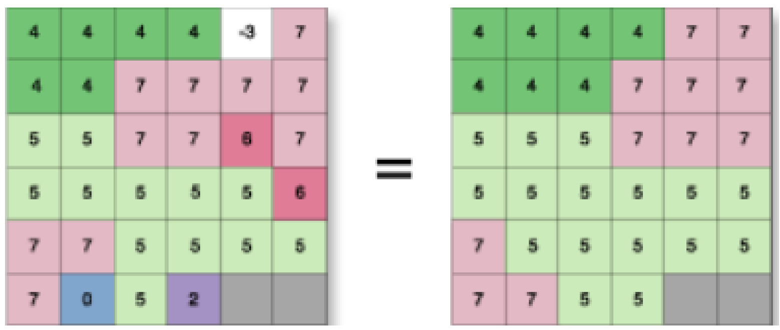


Figure 4.14： Majority smoothing algorithm using eight pixels as a filter

An important aspect that must be taken into account during the implementation of the majority algorithm is the difficulty which arises when the center of the filter coincides with various boundaries of the image under study. For example, suppose the size of the filter is n × n, is possible to approximate and find the range extremely away from the middle pixel of the immediate area is (n-1) / 2. When the distance between the center of the smooth filter and the boundary pixel is exactly (n-1) / 2, the scanning result is normal, It is well observed that this condition generally does not trigger any problems since the pixels in the filter are well contained in the image boundaries. Though, if the filter center is less than (n-1)/2 from the picture edge, the edge pixels of the image cannot be scanned. Figure 4.15 shows the challenge that occurs when the scanned image is smoothed.

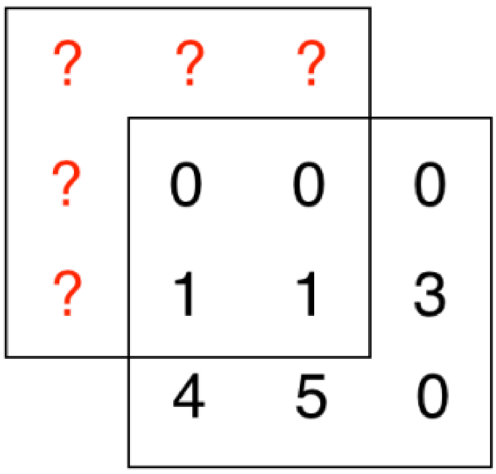


Figure 4.15: The border problem

Figure 4.15 demonstrates that when smoothing with a three by three filter if the center of the filter is on the edge of the image, 5 pixels of the edge will be lost and will not be scanned.

In this project, the idea to solve the boundary problem is to add the pixel matrix of image processing by adding rows and columns. If the height and width of the matrix are increased by (n-1) / 2, the improved majority algorithm will scan the pixels contained in the image, and then read the scan results into the new matrix, which is the same size as the processed image matrix, and finally discard the matrix with an increase of (n-1) / 2 times.

4.4 The results of the assessment

The expected result of improved majority smoothing algorithms is that the user can visually observe the effect of smoothing on the image. As the filter size increases, the smoothing effect becomes more obvious. On the other hand, the disadvantage is that most filter values cannot be increased all the time. After testing, the filter value is 10, the image smoothing effect is the best. If the filter size keeps increasing, so does the number of other pixel values in the neighborhood. As a result, the center pixel is more easily replaced by other pixel values, so some details of the image will be lost, which is not allowed in the project.

The acute angle in figure 4.16 (original image) becomes smoother in figure 4.17. Nevertheless, since the outline of the object in the picture is comparatively unfussy, therefore, the image smoothing results obtained by improved majority algorithms are not obvious. And there are only three pixels in the test image, which are not easily affected by another value of the pixel. The following figure shows the different scenarios of smoothing images with various shapes and colors using an improved majority technique, which better sees the performance of the smoothing algorithm.

Figure 4.16: A part of artificial PGM pictures (original image)

Figure 4.17： The image is 5 times larger and the filter is 3 * 3.

Figure 4.18: A part of the picture in DHP (original image)

Figure 4.19: The image is 5 times larger and the filter is 7 * 7.

According to the observation in figure 4.18, the structure in the DHP image is complex and the object has separate layers. The disparity among the initial picture and the resulting picture can be compared from the figure above. Contrasted to the initial picture, the 7×7 majority filter smoothed the image significantly smoother and with fewer jagged edges. In a smooth image, some corners become very smooth.

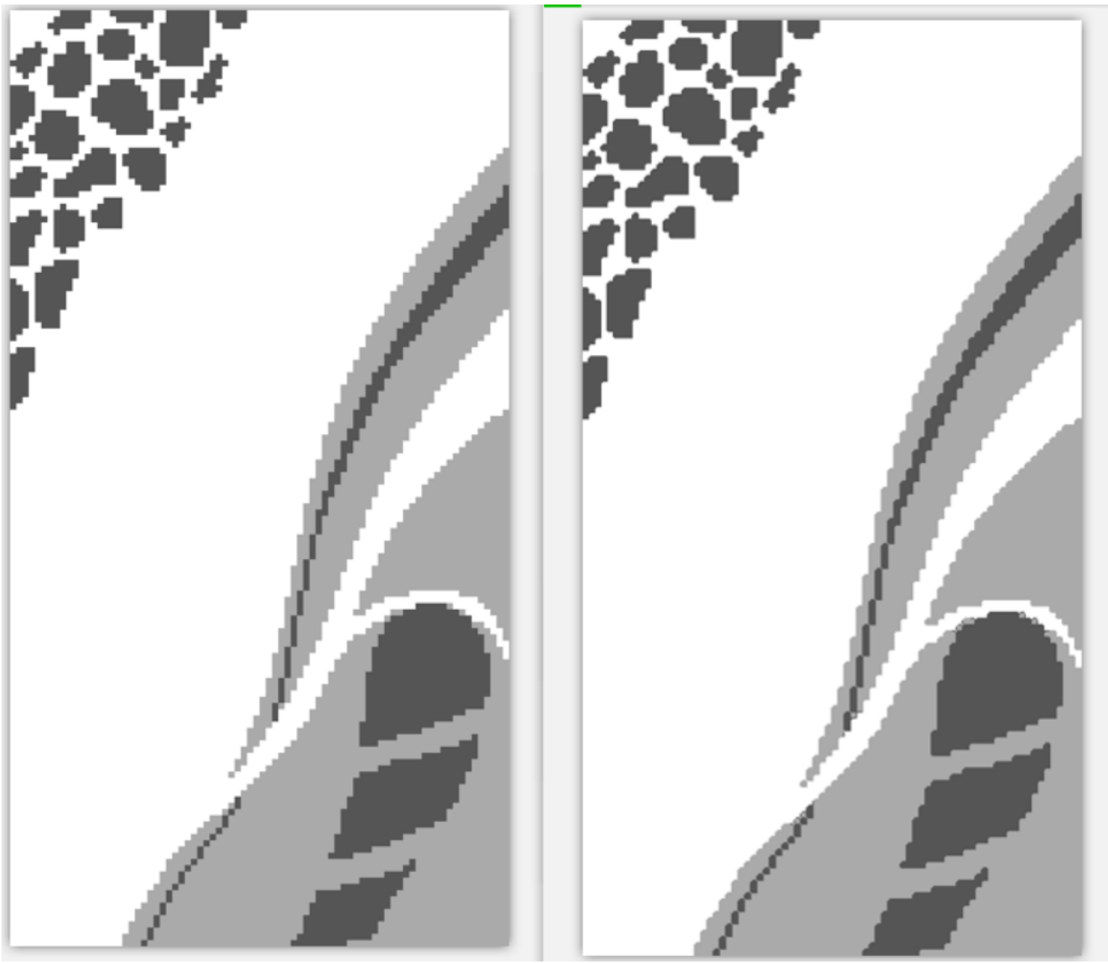


Figure 4.20：The figure on the left is the original figure, the figure on the right is the result of 5 times larger and the filter is 10 * 10.

4.4.1 Smoothing of 2D DHP images

Part 4.4.1 shows multiple DHP pictures processed by an improved majority smoothing algorithm. DHP pictures occur greatly compound than test pictures. The cross-section of the human physique encompasses a huge amount of tissue which includes skin, air, heart, fat, lungs and heart. The structure of various tissues or organs is diverse. The similar organization could also be dispersed in various parts. The results obtained through the improved majority algorithm are not much different from the expected results. The user can visually and clearly observe the smoothing effect of the image. This part of the image is selected from DHP head to foot representative images for smoothing test.

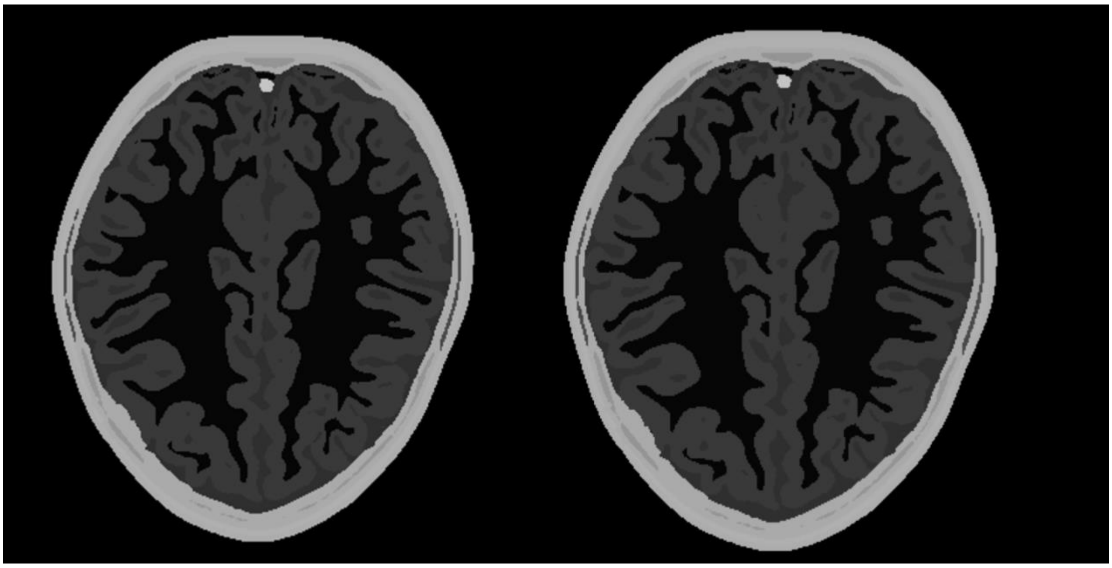


Figure 4.21: The figure on the left is the original figure, the figure on the right is the result of 5 times larger and the filter is 10 * 10. (The head in the DHP pictures)

Figure 4.22: This image is the smoothed part of 4.21 which shows the smoothed effect of the image.

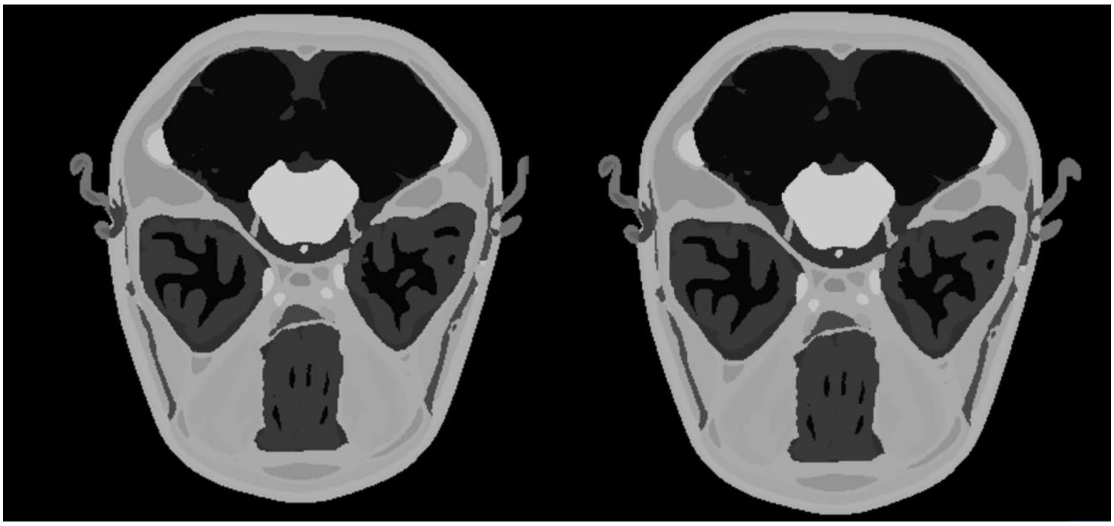


Figure 4.23： The figure on the left is the original figure, the figure on the right is the result of 5 times larger and the filter is 10 * 10. (The head in the DHP pictures).

Figure 4.24: This image is the smoothed part of 4.23 which shows the smoothed effect of the image.

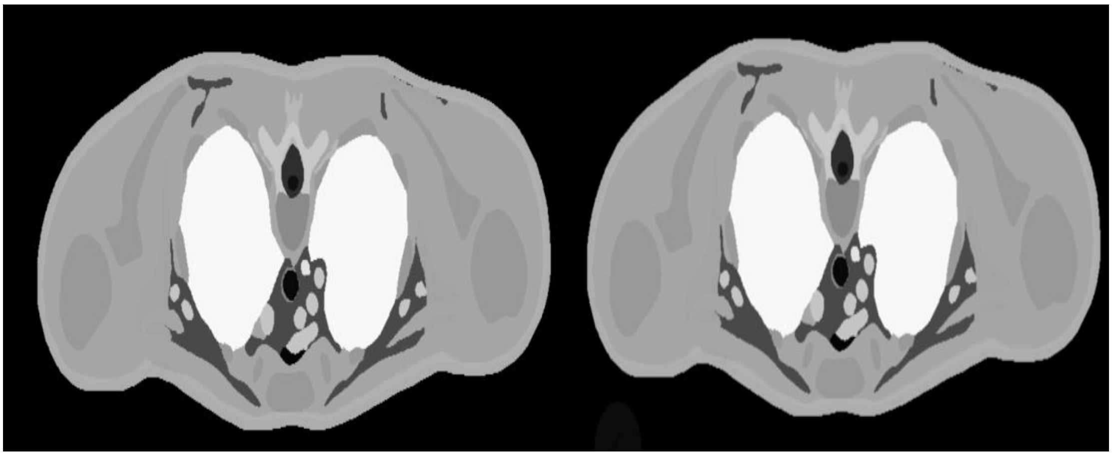


Figure 4.25： The figure on the left is the original figure, the figure on the right is the result of 5 times larger and the filter is 10 * 10. (A cross-section of the body in the DHP image).

Figure 4.26：This image is the smoothed part of 4.25 which shows the smoothed effect of the image.

Figure 4.27： The figure on the left is the original figure, the figure on the right is the result of 5 times larger and the filter is 10 * 10. (A cross-section of the foot in the DHP image).

Figure 4.28: This image is the smoothed part of 4.27 which shows the smoothed effect of the image.

4.4.2 Graphic interface construction

This project uses PyQt to create a graphical interface, and PyQt is equipment for generating GUI functions. It is an effective synthesis of the Python programming language and the Qt library. The Qt library remains to be one of the supreme libraries present today. PyQt was created by Phil Thompson.

PyQt executes a Python component program. It contains over 300 classes and approximately 6,000 methods and functions. It is a multi-platform toolkit that runs on all major operating systems, including UNIX, Windows, and Mac. Its main composition can be partitioned into numerous units. The QtCore unit comprises essential non-GUI features. This unit is utilized for directories and files, time, several data types, streams, URLs, MIME types, threads or processes. The QtGui unit comprises graphical components and connected divisions for example forms, status bars, buttons, bitmaps, colors, toolbars, fonts, scrollbars, and more. The QtNetwork unit includes divisions for network encoding that enables the user to inscribe TCP/IP and UDP clients and servers that render network encoding simpler and lighter. QtXml comprises divisions that use XML files. This unit offers implementations of the SAX and DOM APIs. The QtSvg unit offers divisions for exhibiting SVG files. Scalable Vector Graphics (SVG) is an XML jargon used to describe 2D graphics and graphics operations. The QtOpenGL unit uses OpenGL libraries to render 2D and 3D graphics, and the component seamlessly integrates Qt's GUI libraries and OpenGL libraries. The QtSql module offers divisions for the database.

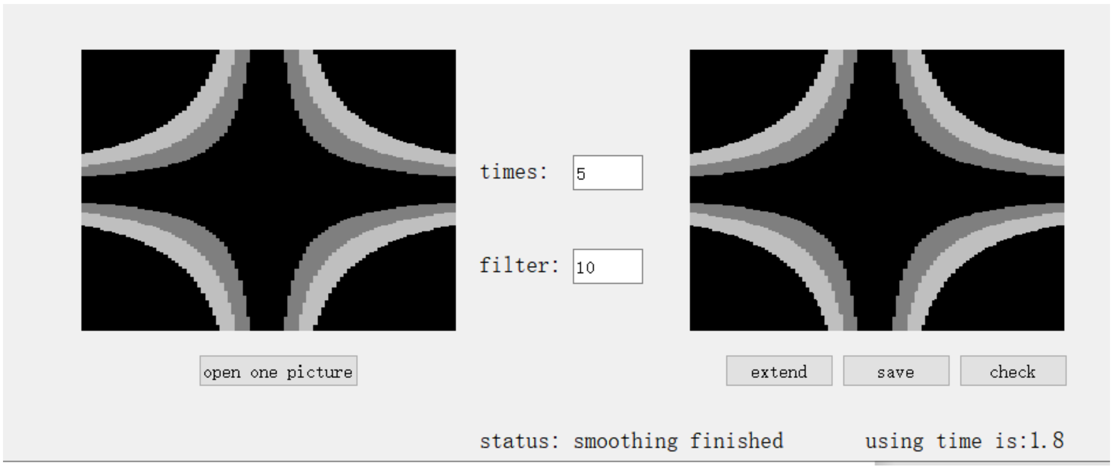


Figure 4.29：User interface diagram in the project

Function introduction: Open an image, input the amplification factor and filter size, click the extend button to enlarge the image; click check to ensure the missing feature;

add the image processing result save function, click extend, save the amplification Image, click check, save the image result after the missing check.

4.4.3 Time Consumption

Verify the performance of the inscribed code in the relations of computation interval by using two different images, artificial PGM images, and DHP images. The size and expansion factor of the image is highly correlated with the calculation time, obviously, a larger image requires a bit longer to calculate the processing results. For instance, in a scenario where an image is expanded by 3 times and the value of the filter is 5, the calculation time of the processing results for the artificial PGM picture is 0.5 second, and the smoothing time of the DHP image is 13 seconds. Another factor which influences the calculation of the processing results remains to be the expansion factor of the image. As the image expands, the number of pixels increases. In the case of DHP, the original image is expanded by 6 times, and the time consumed is 52 seconds. Compared with the expansion of 2 times, the time consumed is increased by about 8 times.

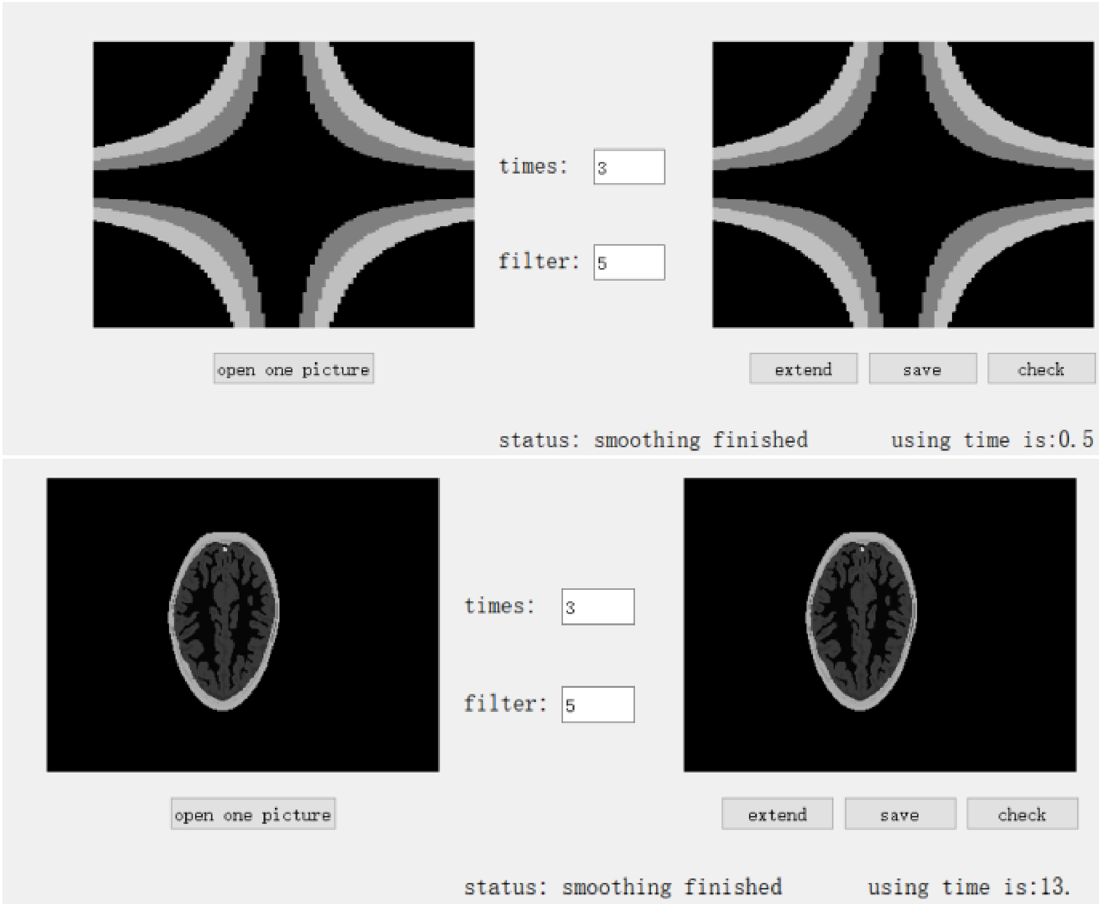


Figure 4.30：When the expansion multiples and filter values are equal, the required smoothing time is compared between the artificial PGM image and the DHP image.

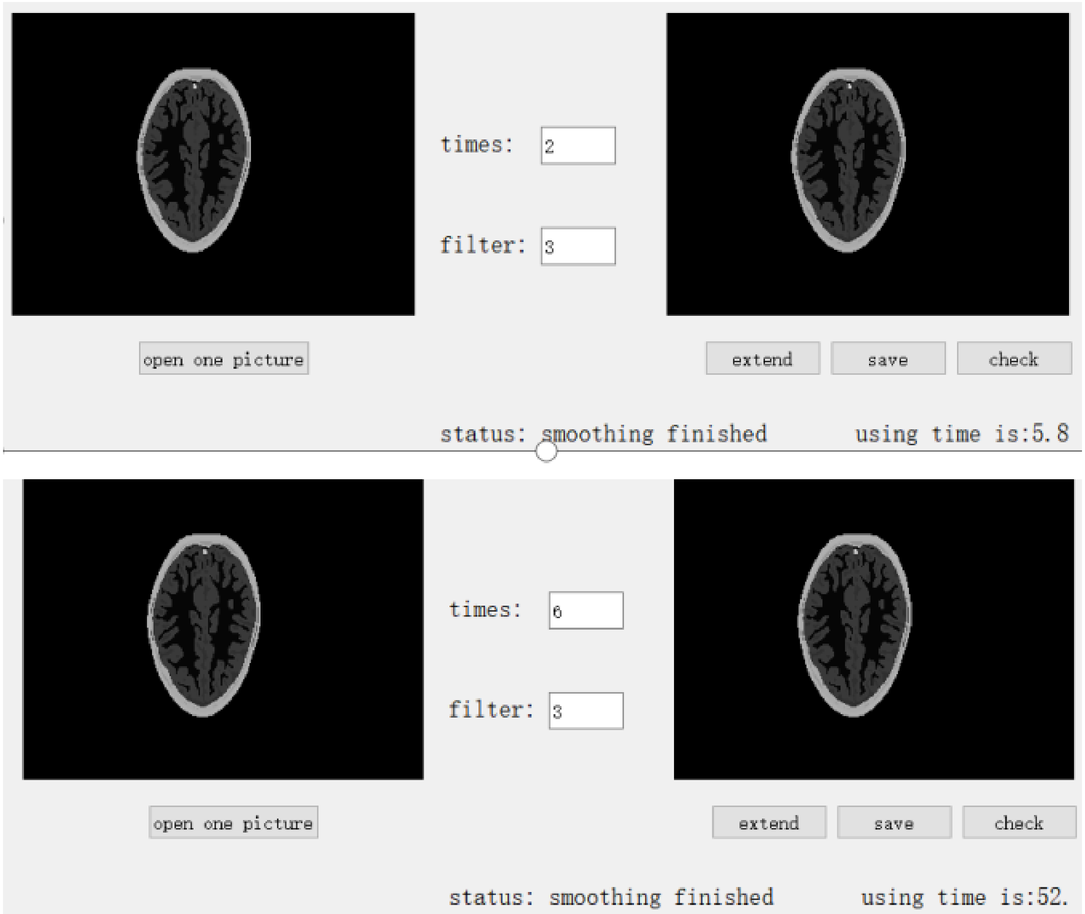


Figure 4.31: When the filter value is the same, the expansion multiple is different, and the smoothing time required for the artificial PGM image and DHP image is compared

5. Conclusion

DHP is produced by MRI scanning and segmentation, which is an image of the human body, showing the internal structure of the human body. According to different simulation requirements, the spatial resolution must be flexible. However, acute angles are overstated as the dimensions of DHP rises. Consequently, key objective of this study is to create a smoothing algorithm to level and enhance the visual quality of

DHP pictures. This paper presents an improved majority smoothing algorithm, which remains a good method. The assumption of the improved majority technique is to conduct canny edge recognition on the picture first and adopt sensible edge value to remove the tiny edge parts to stop them from being smoothed out in the subsequent smoothing operation. In the next phase, the linked domain technique is assumed to mark the connected domain of the binarized edge images (that is, mark the connected images). Afterward, erase the picture with an area less than 10*n (n magnification), and delete the fragment with a comparatively small area. Finally, the number of occurrence of each pixel value in the filter is calculated by the edge position, and the pixel value with the highest occurrence frequency is assigned as the new value of the focus pixel. The main benefit of the improved majority technique is that it cannot generate any fresh value in the picture, which refers to the new tissue that is not permitted to be generated by the smooth system. The improved majority technique has good smoothing influence on DHP image and decreases the acute angle of DHP image once the image size rises. This project also introduced the user interface function introduction, the user can be the simple and convenient operation, but also can get the smoothing process needs time when the picture processing is completed, set the save button, can according to the user needs after the smooth picture is saved.

5.1 Future work

In this scheme, the coding is still limited. The filter value of this algorithm can achieve good results when it is between 5 and 15. When the filter value of the improved majority algorithm is too large, such as 20×20, as shown in figure 5.1, the edge of image processing may be adulterated with other pixel colors. This problem cannot currently be solved by code. In addition, in order to ensure that the fine tissues in the picture are not smoothed out, the project chooses to smooth the edges, so the smoothing result is not very obvious, which leads to the complex comparison of results, only when the scale is large enough, can the smoothed image effect be compared. There is not enough time to solve or optimize these two deficiencies, so they need to be solved in future work.

Moreover, 3d smoothing remains to be one of the major undertakings impending. Currently, users can only get the smooth results of two-dimensional images, but cannot directly obtain smooth three-dimensional data, because only two-dimensional pictures of distinct planes are generated after processing. This procedure can be upgraded in the future by directly smoothing 3D information devoid of highlighting a particular plane so that the individual can obtain smooth 3D data and produce smooth images of whichever plane.

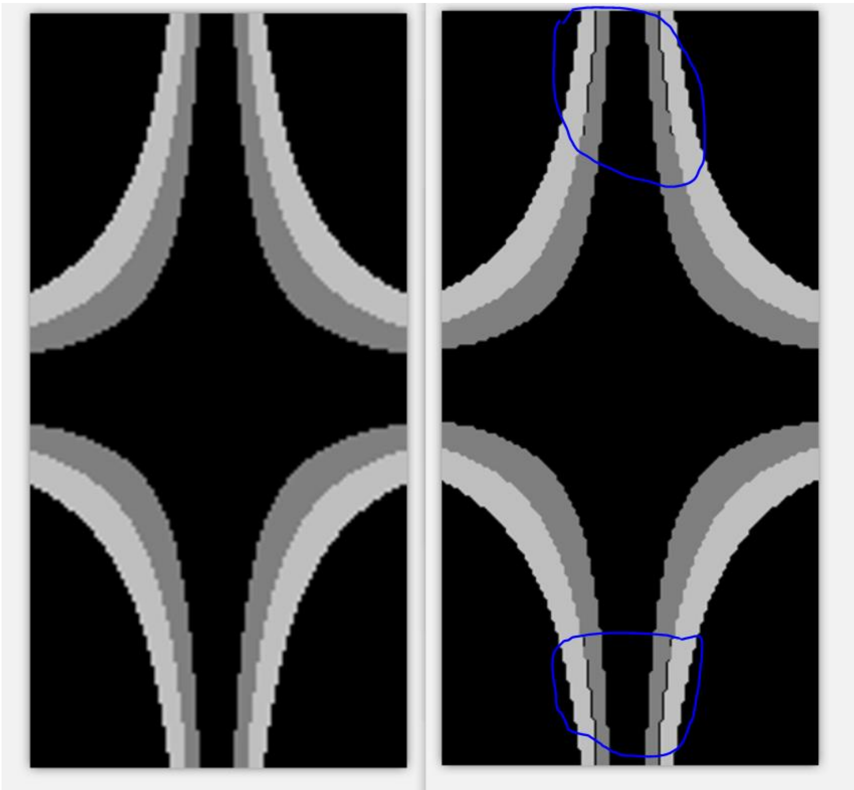


Figure 5.1: When the filter value is 20×20, the image edge rendering.

References

[1] Anon (1982) Magnetic resonance imaging (Online). New York, NY: Pergamon Press.

[2] Anon (1960) Radiation protection / report of Committee 2 on permissible dose for internal radiation.

[3] Anon (1975) Report of the Task Group on Reference Man: a report. Oxford; Pergamon Press.

[4] B. Graham and D. Cairns. CSCU9N5: Multimedia and HCL – Grapgics File Formats [Online]. Available: http://www.cs.stir.ac.uk/courses/CSC9N5/lectures/bpg/FileFormats1G.pdf Accessed on 2016

[5] Cameron, Elspeth (2013) Python. Reading Time. 57 (1), 37. [online]. Available from: http://search.proquest.com/docview/1370349915/.

[6] Clapham, Christopher & Nicholson, James (2014) linear interpolation. The Concise Oxford Dictionary of Mathematics [online]. Available from: http://www.oxfordreference.com/view/10.1093/acref/9780199679591.001.0001/acref-9780199679591-e-1693.

[7] Gonzalez, Rafael C. & Woods, Richard E. (2008) Digital image processing. 3rd ed. Upper Saddle River, N.J: Pearson/Prentice Hall.

[8] Gualdrini, Gianfranco & Ferrari, Paolo (2011) Handbook of anatomical models for

radiation dosimetry. Radiation Protection Dosimetry. [Online] 143 (1), 125–127.

[9] J. Burkardt. PGMA Files: ASCII PGM Files (Portable Gray Map) [Online]. Available:
http://people.sc.fsu.edu/~jburkardt/data/pgma/pgma.html Accessed on 10 June 2011

[10] Jiang, Nan & Wang, Luo (2015) Quantum image scaling using nearest neighbor interpolation. Quantum Information Processing. [Online] 14 (5), 1559–1571.

[11] Jing Lu et al. (2013) An Improved Bilinear Interpolation Algorithm Using Center Coordinates of Pixels. Sensors & Transducers. 19 (Special Issue), 89–93.

[12] Koay, Cheng Guan et al. (2007) Three-dimensional analytical magnetic resonance imaging phantom in the Fourier domain. Magnetic Resonance in Medicine. [Online] 58 (2), 430–436.

[13] Kramer, R et al. (2004) All about fax: a female adult voxel phantom for monte carlo calculation in radiation protection dosimetry. Physics in Medicine and Biology. [Online] 49 (23), 5203–5216.

[14] Lee, Juseong et al. (2018) Energy Efficient Canny Edge Detector for Advanced Mobile Vision Applications. IEEE Transactions on Circuits and Systems for Video Technology. [Online] 28 (4), 1037–1046.

[15] López-Rubio, Ezequiel (2016) Superresolution from a Single Noisy Image by the Median Filter Transform. SIAM Journal on Imaging Sciences. [Online] 9 (1), 82–115.

[16] Medina-Carnicer, R. et al. (2011) A novel method to look for the hysteresis thresholds for the Canny edge detector. Pattern Recognition. [Online] 44 (6), 1201–1211.

[17] Na, Yong Hum et al. (2010) Deformable adult human phantoms for radiation protection dosimetry: anthropometric data representing size distributions of adult worker populations and software algorithms. Physics in Medicine and Biology. [Online] 55 (13), 3789–3811.

[18] Spagnolo, Fanny et al. (2018) An Efficient Connected Component Labeling Architecture for Embedded Systems. Journal of Low Power Electronics and Applications. [Online] 8 (1). [online]. Available from: http://search.proquest.com/docview/2026650331/.

[19] Tarkia, Miikka et al. (2014) Abstract 15108: Imaging Coronary Atherosclerosis Using [18F] FDG PET-CT: Validation Study in Pigs. Circulation. 130 (Suppl_2 Suppl 1), A15108–A15108.

[20] Wang Lei et al. (2015) 'A Two-Step Robust Filter for Mean Line Extraction Based on the Median Filter', in 2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC). [Online]. September 2015 IEEE. pp. 25–28

[21] Wüthrich, Manuel et al. (2016) A new perspective and extension of the Gaussian Filter. The International Journal of Robotics Research. [Online] 35 (14), 1731–1749.

[22] Xu, G., X. et al. (2000) VIP-man: an image-based whole-body adult male model constructed from color photographs of the visible human project for multi-particle monte Carlo calculations. Health Physics: The Radiation Safety Journal. [Online] 78 (5), 476–486.

[23] Yu, Han et al. (2016) Alternative framework of the Gaussian filter for non-linear

systems with synchronously correlated noises. IET Science, Measurement & Technology. [Online] 10 (4), 306–315. [online]. Available from: http://search.proquest.com/docview/1845804814/.

[24] Zaidi, Habib & Xu, Xie George (n.d.) Computational Anthropomorphic Models of the Human Anatomy: The Path to Realistic Monte Carlo Modeling in Radiological Sciences. [Online] 9 (1), 471–500.

[25] Zankl, M. et al. (1988) The construction of computer tomographic phantoms and their application in radiology and radiation protection. Radiation and Environmental Biophysics. [Online] 27 (2), 153–164.