CENTRO DE INVESTIGACIONES EN OPTICA, A.C.

Segmentation of molars in noisy pantomograms using digital image processing techniques

Author:

José Luis Reynoso Rodríguez

Supervisor:

Dr. Francisco Javier Cuevas de la Rosa

Thesis submitted in partial fulfillment of the requirements for the degree of Master in Optomechatronics

"Let the future tell the truth, and evaluate each one according to his work and accomplishments. The present is theirs; the future, for which I have really worked, is mine."

Nikola Tesla

Acknowledgments

I would first like to thank my thesis advisor PhD Francisco Javier Cuevas de la Rosa at Centro de Investigaciones en Óptica. The door to PhD Cuevas' office was always open whenever I ran into a trouble spot or had a question about my research or writing. He consistently allowed this paper to be my own work, but steered me in the right the direction whenever he thought I needed it.

I would also like to thank the advisors who were involved in the validation for this research project: PhD Carlos Pérez López and PhD María del Socorro Hernández Montes. Their advice helped in the creation of a clearer documentation.

I would also like to acknowledge PhD Abundio Dávila Álvarez at Centro de Investigaciones en Óptica for some corrections on this thesis, for his help in digital image processing techniques, and for motivating me to use LaTeXto write this document.

I'm also grateful with Consejo Nacional de Ciencia y Tecnología (CONACYT). My work on this thesis could not have been done without their economic support through a scholarship.
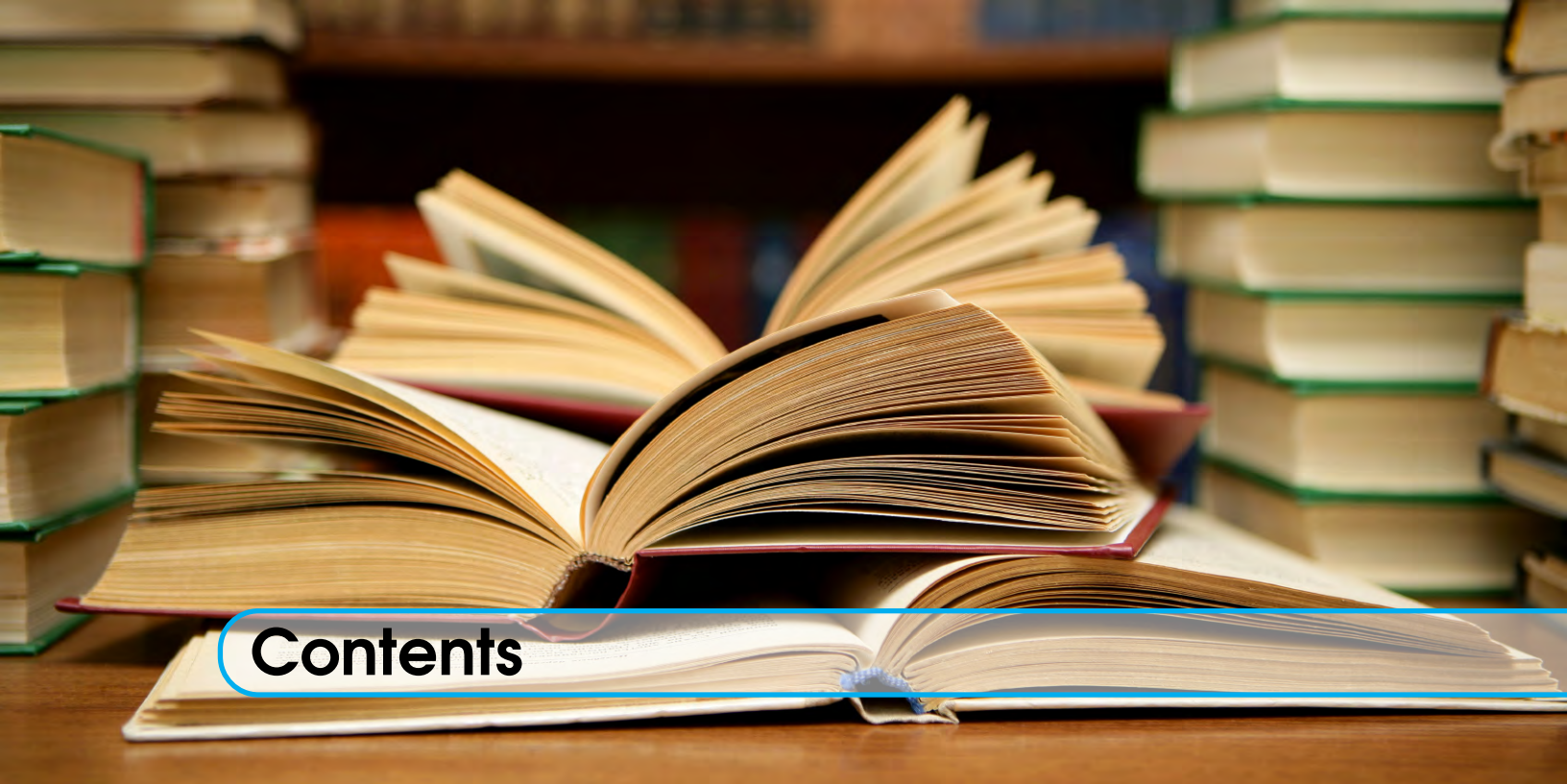
A big thank you to my favorite Colombians and classmates: Leidy Johana Quintero Rodríguez and Jorge Mario Uribe Martínez, and, in particular, to my classmate Francisco Javier Vargas Muñoz ("Paquito") in whom I found a valuable friend.

Special thanks to my friend Oscar Gónzalez Chavarría, whom I have known since kindergarten and our friendship still stands strong. Both academically and personally, his support has been unmeasurable. May we have many more gaming and pancake nights.

Finally, I must express my very profound gratitude to my parents and to my girlfriend Bethel Manrique for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without their faith in me.
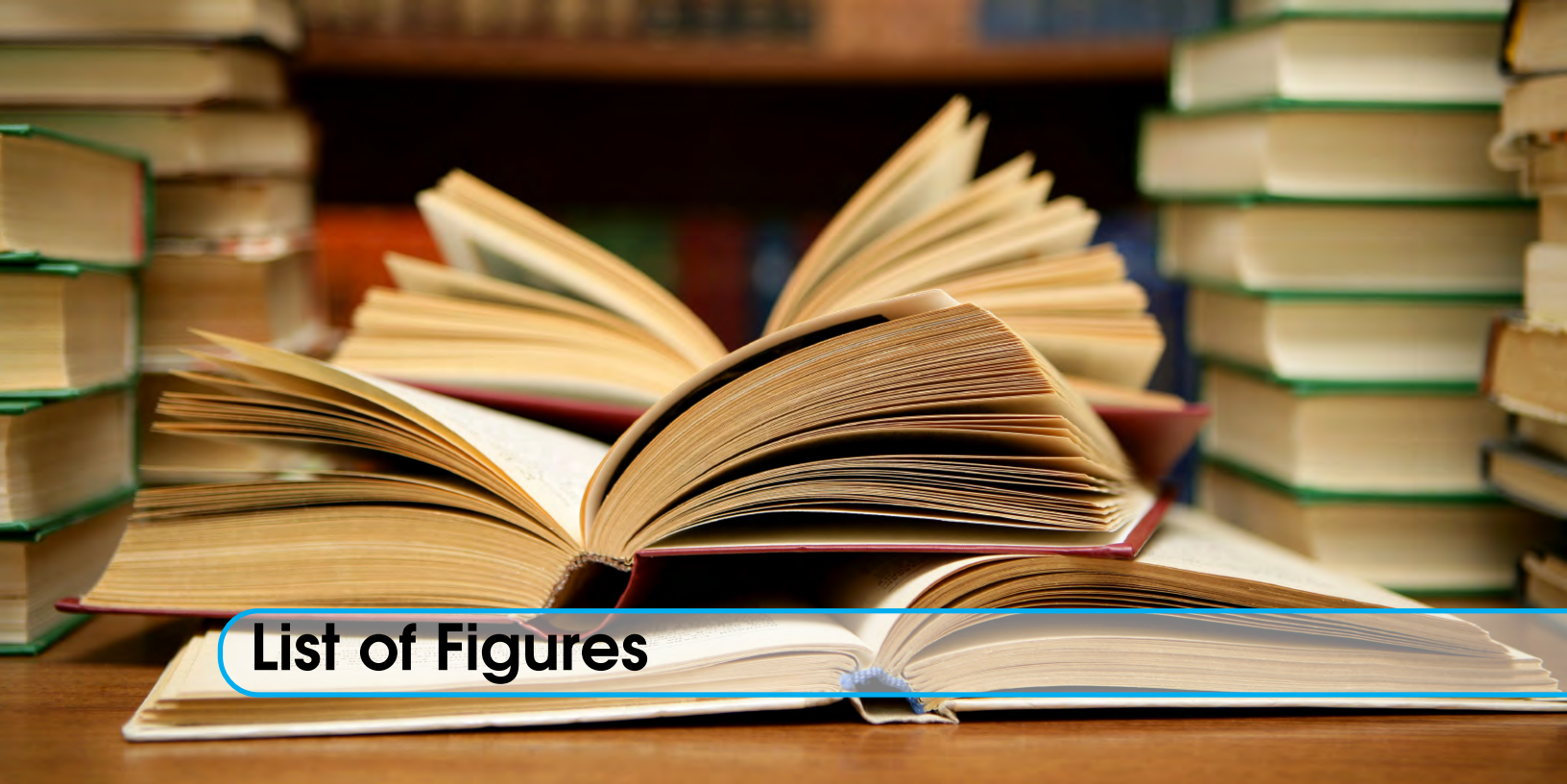
Thank you.

José Luis Reynoso Rodríguez
"Pepis"

Contents

List of Figures

List of Tables

1. Introduction

Contents

Over the past 20 years, technology has advanced exponentially. This has lead to some dramatic changes in human (and even non-human) lifestyles. When it comes to medicine, technology has made a huge impact on this field. Radiographs are so often used that all modern hospitals possess radiological facilities. According to the United States Food and Drug Administration, some applications of radiographs are [1]:

- Dental examinations
- Verification of correct placement of surgical markers prior to invasive procedures
- Mammography
- Orthopedic evaluations
- Spot film or static recording during fluoroscopy
- Chiropractic examinations

The work presented in this thesis is focused on dental radiographs. Most modern dental clinics have the capability of getting radiographs from patients. They are a great way to analyze teeth and find problems which could not be seen with the naked eye, which is why are are so fundamental in dental procedures.

Digital radiography was introduced in the mid 1980s [2] and is competing with conventional screen film radiography in radiographic applications [3].

There are two categories in which dental x-rays can be divided:

Intraoral x-rays. This is the most common type. They provide detail about specific teeth so they are used for identifying problems with individual teeth.

Extraoral x-rays. Their main focus is the jaw and skull. They are used to monitor problems between these last two and teeth.

The work on this thesis is done on **pantomograms**, also known by its trade names Panorex or Orthopantomograph or more typically called panoramic dental radiographs. Pantomograms are two-dimensional images that capture the entire mouth, including the teeth, upper and lower jaws, and surrounding structures and tissues. It is important to note that this procedure causes some geometric distortion because it produces a flat image of a curved structure just like a map projection.
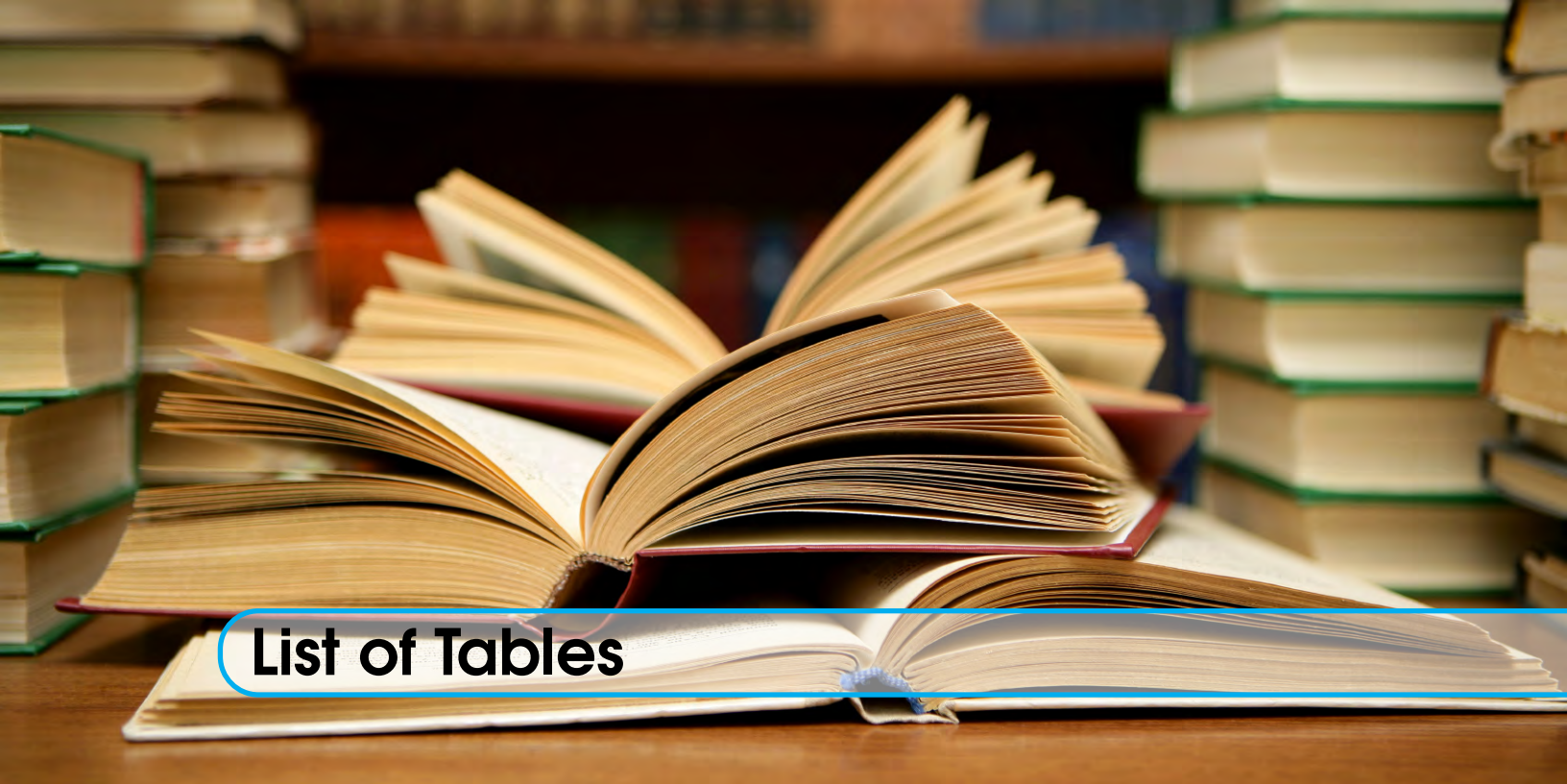
Figure 1.1: Example of a pantomogram. Image courtesy of the dentistry faculty of Universidad La Salle Bajío.

In this type of tomography, narrow-beam equipment is used, synchronizing the movement of the x-ray tubehead and the cassette carrier, which are designed to rotate in the horizontal plane, in a circular path around the head, with a single center of rotation [4]. The resultant focal trough is curved and forms the arc of a circle. The plane of the object that is not blurred on the radiograph is called the plane of acceptable detail or **focal trough**. It is also called the image layer.

In the case of digital radiography, a digital sensor approximately the size of the slit beam collimating device is placed in the carrier instead of a film/screen combination. As the carrier moves around the patient, the electronic impulses are sent back to the computer and an image is generated.

(a) Principles of tomography. Note that only objects in the focal trough (square) project onto the same area of the film and are not blurred out.

(b) Rotational tomography procedure. Note the slit opening in the film carrier and that the film and the tube travel around the patient in opposite directions.
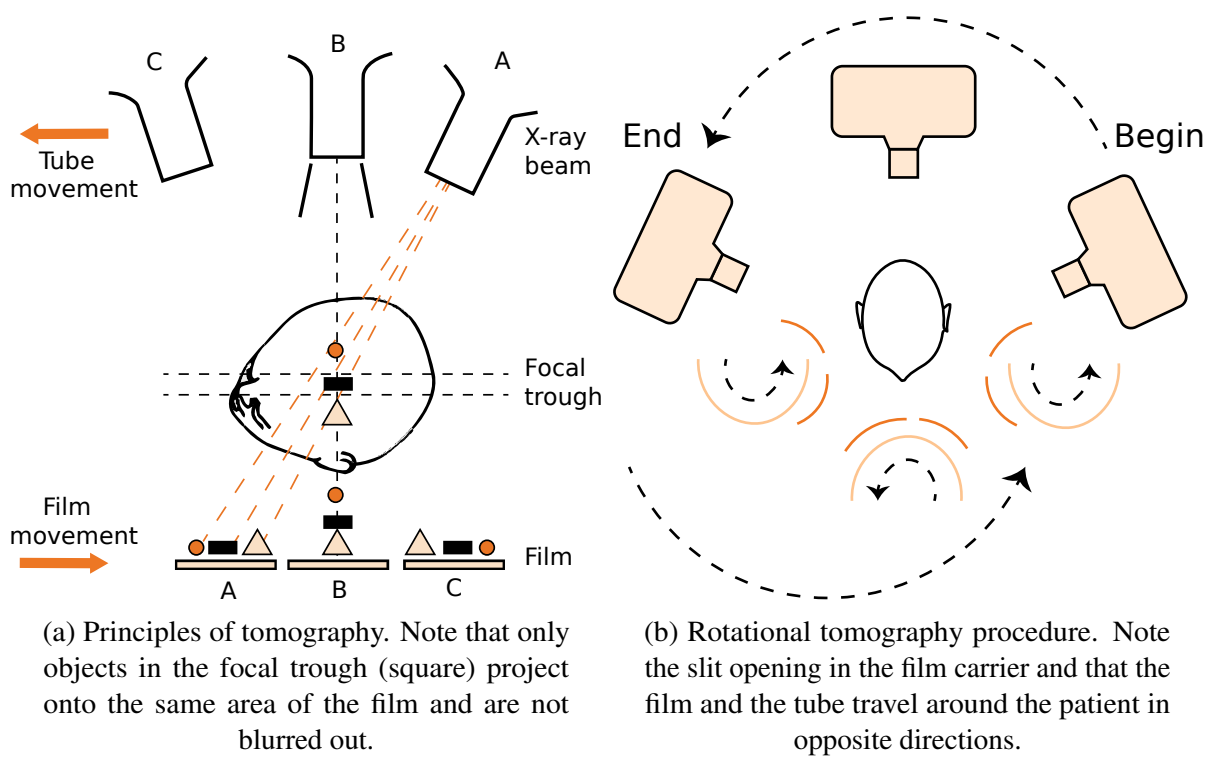
Figure 1.2: Midsagittal plane of a human being.

Although most of the panoramic machine units are different, they follow some common positioning requirements for patients and operators, which are explained below [5]:

1. The patient's midsagittal plane (Figure 1.3a) should be perpendicular to the floor.
2. In most units, the Frankfort plane (Figure 1.3b) should be parallel to the floor.
3. The anterior teeth should be positioned in the proper groove in the bite block and not forward or posterior to the groove (Figure 1.3c).
4. The chin should not be angled up or down.
5. The head-positioning devices should be firm to prevent tipping or rotation during the exposure and loss of proper midsagittal orientation.
6. The patient must not move during the exposure or a blurred image will result.
7. The patient should maintain his or her tongue against the roof of the mouth.
8. The patient must keep his or her spine erect.
9. A lead apron with no thyroid collar should be used as a radiation shield in order to protect the patient and not block the primary beam.

(a) Midsagittal plane of a human being (blue plane).

(b) Orbito-meatal (Frankfort) plane of a human being (blue dotted line).

(c) Anterior teeth positioned correctly.
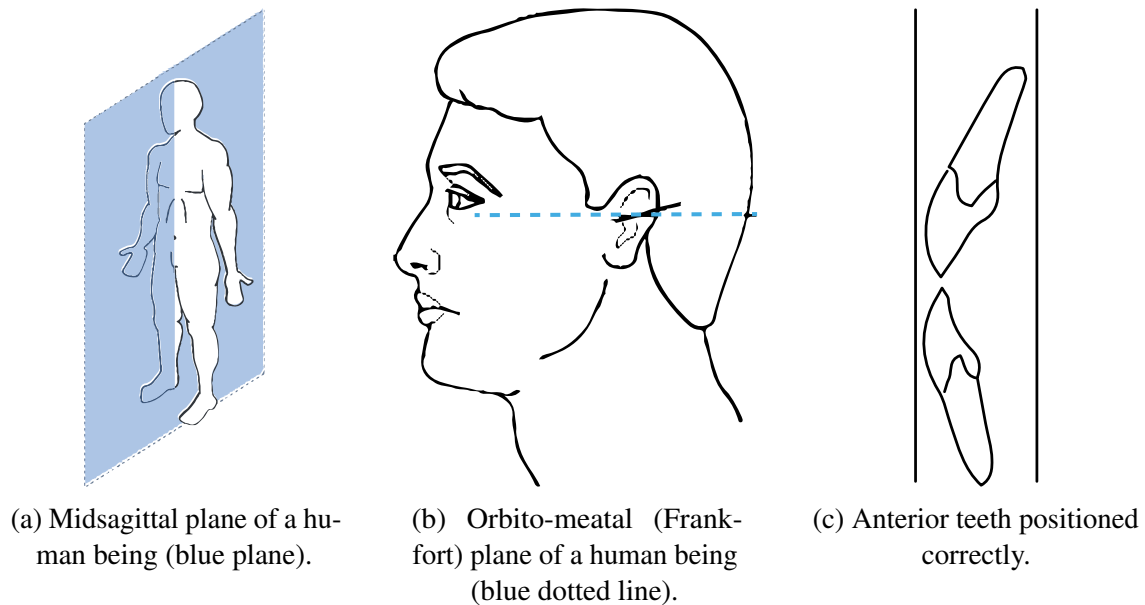
Figure 1.3: Midsagittal plane of a human being.

One of the main disadvantages of pantomographic units is that they have a tendency to produce overlapping images, particularly in the premolar area. An approach to correct this problem is presented in this thesis.

1.1 Problem background

There exists little work done on dental radiographs using digital image processing, with most of the work consisting in post-processing results such as enhancing contrast for better human appreciation [6, 7].

Some morphological teeth features can be related to the development of certain diseases, such as the case of enamel pearls and periodontal disease [8, 9, 10]. Finding these features in time is essential to avoid complications; however, in the case of enamel pearls and periodontal disease, given a sufficiently big database, this may prove difficult to do manually considering the enamel pearls low rate of appearance [11], hence the importance of automated methods of detecting these features. Segmentation of molars if the first step towards detecting these morphological features of interest. In the case of periodontal disease, some of these morphological features of interest are enamel projections and enamel pearls.

1.1.1 Enamel

Dental enamel is one of the four main tissues (along with dentin, dental cementum and dental pulp) that make up the tooth in humans and other animals, including some species of fish.

Dental enamel is the part of the teeth that we normally see and that covers the crown. It is usually white or slightly yellow in color and quite resistant, thus acting as a protective barrier for the tooth, although it may be subject to degradation, especially from food acids or beverages. Because the ameloblasts (cells responsible for producing the enamel) are lost once the tooth springs into the oral cavity, dental enamel is impossible to renew naturally.
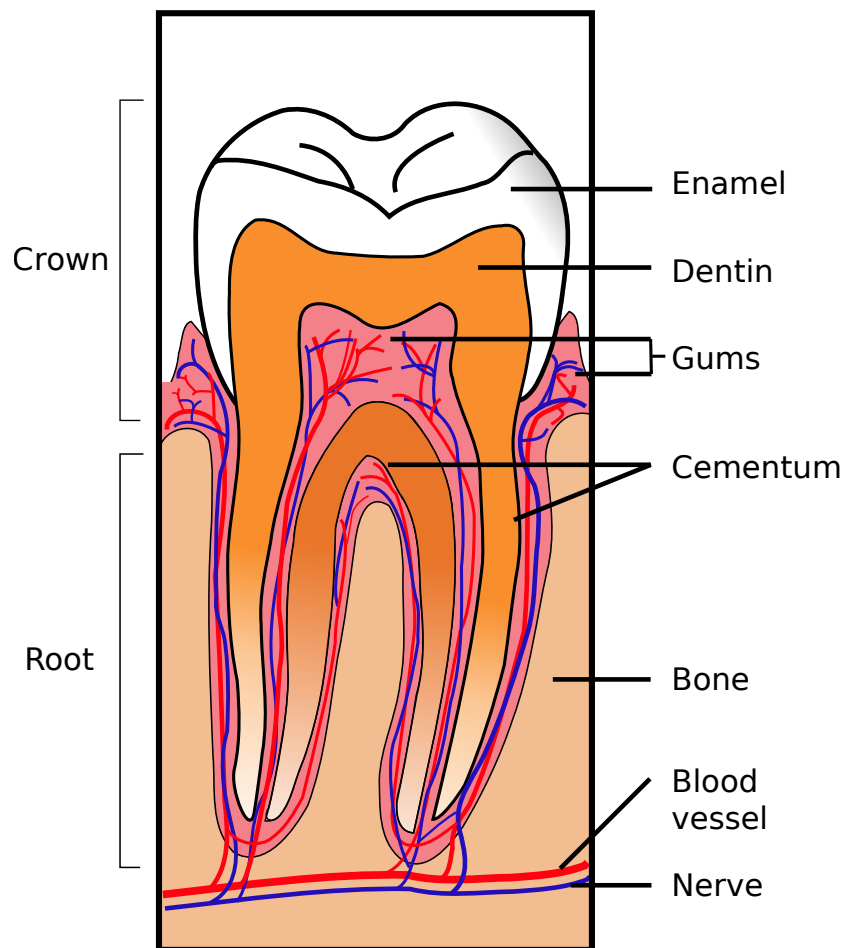
Figure 1.4: Cross section of a tooth where some of its most relevant parts are shown.

In humans, the enamel varies in thickness along the surface of the tooth, reaching its greatest thickness in the cusps of the teeth (elevations of enamel with foundation in the dentin), where it can be 2.5 mm thick; while the thinnest part can be found on the edge of the dental

cement and the cement-enamel junction [12].

Enamel is the hardest substance in the human body and contains the highest percentage of minerals (96 %), while the remaining 4 % is composed of water and organic material [12]. This large amount of minerals in the enamel contribute to its strength and fragility. Enamel is at level 5 of the Mohs hardness scale and has a Young's modulus of 83 GPa [13]. This composition of materials, together with its organizational structure, allows the enamel to withstand the mechanical forces applied by the teeth.

In radiographs, the differences in mineralization between the different tooth portions can be appreciated; enamel appears in a brighter shade than dentin or pulp, as it is denser than these and more radiopaque [14].

Figure 1.5: X-ray image where the contrast in luminosity of the different parts of a tooth can be appreciated.

1.1.2 Emamel projections

Amelogenesis is the process of enamel formation on the teeth. The process begins in the teeth while they are still developing inside the gums, before they erupt [15]. Also known as enamel extensions, enamel projections occur when amelogenesis continues even after the crown of the tooth has been covered, which causes enamel to overcome the cervical border (amelocemental junction) and extend into the furcation of the tooth [16].

Cervical enamel projections are usually classified into three grades according to the scale published by Masters and Hoskins [17]. This scale is shown in Figure 1.6.
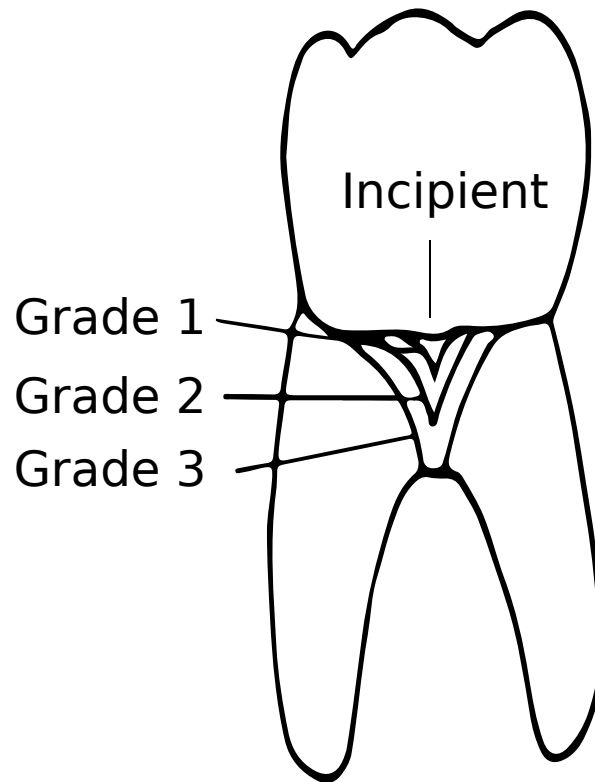
Figure 1.6: Classification of cervical enamel projection according to Masters and Hoskins on "Projection of Cervical Enamel into Molar Furcations".

In a study carried out by Prashanth, Nagarathna, and D'Souza from the Institute of Dental Sciences, the average length of enamel extensions was found to be 3.7 mm with a standard deviation of 1.1 mm [18].

Its clinical significance is related to the fact that they could contribute to the formation of periodontal pockets, which in turn could evolve into periodontal disease [17, 19, 20].

1.1.3   Enamel pearls

An enamel pearl is an abnormal localized developmental condition of enamel somewhere in the tooth, usually found on the dental root surface [16, 21, 22] and the molar furcations [23]. In an anthropological context, they are usually related to enamel projections, and can be found even attached to these [24].

Figure 1.7: Enamel pearl visible on a third molar (wisdom tooth).

According to Matthews and Tabesh, enamel pearls have a prevalence between 1.1 % and 9.7 % and are most commonly found in upper second molars and lower third molars [11]. A retrospective study by Colak et al. evaluated radiographs of Turkish patients, from which a 5.1 % exhibited enamel pearls. The researchers found a higher percentage of occurrence in men (6.58 %) compared to the percentage of occurrence in women (3.96 %). Similarly, a higher percentage of enamel pearls appearance was found in the mandibular region of the first molars without an appreciable statistical difference on the right side or the left side [25].

Enamel pearls are most likely caused by an anomaly in the Hertwig epithelial sheath, in which a stellate reticulum area (the middle layer of the enamel organ) is retained between the two layers of epithelium that normally make up the root sheath Cite pmid10760733. The presence of stellate reticulum allows the differentiation of the cells of the root sheath into the ameloblasts, leading to the formation of a small area of enamel on the external root surface [23, 26].

Enamel pearls are linked to the origin of periodontal disease because they interrupt the insertion of periodontal fibers into the cementum and contribute to the accumulation of plaque or biofilm, development of subgingival calculi and subsequent formation of an active periodontal pocket [21, 23, 27, 28].

In an radiograph, enamel pearls can be seen as a soft, radiopaque structure in the tooth root. Because the nodule is below the tissue, it is not noticed during the review of usual oral tissue.
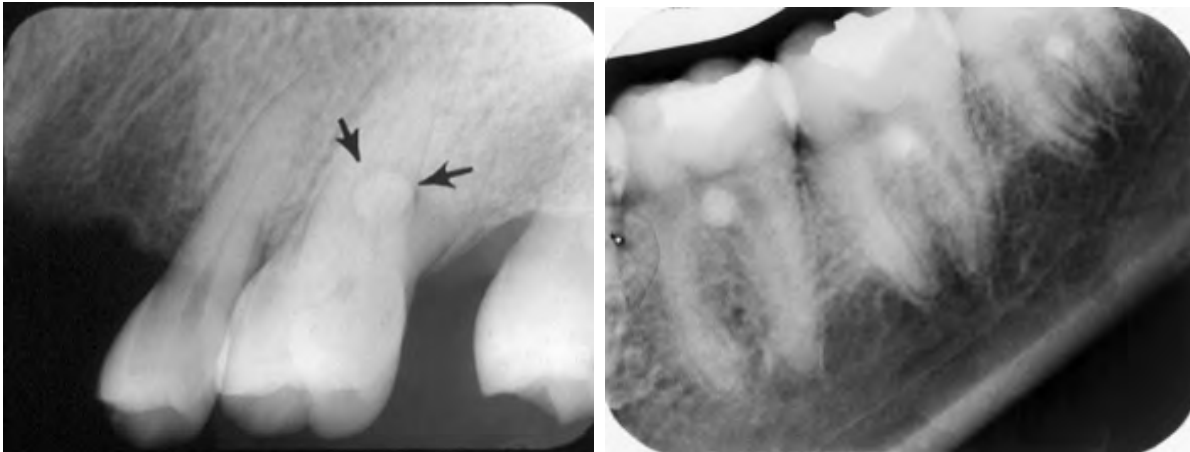
Figure 1.8: Enamel pearls shown in a radiograph.

Some other tasks, like classification of difficulty of third molars extraction, require methods which may result cumbersome and slow when doing manually, making computer-assisted methods really helpful. Additionally, measurements of morphological features may be more accurately acquired using digital image processing methods.

1.2   Work summary

In this thesis, digital image processing techniques were used in order to isolate and segment individual molars in pantomograms; in particular, second and third molars. This is done using two different approaches pre-processing the image. The general area of the tooth is cropped beforehand manually in order to speed-up the digital image processing techniques. The orientation of the segmented molars is also determined using image moments.

In both implemented methods, the pantomograms are preprocessed by applying a Gaussian filter to smooth some edges of the image. The smoothed image is then clustered into superpixels, and the mean color of each cluster is computed. Each pixel is then assigned its cluster's mean color. This helps to achieve color uniformity and reduces noise while preserving important image information such as edges.

1.2.1   Using Otsu's method

The first method presented in this work uses fast local Laplacian filtering in order to produce a contrast-enhanced image [29, 30]. Fast local Laplacian filtering greatly helps to make the teeth stand out from the noisy background in the pantomograms. This makes it easy to apply Otsu's method to binarize the image and get the teeth isolated from the background.

The previous algorithm successfully extracts the molars from the background of the pantomograms; however, most of the times, the teeth overlap each other, and a method to separate the different teeth is required. A method for separating the overlapping teeth is proposed using the distance transform on the binary image obtained from Otsu's method and applying watershed segmentation to the resulting distance transform. The teeth are further separated from each other using image gradients. The images are cleaned using morphological operations between steps.

1.2.2   Using active contour

This method applies the active contour algorithm directly to the contrast-enhanced image created by using fast local Laplacian filtering. The initial mask for the active contour algorithm is created automatically while cropping the image, so it is not necessary to specify the initial mask manually.

The active contour algorithm does not produce perfect segmentations either and also creates masks where the teeth overlap each other, so the proposed algorithm for separating the overlapping teeth using the distance transform and watershed segmentation applied in the segmentation using Otsu's method is once again used in this approach.

The advantage of using active contouring over Otsu's method is that the latter requires different morphological operations for different images, while the same morphological operations can be applied to active contouring to achieve a final segmentation.

1.3   Related research

There has been research about the challenges of edge extraction from dental radiographs using conventional digital image processing algorithms [31], and research on teeth segmentation [32, 33]. Teeth segmentation has been performed in x-rays radiographs in order to identify subjects post-mortem [34]; however, there is no work done for automated measurements (such as orientation) or segmentation of molars in-vivo patients.

1.4 Thesis structure

The following is a general summary of the different chapters contained in this thesis, in order to give the reader a preamble of the content of this thesis. The digital image processing techniques used are all explained and some may contain some MATLAB code examples or pseudocode.

Chapter 1: Introduction

A brief introduction about the problem background and why is it relevant for more future work, the work done in this thesis and any similar work done by others.

Chapter 2: Digital image formation

This chapter briefly describes how a digital image is formed, which components affect image formation, how a digital image is represented and why it is represented like that.

Chapter 3: Filtering and morphological operations

In this chapter, the digital image processing techniques used in this thesis are described. Some example code is included for some of the operations, as well as actual results from each of the operations. In the case of more complicated procedures such as superpixel segmentation, the pseudocode is listed instead.

Chapter 4: Segmentation results

The digital image processing techniques of the previous chapter are applied to the pantomograms supplied by the Faculty of Dentistry of Universidad de la Salle Bajío. Each operation done is documented with the parameters that were used to acquire the output image shown in the figures.

3D Object Lens CCD/CMOS Digitizer 2D Digital Image
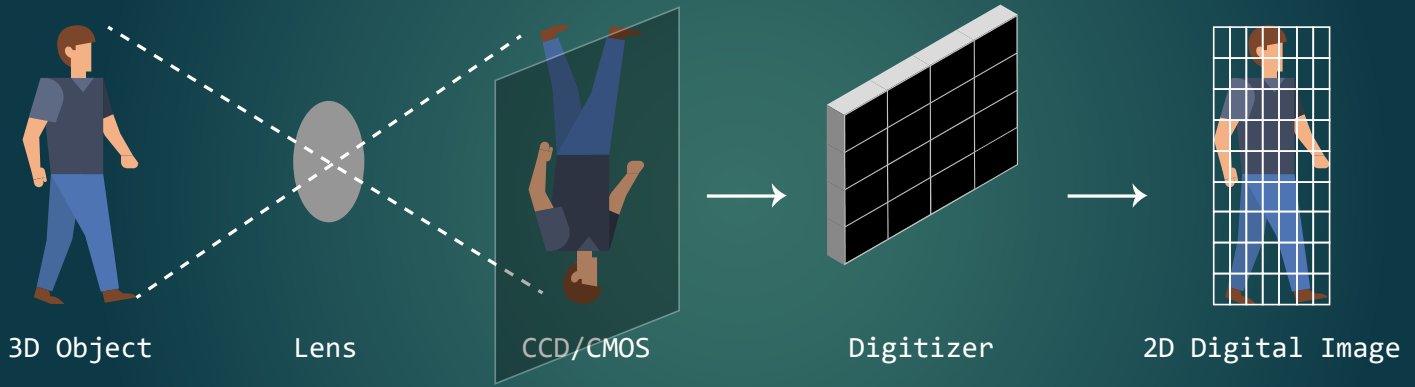
2. Digital image formation

Contents

2.1 Digital Cameras

A digital image can be represented as a two-dimensional signal $I(x,y)$, where x and y are discrete values which correspond to the number of samples taken horizontally and vertically from the real image respectively.

Each sample forms a point in the image called pixel (a contraction of *picture element*) and represents the smallest homogeneous unit in a digital image. When a digital image is said to be M pixels wide and N pixels tall that means the image has got M columns of samples and N rows of samples for a total of $M \times N$ samples. Each pixel in the image is given a discrete value which represents the luminosity captured by the camera sensor.

In order to reference a specific pixel, a coordinate system is required. Unlike the Cartesian coordinate system, the origin is located at the top left pixel of the image with increasing x values to the right and increasing y values towards the bottom. This coordinate system can be seen in Figure 2.1.
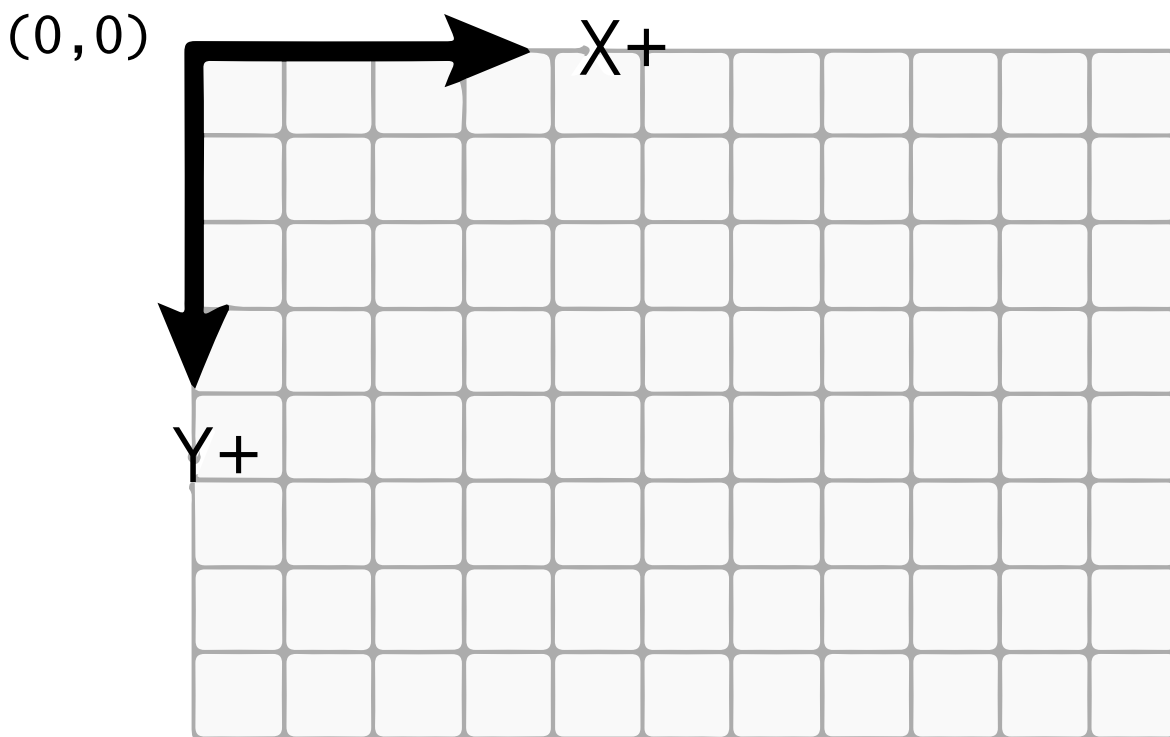
Figure 2.1: Coordinate system used for digital images.

2.1.1 Pixel values assignation

The value acquired by a pixel depends basically on three components:

1. The color of the light which illuminates the environment
2. The absorption and reflection properties of the object on which light is striking
3. The sensibility of the camera sensor that is being used to capture the picture
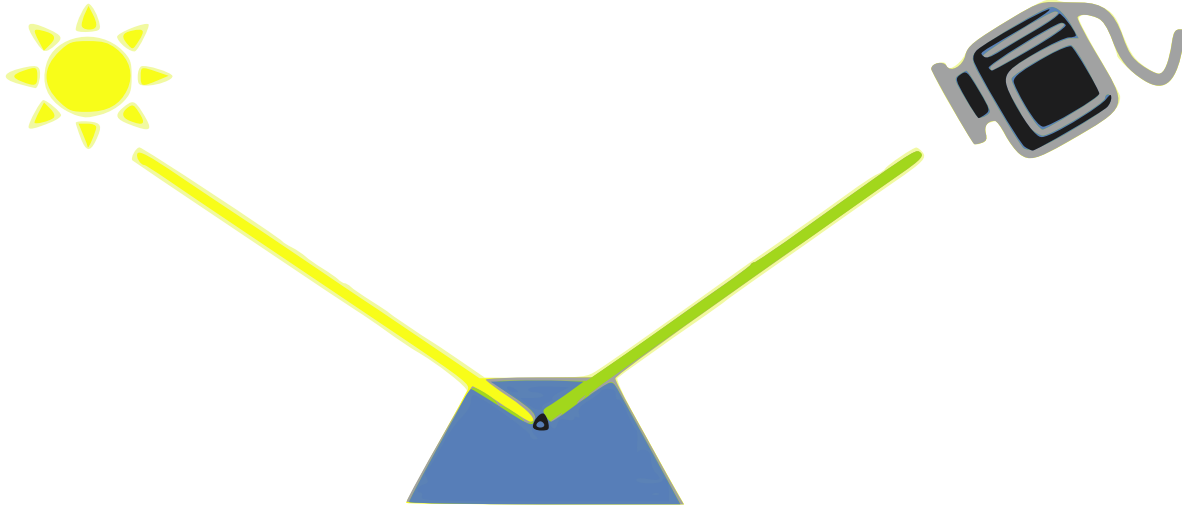
Figure 2.2: Elements that affect the composition of a digital image.

2.1.2 Light color

Light is a form of energy that can be represented as an electromagnetic wave with a specific wavelength or frequency. The wavelength and frequency correspond to the periodicity to which the signal is repeated. The perceived color of an object by the human eye is subject to this wavelength or frequency. The human eye is only able to see a subset of electromagnetic wavelengths; this subset is called the visible spectrum.

The visible spectrum for humans is located approximately between the wavelength interval of 390 nm to 700 nm [35]. Shorter wavelengths are associated to violet and blue colors, while longer wavelengths are associated to red colors.
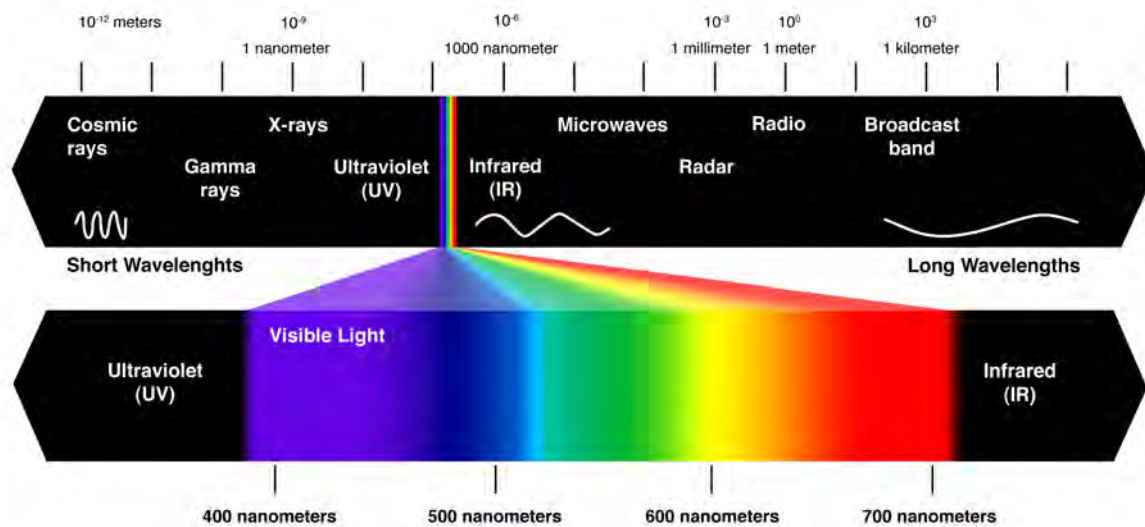
Figure 2.3: Electromagnetic spectrum. Human visible spectrum is amplified for easier appreciation.

An image is not only formed by primary light sources such as the Sun, lamps and bulbs, among others. Light strikes other objects and these objects will reflect a part of this light while reflecting another, which will make the reflected light have different components that the primary light source [36]. The reflected light makes an object look a certain color; blue objects reflect blue light, red objects reflect red light and so on.

2.1.3 Object properties

When light hits an object, some light wavelengths are absorbed and some other wavelengths are reflected by the object depending on its properties. The reflected wavelengths determine the perceived color of the object. Let $I(\lambda)$ be the light source illuminating the scene and $S(\lambda)$ be the reflected light by the object; thus the light captured by a camera sensor is the product of $S(\lambda) \cdot I(\lambda)$, as shown in Figure 2.4.
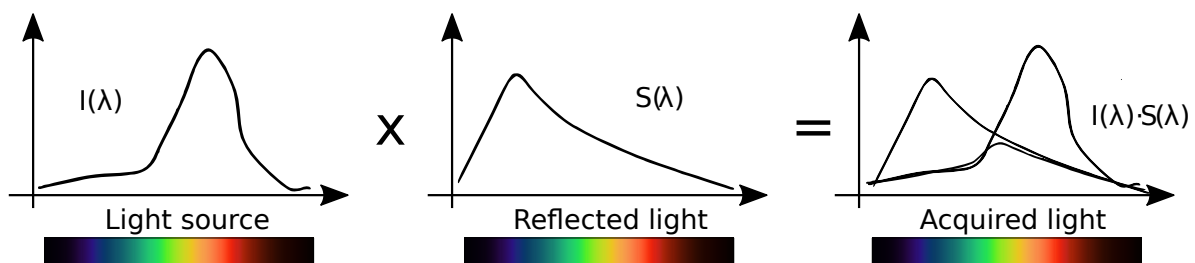
Figure 2.4: Light acquired by a camera sensor.

2.1.4 Camera sensitivity

Typical digital cameras have three different color sensors; one corresponding to red, one corresponding to green and one corresponding to blue. The acronym RGB comes from each of these colors' initials and is the most frequent color space used for the creation of digital images. These sensors are based on silicon and use specific filters to only let thought certain light wavelengths. The final values assigned to each pixel are the result of the integration of the light perceived by each sensor, like so:

$$(R,G,B) = \left(\int I(\lambda)S(\lambda)R(\lambda)d\lambda, \int I(\lambda)S(\lambda)G(\lambda)d\lambda, \int I(\lambda)S(\lambda)B(\lambda)d\lambda \right)$$

Where $R(\lambda)$, $G(\lambda)$ y $B(\lambda)$ are the red, green and blue sensors of the camera.
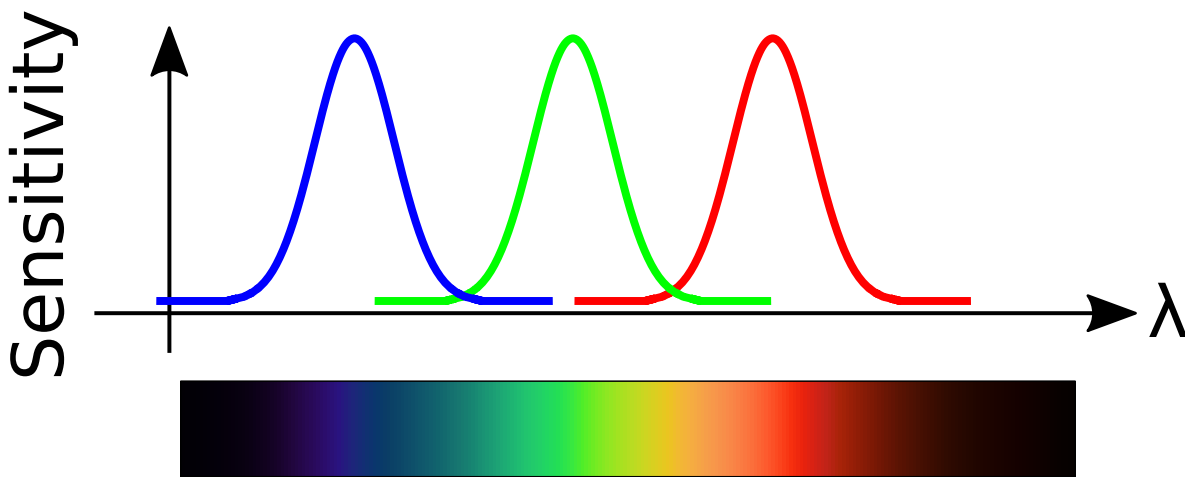
Figure 2.5: Typical camera sensors and their sensitivity.

Most digital images have a depth of 8 bit per channel. In binary and grayscale digital images, one channel is used to store the pixel value; in color images three channels are used to store each of the pixel's red, green and blue values. Having a bit depth of 8 means that each pixel in a channel can have total of $2^8 = 256$ different values, which are usually represented by the numbers 0 to 255. A value of 0 represents the lowest brightness in that channel, while a value of 255 represents full brightness. In the case of grayscale images, a value of 0 would be displayed as black and a value of 255 would be displayed as white. In the case of color images, a pixel with a value of $\{255,0,0\}$ would be displayed as red; $\{0,255,0\}$ would be displayed as green and $\{0,0,255\}$ would be displayed as blue; any combination in between is displayed using an additive color model.

2.2   Why three sensors?

The RGB color space is inspired by the human visual system, which has three cell types sensitive to color called cones. Cones are photosensitive cells located in the retina. In the case of human beings, there are three different types of cones and they are labeled **L**arge, **M**edium or **S**hort according to the wavelengths they are most sensitive to [36]. Cone distribution is not uniform; photobleaching results illustrate that S cones are randomly placed and appear much less frequently than the M and L cones. The ratio of M and L cones varies greatly among different people with regular vision (e.g. values of 75.8 % L with 20 % versus 50.6 % L with 44.2 % M in two male subjects) [57].

Sensitivity to different wavelengths is due to the presence of pigments composed of an apoenzyme called opsin: [37, 58]

- Erythropsin is sensitive to a range of 500 nm to 700 nm, with a maximum located between 564 nm to 580 nm (red color) [38, 39].
- Chloropsin is sensitive to a range of 450 nm to 630 nm, with a maximum located between 534 nm to 550 nm (green color) [38, 39].
- Lastly, cyanopsin is sensitive to a range of 400 nm to 500 nm, with a maximum located between 420 nm to 440 nm (blue color) [38, 39].
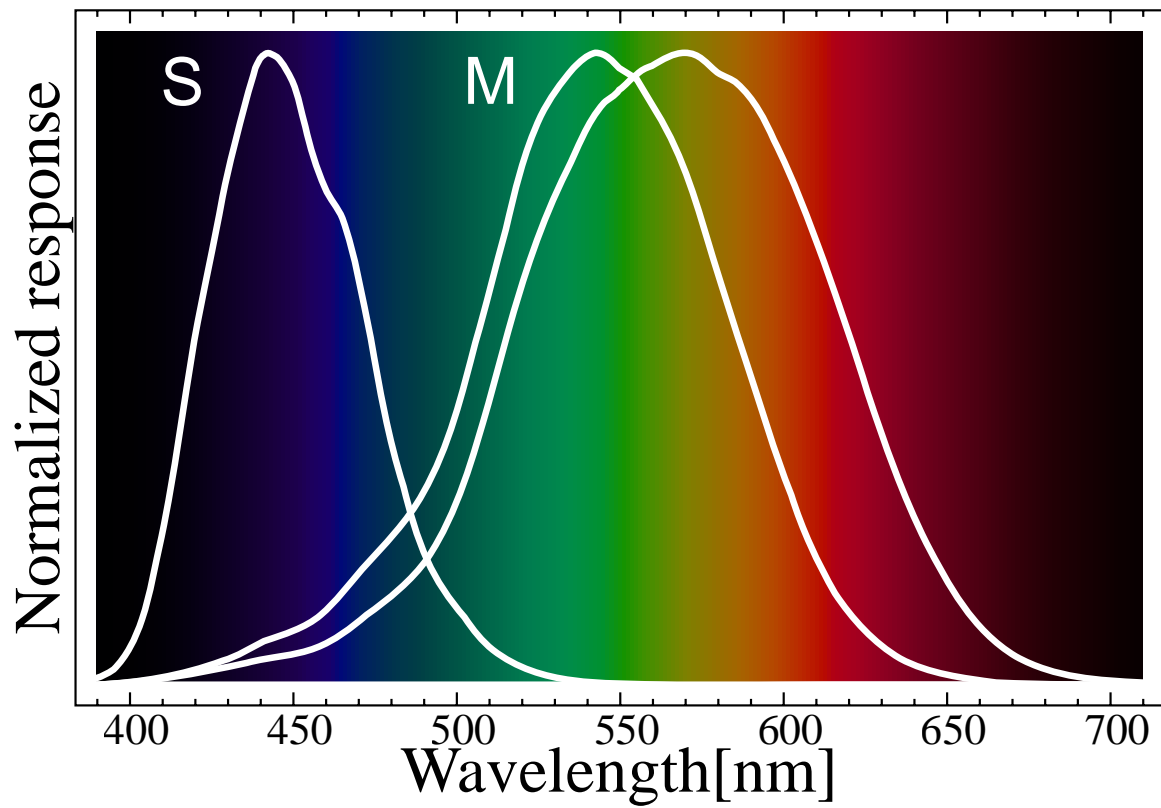
Figure 2.6: Normalized cone response to the visible light spectrum.

3. Filtering and morphological operations

Contents

3.1   Image filtering

Filtering in digital image processing is a diverse topic which covers several areas. In this work, the filtering done is intended primarily to reduce noise in the working image. There are many types of noise, such as salt-and-pepper and Gaussian noise. These types of noise can be simulated in order to evaluate which algorithm may perform better in reducing a particular type of noise.

3.1.1   Mean filtering

This filter works by assigning each pixel a value equal to the mean of the pixels value around it in a neighborhood of $N \times M$[1]. It has an effect of smoothing the picture. It may be used to reduce noise in an image, but the median filter in section 3.1.2 usually does a better job [40]. Mathematically, this can be represented like so:

$$\mu(x,y) = \frac{1}{N} \sum_{(s,t) \in S_{(x,y)}} I(s,t) \tag{3.1}$$

Where $\mu(x,y)$ is the new pixel value of the image, N is the number of pixels in the neighborhood, and $S_{(x,y)}$ is the neighborhood of the pixel at (x,y).

It is important to note that while mean filtering reduces noise somewhat, high-frequency image detail is lost (image edges), which will make the image look out of focus. These effects are proportional to the used kernel size.

Mean filtering is not directly used in this work, but it is mentioned to better introduce Gaussian filtering.

MATLAB code implementation

```matlab
1  % img = imread('lena.jpg');   % Read image
2  img = imread('http://www.ece.rice.edu/~wakin/images/lenaTest3.jpg');
3  % img = rgb2gray(img);   % Convert image to grayscale
4
5  % Add speckle noise to original image
6  noisyImg = imnoise(img,'speckle');
7
8  % Neighborhood size
9  neighborhoodSize = 5;
10
11 % Apply mean filtering
```

---

[1]This is usually called the **kernel size**.

```
12  filteredImg = imfilter(noisyImg, ...
        ones(neighborhoodSize)/(neighborhoodSize^2));
13
14  % Display noisy image and filtered image
15  imshowpair(noisyImg,filteredImg,'montage')
```


(a) Image with speckle noise.

(b) Pixel inspection of image with speckle noise starting at row = 254, colum = 253.
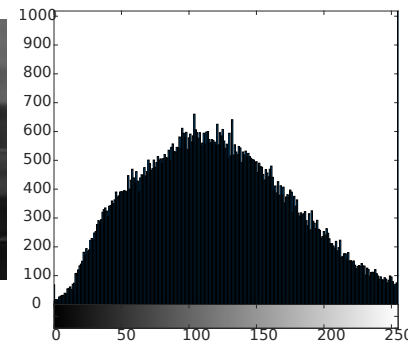
(c) Histogram of image with speckle noise.

(d) Image after mean filtering.
Kernel size = 5×5.

(e) Pixel inspection of image region after mean filtering starting at row = 254, colum = 253.
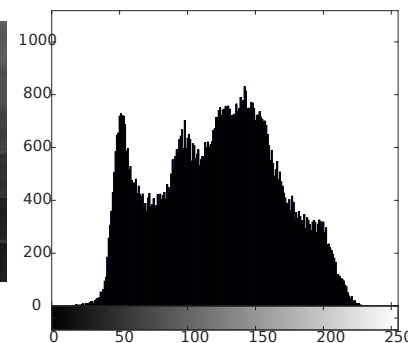
(f) Histogram of filtered image.

Figure 3.1: Mean filtering of an image.

3.1.2 Median filtering

The median filter is a very commonly used filter in digital image processing. The filter takes the pixel values and sorts them by ascending pixel value as is the case of rank value filters such as this one [41]. The median filter selects the medium or median value and assigns it to the current pixel. This filter is better for preserving image edges because outliers do not

modify the final pixel value, which is the case of the mean filter. [40]. Median filtering is exceptionally good at reducing salt-and-pepper noise [59].

Median filtering is used in this work as a morphological operation in binary images. The filter helps to increase uniformity of the image by increasing the number of the pixels that are in the majority in a region of the image whether they are black or white pixels.

MATLAB code implementation

```matlab
% Read image
img = imread('CoinsWithSaltAndPepper.jpg');

% Convert image to grayscale
img = rgb2gray(img);

% Apply median filtering with a kernel size of 5x5
filteredImg = medfilt2(img,[5 5]);

% Display noisy image and filtered image
imshowpair(img,filteredImg,'montage')
```
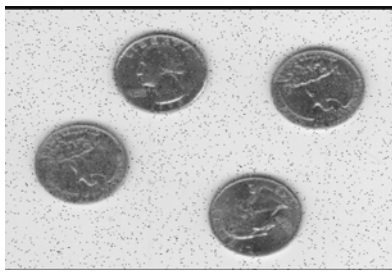
(a) Image with salt-and-pepper noise.

(b) Pixel inspection of image region with salt-and-pepper noise starting at row $= 313$, colum $= 530$.
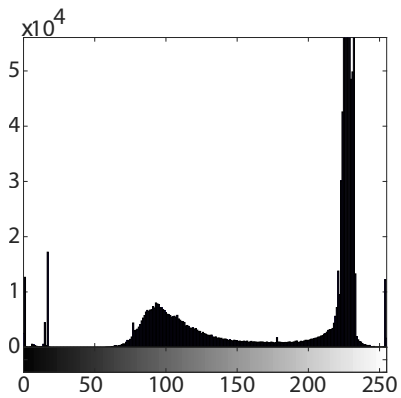
(c) Histogram of image with salt-and-pepper noise.

(d) Image after median filtering. Kernel size $= 5 \times 5$.

(e) Pixel inspection of image region after median filtering starting at row $= 313$, colum $= 530$.
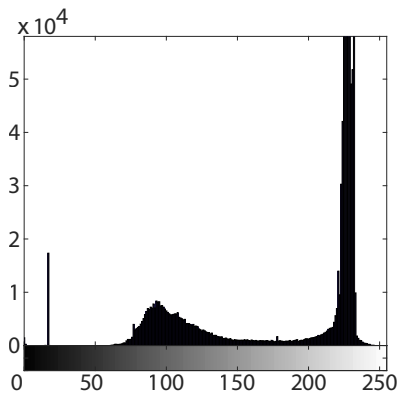
(f) Histogram of filtered image.

Figure 3.2: Median filtering of an image.

3.1.3 Gaussian filtering

A Gaussian filter is similar to a mean filter in the sense that both smooth a picture and degrade high-frequency details. A Gaussian filter takes after a continuous 2D Gaussian function, defined as:

$$P(x,y) = \frac{1}{2\pi\sigma^2}e^{-\left(x^2+y^2\right)\big/2\sigma^2} \tag{3.2}$$

A discrete-valued 2D Gaussian function takes two arguments:

1. The kernel size
2. The standard deviation σ of the Gaussian function

The main difference between a mean filter and a Gaussian filter is that the degree of smoothing if controlled by σ in a Gaussian filter, not by the kernel size like in a mean filter [40, 42]. A Gaussian filter is useful in frequency-domain analysis because its Fourier transform is also a Gaussian function [40, 42].

Gaussian filtering in this work is used to smooth out the image before applying superpixel clustering, resulting in smoother superpixel clusters.

MATLAB code implementation

```matlab
% Read image
img = imread('Rue_de_la_Forge.jpg');

% Creation of Gaussian filter with a kernel size of [5 5] and
% standard deviation of 2
gaussianFilter = fspecial('gaussian',[5 5],2);

% Apply Gaussian filtering
filteredImg = imfilter(img,gaussianFilter);

% Display noisy image and filtered image
imshowpair(img,filteredImg,'montage')
```

(a) Image with Gaussian noise; kernel size $= 5 \times 5$, $\sigma = 2$.

(b) Pixel inspection of image region with Gaussian noise starting at row $= 522$, colum $= 817$.

(c) Histogram of image with Gaussian noise.

(d) Image after Gaussian filtering. Kernel size $= 5 \times 5, \sigma = 2$.

(e) Pixel inspection of image region after Gaussian filtering starting at row $= 522$, colum $= 817$.

(f) Histogram of filtered image.
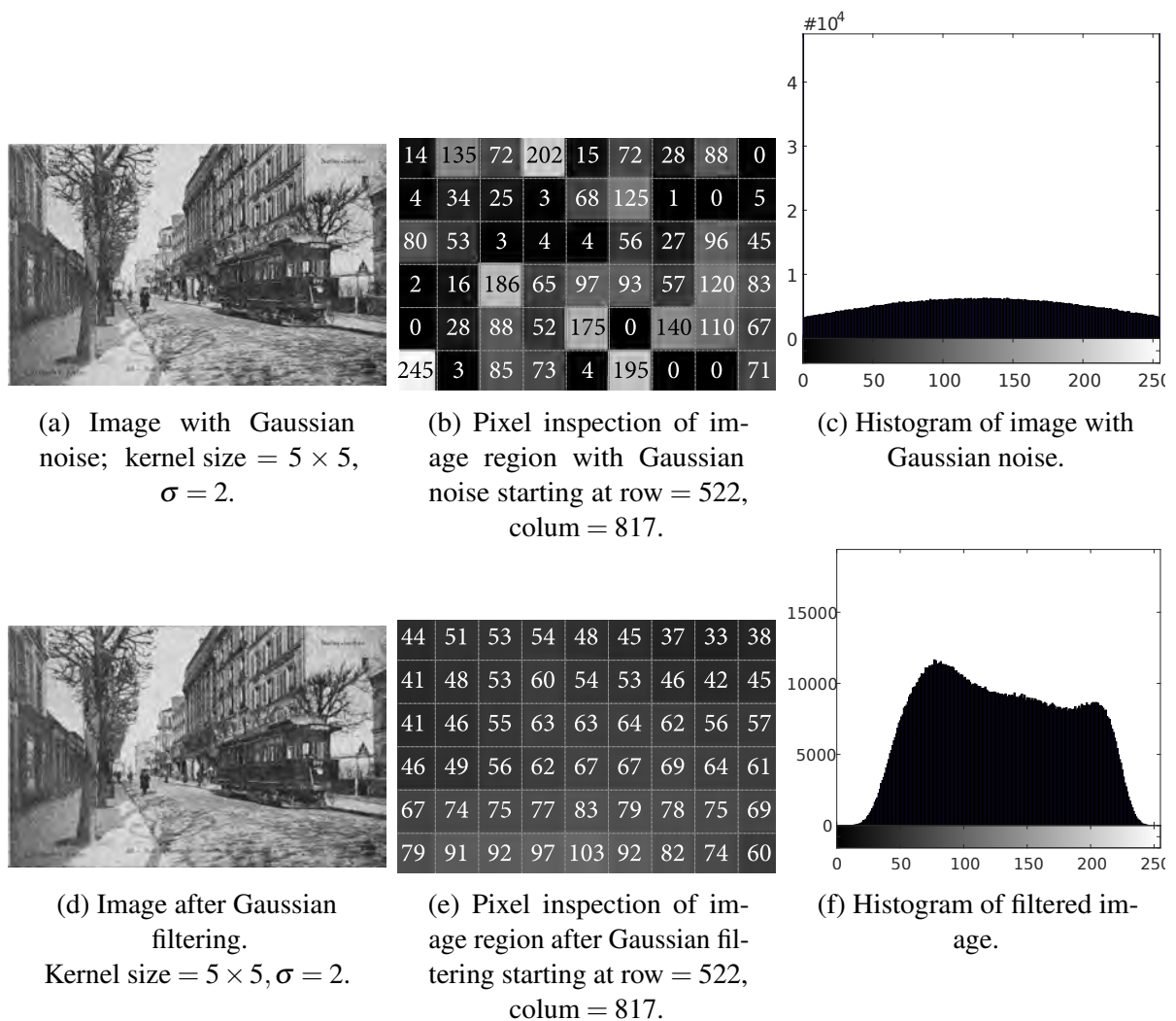
Figure 3.3: Gaussian filtering of an image.

3.2 Morphological operations

Morphological operations can detect or quantify geometrical properties in signal (such as a digital image) in a similar way that human perception may do it. Its algebraic system of operators of mathematical morphology and their compositions are capable of decomposing a complex shape into smaller meaningful parts and, therefore, simplify image data processing [0].

Morphological operations have been used for multiple purposes, such as: interactive 3D segmentation of MRI and CT volumes [60], generation of volumes of left ventricle chambers from cardiac CT volumetric images [61], Frontal-view face detection and facial feature

extraction [62], license plate detection in complex scenes [0], separation of characters from maps/graphics [0], automatic brain extraction methods from T1 MRI [63], detection and geometric modeling of molecular surfaces and cavities [64], film restoration [43], among others.

In this work, morphological operations are used to clean results from other techniques by preserving or removing some geometric components in the digital image.

3.2.1   Binary images

Morphological operations are usually done on binary images, which are images in which each pixel takes a discrete value of either 0 or 1 [44] and are displayed as black (for logical values equal to 0; also called `false`) and white (for logical values equal to 1; also called `true`) images. The output of these operations is another binary image.

In binary images, pixels with a logical value of 1 are usually referred as the foreground pixels, while pixels with a logical value of 0 are usually referred as the background pixels.

3.2.2   Thresholding

Binary images are often created from grayscale images by thresholding. There are two different general methods of thresholding: global and adaptive thresholding [42].

Definition 3.2.1   Given a grayscale image G with x, y elements, then a binary image B may be created from G using thresholding by establishing a value T. All values of G that are equal or above T will be set to logical 1, while all values of G below T will be set to logical 0. This can be mathematically represented like so [45]:

$$B(x,y) = \begin{cases} 1, & \text{if } G(x,y) \geq T \\ 0, & \text{if } G(x,y) < T \end{cases} \tag{3.3}$$

(a) Original image.   (b) Thresholded im-   (c) Thresholded im-   (d) Thresholded im-
                           age. $T = 51$.   age. $T = 153$.   age. $T = 204$.
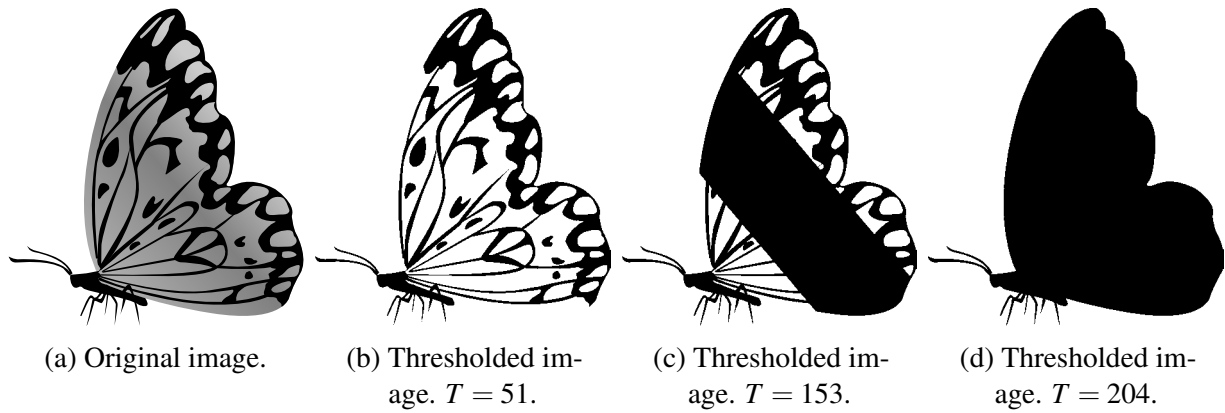
Figure 3.4: Binarization of images with different threshold magnitudes.

As shown in Figure 3.4, the selected threshold plays a major part in the creation of a binary image. Selecting an appropriate threshold, thus, is of utmost importance.

Otsu's method

Otsu's Method is an algorithm for automatically selecting the most appropriate threshold T for an image. Named after Nobuyuki Otsu, the algorithm assumes the following:

1. The image is made of two types of pixels: foreground pixels and background pixels (binary image).
2. The pixels belonging to either of the two type of pixels are very similar to each other.

The algorithm computes the threshold level which optimizes the separation between these two groups of pixels. Optimization is understood as the minimization of the intra-class variance between the two groups, defined as the weighted sum of variances between the two groups (equation 3.4), where $\omega_{0,1}$ are the probabilities of the two groups and $\sigma_{0,1}^2$ are the variances of the two groups [65].

$$\sigma_w^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t) \tag{3.4}$$

Class probabilities $\omega_{0,1}$ are computed by:

$$\omega_0(t) = \sum_{i=0}^{t-1} p(i) \qquad \omega_1(t) = \sum_{i=t}^{L-1} p(i) \tag{3.5}$$

Minimizing the intra-class variance is the same as maximizing inter-class variance [65].

Expressing this in terms of class probabilities ω and class means μ yields:

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 \qquad (3.6)$$

$$= \omega_0(t)\omega_1(t)\left[\mu_0(t) - \mu_1(t)\right]^2 \qquad (3.7)$$

The class mean $\mu_{0,1,T}(t)$ is:

$$\mu_0(t) = \sum_{i=0}^{t-1} i\frac{p(i)}{\omega_0} \qquad (3.8)$$

$$\mu_1(t) = \sum_{i=t}^{L-1} i\frac{p(i)}{\omega_1} \qquad (3.9)$$

$$\mu_T = \sum_{i=0}^{L-1} ip(i) \qquad (3.10)$$

An implementation of this algorithm can be consulted in the following MATLAB code. This function doesn't include error-checking. MATLAB's built-in function for computing the threshold T using Otsu's method is called `graythresh`, after which you may run `imbinarize` with the computed threshold. These functions include error-checking, so use of these functions is recommended instead.

Otsu's method is used in this work to create a binary mask that contains only teeth information with no background from the pantomograms.

```matlab
function  [threshold_otsu] = Thresholding_Otsu(image)
%    Inputs:
%          nbins: number of gray levels
%          counts: image histogram
%          p: probability of each gray level
%          omega: cumulative sum of probabilites
%          mu: classes mean
%          mu_t: total mean level of the original image
%          sigma_b_squared: intra-class variance

nbins = 256;
counts = imhist(image,nbins);     % image histogram
p = counts / sum(counts);         % probability of each gray level
omega = cumsum(p);
mu = cumsum(p .* (1:nbins)');
mu_t = mu(end);

sigma_b_squared = (mu_t * omega - mu).^2 ./ (omega .* (1 - omega));

maxval = max(sigma_b_squared);
threshold_otsu = mean(find(sigma_b_squared == maxval));
```

```
22   end
```


(a) Original image.

(b) Thresholded image using Otsu. Computed threshold $= 110$.
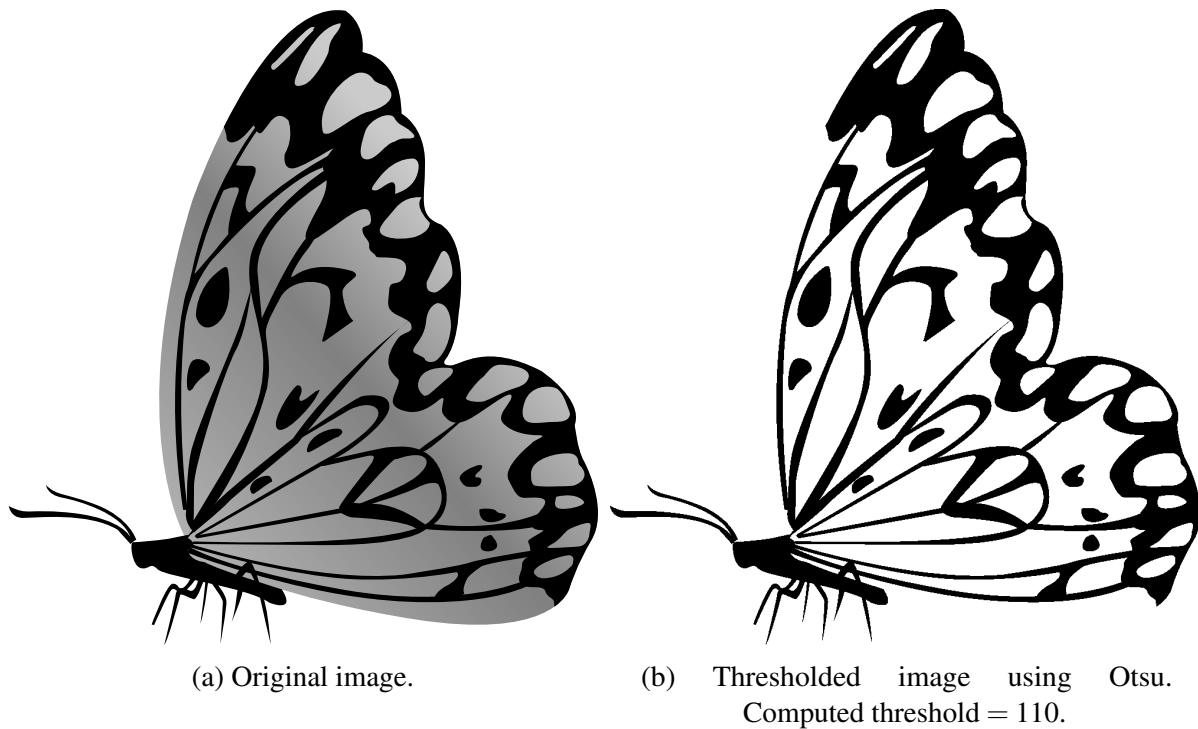
Figure 3.5: Binarization of an image using Otsu's method.

3.2.3 Structuring elements

Structuring elements are to morphological operations what kernels are to filters; they describe a neighborhood of logical ones and zeros which will affect the pixels in an image. These structuring elements are usually shaped like a diamond, disk, line, or any other polygon [46]. A structuring element basically controls which pixels are affected by a morphological operation [47].

Structuring elements are used in this work to affect only some specific pixel patterns with morphological operations.

(a) Structuring element of type `diamond`, radius $= 3$.

0	0	0	1	0	0	0
0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1	1	0	0
0	0	0	1	0	0	0

(b) Structuring element of type `disk`, radius $= 5$.

0	0	1	1	1	1	1	0	0
0	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	0	0

(c) Structuring element of type `line`, length $= 10$, angle $= 45°$.

0	0	0	0	0	0	1
0	0	0	0	0	1	0
0	0	0	0	1	0	0
0	0	0	1	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
1	0	0	0	0	0	0

(d) Structuring element of type `octagon`, radius $= 3$.

0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1	1	0	0

(e) Structuring element of type `rectangle`, width $= 5$, height $= 4$.

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

(f) Structuring element of type `square`, size $= 5$.

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Figure 3.6: Different structuring elements examples.

3.2.4 Erosion and dilation

Dilation and erosion are the basic morphological operations [48]. Any other morphological operation may be defined in terms of these primitive operators [40]. These operations are used to clean up the image after doing some operations.

Definition 3.2.2 Let A be a general binary image and B be an arbitrary structuring element. We may define erosion and dilation as follows:

Erosion The erosion of A by B is mathematically represented as $A \ominus B = \bigcap_{b \in B} A_{-b}$, where A_b is the translation of A by b. This means that in order to perform erosion, the anchor pixel of the structuring element (usually the center pixel) must be positioned on each

foreground pixel of the image. If **any** of the of foreground pixels of the structuring element is on top of a **background** pixel of the image, then the foreground pixel of the image is changed to a background pixel.

Dilation The dilation of A by B is mathematically represented as $A \oplus B = \bigcup_{b \in B} A_b$, where A_b is the translation of A by b. This means that in order to perform dilation, the anchor pixel of the structuring element (usually the center pixel) must be positioned on each **background** pixel of the image. If **any** of the of foreground pixels of the structuring element is on top of a **foreground** pixel of the image, then the background pixel of the image is changed to a foreground pixel.
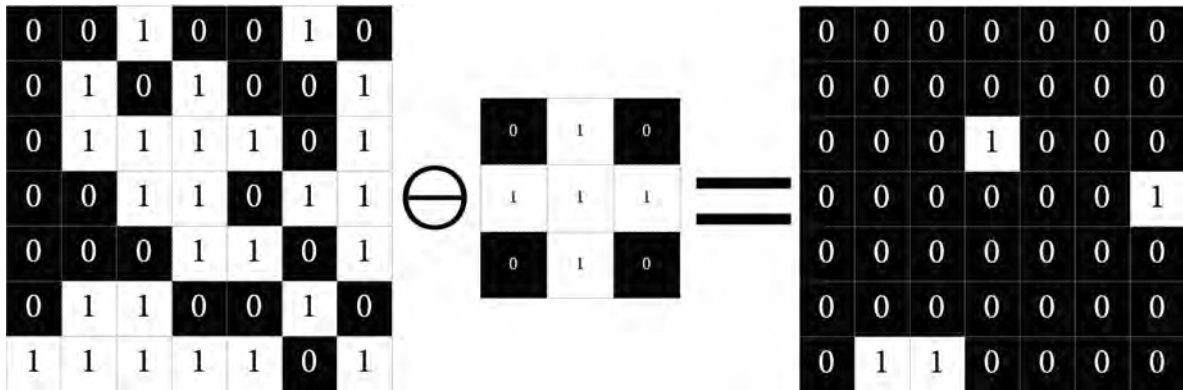
(a) Erosion morphological operation on a binary image.
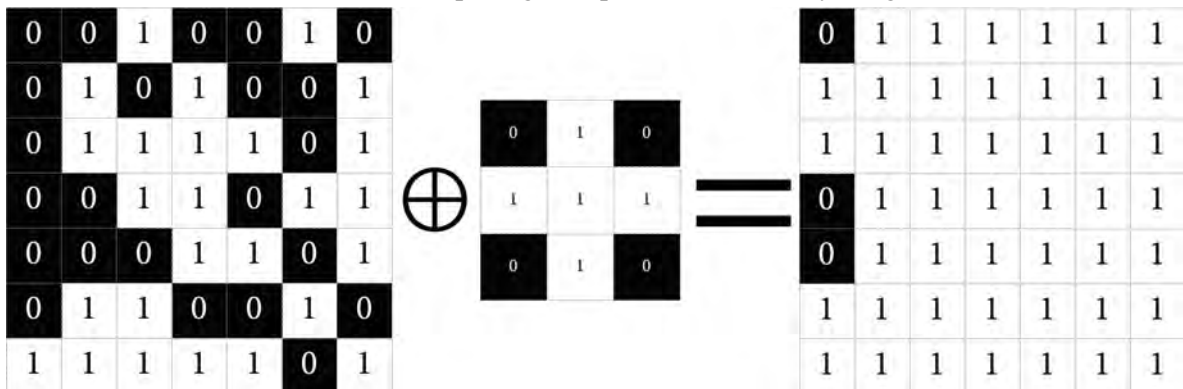
(b) Dilation morphological operation on a binary image.

Figure 3.7: Morphological erosion and dilation of an image.

3.2.5 Opening and closing

Definition 3.2.3 Opening is the name given to the morphological operation of **erosion** followed by **dilation** [40, 42]. Given an image A and a structuring element B, morphological

opening may be defined as [40, 42, 49]:

$$A \circ B = (A \ominus B) \oplus B \tag{3.11}$$

Morphological opening will remove all the pixels in the image that are too small to contain the structuring element. It smoothes outward bumps such as corners, breaks narrow sections and eliminates thin portions of the image [42, 50].

Definition 3.2.4 Closing is the name given to the morphological operation of **dilation** followed by **erosion** [40, 42]. Given an image A and a structuring element B, morphological closing may be defined as [40, 42, 49]:

$$A \bullet B = (A \oplus B) \ominus B \tag{3.12}$$

Morphological closing fills holes and gaps smaller than the structuring element [42, 50].

These two filters can be used similarly as a low-pass or high-pass filter. Morphological opening of an image is equivalent to a low-pass filter and its residue is equivalent to a high-pass filter [50]. Additionally, a band-pass filter may be created by opening an image with structuring elements of different diameter. Mathematically, this can be represented like so:

$$\text{Low-pass filter} = A \circ B \tag{3.13}$$
$$\text{High-pass filter} = A - (A \circ B) \tag{3.14}$$
$$\text{Band-pass filter} = (A \circ B_1) - (A \circ B_2) \tag{3.15}$$

(a) Original binary image.

(b) Structuring element used for opening.

(c) Result of morphological opening.

(d) Original binary image.

(e) Structuring element used for closing.

(f) Result of morphological closing.
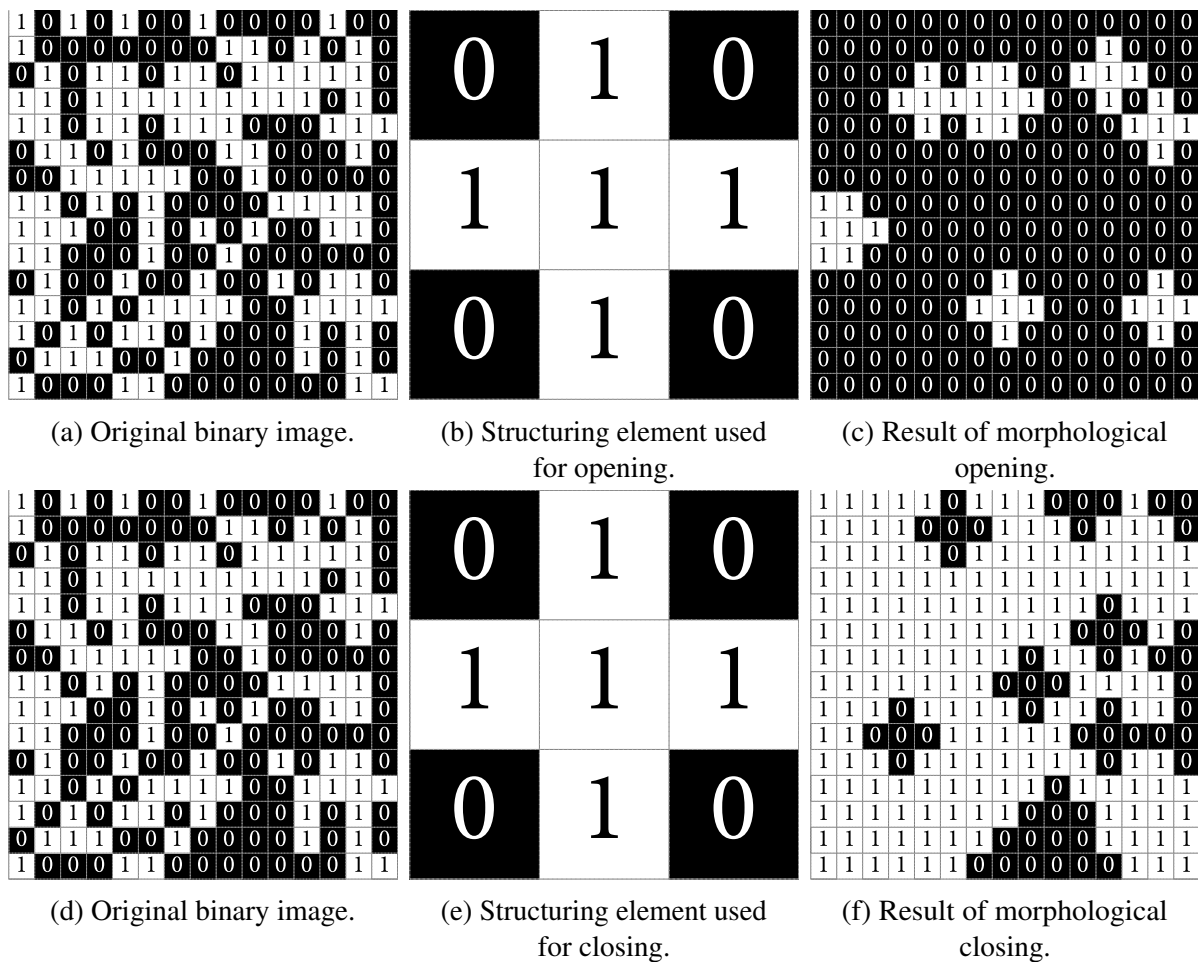
Figure 3.8: Morphological opening and closing of a binary image.

3.3 Watershed segmentation

Watershed segmentation is a popular tool to segment objects in images. This method works by considering a gray-scale image as a topographic surface, where each pixel is considered to be in a height equal to its gray value $I(x, y)$ [51, 52]; this means darker gray tones represent lower zones while brighter gray tones represent higher zones. Watershed segmentation is used to separate teeth that are overlapping each other after some initial segmentation.

(a) Original gray scale image.   (b) Topographic representation of the image.
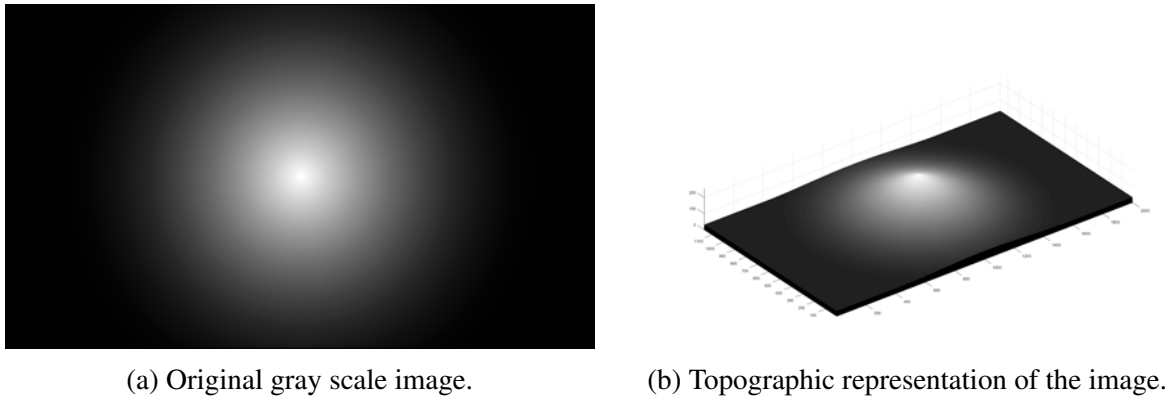
Figure 3.9: Gray scale image and its topographic representation.

Watershed segmentation stars by "flooding" the image, as if heavy rain was falling. This creates catchment basins where the water is accumulated. These catchment basins are separated by watersheds. These catchments basins describe the area of an object and the watersheds represent the outline of the object. An illustrated example of this process can be seen in Figure 3.10.
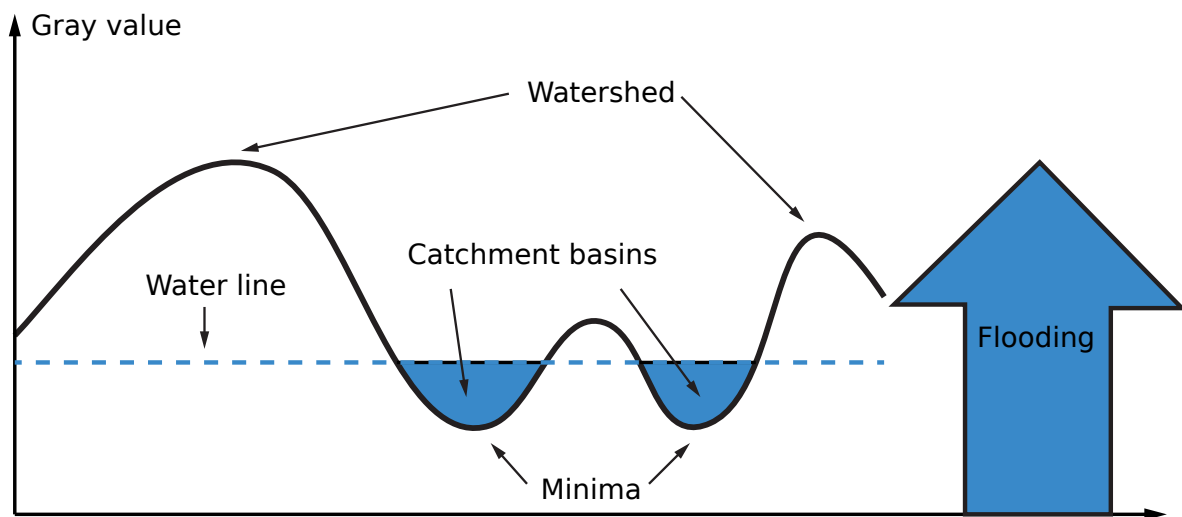
Figure 3.10: Flooding process from the watershed transform.

Due to local irregularities, watershed segmentation may cause oversegmentation of an image, so a height threshold may be incorporated in order to reduce this oversegmentation [53].

3.3.1 Distance transform

Watershed segmentation is commonly used with the distance transform. The distance transform of an image is defined by Gonzalez and Woods as "the distance from every pixel to the nearest nonzero-valued pixel". An example of this can be seen in Figure 3.11.

1	1	0	0	0
1	1	0	0	0
0	0	0	0	0
0	0	0	0	0
0	1	1	1	0

(a) Original pixel values of an image.

0.00	0.00	1.00	2.00	3.00
0.00	0.00	1.00	2.00	3.00
1.00	1.00	1.41	2.00	2.24
1.41	1.00	1.00	1.00	1.41
1.00	0.00	0.00	0.00	1.00

(b) Pixel values of an image after distance transform.

Figure 3.11: Pixel values of an image before and after distance transform.

The distance transform is useful when used alongside the watershed segmentation in cases where two objects of interest are overlapping each other (Figure 3.12a). However, the method does not separate the two full objects separately; instead, it divides them according to their catchment basins (Figure 3.12b).
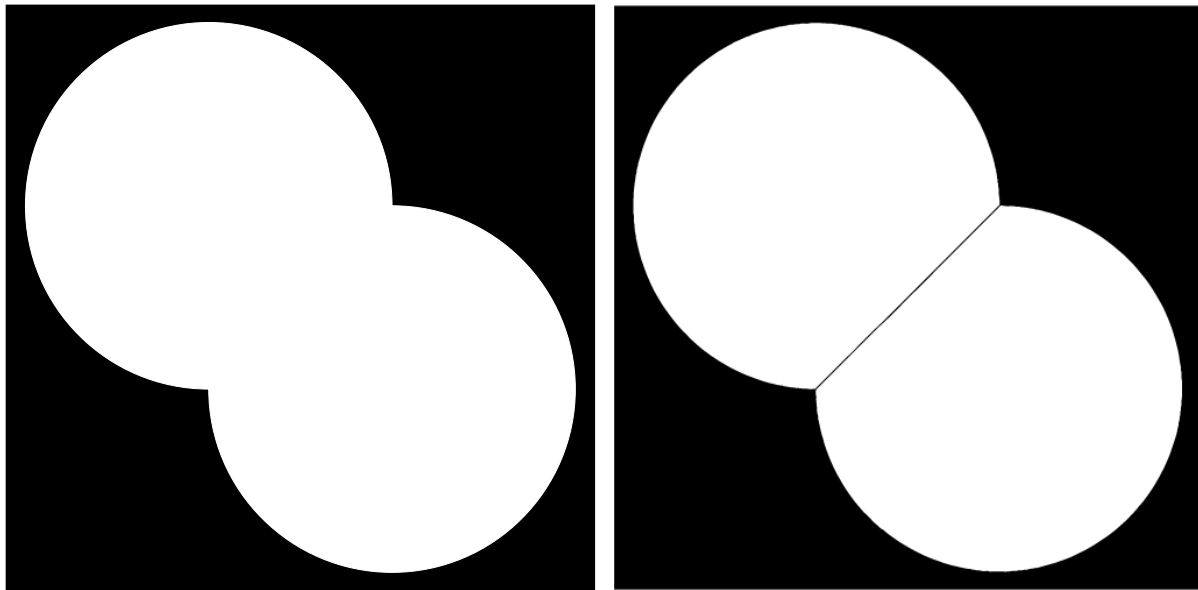
(a) Original image.

(b) Original image using distance transform and watershed segmentation.

Figure 3.12: Watershed segmentation used to separate two objects.

The distance transform was applied to the complement of the original image (seen in fig 3.13b) because the distance transform of the original picture (Figure 3.13a) is not very useful due to having only one catchment basin spanning the entire image. The basic steps for the process may be described as follows [52]:

1. Convert the original image to a binary image if this hasn't been done already
2. Complement binary image (Figure 3.13b)
3. Compute distance transform of inverted binary image (Figure 3.13c)
4. Negate computed distance transform (Figure 3.13d)
5. Compute watershed transform for negated distance transform
6. Set to zero in the original image the elements where the catchments basins collide in the label matrix.
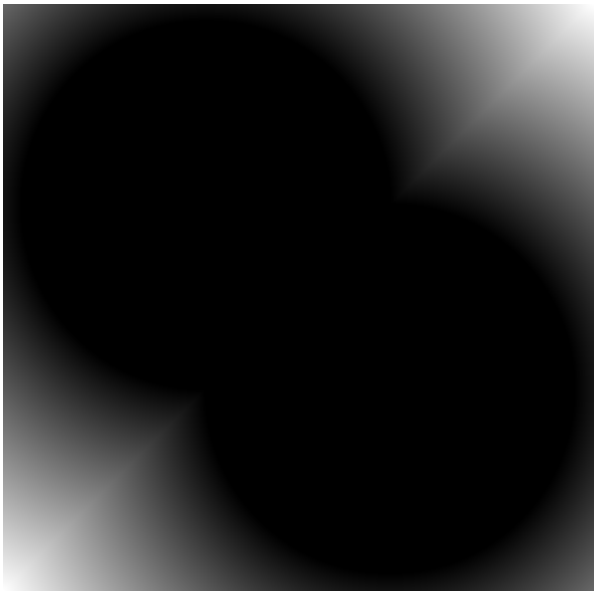
(a) Distance transform of original picture (Figure 3.12a).
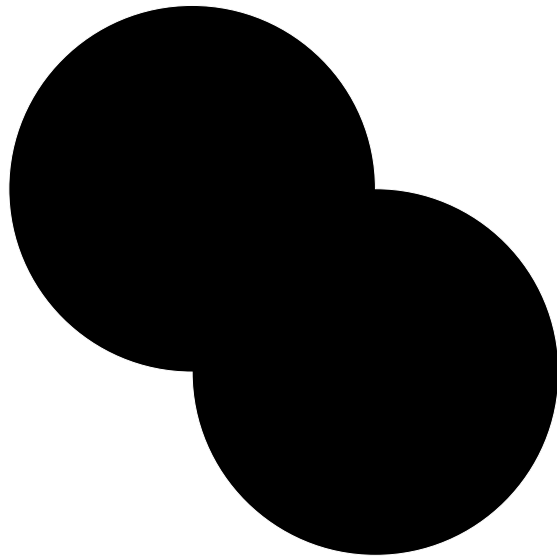
(b) Inverted original image.
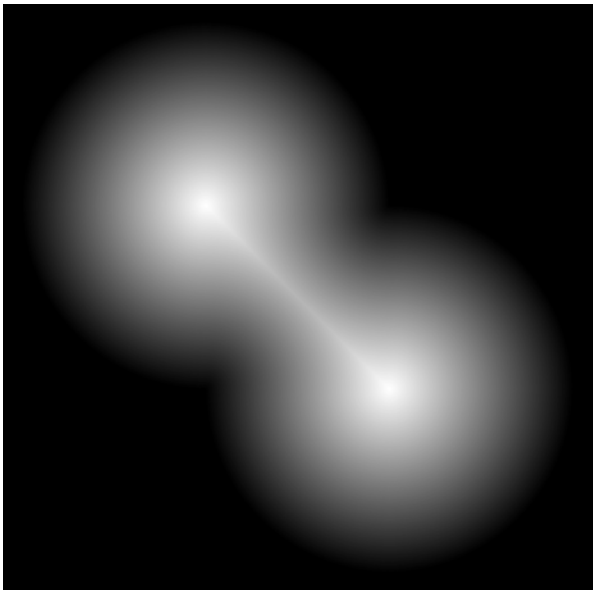
(c) Distance transform of inverted image.

(d) Negated image of Figure 3.13c.

Figure 3.13: Distance transform steps to generate an appropriate image for watershed segmentation.

Having computed the negated distance transform of the inverted image (Figure 3.13d), the MATLAB function `watershed` may be used to acquire a label matrix of the catchments basins. The catchment basins can be displayed using the `label2rgb` function in MATLAB.
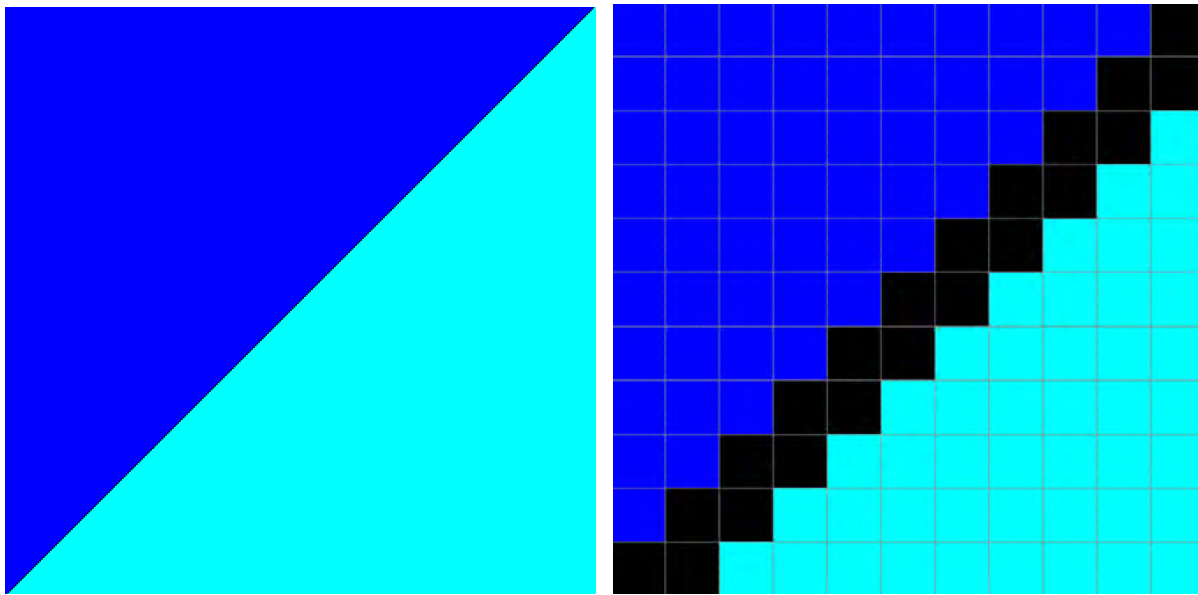
(a) Catchments basins formed by watershed transform.

(b) Zoomed in catchment basins.

Figure 3.14: Label matrix of computed watershed transform.

Figure 3.14b shows that the pixels where the catchment basins join are assigned a logical value of 0. Assigning this value to the original image creates the final image seen in Figure 3.12b. This can be done in MATLAB with the following code `img(L == 0)= 0;`, where `img` is the original image and `L` is the label matrix computed by the `watershed` function.

The full MATLAB code for this example is shown below.

```matlab
% Read image and binarize it
img = imread('CirclesOverlapping.png');
img = rgb2gray(img);
img = imbinarize(img);


imgDistTrans = bwdist(img);   % Distance transform of original picture
DistTransOfInv = bwdist(~img); % Distance transform of inverted picture
NegDistTransOfInv = -DistTransOfInv; % Negated transform of inv image

% Compute label matrix
L = watershed(NegDistTransOfInv);

% Set catchments basins limits to zero effectively separating the ...
    objects
finalImg = img;
finalImg(L == 0) = 0;
```

```
17
18  imshow(finalImg)
```

3.4 Superpixels

Superpixels are a recent development in digital image processing with their most modern variant appearing in 2010 [0]. Superpixels are, essentially, a controlled segmentation, where the metric of control might be a selected intensity, texture similarity, boundary preservation, or number of clusters created [0].

In this work, simple linear iterative clustering (SLIC) superpixels [0] (compact) will be used to make the regions in the radiographs less noisy and more uniform. This method was selected due to its speed compared to other methods and because SLIC superpixels are easy to use since the algorithm requires only one parameter: the desired number of superpixels. The methods compared are: "Efficient graph-based image segmentation" (GS04) [66], "Guiding model search using segmentation" NC05 [0], "Superpixel lattices" SL08 [0], "Watersheds in digital spaces: an efficient algorithm based on immersion simulations" WS91 [67], "Mean shift: A robust approach toward feature space analysis" MS02 [68], "Turbopixels: Fast superpixels using geometric flows" TP09 [69] and "Quick shift and kernel methods for mode seeking" QS08 [0].

Properties	Graph-based			Gradient-ascent-based				
	GS04	NC05	SL08	WS91	MS02	TP09	QS08	SLIC
Superpixel no. control	No	Yes	Yes	No	No	Yes	No	Yes
Compactness control	No	Yes	Yes	No	No	Yes	No	Yes
Complexity $O()$	$N \log N$	$N^{3/2}$	$N^2 \log N$	$N \log N$	N^2	N	dN^2	N
Parameters	2	1	3	1	3	1	2	1

Table 3.1: Superpixels algorithms comparison [0].

(a)   Superpixel segmentation with NC05 [0].

(b)   Superpixel segmentation with QS08 [0].

(c)   Superpixel segmentation with TP09 [69].

(d)   Superpixel segmentation with SLIC [0].
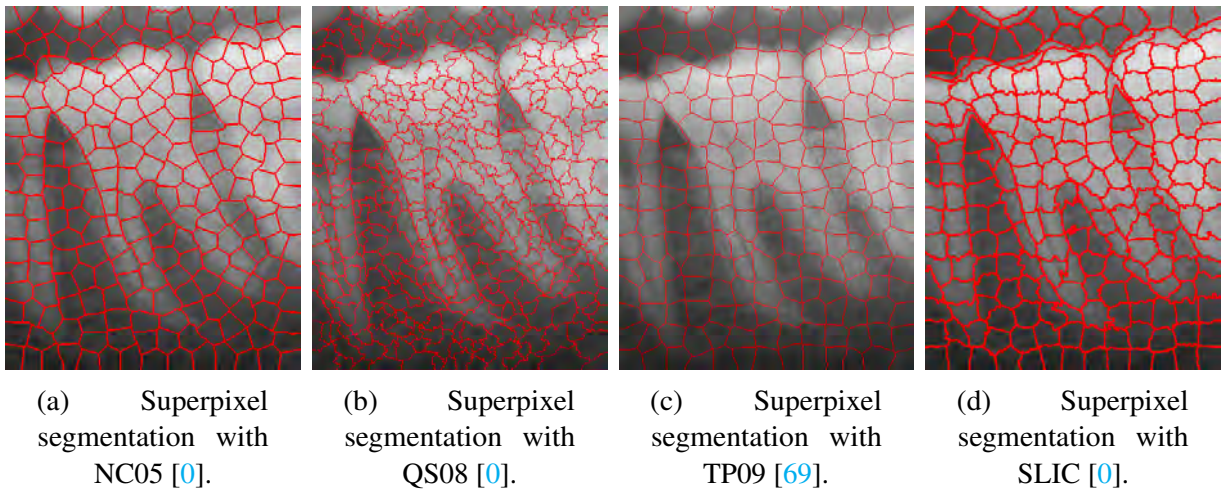
Figure 3.15: Comparison of superpixel segmentation with some of the methods compared in table 3.1.

Algorithm	Number of superpixels	Time taken
NC05	200	42.5303 s
QS08	N/A	1.7469 s
TP09	200	6.3923 s
SLIC	200	0.1053 s

Table 3.2: Computation time comparison between different methods for superpixel segmentation.

SLIC is a k-means clustering algorithm producing a number of compact superpixels by limiting the search space proportional to the desired superpixel size [0]. This is done in a 5-dimensional space [$labxy$], where lab is the CIELAB color space (used because of its perceptually-uniform distances) and xy is the pixel position [0].

The SLIC algorithm does not use the Euclidean distance as the metric for cluster assignation because the CIELAB color space's perceptually-uniform distances only holds true for small distances. If spatial pixel distances exceed this perceptual color distance limit, then they begin to outweigh pixel color similarities (resulting in superpixels that do not respect region boundaries, only proximity in the image plane), so the SLIC algorithm uses its own distance metric defined below:

$$d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2} \tag{3.16}$$

$$d_{xy} = \sqrt{(x_k - x_i)^2 + (y_i - y_k)^2} \tag{3.17}$$

$$D_s = d_{lab} + \frac{m}{S} d_{xy} \tag{3.18}$$

Where d_{lab} and d_{xy} are the *lab* and *xy* plane distances respectively. D_s is the sum of the *lab* distance and the *xy* plane distance normalized by the grid interval S. A k subindex refers to cluster centers components and an i subindex refers to the current pixel evaluated. A variable m is introduced to control the compactness of a superpixel; the greater the value of m, the more spatial proximity is emphasized and the more compact the cluster. This value m can be in the range 1 to 20 inclusive [0].

3.4.1 SLIC pseudocode

Algorithm 1 SLIC superpixel segmentation

1: Initialize cluster centers $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ by sampling pixels at regular grid steps S.
2: Perturb clusters centers in an $n \times n$ neighborhood to the lowest gradient position.
3: **repeat**
4:     **for** each cluster center C_k **do**
5:         Assign the best matching pixels from a $2S \times 2S$ square neighborhood around the cluster center according to the distance measure (equation 3.18).
6:     **end for**
7:     Compute new cluster centers and residual error E (distance between previous centers and recomputed centers)
8: **until** $E \leq threshold$
9: Enforce connectivity.

3.4.2 Fast local Laplacian filtering

Fast local Laplacian filtering is a very recent digital image processing approach first suggested by Paris, Hasinoff, and Kautz in 2011 [70] with technical improvements by Aubry et al. in 2014 [71]. It works by characterizing edges with a simple threshold on pixel values to allow to differentiate large-scale edges from small-scale ones. Its main parameter α controls the detail smoothing or detail enhancement done to the image. If $0 < \alpha < 1$, then the details are enhanced; if $\alpha > 1$, then the details will be smoothed.
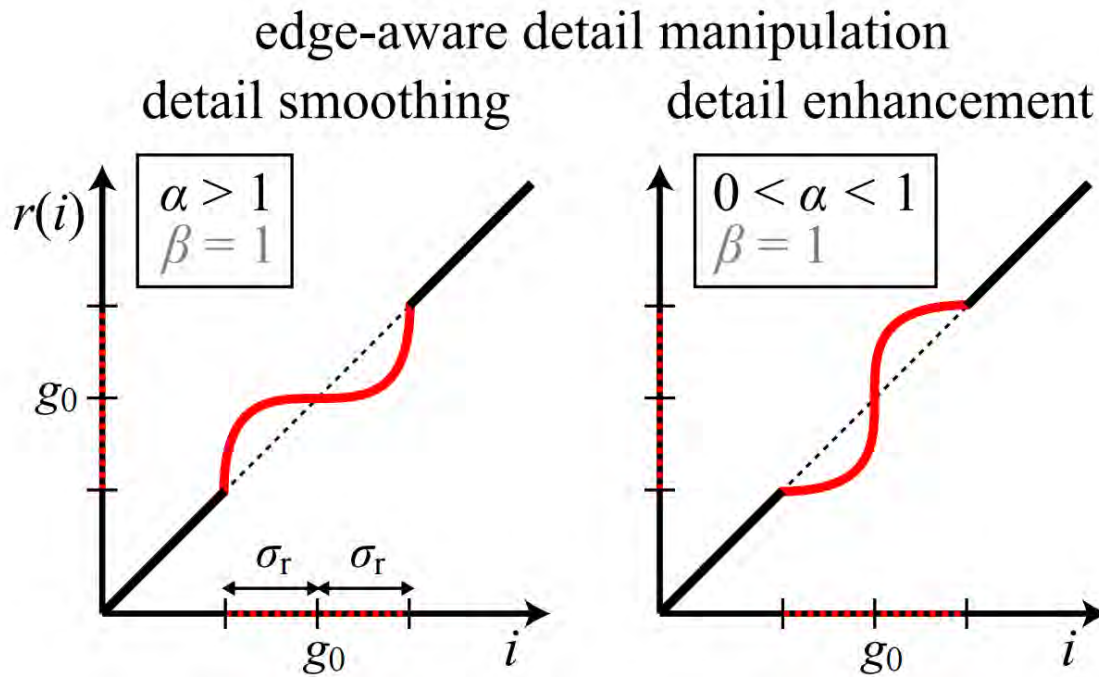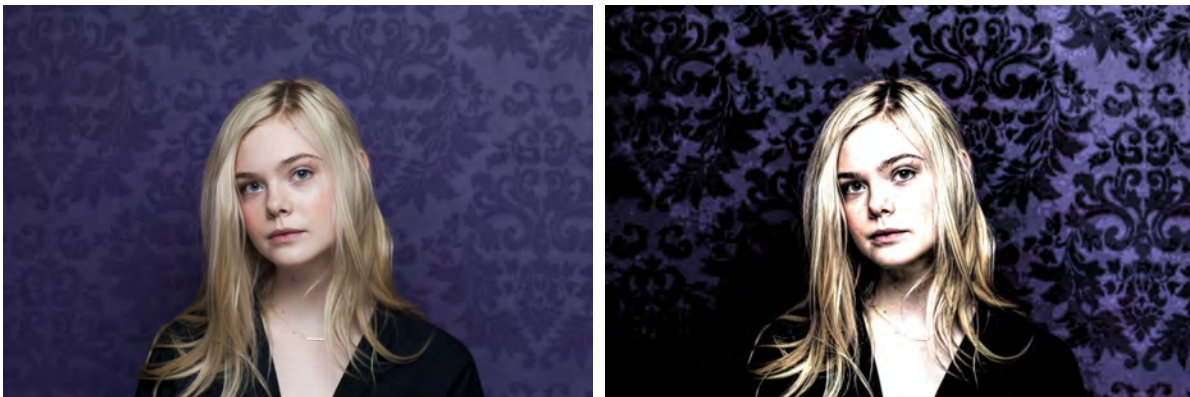
Figure 3.16: Point-wise functions for edge-aware manipulations described in "Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid."

(a) Original color image.

(b) Contrast-enhanced image with fast local Laplacian filtering. $\alpha = 0.5$.

Figure 3.17: Fast local Laplacian filtering example with color image.

Algorithm

Algorithm 2 Local Laplacian filtering $O(N \log N)$

---

  input: Image I, parameter σ_r, remapping function r
  output: Image I'
1: Compute input Gaussian pyramid $\{G[I]\}$
2: **for all** (x_0, y_0, ℓ_0) **do**
3:    $g_0 \leftarrow G_{\ell_0}(x_0, y_0)$
4:    Determine sub-region R_0 needed to evaluate $L_{\ell_0}(x_0, y_0)$
5:    Apply remapping function: $\tilde{R}_0 \leftarrow r_{g_0, \sigma_r}(R_0)$
6:    Compute sub-pyramid $\{L_{\ell_0}[\tilde{R}_0]\}$
7:    Update output pyramid: $L_{\ell_0}[I'](x_0, y_0) \leftarrow L_{\ell_0}[\tilde{R}_0](x_0, y_0)$
8: **end for**
9: Collapse output pyramid: $I' \leftarrow \text{collapse}(\{L_\ell[I']\})$

For a more detailed explanation of the pseudocode, see [70].

3.5   Active contours

  Active contours is a method that uses an initial binary mask which is guided by external constraint forces and influenced by image forces that pull it toward features such as lines and edges [72]. They lock onto nearby edges, localizing them accurately. In this work active contours are used to isolate a molar from the rest of the teeth.

  Figure 3.18 shows how a rectangular mask can be used to isolate objects after several iterations of active contours. Since active contours is a computationally expensive procedure, which is why it is best to avoid using it on high-resolution pictures. In this work active contours is used in cropped pictures instead of whole pantomograms.

(a) Original image of coins.

(b) Initial mask done manually.

(c) Initial mask after 75 iterations.

(d) Initial mask after 150 iterations.

(e) Initial mask after 225 iterations.

(f) Initial mask after 300 iterations.
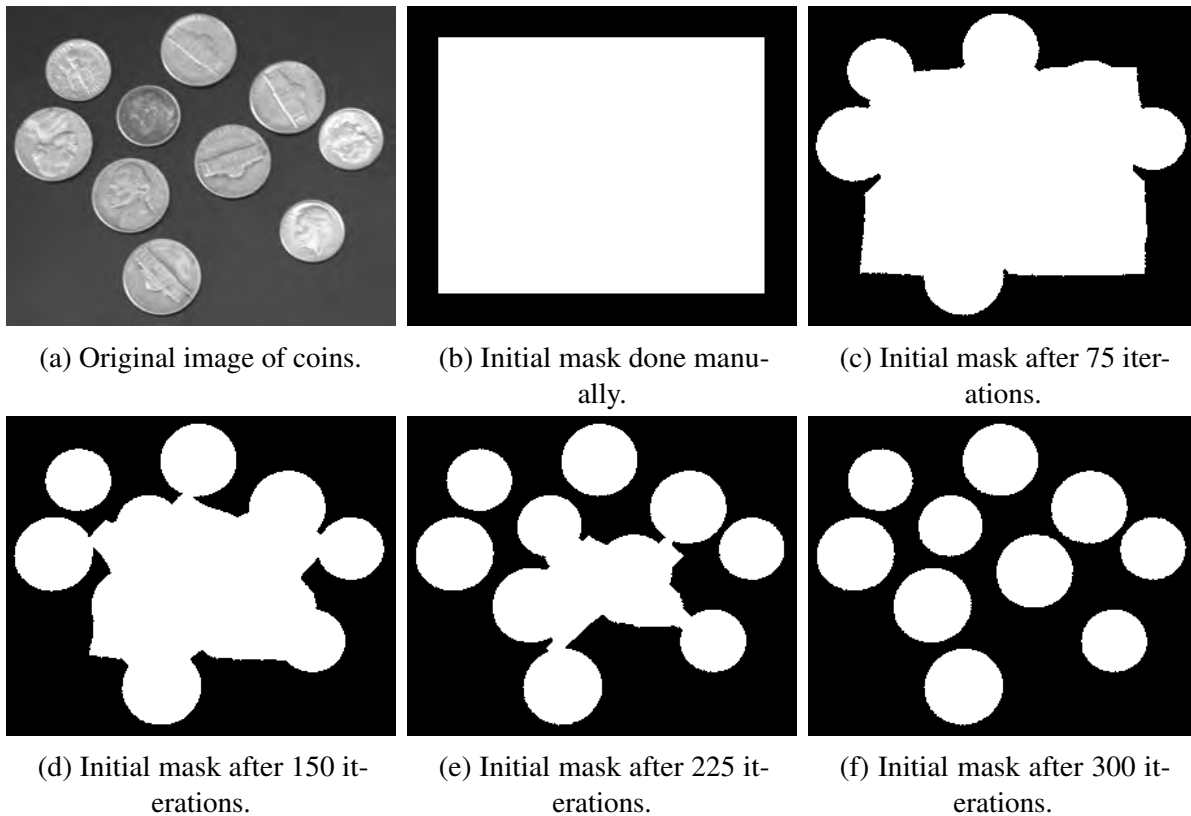
Figure 3.18: Active contours applied to a picture of some coins.

The active contour algorithm used in this work uses the Chan-Vese (algorithm 3) method for segmentation [73] and the Sparse-Field level-set method [2], similar to the method described in [75], for implementing active contour evolution.

Algorithm 3 Chan-Vese method for segmentation (check [0, 76] for an in-depth explanation).

1: Initialize φ
2: **for** $n = 1, 2, ...$ **do**
3:      Compute C_1 and C_2 as the region averages.
4:      Evolve φ with one semi-implicit time-step.
5:      **if** $\frac{\left\| \varphi^{n+1} - \varphi^n \right\|_2}{|\Omega|} \leq$ tol **then stop**
6:      **end if**
7:      (Optional) If n is divisible by N, reinitialize φ.
8: **end for**

---

[2]For a complete list of pseudocode and MATLAB source code, see [74]

```matlab
function phi = SimpleCVsegment(I,phi0,dt,numIts,epsilon)
% Perform a simplified version of Chan-Vese segmentation algorithm, as
% described in "Solving Chan-Vese..." paper by He & Osher

    function f = myFunc(t,y)
        f = PhiT(y,reshape(I,numel(I),1),epsilon);
    end

% Set up coefficients for 4th order Runge-Kutta method
A = [0 0 0 0; 1/2 0 0 0; 0 1/2 0 0; 0 0 1 0];
b = [1/6 1/3 1/3 1/6]';
c = [0 1/2 1/2 1]';

% Solve segmentation problem using a fixed number of steps of
[m,n] = size(phi0);
phi = reshape(phi0,m*n,1);
for i = 1:numIts
    phi = RKstep(0,phi,dt,@myFunc,A,b,c);
end

phi = reshape(phi,m,n);
end
```

4. Segmentation results

Contents

4.1 Preprocessing

The panoramic dental x-rays provided by the faculty of dentistry of Universidad La Salle Bajío have a pixel resolution of $1,935 \times 1,024$ pixels; they may not be the largest pictures, but they are considerably big. They are also noisy, which further complicates image analysis. In order to mitigate these two factors, superpixels will be used.

The main advantage of using superpixels is that the image can be simplified while preserving its structure; this means edges and object boundaries are preserved.
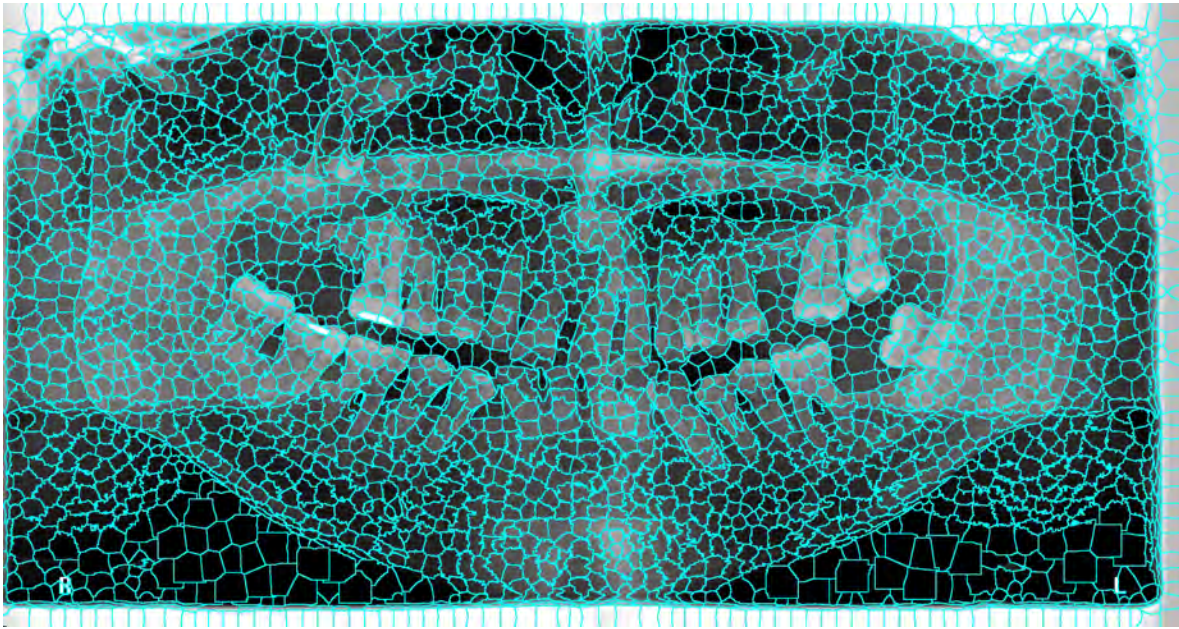
Figure 4.1: Superpixels oversegmentation applied to one of the supplied panoramic dental x-rays.

Resolution [px]	Average time per radiograph [s]	No. superpixels
1935×1024	1.78	1000
968×512	0.41	1000
484×256	0.12	1000

Table 4.1: Average time taken for superpixel clustering.

In these results, superpixels were applied to the x-rays and then all the pixels in each cluster were assigned the cluster's mean color in order to reduce noise as much as possible. The result

of this can be seen in Figure 4.2, where noise is reduced and most image components are preserved.

(a) Original panoramic x-ray.

(b) Original cropped tooth.
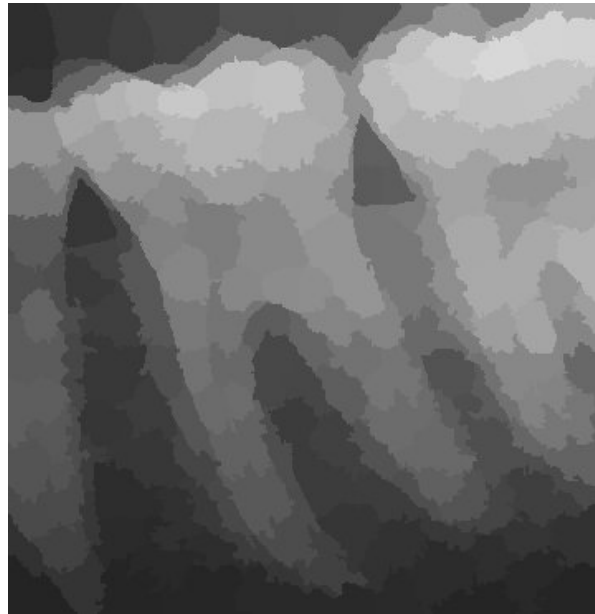
(c) Cropped tooth after having its superpixels clusters computed and each cluster assigned its mean color. Number of clusters $= 200$

Figure 4.2: Cluster pixels assigned its cluster's mean color.

The resulting superpixel clustering can be smoothed by using a Gaussian or a median

filter on the cropped tooth before computing superpixels. Figure 4.3 shows the results of different Gaussian filters applied on the original cropped tooth and its corresponding superpixel clustering. The filter size was computed using `2*ceil(2*sigma)+1`, where `sigma` (σ) is the specified standard deviation of the filter.
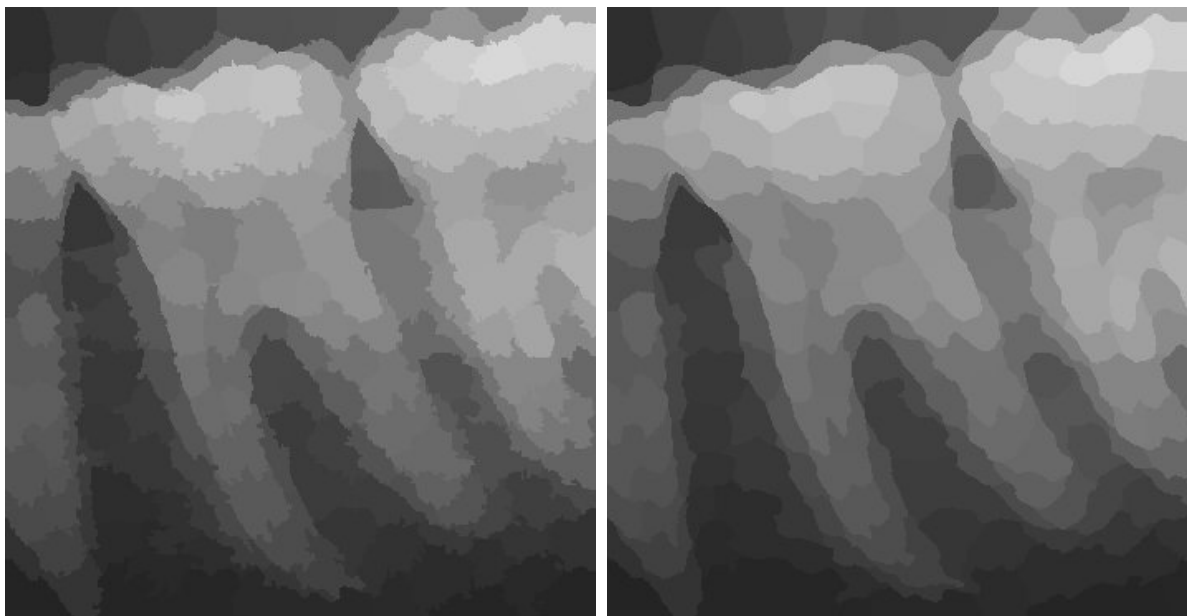
(a) Superpixel clustering with no Gaussian filter.

(b) Superpixel clustering with Gaussian filter: $\sigma = 2$, filter size $= 9$
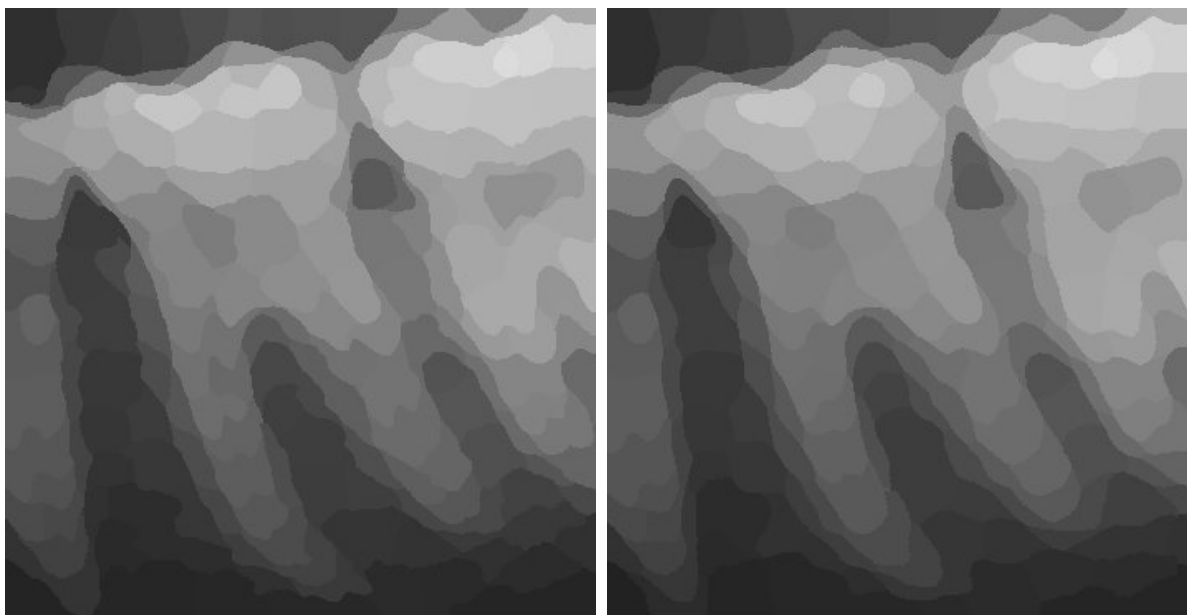
(c) Superpixel clustering with Gaussian filter: $\sigma = 3$, filter size $= 13$

(d) Superpixel clustering with Gaussian filter: $\sigma = 5$, filter size $= 21$

Figure 4.3: Superpixel clustering after applying varying Gaussian filter sizes.

4.2 **Teeth segmentation**

4.2.1 **Using Otsu's method**

This approach uses fast local Laplacian filtering in order to produce a contrast-enhanced image [29, 30] and then proceeds to use Otsu's method to create a threshold value to binarize the image. The process will begin with a cropped molar and show each individual step taken in every subfigure from left to right, top to bottom.

The results of these two operations are shown in Figure 4.4.
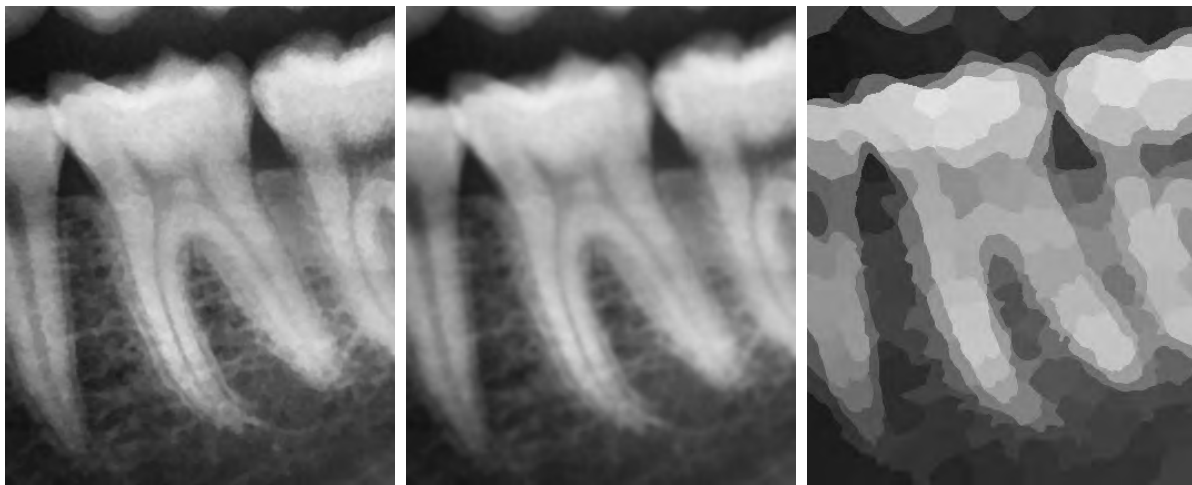
(a) Original cropped molar.

(b) Gaussian filter applied to original cropped molar. $\sigma = 2$, filter size $= 9$.

(c) Superpixel clustering applied to Gaussian-filtered image. Number of superpixels $= 200$.

(d) Fast Local Laplacian filtering applied to image with superpixel clustering. $\alpha = 0.1$.

(e) Otsu's method applied to image with fast local Laplacian filtering.

Figure 4.4: Fast local Laplacian filtering and Otsu's method applied to the cropped molar.

Otsu's method provides a very good image to work with. To fully segment the tooth, the image will be cleaned-up using morphological operations before separating the teeth. These morphological operations and its parameters are shown in Figure 4.5.

(a) Median filtering applied with a kernel size of 3×3.

(b) Largest connected object kept.

(c) Black pixels inside white pixels boundaries set to logical 1 (fill holes).
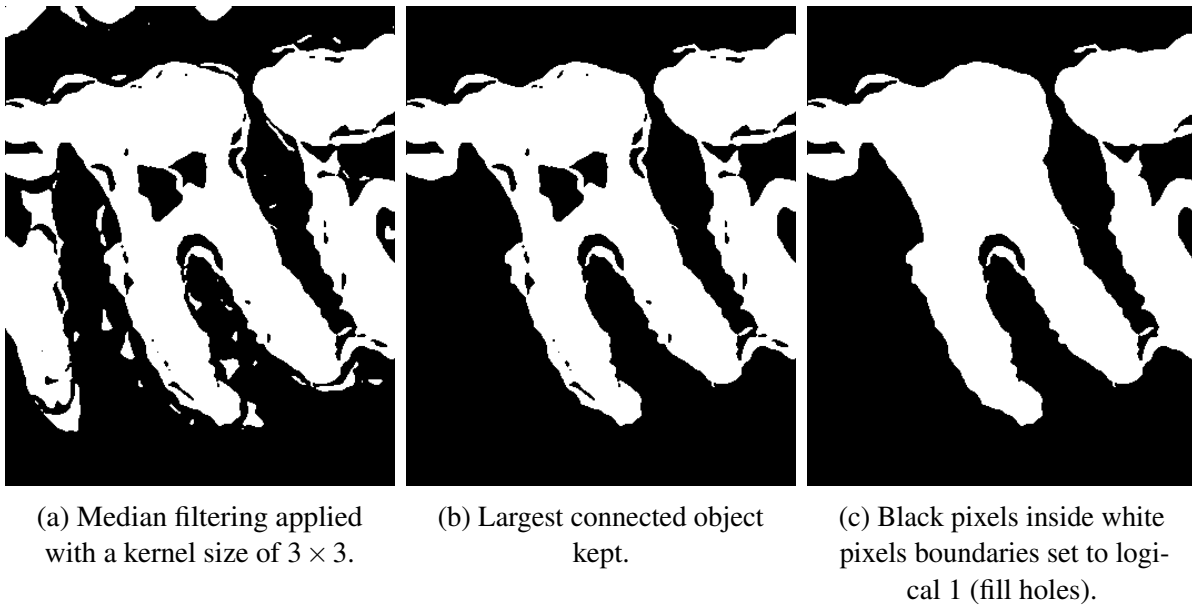
Figure 4.5: Results of morphological operations used to clean the image.

The biggest problem with the current image is the overlapping teeth that have been masked together because of their similar values and connectivity. A proposed approach to split the different teeth is to apply the distance transform in a similar way as described in 3.3.1. The preparation process for watershed segmentation can be seen in Figure 4.6.

(a) Distance transform of the image in Figure 4.5c.

(b) Inverted image in Figure 4.5c.

(c) Distance transform of inverted image (4.5c).

(d) Complement of image of Figure 4.6c.
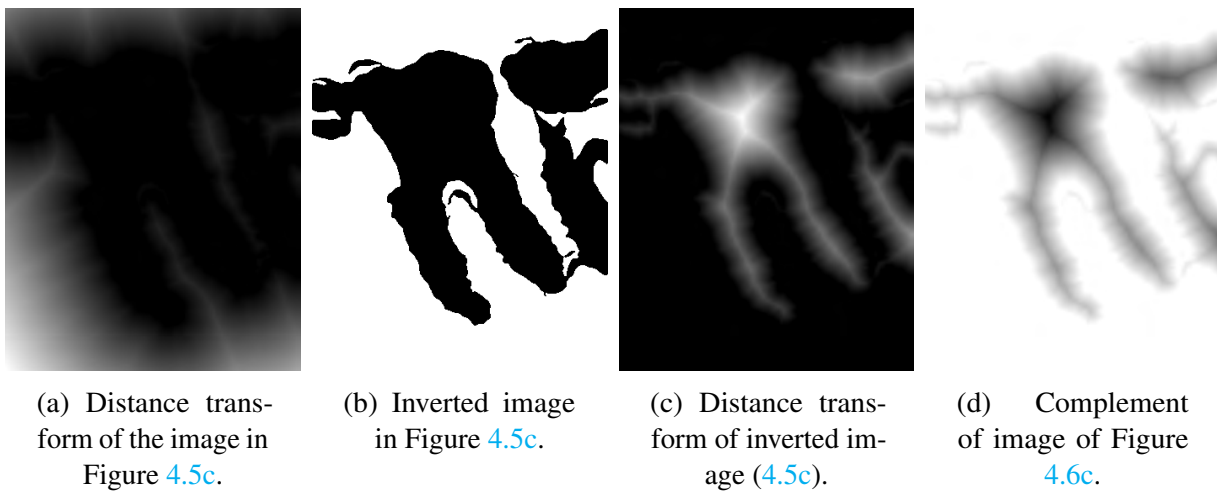
Figure 4.6: Distance transforms of the cleaned Otsu mask.

Applying watershed segmentation to Figure 4.6d results in Figure 4.7.

Figure 4.7: Watershed segmentation of Figure 4.6d.

As shown in Figure 4.7, the watershed segmentation does separate the teeth, but also introduces additional gaps in the teeth; the teeth are split with vertical gaps, while horizontal gaps are created inside the teeth's boundary. In order to attempt to fix this, image gradients are used in order to fill or thicken the empty space between lines. The computed image gradients for the image in Figure 4.7 are shown in Figure 4.8.
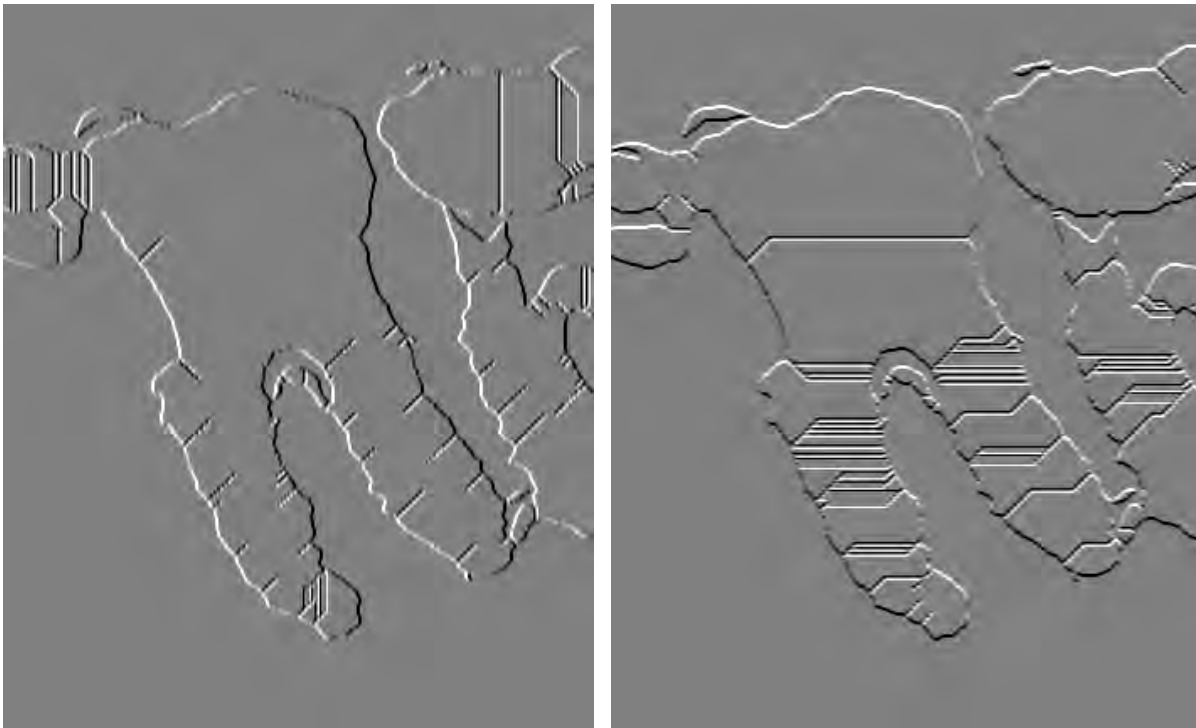
(a) X-directional gradient of Figure 4.7.   (b) Y-directional gradient of Figure 4.7.

Figure 4.8: Directional gradients of Figure 4.7. Gray pixels have a small gradient; black or white pixels have a large gradient.

We can filter these gradients to select only the strongest gradients. This can be done in MATLAB with the following commands:

```
spotsToRemove = Gx >= max(Gx(:))| Gx <= min(Gx(:));
spotsToFill = Gy >= max(Gy(:))| Gy <= min(Gy(:));
```

where `Gx`, `Gy` are the directional *x*, *y* gradients respectively and `spotsToFill` and `spotsToRemove` are the resulting *x*, *y* filtered gradients respectively.

These filtered gradients will be used to fill and further separate the teeth. The filtered x-directional gradient will be filled while the y-directional gradient will be removed. This can be done easily in MATLAB using linear indexing like so:

```
tooth(spotsToFill == 1)= 1;
tooth(spotsToRemove == 1)= 0;
```

where `tooth` is the segmented tooth image. To further enhance this operation, the `spotsToFill`

and `spotsToRemove` images can be thickened before applying this. The result of this is shown in Figure 4.9.

Figure 4.9: Tooth image with horizontal gradient added and vertical gradient removed.

Now that the teeth are separated and image gradients added and removed, some final morphological operations are done to completely isolate the molar. The final morphological operations done and their parameters are shown in Figure 4.10. These operations were carried out from left to right.

(a) Morphological closing with a structuring element of type `line`, length $= 2\,\text{px}$ and angle $= 90°$.

(b) Largest connected object kept.

(c) Morphological closing with a structuring element of type `line`, length $= 5\,\text{px}$ and angle $= 0°$.

(d) Black pixels inside white pixels boundaries set to logical 1 (fill holes).

(e) Morphological closing with a structuring element of type `disk`, radius $= 10\,\text{px}$.

Figure 4.10: Morphological operations done to further isolate the tooth.

The final result of the segmentation (Figure 4.10e) laid over the initial cropped tooth (Figure 4.4a) is shown in Figure 4.11. The segmentation is shown with a higher brightness in

order to easily differentiate it from the initial cropped image.
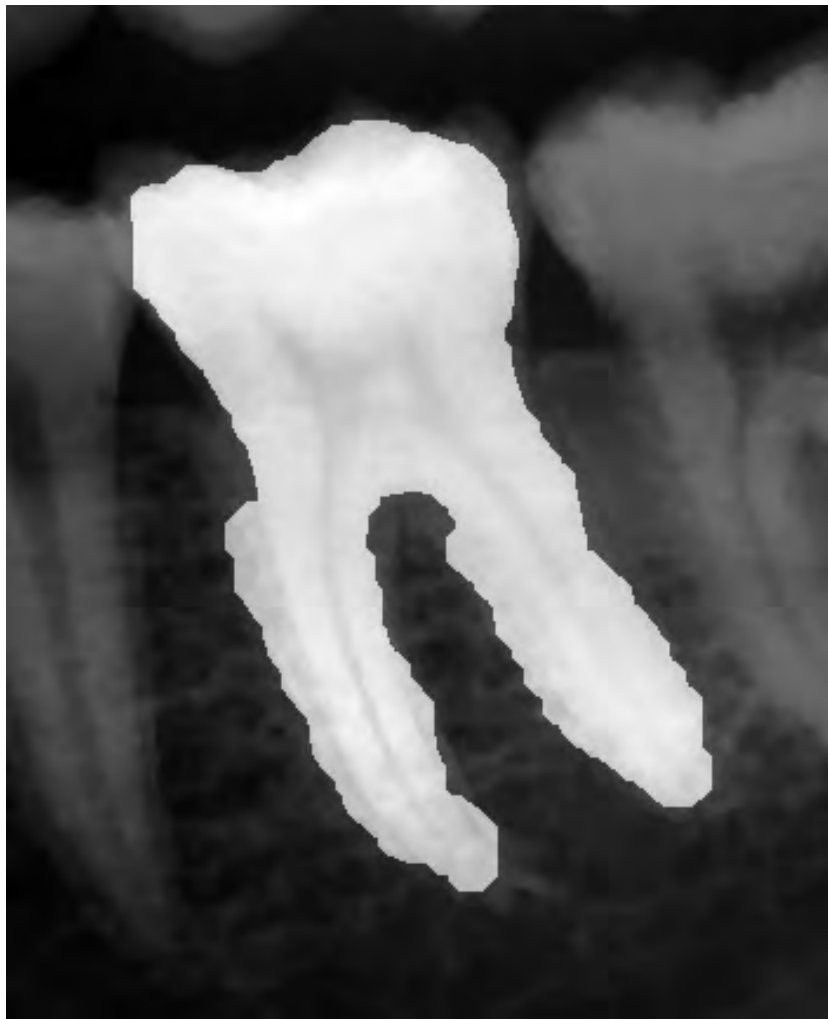
Figure 4.11: Final segmentation mask laid over initial cropped tooth.

4.2.2 Using active contour

In order to accurately compare the two methods, this method will begin with Figure 4.2a and the same tooth will be segmented just like the previous approach.

Applying active contour directly on the unmodified cropped tooth results in the image shown in Figure 4.12b. A root-mean-square error check was implemented to stop the algorithm in case the images were too similar.
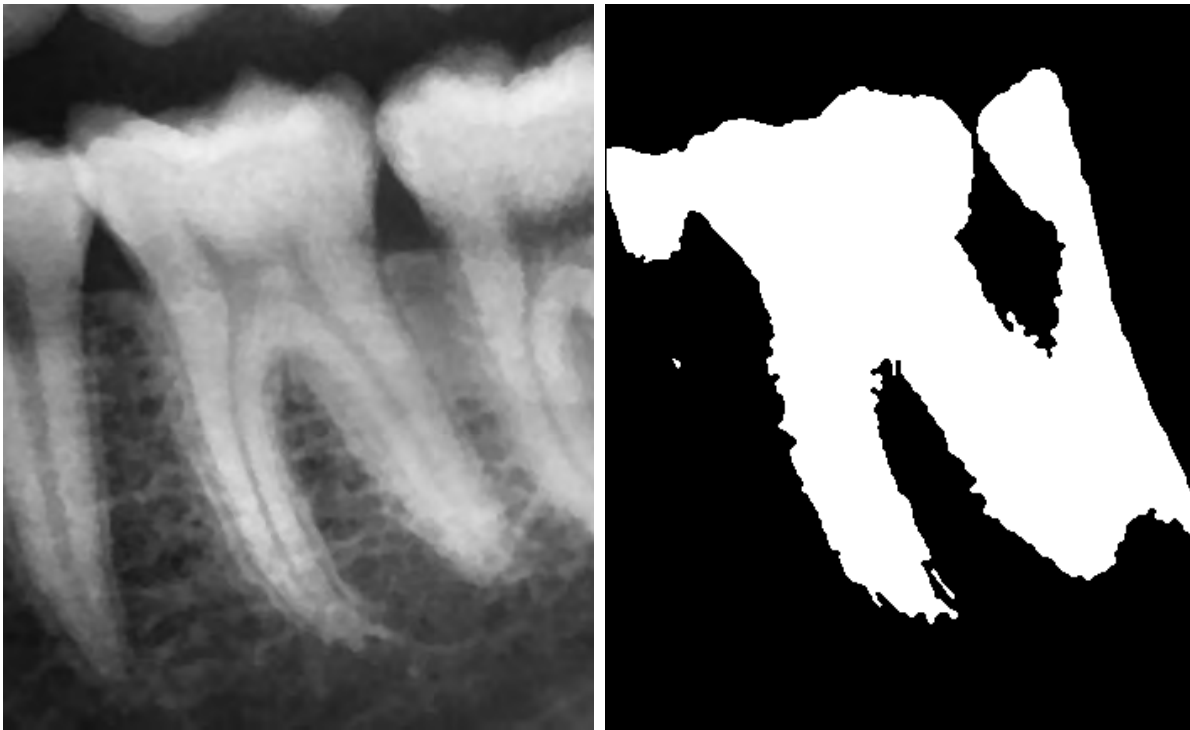
(a) Unmodified cropped tooth.   (b)   Active   contours   applied   directly.
                                              Smooth factor $= 0$, contraction bias $= 0$.

Figure 4.12: Active contour applied to unmodified cropped tooth.

The result is a decent segmentation of the tooth; however, it remains joined with the two teeth to its sides. This particular result took 268 iterations of the active contour algorithm.

Enhancing the contrast using fast local Laplacian filtering improves the segmentation, while increasing the required iterations of active contouring to 291.
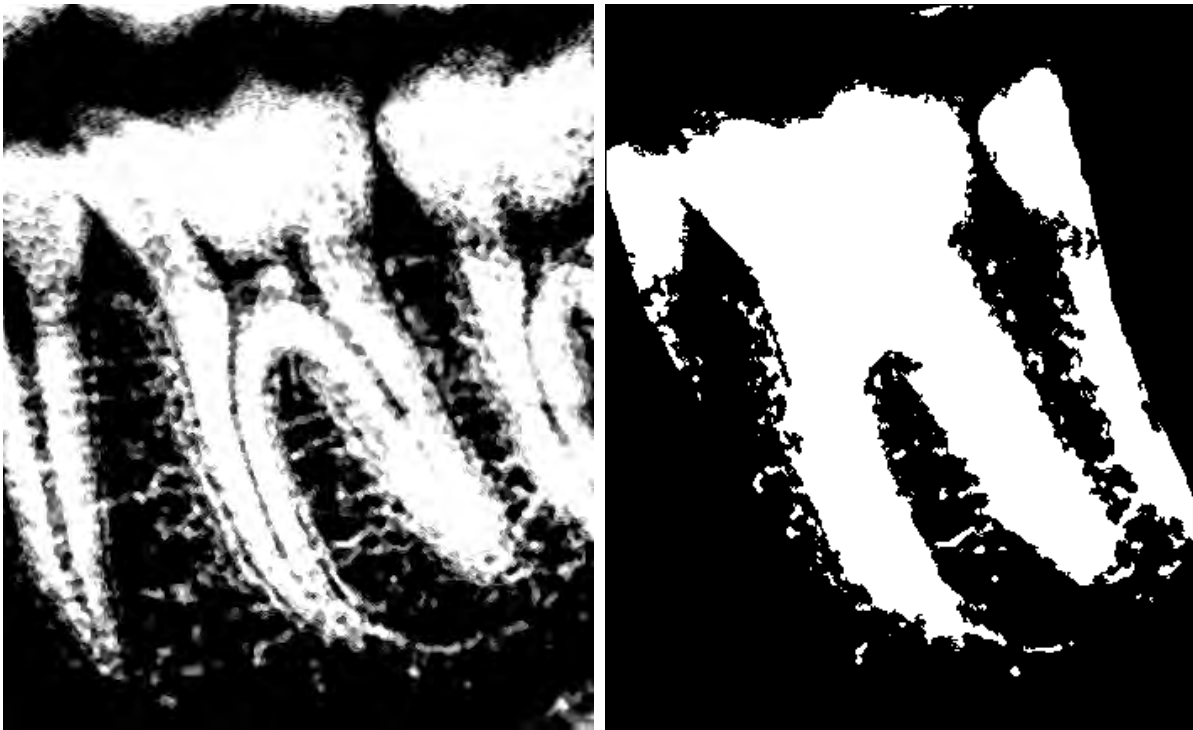
(a) Contrast-enhanced cropped tooth with fast local Laplacian filtering. $\alpha = 0.1, \beta = 1, \sigma = 1$.

(b) Active contour applied to contrast-enhanced cropped tooth. Smooth factor $= 0$, contraction bias $= 0$.

Figure 4.13: Active contour applied to contrast-enhanced cropped tooth.

Increasing the smooth factor helps to reduce the noise in the segmentation; however, selecting a value too large will make the algorithm lose a considerable amount of tooth pixels.

(a) Active contour applied to contrast-enhanced cropped tooth. Smooth factor $= 0$, contraction bias $= 0$.

(b) Active contour applied to contrast-enhanced cropped tooth. Smooth factor $= 1.5$, contraction bias $= 0$.

Figure 4.14: Comparison of smoothing factor.

Superpixel clustering changes the segmentation done by active contour, but does not necessarily improve it. Morphological post-processing is preferred as it improves the result and most of the times it is computationally faster.

Post-processing

Some post-processing operations were done on the image to further improve the result. The operations done and their parameters are illustrated in Figure 4.15.
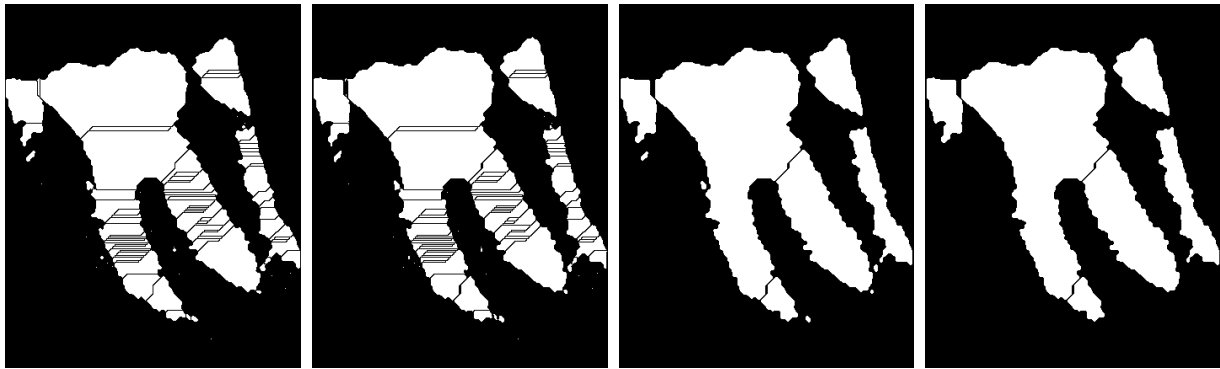
(a) Distance transform to separate teeth.

(b) Largest vertical gradients removed and largest horizontal gradients filled in.

(c) Median filtering applied with a kernel size of 3×3.

(d) Areas with less than $100\,\text{px}$ removed.

(e) Morphological closing with structuring element of type `line`, length $= 10\,\text{px}$, angle $= 120°$.

(f) Largest area kept.

(g) Morphological opening with structuring element of type `disk`, radius $= 5\,\text{px}$.

Figure 4.15: Post processing operations done to further isolate the segmented molar.

The final result from the segmentation can be seen overlaid on the initial panoramic dental radiograph in Figure 4.16.

Figure 4.16: Panoramic x-ray with segmentation mask overlaid.

4.3 Computing molar orientation

In order to find the orientation of a particular molar (that is, the angle between the x-axis and the major axis of the ellipse that has the same second-moments as the region), the centroid of the segmented mask must be found first.

Definition 4.3.1 Given a binary or grayscale image $I(x,y)$, its ordinary moment of order i,j can be defined as [54, 55, 56]:

$$m_{ij} = \sum_x \sum_y x^i y^j I(x,y) \tag{4.1}$$

The centroid of a binary or grayscale image is the arithmetic mean of the coordinates in the x and y directions times its pixel value and thus can be defined as:

$$\bar{x} = \frac{m_{10}}{m_{00}} \qquad \bar{y} = \frac{m_{01}}{m_{00}} \tag{4.2}$$

The orientation of the molar is the same as the orientation of the major axis; that is, the axis that runs through the centroid and along the widest part of the region [54]. This major axis can be computed using the image moments.

Definition 4.3.2 We can shift the image moments using its centroid in order to compute the image central moments of order p,q like so:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I(x,y) \tag{4.3}$$

Normalizing these central moments can be done by dividing by the area of the region,

which is equal to m_{00} and μ_{00}.

$$\mu'_{pq} = \frac{\mu_{pq}}{\mu_{00}} \tag{4.4}$$

A covariance matrix can be built from these normalized central moments:

$$\text{cov}[I(x,y)] = \begin{bmatrix} \mu'_{20} & \mu'_{11} \\ \mu'_{11} & \mu'_{02} \end{bmatrix} \tag{4.5}$$

The eigenvectors of this matrix correspond to the major and minor axes of the image intensity, so the orientation can thus be extracted from the angle of the eigenvector associated with the largest eigenvalue towards the axis closest to this eigenvector. The angle of orientation Θ is then given by the following equation [54, 55]:

$$\Theta = \frac{1}{2} \arctan \left(\frac{2\mu'_{11}}{\mu'_{20} - \mu'_{02}} \right) \tag{4.6}$$

Having defined these moments, it is possible to compute the orientation angle of any binary region. Shown in Figure 4.17 is the segmented tooth with its major axis plotted in red along with its length and the major axis orientation and its angle magnitude.
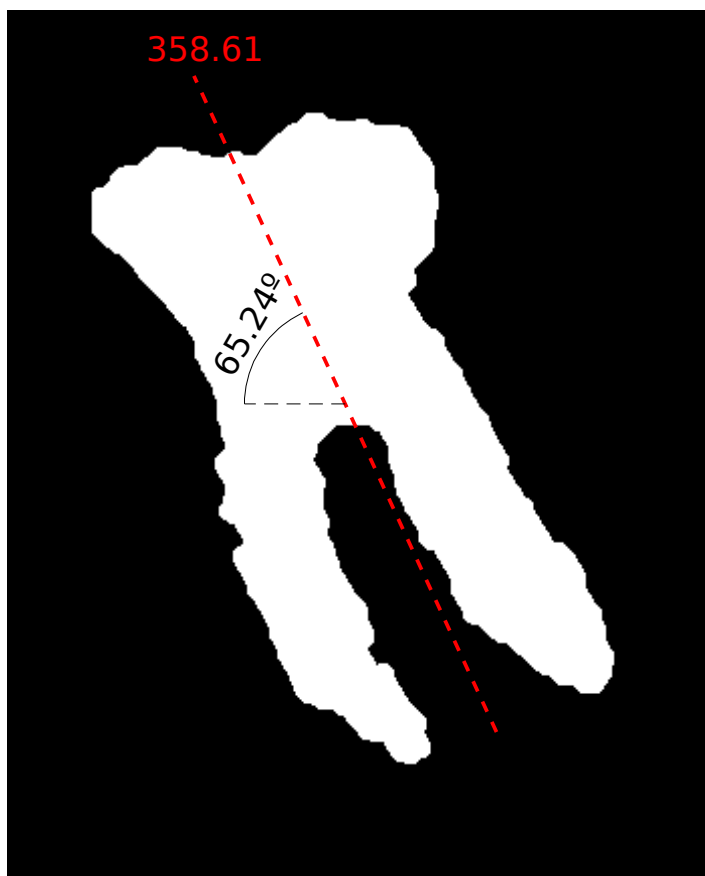
Figure 4.17: Segmented molar with its major axis length and orientation plotted.

References

[1] U.S. Food and Drug Administration. *Medical X-ray Imaging*. URL: http://www.fda.gov/Radiation-EmittingProducts/RadiationEmittingProductsandProcedures/MedicalImaging/MedicalX-Rays/ucm175028.htm (visited on 11/30/2016) (cited on page 2).

[2] M Sonoda et al. "Computed radiography utilizing scanning laser stimulated luminescence." In: *Radiology* 148.3 (Sept. 1983), pages 833–838. DOI: 10.1148/radiology.148.3.6878707. URL: http://dx.doi.org/10.1148/radiology.148.3.6878707 (cited on page 2).

[3] G J Bansal. "Digital radiography. A comparison with modern conventional imaging". In: *Postgraduate Medical Journal* 82.969 (July 2006), pages 425–428. DOI: 10.1136/pgmj.2005.038448. URL: http://dx.doi.org/10.1136/pgmj.2005.038448 (cited on page 2).

[4]   Eric Whaites. *Essentials of dental radiography and radiology*. Edinburgh: Churchill Livingstone/Elsevier, 2013. ISBN: 0702045993 (cited on page 3).

[5]   Herbert Frommer. *Radiology for the dental professional*. London: Elsevier Health Sciences, 2011. ISBN: 9780323291156 (cited on page 4).

[6]   Minoru Fujita et al. "Digital image processing of periapical radiographs". In: *Oral Surgery, Oral Medicine, Oral Pathology* 65.4 (1988), pages 490–494 (cited on page 5).

[7]   Ozlem Gormez and Hasan Huseyin Yilmaz. "Image post-processing in dental practice". In: *Eur J Dent* 3.4 (2009), pages 343–7 (cited on page 5).

[8]   Avrum R Goldstein. "Enamel pearls as a contributing factor in periodontal breakdown". In: *The Journal of the American Dental Association* 99.2 (1979), pages 210–211 (cited on page 5).

[9]   Debora C Matthews and Moe Tabesh. "Detection of localized tooth-related factors that predispose to periodontal infections". In: *Periodontology 2000* 34.1 (Feb. 2004), pages 136–150. DOI: 10.1046/j.0906-6713.2003.003429.x. URL: http://dx.doi.org/10.1046/j.0906-6713.2003.003429.x (cited on page 5).

[10] U. Romeo et al. "Enamel pearls as a predisposing factor to localized periodontitis". In: *Quintessence Int* 42.1 (Jan. 2011), pages 69–71 (cited on page 5).

[11] D. C. Matthews and M. Tabesh. "Detection of localized tooth-related factors that predispose to periodontal infections". In: *Periodontol. 2000* 34 (2004), pages 136–150 (cited on pages 5, 9).

[12] Antonio Nanci. *Ten Cate's Oral Histology: Development, Structure, and Function*. 8th edition. Mosby, 2012, pages 121–122. ISBN: 9780323078467. URL: http://gen.lib.rus.ec/book/index.php?md5=A5BBADDA16560F63451B1AAB8B901D07 (cited on page 7).

[13] M. Staines, W. H. Robinson, and J. A. A. Hood. "Spherical indentation of tooth enamel". In: *J Mater Sci* 16.9 (Sept. 1981), pages 2551–2556. DOI: 10.1007/bf01113595. URL: http://dx.doi.org/10.1007/BF01113595 (cited on page 7).

[14] Mary Bath-Balogh BA BS MS and Margaret J. Fehrenbach RDH MS. *Illustrated Dental Embryology, Histology, and Anatomy*. 3rd edition. Saunders, Jan. 2011, page 180. ISBN: 9781437717303. URL: http://amazon.com/o/ASIN/1437717306/ (cited on page 7).

[15] P. M. Butler. "Ontogenetic aspects of dental evolution". In: *Int. J. Dev. Biol.* 39.1 (Feb. 1995), pages 25–34 (cited on page 7).

[16] S. Risnes et al. "Enamel pearls and cervical enamel projections on 2 maxillary molars with localized periodontal disease: case report and histologic study". In: *Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology* 89.4 (Apr. 2000), pages 493–497 (cited on pages 7, 8).

[17] Donald H. Masters and Sam W. Hoskins. "Projection of Cervical Enamel into Molar Furcations". In: *Journal of Periodontology* 35.1 (Jan. 1964), pages 49–53. DOI: 10.1902/jop.1964.35.1.49. URL: http://dx.doi.org/10.1902/jop.1964.35.1.49 (cited on pages 7, 8).

[18] G.V. Prashanth, D.V. Nagarathna, and Melba D'Souza. "Congenital grooves and cervical enamel projections on the teeth and their relation to periodontal status". In: *Al Ameen Journal of Medical Sciences* 7.2 (2014), pages 112–117 (cited on page 8).

[19] J.P. Ennes and V.S. Lara. "Comparative morphological analysis of the root developmental groove with the palato-gingival groove". In: *Oral Diseases* 10.6 (Nov. 2004), pages 378–382. DOI: 10.1111/j.1601-0825.2004.01009.x. URL: http://dx.doi.org/10.1111/j.1601-0825.2004.01009.x (cited on page 8).

[20] J. Simon et al. "The Radicular Groove: Its Potential Clinical Significance". In: *Journal of Endodontics* 26.5 (May 2000), pages 295–298. DOI: 10.1097/00004770-200005000-00012. URL: http://dx.doi.org/10.1097/00004770-200005000-00012 (cited on page 8).

[21] D. D. Bosshardt and A. Nanci. "Immunocytochemical characterization of ectopic enamel deposits and cementicles in human teeth". In: *Eur. J. Oral Sci.* 111.1 (Feb. 2003), pages 51–59 (cited on pages 8, 10).

[22] S. Risnes. "Ectopic tooth enamel. An SEM study of the structure of enamel in enamel pearls". In: *Adv. Dent. Res.* 3.2 (Sept. 1989), pages 258–264 (cited on page 8).

[23] Leslie DeLong and Nancy Burkhart. *General and Oral Pathology for the Dental Hygienist*. 1 Pap/Psc. Lippincott Williams and Wilkins, 2007, pages 498–499. ISBN: 9780781755467. URL: http://gen.lib.rus.ec/book/index.php?md5=03C2DC701635F39190CB5F9CEA189FBE (cited on pages 8–10).

[24] G.R. Scott and C.G. Turner. *The Anthropology of Modern Human Teeth: Dental Morphology and Its Variation in Recent Human Populations*. Cambridge Studies in Biological and Evolutionary Anthropology. Cambridge University Press, 2000. ISBN: 9780521784535. URL: https://books.google.com.mx/books?id=HuRcAyXWJxIC (cited on page 8).

[25] H. Colak et al. "Radiographic evaluation of the prevalence of enamel pearls in a sample adult dental population". In: *European Review for Medical and Pharmacological Sciences* 18.3 (2014), pages 440–444 (cited on page 9).

[26] T.S. Leeson, C.R. Leeson, and A.A. Paparo. *Text/atlas of histology*. Saunders, 1988. ISBN: 9780721623863. URL: https://books.google.com.mx/books?id=881qAAAAMAAJ (cited on page 9).

[27] M. DeSanctis and K. G. Murphy. "The role of resective periodontal surgery in the treatment of furcation defects". In: *Periodontol. 2000* 22 (Feb. 2000), pages 154–168 (cited on page 10).

[28] A. R. Goldstein. "Enamel pearls as contributing factor in periodontal breakdown". In: *J Am Dent Assoc* 99.2 (Aug. 1979), pages 210–211 (cited on page 10).

[29]   Mathieu Aubry et al. "Fast Local Laplacian Filters: Theory and Applications". In: *ACM Trans. Graph.* 33.5 (Sept. 2014), 167:1–167:14. ISSN: 0730-0301. DOI: 10.1145/2629645. URL: http://doi.acm.org/10.1145/2629645 (cited on pages 11, 53).

[30]   Sylvain Paris, Samuel W. Hasinoff, and Jan Kautz. "Local Laplacian filters". In: *Communications of the ACM* 58.3 (Feb. 2015), pages 81–91. DOI: 10.1145/2723694. URL: http://dx.doi.org/10.1145/2723694 (cited on pages 11, 53).

[31]   Gayathri V and Hema P Menon. "Challenges in Edge Extraction of Dental X-Ray Images Using Image Processing Algorithms – A Review". In: *International Journal of Computer Science and Information Technologies* 5(4) (2014), pages 5355–5357 (cited on page 11).

[32]   Samir Shah et al. "Automatic Tooth Segmentation Using Active Contour Without Edges". In: *2006 Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*. Institute of Electrical and Electronics Engineers (IEEE), Sept. 2006. DOI: 10.1109/bcc.2006.4341636. URL: http://dx.doi.org/10.1109/BCC.2006.4341636 (cited on page 11).

[33]   Yusra Y. Amer and Musbah J. Aqel. "An Efficient Segmentation Algorithm for Panoramic Dental Images". In: *Procedia Computer Science* 65 (2015), pages 718–725. DOI: 10.1016/j.procs.2015.09.016. URL: http://dx.doi.org/10.1016/j.procs.2015.09.016 (cited on page 11).

[34]   D.E. Nassar et al. "Automatic Construction of Dental Charts for Postmortem Identification". In: *IEEE Transactions on Information Forensics and Security* 3.2 (June 2008), pages 234–246. DOI: 10.1109/tifs.2008.922452. URL: https://doi.org/10.1109%2Ftifs.2008.922452 (cited on page 11).

5. Discussion and conclusions

Contents

5.1   Main concern of the study

The two approaches presented in this work offer solutions to the problem of segmenting molars in noisy pantomograms. The two methods were tested on 66 different pantomograms kindly provided by Universidad de la Salle Bajío.

The goal of this work was to show that automated feature recognition in teeth is possible by being able to segment teeth (molars) individually and extract information from this image. This is important because an odontologist can only look and analyze one x-ray at a time, while an automated approach could use several processors in parallel for analyzing multiple x-rays at the same time.

For studies that involve quantitative variables, the number of samples examined (in this case, x-rays) must be large enough in order to satisfy some confidence level. For some studies this sample size may be required to be over hundreds or thousands of patients, which would take a huge amount of work to complete for odontologists, but would only take a fraction of that time for computers.

5.2   Main findings of the study

For the work done in this thesis, each pantomogram took an average of $0.66\,\mathrm{s}$ to analyze using Otsu's method with a total of $0.0\,\mathrm{s}$ for all 66 pantomograms. For active contours, each pantomogram took an average of $10.90\,\mathrm{s}$ to analyze with a total of $0.0\,\mathrm{s}$ for all 66 pantomograms.

To calculate an error rate, a ground truth image was segmented manually, which was compared pixel-wise to the masks created by both methods. The error rate using Otsu's method was found to be $6.64\,\%$, while the average error rate using active contours was found to be $6.19\,\%$. This is in large part due to the fact that Otsu's method requires more tuning to each individual case because of its larger number of morphological operations required; however, if done properly, the error rate could be less or equal to the error rate using active contours.

In summary, active contours requires less tuning of the algorithm at the expense of computation time, while Otsu's method requires more manual intervention for every case but can be magnitudes faster compared to active contours. Of course, both methods represent a massive amount of time saved for odontologists, which would take days to analyze every pantomogram by themselves.

Watershed segmentation, alongside the distance transform, is an invaluable tool for separating teeth that are overlapping each other in the pantomograms, which is almost always unless a tooth is missing.

Superpixel clustering helps reduce the noise in the pantomograms in order to produce better results with both methods, though it certainly is more useful for active contours; on the other hand, fast local Laplacian filtering is more essential to Otsu's method due to its nature.

5.3 Limitations of the study

The work on this thesis is focused on developing a method for automating teeth segmentation in pantomograms. Any additional measurement, such as the teeth's angle, is meant to be a showcase of what can be done once the teeth are extracted from the x-rays.

Due to time constrains, The sample size in this work is 66 pantomograms, a very small number that does not indicate that the work done here will output similar results in other x-rays; however, only small adjustments should be needed, particularly in the morphological operations applied and their order. It is recommended for other researchers to increase this number as much as possible. Traditional digital image processing techniques were used because of this limitation instead of more complex approaches.

5.4 Future work and recommendations

Another objective of this work is to lay the foundations for future work regarding odontological studies; in particular, those related to periodontal disease. Follow-up to this work related to periodontitis includes detection of enamel pearls and measurement of enamel projection.

Of course, periodontal disease is not the only area which can benefit from the work done here. Further work is already being planned for classification of third molars according to their extraction difficulty using the Pell-Gregory system.

Whenever measurements need to be accurate, periapical x-rays should be used instead of pantomograms. This is because pantomograms possess an inherent distortion because they essentially project a curved surface (the jaw) into a flat surface (a 2D image); this is identical to the problems that world map projections have, such as the case of the Mercator projections where landmasses such as Greenland and Antarctica appear much larger than they actually are relative to land masses near the equator, such as Central Africa.

A larger sample size should be used in order to achieve more robustness. Also, given a large enough sample size, more complex techniques such as machine learning algorithms could be used to segment the molars, with more effectiveness and efficiency.

Furthermore, the techniques in this work, as well as more robust approaches, could be

included in a system which processes pantomograms as they are acquired in real-time. This would imply the need of some new hardware (particularly electronic components) in order to fully automate the process. If the original pantomograms are analog-based, then a mechanical and optical system could be included to digitize the samples.

Bibliography

Books

[4] Eric Whaites. *Essentials of dental radiography and radiology*. Edinburgh: Churchill Livingstone/Elsevier, 2013. ISBN: 0702045993 (cited on page 3).

[5] Herbert Frommer. *Radiology for the dental professional*. London: Elsevier Health Sciences, 2011. ISBN: 9780323291156 (cited on page 4).

[12] Antonio Nanci. *Ten Cate's Oral Histology: Development, Structure, and Function*. 8th edition. Mosby, 2012, pages 121–122. ISBN: 9780323078467. URL: http://gen.lib.rus.ec/book/index.php?md5=A5BBADDA16560F63451B1AAB8B901D07 (cited on page 7).

[14] Mary Bath-Balogh BA BS MS and Margaret J. Fehrenbach RDH MS. *Illustrated Dental Embryology, Histology, and Anatomy*. 3rd edition. Saunders, Jan. 2011, page 180. ISBN: 9781437717303. URL: http://amazon.com/o/ASIN/1437717306/ (cited on page 7).

[23] Leslie DeLong and Nancy Burkhart. *General and Oral Pathology for the Dental Hygienist*. 1 Pap/Psc. Lippincott Williams and Wilkins, 2007, pages 498–499. ISBN: 9780781755467. URL: http://gen.lib.rus.ec/book/index.php?md5=03C2DC701635F39190CB5F9CEA189FBE (cited on pages 8–10).

[24] G.R. Scott and C.G. Turner. *The Anthropology of Modern Human Teeth: Dental Morphology and Its Variation in Recent Human Populations*. Cambridge Studies in Bi-

ological and Evolutionary Anthropology. Cambridge University Press, 2000. ISBN: 9780521784535. URL: https://books.google.com.mx/books?id=HuRcAyXWJxIC (cited on page 8).

[26]   T.S. Leeson, C.R. Leeson, and A.A. Paparo. *Text/atlas of histology*. Saunders, 1988. ISBN: 9780721623863. URL: https://books.google.com.mx/books?id=881qAAAAMAAJ (cited on page 9).

[35]   C. Starr, C. Evers, and L. Starr. *Biology: Concepts and Applications*. Available Titles CourseMate Series. Cengage Learning, 2010. ISBN: 9781439046739. URL: https://books.google.com.mx/books?id=%5C_16xbB2Py%5C_UC (cited on page 15).

[36]   Henri Maitre. *From Photon to Pixel: The Digital Camera Handbook*. Wiley-ISTE, 2015, pages 148–149. ISBN: 1848218478. URL: http://www.amazon.com/Photon-Pixel-Digital-Camera-Handbook/dp/1848218478%3FSubscriptionId%3D0JYN1NVW651KCA56C102%26tag%3Dtechkie-20%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3D1848218478 (cited on pages 16, 18).

[37]   J.E. Hall. *Guyton and Hall Textbook of Medical Physiology*. Guyton Physiology. Elsevier Health Sciences, 2010. ISBN: 9781437726749. URL: https://books.google.com.mx/books?id=Po0zyO0BFzwC (cited on page 18).

[38]   G. Wyszecki and W.S. Stiles. *Color science: concepts and methods, quantitative data and formulae*. Wiley classics library. Wiley, 1982. ISBN: 9780471021063. URL: https://books.google.com.mx/books?id=HkjM5HNB6jIC (cited on page 18).

[39]   R.W.G. Hunt. *The Reproduction of Colour*. The Wiley-IS&T Series in Imaging Science and Technology. Wiley, 2004. ISBN: 9780470024256. URL: https://books.google.com.mx/books?id=1%5C_MeAQAAIAAJ (cited on page 18).

[40]   Chris Solomon. *Fundamentals of digital image processing : a practical approach with examples in Matlab*. Chichester: Wiley-Blackwell, 2011. ISBN: 978-0-470-84472-4 (cited on pages 22, 24, 26, 32–34).

[41]   Bernd Jähne. *Digital image processing*. Berlin New York: Springer, 2005. ISBN: 9783540240358 (cited on page 23).

[42]   S Jayaraman. *Digital image processing*. New Delhi: Tata McGraw Hill Education, 2009. ISBN: 9780070144798 (cited on pages 26, 28, 33, 34).

[43]   John Goutsias. *Mathematical morphology and its applications to image and signal processing*. Boston: Kluwer Academic Publishers, 2000. ISBN: 9780306470257 (cited on page 28).

[44]   Scott Umbaugh. *Computer imaging : digital image analysis and processing*. Boca Raton: Taylor & Francis, 2005. ISBN: 9780849329197 (cited on page 28).

[45]   Alan Bovik. *Handbook of image and video processing*. Amsterdam Boston, MA: Elsevier Academic Press, 2005. ISBN: 9780080533612 (cited on page 28).

[46] Michael Seul. *Practical algorithms for image analysis : description, examples, and code*. Cambridge New York: Cambridge University Press, 2000. ISBN: 9780521660655 (cited on page 31).

[47] Thomas Deserno. *Biomedical image processing*. Berlin Heidelberg New York: Springer, 2011. ISBN: 9783642158162 (cited on page 31).

[48] Gary Bradski. *Learning OpenCV : computer vision with the OpenCV library*. Sebastopol, Calif: O'Reilly, 2008. ISBN: 0596554044 (cited on page 32).

[49] Oge Marques. *Practical image and video processing using MATLAB*. Hoboken, NJ: Wiley-IEEE Press, 2011. ISBN: 9781118093474 (cited on page 34).

[50] Frank Shih. *Image processing and mathematical morphology : fundamentals and applications*. Boca Raton: CRC Press, 2009. ISBN: 9781420089448 (cited on page 34).

[51] K.A. Navas and R. Jayadevan. *Lab Primer Through Matlab®: Digital Signal Processing, Digital Image Processing, Digital Signal Processor and Digital Communication*. Eastern economy edition. PHI Learning, 2014. ISBN: 9788120349322. URL: https://books.google.com.mx/books?id=9I5eBAAAQBAJ (cited on page 35).

[52] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing (3rd Edition)*. Pearson, 2007. ISBN: 013168728X. URL: http://www.amazon.com/Digital-Image-Processing-Rafael-Gonzalez/dp/013168728X%3FSubscriptionId%3D0JYN1NVW651KCA56C102%26tag%3Dtechkie-20%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3D013168728X (cited on pages 35, 37, 38).

[53] Pierre Soille. *Morphological Image Analysis: Principles and Applications*. Springer, 2010. ISBN: 3642076963. URL: http://www.amazon.com/Morphological-Image-Analysis-Principles-Applications/dp/3642076963%3FSubscriptionId%3D0JYN1NVW651KCA56C102%26tag%3Dtechkie-20%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3D3642076963 (cited on page 36).

[54] Wilhelm Burger. *Principles of digital image processing : core algorithms*. London: Springer, 2009. ISBN: 1848001959 (cited on pages 65, 66).

[55] R Mukundan. *Moment functions in image analysis : theory and applications*. Singapore River Edge, NJ: World Scientific, 1998. ISBN: 9789810235246 (cited on pages 65, 66).

[56] Lawrence Gorman. *Practical algorithms for image analysis : description, examples, programs, and projects*. Cambridge New York: Cambridge University Press, 2008. ISBN: 9780521884112 (cited on page 65).

Articles

[2] M Sonoda et al. "Computed radiography utilizing scanning laser stimulated luminescence." In: *Radiology* 148.3 (Sept. 1983), pages 833–838. DOI: 10.1148/radiology.

148.3.6878707. URL: http://dx.doi.org/10.1148/radiology.148.3.6878707 (cited on page 2).

[3]   G J Bansal. "Digital radiography. A comparison with modern conventional imaging". In: *Postgraduate Medical Journal* 82.969 (July 2006), pages 425–428. DOI: 10.1136/pgmj.2005.038448. URL: http://dx.doi.org/10.1136/pgmj.2005.038448 (cited on page 2).

[6]   Minoru Fujita et al. "Digital image processing of periapical radiographs". In: *Oral Surgery, Oral Medicine, Oral Pathology* 65.4 (1988), pages 490–494 (cited on page 5).

[7]   Ozlem Gormez and Hasan Huseyin Yilmaz. "Image post-processing in dental practice". In: *Eur J Dent* 3.4 (2009), pages 343–7 (cited on page 5).

[8]   Avrum R Goldstein. "Enamel pearls as a contributing factor in periodontal breakdown". In: *The Journal of the American Dental Association* 99.2 (1979), pages 210–211 (cited on page 5).

[9]   Debora C Matthews and Moe Tabesh. "Detection of localized tooth-related factors that predispose to periodontal infections". In: *Periodontology 2000* 34.1 (Feb. 2004), pages 136–150. DOI: 10.1046/j.0906-6713.2003.003429.x. URL: http://dx.doi.org/10.1046/j.0906-6713.2003.003429.x (cited on page 5).

[10]   U. Romeo et al. "Enamel pearls as a predisposing factor to localized periodontitis". In: *Quintessence Int* 42.1 (Jan. 2011), pages 69–71 (cited on page 5).

[11]   D. C. Matthews and M. Tabesh. "Detection of localized tooth-related factors that predispose to periodontal infections". In: *Periodontol. 2000* 34 (2004), pages 136–150 (cited on pages 5, 9).

[13]   M. Staines, W. H. Robinson, and J. A. A. Hood. "Spherical indentation of tooth enamel". In: *J Mater Sci* 16.9 (Sept. 1981), pages 2551–2556. DOI: 10.1007/bf01113595. URL: http://dx.doi.org/10.1007/BF01113595 (cited on page 7).

[15]   P. M. Butler. "Ontogenetic aspects of dental evolution". In: *Int. J. Dev. Biol.* 39.1 (Feb. 1995), pages 25–34 (cited on page 7).

[16]   S. Risnes et al. "Enamel pearls and cervical enamel projections on 2 maxillary molars with localized periodontal disease: case report and histologic study". In: *Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology* 89.4 (Apr. 2000), pages 493–497 (cited on pages 7, 8).

[17]   Donald H. Masters and Sam W. Hoskins. "Projection of Cervical Enamel into Molar Furcations". In: *Journal of Periodontology* 35.1 (Jan. 1964), pages 49–53. DOI: 10.1902/jop.1964.35.1.49. URL: http://dx.doi.org/10.1902/jop.1964.35.1.49 (cited on pages 7, 8).

[18]   G.V. Prashanth, D.V. Nagarathna, and Melba D'Souza. "Congenital grooves and cervical enamel projections on the teeth and their relation to periodontal status". In: *Al Ameen Journal of Medical Sciences* 7.2 (2014), pages 112–117 (cited on page 8).

[19] J.P. Ennes and V.S. Lara. "Comparative morphological analysis of the root developmental groove with the palato-gingival groove". In: *Oral Diseases* 10.6 (Nov. 2004), pages 378–382. DOI: 10.1111/j.1601-0825.2004.01009.x. URL: http://dx.doi.org/10.1111/j.1601-0825.2004.01009.x (cited on page 8).

[20] J. Simon et al. "The Radicular Groove: Its Potential Clinical Significance". In: *Journal of Endodontics* 26.5 (May 2000), pages 295–298. DOI: 10.1097/00004770-200005000-00012. URL: http://dx.doi.org/10.1097/00004770-200005000-00012 (cited on page 8).

[21] D. D. Bosshardt and A. Nanci. "Immunocytochemical characterization of ectopic enamel deposits and cementicles in human teeth". In: *Eur. J. Oral Sci.* 111.1 (Feb. 2003), pages 51–59 (cited on pages 8, 10).

[22] S. Risnes. "Ectopic tooth enamel. An SEM study of the structure of enamel in enamel pearls". In: *Adv. Dent. Res.* 3.2 (Sept. 1989), pages 258–264 (cited on page 8).

[25] H. Colak et al. "Radiographic evaluation of the prevalence of enamel pearls in a sample adult dental population". In: *European Review for Medical and Pharmacological Sciences* 18.3 (2014), pages 440–444 (cited on page 9).

[27] M. DeSanctis and K. G. Murphy. "The role of resective periodontal surgery in the treatment of furcation defects". In: *Periodontol. 2000* 22 (Feb. 2000), pages 154–168 (cited on page 10).

[28] A. R. Goldstein. "Enamel pearls as contributing factor in periodontal breakdown". In: *J Am Dent Assoc* 99.2 (Aug. 1979), pages 210–211 (cited on page 10).

[29] Mathieu Aubry et al. "Fast Local Laplacian Filters: Theory and Applications". In: *ACM Trans. Graph.* 33.5 (Sept. 2014), 167:1–167:14. ISSN: 0730-0301. DOI: 10.1145/2629645. URL: http://doi.acm.org/10.1145/2629645 (cited on pages 11, 53).

[30] Sylvain Paris, Samuel W. Hasinoff, and Jan Kautz. "Local Laplacian filters". In: *Communications of the ACM* 58.3 (Feb. 2015), pages 81–91. DOI: 10.1145/2723694. URL: http://dx.doi.org/10.1145/2723694 (cited on pages 11, 53).

[31] Gayathri V and Hema P Menon. "Challenges in Edge Extraction of Dental X-Ray Images Using Image Processing Algorithms – A Review". In: *International Journal of Computer Science and Information Technologies* 5(4) (2014), pages 5355–5357 (cited on page 11).

[33] Yusra Y. Amer and Musbah J. Aqel. "An Efficient Segmentation Algorithm for Panoramic Dental Images". In: *Procedia Computer Science* 65 (2015), pages 718–725. DOI: 10.1016/j.procs.2015.09.016. URL: http://dx.doi.org/10.1016/j.procs.2015.09.016 (cited on page 11).

[34] D.E. Nassar et al. "Automatic Construction of Dental Charts for Postmortem Identification". In: *IEEE Transactions on Information Forensics and Security* 3.2 (June 2008), pages 234–246. DOI: 10.1109/tifs.2008.922452. URL: https://doi.org/10.1109%2Ftifs.2008.922452 (cited on page 11).

[57] Austin Roorda and David R. Williams. "The arrangement of the three cone classes in the living human eye". In: *Nature* 397 (Feb. 1999), pages 520–522. DOI: 10.1038/17383. URL: http://www.nature.com/nature/journal/v397/n6719/abs/397520a0.html (cited on page 18).

[58] J. Nathans, D. Thomas, and D. S. Hogness. "Molecular Genetics of Human Color Vision: The Genes Encoding Blue, Green, and Red Pigments". In: *Science* 232 (Apr. 1986), pages 193–202. DOI: 10.1126/science.2937147 (cited on page 18).

[59] Zhou Wang and D. Zhang. "Progressive switching median filter for the removal of impulse noise from highly corrupted images". In: *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 46.1 (1999), pages 78–80. DOI: 10.1109/82.749102. URL: https://doi.org/10.1109%2F82.749102 (cited on page 24).

[60] Karl Heinz Höhne and William A. Hanson. "Interactive 3D Segmentation of MRI and CT Volumes using Morphological Operations". In: *Journal of Computer Assisted Tomography* 16.2 (Mar. 1992), pages 285–294. DOI: 10.1097/00004728-199203000-00019. URL: https://doi.org/10.1097%2F00004728-199203000-00019 (cited on page 27).

[61] C.W. Chen, J. Luo, and K.J. Parker. "Image segmentation via adaptive K-mean clustering and knowledge-based morphological operations with biomedical applications". In: *IEEE Transactions on Image Processing* 7.12 (1998), pages 1673–1683. DOI: 10.1109/83.730379. URL: https://doi.org/10.1109%2F83.730379 (cited on page 27).

[62] Jian-Gang Wang and Eric Sung. "Frontal-view face detection and facial feature extraction using color and morphological operations". In: *Pattern recognition letters* 20.10 (1999), pages 1053–1068 (cited on page 28).

[63] K Somasundaram and T Kalaiselvi. "Automatic brain extraction methods for T1 magnetic resonance images using region labeling and morphological operations". In: *Computers in biology and medicine* 41.8 (2011), pages 716–725 (cited on page 28).

[64] Masato Masuya and Junta Doi. "Detection and geometric modeling of molecular surfaces and cavities using digital mathematical morphological operations". In: *Journal of molecular graphics* 13.6 (1995), pages 331–336 (cited on page 28).

[65] Nobuyuki Otsu. "A Threshold Selection Method from Gray-Level Histograms". In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (1979), pages 62–66. DOI: 10.1109/tsmc.1979.4310076. URL: http://dx.doi.org/10.1109/TSMC.1979.4310076 (cited on page 29).

[66] Pedro F Felzenszwalb and Daniel P Huttenlocher. "Efficient graph-based image segmentation". In: *International journal of computer vision* 59.2 (2004), pages 167–181 (cited on page 41).

[67]   Luc Vincent and Pierre Soille. "Watersheds in digital spaces: an efficient algorithm based on immersion simulations". In: *IEEE transactions on pattern analysis and machine intelligence* 13.6 (1991), pages 583–598 (cited on page 41).

[68]   Dorin Comaniciu and Peter Meer. "Mean shift: A robust approach toward feature space analysis". In: *IEEE Transactions on pattern analysis and machine intelligence* 24.5 (2002), pages 603–619 (cited on page 41).

[69]   Alex Levinshtein et al. "Turbopixels: Fast superpixels using geometric flows". In: *IEEE transactions on pattern analysis and machine intelligence* 31.12 (2009), pages 2290–2297 (cited on pages 41, 42).

[70]   Sylvain Paris, Samuel W Hasinoff, and Jan Kautz. "Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid." In: () (cited on pages 43–45).

[71]   M. Aubry et al. "Fast Local Laplacian Filters: Theory and Applications". In: *ACM Transactions on Graphics* (2014) (cited on page 43).

[72]   Michael Kass, Andrew Witkin, and Demetri Terzopoulos. "Snakes: Active contour models". In: *International Journal of Computer Vision* 1.4 (Jan. 1988), pages 321–331. DOI: 10.1007/bf00133570. URL: https://doi.org/10.1007%2Fbf00133570 (cited on page 45).

[73]   T.F. Chan and L.A. Vese. "Active contours without edges". In: *IEEE Transactions on Image Processing* 10.2 (2001), pages 266–277. DOI: 10.1109/83.902291. URL: http://dx.doi.org/10.1109/83.902291 (cited on page 46).

[74]   Shawn Lankton. "Sparse field methods-technical report". In: (2009) (cited on page 46).

[75]   Ross T. Whitaker. "A Level-Set Approach to 3D Reconstruction from Range Data". In: *Int. J. Comput. Vision* 29.3 (Sept. 1998), pages 203–231. ISSN: 0920-5691. DOI: 10.1023/A:1008036829907. URL: http://dx.doi.org/10.1023/A:1008036829907 (cited on page 46).

[76]   Pascal Getreuer. "Chan-vese segmentation". In: *Image Processing On Line* 2 (2012), pages 214–224 (cited on page 46).