EXTRAVASCULAR ULTRASOUND ANALYSIS

PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Biomedical Engineering.

by

T.UDAY KIRAN (03241A1126)
B.ADITYA (03241A1101)
P.SRINIVAS (03241A1123)
CH.SURESH (04245A1102)

Department of Biomedical Engineering
Gokaraju Rangaraju Institute of Engineering and Technology
(Affiliated to Jawaharlal Nehru Technological University)
Hyderabad
2007
Heartful salutations to the intellectuals who guided us for the successful completion of project titled “Extra Vascular Ultrasound Analysis (EVUS)”. We are thankful and express gratitude towards Prof. Ram Murthy garu (General Mathematics, GRIET), Prof. Jayanthi Sivaswamy garu (General Mathematics, IIIT), Dr. Chalapathi Rao garu (Chief Radiologist, KIMS), and Dr. Rajeshwar Rao garu (ABC Imaging) for extending technical guidance.

We are grateful to Prof. T.Padma (HOD, Biomedical Department) for giving the ‘birds eye view’ of the project. We are highly thankful to T. Purna Radhika, Asst. Prof (Project Guide) for her personal attention in guiding different sectors of the project and for guiding us at every aspect of our project development and duly explaining the importance and applications in day to day life.

Last but not the least we are blessed with the moral support of our parents in completion of the project.
ABSTRACT

Extra Vascular Ultrasound Analysis (EVUS) is a catheter based medical imaging technique particularly useful for studying atherosclerotic disease. It produces cross-sectional images of blood vessels that provide quantitative assessment of the vascular wall, information about the nature of atherosclerotic lesions as well as plaque shape and size. Automatic processing of large EVUS data sets represents an important challenge due to ultrasound speckle, transducer artifacts. The tomographical orientation of ultrasound enables visualization of the full 360° circumference of the vessel wall, so that lumen dimensions can be directly planimetered on a cross-sectional image. This allows precise assessment of the extent of disease in vessels that are often difficult to assess with angiography. It provides a reproducible, safe and sensitive method for assessing the development and extent of atherosclerosis, particularly in its earlier, pre-symptomatic stages.

It is an ideal technology for studying the progression, stabilization and potential regression of coronary atherosclerosis. Unlike angiography, EVUS is able to image atheroma within the vessel wall directly, allowing measurement of atheroma size, distribution and, to some extent, composition. In pathology, an atheroma (plural: atheromata) is an unhealthy (though typical for most humans) accumulation and swelling (-oma) with cells, or cell debris, which contain lipids (cholesterol and fatty acids), calcium mineral and a variable amount of fibrous connective tissue within the walls of arteries. In the context of heart or artery terminology, atheromata are commonly referred to as atheromatous plaques. Veins do not develop atheromata, unless surgically moved to function as an artery, as in bypass surgery.

It can assess arterial wall architecture and localize large intravascular deposits, but it does not provide quantitative chemical information, which is essential in the evaluation of atherosclerotic lesions. In the present study, we explore some benefits of EVUS to evaluate the intact arterial wall.
# CONTENTS

<table>
<thead>
<tr>
<th>Chapters</th>
<th>Page no</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledgement</td>
<td>i</td>
</tr>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Contents</td>
<td>iii</td>
</tr>
<tr>
<td>List of Figures</td>
<td>v</td>
</tr>
<tr>
<td>1. Introduction</td>
<td></td>
</tr>
<tr>
<td>1.1. Anatomy of blood vessel</td>
<td></td>
</tr>
<tr>
<td>1.1.1. Arteries</td>
<td></td>
</tr>
<tr>
<td>1.1.1.1. Elastic arteries</td>
<td>2</td>
</tr>
<tr>
<td>1.1.1.2. Muscular arteries</td>
<td>3</td>
</tr>
<tr>
<td>1.1.2. Veins</td>
<td>3</td>
</tr>
<tr>
<td>1.2. Blood flow</td>
<td>4</td>
</tr>
<tr>
<td>1.2.1. Strain</td>
<td>5</td>
</tr>
<tr>
<td>1.2.2. Laminar flow of blood</td>
<td>6</td>
</tr>
<tr>
<td>2. Video, Frames and Images</td>
<td></td>
</tr>
<tr>
<td>2.1. Video format</td>
<td></td>
</tr>
<tr>
<td>2.1.1. Common organisation of video formats</td>
<td>8</td>
</tr>
<tr>
<td>2.1.2. Blanking Region</td>
<td>9</td>
</tr>
<tr>
<td>2.2. Digital video formats</td>
<td></td>
</tr>
<tr>
<td>2.2.1. NTSC</td>
<td>9</td>
</tr>
<tr>
<td>2.2.2. PAL</td>
<td>10</td>
</tr>
<tr>
<td>2.2.3. SECAM</td>
<td>11</td>
</tr>
<tr>
<td>2.3. Video Compressions</td>
<td>13</td>
</tr>
<tr>
<td>2.4. Video structure in MATLAB</td>
<td>15</td>
</tr>
<tr>
<td>3. Tempoaral Median Filtering</td>
<td></td>
</tr>
<tr>
<td>3.1 Median</td>
<td>16</td>
</tr>
<tr>
<td>3.1.1. Definition</td>
<td>17</td>
</tr>
<tr>
<td>3.1.2. Approach</td>
<td>17</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>3.2 Steps involved</td>
<td>18</td>
</tr>
<tr>
<td>4. Image Segmentation</td>
<td>19</td>
</tr>
<tr>
<td>4.1. Standard Deviation</td>
<td>19</td>
</tr>
<tr>
<td>4.2. Mean</td>
<td>21</td>
</tr>
<tr>
<td>4.3. Decision Function</td>
<td>21</td>
</tr>
<tr>
<td>5. Noise Reduction</td>
<td>23</td>
</tr>
<tr>
<td>6. Convex Hull</td>
<td>25</td>
</tr>
<tr>
<td>6.1. Morphological operations based convex hull</td>
<td>25</td>
</tr>
<tr>
<td>6.1.1. Dilations &amp; Erosions</td>
<td>25</td>
</tr>
<tr>
<td>6.1.2. Hit or Miss transformation</td>
<td>26</td>
</tr>
<tr>
<td>6.2. Quick Hull algorithm</td>
<td>27</td>
</tr>
<tr>
<td>6.3. Gift wrapping algorithm</td>
<td>28</td>
</tr>
<tr>
<td>6.4. Graham’s scan</td>
<td>29</td>
</tr>
<tr>
<td>6.5. Convex hull in ultrasound images</td>
<td>32</td>
</tr>
<tr>
<td>7. Skeletonisation</td>
<td>33</td>
</tr>
<tr>
<td>7.1. Pre – processing</td>
<td>33</td>
</tr>
<tr>
<td>7.1.1. Binary Thresholding</td>
<td>33</td>
</tr>
<tr>
<td>7.1.2. Convex Hull</td>
<td>34</td>
</tr>
<tr>
<td>7.1.3. Region Filling</td>
<td>34</td>
</tr>
<tr>
<td>7.2. Procedure</td>
<td>35</td>
</tr>
<tr>
<td>7.3. Steps involved</td>
<td>36</td>
</tr>
<tr>
<td>8. Camera Correction Using Affine Transformation</td>
<td>37</td>
</tr>
<tr>
<td>8.1. Translation</td>
<td>37</td>
</tr>
<tr>
<td>8.2. Rotation</td>
<td>38</td>
</tr>
<tr>
<td>8.3. Shearing</td>
<td>38</td>
</tr>
<tr>
<td>8.4. Perspective projection</td>
<td>39</td>
</tr>
<tr>
<td>9. Polar Transformation</td>
<td>40</td>
</tr>
<tr>
<td>9.1. Definition</td>
<td>40</td>
</tr>
<tr>
<td>9.2. Polar transformation of geometric figures</td>
<td>41</td>
</tr>
<tr>
<td>Appendix</td>
<td>43</td>
</tr>
<tr>
<td>References</td>
<td>56</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure Number</th>
<th>Description</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Vessel characteristics along a blood pulse</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>Ultrasound image of vessel at different instants of time</td>
<td>2</td>
</tr>
<tr>
<td>1.3</td>
<td>Layers of an artery</td>
<td>3</td>
</tr>
<tr>
<td>1.4</td>
<td>Structure of a artery and vein</td>
<td>4</td>
</tr>
<tr>
<td>1.5</td>
<td>Isotrophic and anisotropic contractions</td>
<td>6</td>
</tr>
<tr>
<td>2.1</td>
<td>Video Compressions</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>Structure of an AVI file in MATLAB</td>
<td>15</td>
</tr>
<tr>
<td>3.1</td>
<td>Median of a 3 x 3 box</td>
<td>16</td>
</tr>
<tr>
<td>3.2</td>
<td>Median Filtered image using a 3 x 3 box</td>
<td>16</td>
</tr>
<tr>
<td>3.3</td>
<td>Shaken frame</td>
<td>17</td>
</tr>
<tr>
<td>3.4</td>
<td>Representation of the temporal median filtering</td>
<td>18</td>
</tr>
<tr>
<td>4.1</td>
<td>STDs and Means at various box in the image</td>
<td>20</td>
</tr>
<tr>
<td>4.2</td>
<td>Distribution of values in various standard deviations of a normal distribution curve</td>
<td>21</td>
</tr>
<tr>
<td>4.3</td>
<td>Image Segmentation for various STDs and Means</td>
<td>22</td>
</tr>
<tr>
<td>5.1</td>
<td>Intersection of 3 consecutive frames</td>
<td>23</td>
</tr>
<tr>
<td>5.2</td>
<td>Intersection of images</td>
<td>24</td>
</tr>
<tr>
<td>6.1</td>
<td>Rubber-band explanation of Convex hull</td>
<td>25</td>
</tr>
<tr>
<td>6.2</td>
<td>Uniform and Non-uniform Dilations</td>
<td>26</td>
</tr>
<tr>
<td>6.3</td>
<td>Morphological Convex Hull operations</td>
<td>27</td>
</tr>
<tr>
<td>6.4</td>
<td>Quick Hull flowchart</td>
<td>27</td>
</tr>
<tr>
<td>6.5</td>
<td>Quick Hull representation</td>
<td>28</td>
</tr>
<tr>
<td>6.6</td>
<td>Gift – wrapping algorithm in 2-D</td>
<td>28</td>
</tr>
<tr>
<td>6.7</td>
<td>Gift – wrapping of a set of data</td>
<td>29</td>
</tr>
<tr>
<td>6.8</td>
<td>Grahams’s Scan</td>
<td>30</td>
</tr>
<tr>
<td>6.9</td>
<td>Flow chart of Graham’s Scan</td>
<td>31</td>
</tr>
<tr>
<td>6.10</td>
<td>Convex of an Image using Qhull</td>
<td>32</td>
</tr>
<tr>
<td>Figure Number</td>
<td>Description</td>
<td>Page No.</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>7.1</td>
<td>Binary Thresholding</td>
<td>34</td>
</tr>
<tr>
<td>7.2</td>
<td>Skeletons of few geometric figure</td>
<td>35</td>
</tr>
<tr>
<td>7.3</td>
<td>(a) Input Image (b) Region Filled Image (c) Skeleton of the image</td>
<td>36</td>
</tr>
<tr>
<td>8.1</td>
<td>Affine Transformations</td>
<td>37</td>
</tr>
<tr>
<td>8.2</td>
<td>Translation &amp; Rotation of images</td>
<td>38</td>
</tr>
<tr>
<td>8.3</td>
<td>Shearing of images</td>
<td>38</td>
</tr>
<tr>
<td>8.4</td>
<td>Perspective Projections</td>
<td>39</td>
</tr>
<tr>
<td>9.1</td>
<td>Cartesian to Polar Conversion</td>
<td>40</td>
</tr>
<tr>
<td>9.2</td>
<td>Polar Plot different geometric figure</td>
<td>41</td>
</tr>
<tr>
<td>9.3</td>
<td>Radial Distance plot of a point on the vessel with $\theta = 30$</td>
<td>42</td>
</tr>
</tbody>
</table>
CHAPTER 1
INTRODUCTION

The anatomical layers of the blood vessel facilitate the pulsatile flow of the blood. The muscular structures of the tunics, namely elastin and collagen by their H-H relaxation process justify the pumping mechanism. The blood flow characteristics depend on these contractions. Abnormalities like artherosclerosis and vascular infarctions have anisotropic contractions along a cross-section. A mathematical approach of determining this elastic property is done by using an ultrasound video of sagittal cross-sectional view of a vessel. The rate of change of a vessel “inside-out” gives the strain relationship at each point on the vessel. This can be achieved by a frame by frame analysis of the video.

Figure 1.1. (a) Vessel characteristics along a blood pulse (b) Cross-sectional view of a blood vessel

The above figure 1(a) shows the contractions and relaxations of the vessel of a blood pulse for various cross-sections. Clearly, it can be noticed that the track of the point on the vessel, about a period of time, gives the stretch observations. The flow of the blood in a vessel is laminar, since blood is a newtonian fluid. The pressure variations across the vessel radially are shown in figure 1(b).
Observing a point on the vessel and plotting its distance from reference in each frame, gives the strain curve of that point. The frequency of this curve gives the elastin-collagen composition.

Speckle is a random, deterministic, interference pattern in an image formed with coherent radiation of a medium containing many sub-resolution scatterers. The texture of the observed speckle pattern does not correspond to underlying structure. The local brightness of the speckle pattern, however, does reflect the local echogenicity of the underlying scatterers.

**1.1. ANATOMY OF BLOOD VESSEL**

Blood vessel can be structurally classified in five types - arteries, arterioles, capillaries, venules, and veins. Arteries carry blood away from the heart to other organs. Large, elastic arteries depart from the heart and divide into medium-sized, muscular arteries that branch out into the various regions of body. These medium sized arteries split up into small arteries and further to, arterioles and lastly to myriad tiny vessels called capillaries. Thin walls of capillaries allow the exchange of substances between the blood and body tissue. Group of capillaries within a tissue reunite progressively to form larger blood vessels called veins. Veins, thus, are the terminating channels of the systemic circulatory path.
1.1.1. ARTERIES

The wall of an artery has three coats or tunics:

1) Tunica Interna: It consists of a lining of simple squamous epithelium called endothelium (a basement membrane). The endothelium is a continuous layer of cells that lines the inner surface of the entire cardiovascular system.

2) Tunica Media: It is the thickest layer. It consists of elastic fibers and smooth muscle fibers that extend circularly around the lumen. Due to their plentiful elastic fibers, arteries normally have high compliance, which means that their walls easily stretch or expand without tearing in response to a small increase in pressure.

3) Tunica Externa or Adventitia: It is mainly composed of elastic and collagen fibers. In muscular arteries, an external elastic lamina composed of elastic tissue separates the tunica externa from the tunica media.

1.1.1.1. Elastic arteries

The largest diameter arteries, termed elastic arteries because the tunica media contains a high proportion of elastic fibers, have walls that are relatively thin in proportion to their overall diameter. Elastic arteries perform an important function. They help propel blood onward while the ventricles are relaxing. As blood is ejected from the heart into elastic arteries, their highly elastic walls stretch, accommodating the surge of blood. By stretching, the elastic fibers store mechanical energy for a short period of time, functioning as a pressure reservoir. These fibers recoil and convert stored potential energy in the vessel into kinetic energy of the blood. Thus, blood continues to move through the arteries even while the ventricles are relaxed. They are also are called conducting arteries because they conduct blood from the heart to medium-sized, muscular arteries.

1.1.1.2. Muscular arteries

Muscular arteries have from 8 to 40 layers of smooth muscle in their media. Arteries have about a 1:1 ratio of media to adventitia, with more than half the wall being
smooth muscle. The adventitia is mostly collagen fibers (some elastic fibers,) fibroblasts, and clusters of smooth muscle cells. They are also called as distributive arteries as they are the channels for the branched flow of blood.

Atherosclerosis may occur in both elastic and muscular arteries, but does not develop in arterioles. In the coronary artery the major extramural vessels are most prone to disease, especially at the sites of branching and turbulent flow. Susceptibility of disease declines as the size of the vessel decreases. Abnormalities in function of arterioles and smaller branches of the coronary arteries may nevertheless account for some forms of angina in which significant atherosclerosis do not appear to be involved.

![Figure 1.4 Structure of a artery and vein](image)

1.1.2. Veins

Veins are composed of essentially the same three coats as arteries; the relative thicknesses of the layers are different. The tunica interna of veins is thinner than that if arteries; the tunica media of the veins is much thinner than arteries, with relatively smooth muscle and elastic fibers. Furthermore, the lumen of a vein is larger than that of a
comparable artery. Many veins especially those situated in limbs also feature valves, which are thin folds of tunica interna that form flap like cusps. These cusps project into the lumen pointing towards the heart.

A vascular sinus is a vein with a thin endothelial wall that has no smooth muscle to alter its diameter. Another example of a vascular sinus is the coronary sinus of the heart.

1.2. BLOOD FLOW

Blood flow is the volume of blood that flows through any tissue in a given time period (in ml/min). Total blood flow is the cardiac output (CO), the volume of blood that circulates through systemic blood vessels each minute. Cardiac output (CO)=heart (HR) x stroke volume (SV).

Factors affecting cardiac output:

1) the pressure difference that drives the blood flow through a tissue
2) the resistance to blood flow in specific blood vessels.

Blood flows from regions of higher to lower pressure; the greater the pressure difference greater the flow. The higher the resistance, by contrast, the smaller the blood flow.

1.2.1. STRAIN

Deformation of a solid that can be related to stresses is described by strain. Take a string of initial length Lo. If it is stretched to a length L, it is natural to describe the change by dimensionless ratios such as L/Lo, (L-Lo)/Lo, (L-Lo)/L. The use of dimensionless ratios eliminates the absolute length from consideration. The ratio L/Lo is called the stretch ratio and is denoted by the symbol λ (lambda).

The ratios e=L-Lo/Lo;  e’=L-Lo/L
In figure (b), the original dimension is $r_o$, and the change in dimension is $r_1 - r_o$. So, the strain of that point is

$$\text{Strain} = \frac{r_1 - r_o}{r_o}$$

The strain for various points on the vessel is calculated and tabulated. The strain for that point at any instant can be observed. The difference between the anisotropic and isotropic contractions can be clearly observed from this table.

### 1.2.2. LAMINAR FLOW OF BLOOD

Assume that blood vessel is of cylindrical in shape and the flow to be laminar. Considering the tube is long and the flow is steady, so that the conditions of flow change neither with the distance along the tube, nor with the time. Under these assumptions we can analyze the flow with a simple adhoc procedure.

Even polar co-ordinates is used for such kind of issues. The polar axis coincides with the axis of the cylinder. The flow obeys Navier-Stokes equations of motion of an incompressible fluid. The boundary condition is that blood adheres to the tube wall. Since the boundary condition is axisymmetric, the flow is also axisymmetric and the only nonvanishing component of velocity is $U(r)$ in the axial direction; $U(r)$ is a function of $r$ alone and not of $x$. 
Isolate a cylindrical body of fluid of radius $r$ and unit length in the axial direction. This body is subjected to a pressure $p_1$ on the left hand end, $p_2$ on the right end and shear stress $\tau$ on the circumferential surface. Since $p_1 - p_2 = -L \frac{dp}{dx}$ acts on an area $\pi r^2$, and $\tau$ acts on an area $L \pi r$ we have for equilibrium, the balance of forces

$$\tau \cdot 2\pi r = - \pi r^2 \cdot \frac{dp}{dx}$$

or

$$\tau = - (r/2) \frac{dp}{dx}$$
CHAPTER 2
VIDEO, FRAMES & IMAGES

2.1. VIDEO FORMAT

A video format describes how one device sends a video picture to another device, such as the way that a DVD player sends pictures to a television or a computer to a monitor. More formally, the video format describes the sequence and structure of frames that create the moving video image.

Video formats are commonly known in the domain of commercial broadcast and consumer devices; most notably to date, these are the analog video formats NTSC, PAL, and SECAM. However, video formats also describe the digital equivalents of the commercial formats, the aging custom military uses of analog video (such as RS-170 and RS-343), the increasingly important video formats used with computers, and even such offbeat formats such as color field sequential.

Video formats were originally designed for display devices such as CRTs. However, because other kinds of displays have common source material and because video formats enjoy wide adoption and have convenient organization, video formats are a common means to describe the structure of displayed visual information for a variety of graphical devices.

2.1.1. COMMON ORGANISATION OF VIDEO FORMATS

A video format describes a rectangular image carried within an envelope containing information about the image. Although video formats vary greatly in organization, there is a common taxonomy. Video formats use a sequence of frames in a specified order. In some formats, a single frame is independent of any other (such as those used in computer video formats), so the sequence is only one frame. In other video formats (such as the Bruch sequence in PAL), frames have an ordered position. Individual frames within a sequence typically have similar construction. However,
depending on its position in the sequence, frames may vary small elements within them to represent additional information.

A frame consists of a rectangular series of lines, sometimes known as scan lines. Lines have a regular and consistent length in order to produce a rectangular image. To achieve this, in analog formats, a line lasts for a given period of time; in digital formats, the line consists of a given number of pixels. When a device sends a frame, the video format usually specifies that devices send each line independently from any others and that all lines are sent in top-to-bottom order.

A frame can consist of two or more fields, sent separately, that assemble together to form a rectangular picture. This kind of assembly is known as interlace. An interlaced video frame distinguishes itself from the progressive scan frame where the entire frame sends as a single intact entity.

Analog video formats:

2.1.2. BLANKING REGION

The video format consists of more information than the visible content of the frame. Preceding and following the image are lines and pixels containing synchronization information or a time delay. This surrounding margin is known as a blanking interval; the horizontal and vertical front porch and back porch are the building blocks of the blanking interval.

2.2. Digital Video Formats

There are three types of formats used worldwide. They are:

1) NTSC (National Television System Committee).
2) PAL (Phase Alternating Line).
3) SECAM (Sequential Couleur Avec Memoire).

2.2.1. NTSC

The NTSC format is used with the M format (see broadcast television systems), which consists of 29.97 interlaced frames of video per second. Each
frame consists of 484 lines out of a total of 525 (the rest are used for sync, vertical retrace, and other data such as captioning). PAL uses 625 lines, and so has a better picture quality. The NTSC system interlaces its scan lines, drawing odd-numbered scan lines in odd-numbered fields and even-numbered scan lines in even-numbered fields, yielding a nearly flicker-free image at its approximately 59.94 hertz (nominally 60 Hz/100.1%) refresh frequency. The refresh compares favorably to the 50 Hz refresh rate of the PAL and SECAM video formats used in Europe, where 50 Hz alternating current is the standard; flicker was more likely to be noticed when using these standards until modern PAL TV sets began using 100 Hz refresh rate to eliminate flicker. This produces a far more stable picture than native NTSC and PAL had, effectively displaying each frame twice. This did, at first, because some motion problems, so it was not universally adopted until a few years ago. Interlacing the picture does complicate editing video, but this is true of all interlaced video formats, including PAL and SECAM.

### Table 1.1 Characteristics of NTSC format

<table>
<thead>
<tr>
<th>Lines/Field</th>
<th>525/60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horizontal frequency</td>
<td>15.734 kHz</td>
</tr>
<tr>
<td>Vertical frequency</td>
<td>60 Hz</td>
</tr>
<tr>
<td>Color subcarrier frequency</td>
<td>3.579545 MHz</td>
</tr>
<tr>
<td>Video bandwidth</td>
<td>4.2 MHz</td>
</tr>
<tr>
<td>Sound carrier</td>
<td>4.5 MHz</td>
</tr>
</tbody>
</table>

#### 2.2.2. PAL

The basics of PAL and the NTSC system are very similar; a quadrature amplitude modulated subcarrier carrying the chrominance information is added to the luminance video signal to form a composite video baseband signal (CVBS). The frequency of this subcarrier is typically 4433618.75 Hz (approximately 4.43 MHz) for PAL, compared to approximately 3.58 MHz for NTSC. The SECAM system, on the other hand, uses a frequency modulation scheme on its color subcarrier. The name "Phase Alternating Line" describes the way that the phase of part of the color information on the
video signal is reversed with each line, which automatically corrects phase errors in the transmission of the signal by cancelling them out. (Lines where the color phase is reversed compared to NTSC are often called PAL or phase-alternation lines, which justifies one of the expansions of the acronym, while the other lines are called NTSC lines.) Early PAL receivers relied on the imperfections of the human eye to do that canceling; however this resulted in a comb-like effect on larger phase errors. Thus, most receivers now use a chrominance delay line, which stores the received color information on each line of display; an average of the color information from the previous line and the current line is then used to drive the picture tube. The effect is that phase errors result in saturation changes, which are less objectionable than the equivalent hue changes of NTSC. A minor drawback is that the vertical color resolution is poorer than the NTSC system's, but since the human eye also has a color resolution that is much lower than its brightness resolution, this effect is not visible. In any case, NTSC, PAL and SECAM all have chrominance bandwidth (horizontal color detail) reduced greatly compared to the luminance signal.

### Table 1.2 Characteristics of PAL format

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>PAL</th>
<th>PAL N</th>
<th>PAL M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lines/Field</td>
<td>625/50</td>
<td>625/50</td>
<td>525/60</td>
</tr>
<tr>
<td>Horizontal frequency</td>
<td>15.625 kHz</td>
<td>15.625 kHz</td>
<td>15.75 kHz</td>
</tr>
<tr>
<td>Vertical frequency</td>
<td>50 Hz</td>
<td>50 Hz</td>
<td>60 Hz</td>
</tr>
<tr>
<td>Color carrier frequency</td>
<td>4.433618 MHz</td>
<td>3.582056 MHz</td>
<td>3.575611 MHz</td>
</tr>
<tr>
<td>Video bandwidth</td>
<td>5 MHz</td>
<td>4.2 MHz</td>
<td>4.2 MHz</td>
</tr>
<tr>
<td>Sound carrier</td>
<td>5.5 MHz</td>
<td>4.5 MHz</td>
<td>4.5 MHz</td>
</tr>
</tbody>
</table>

#### 2.2.3. SECAM

Just as the other color standards adopted for broadcast usage over the world, SECAM is a compatible standard, which means that monochrome television
receivers predating its introduction are still able to show the programs, although only in black and white. Because of this compatibility requirement, color standards add a second signal to the basic monochrome signal, and this signal carries the color information, called chrominance or C in short, while the black and white information is called the luminance (Y in short). Old TV receivers only see the luminance, while color receivers process both signals.

Additionally, for compatibility, it is required to use no more bandwidth than the monochrome signal alone; the color signal has to be somehow inserted into the monochrome signal, without disturbing it. This insertion is possible because the spectrum of the monochrome TV signal is not continuous, hence empty space exists which can be utilized. This lack of continuity results from the discrete nature of the signal, which is divided into frames and lines. Analogue color systems differ by the way in which empty space is used. In all cases, the color signal is inserted at the end of the spectrum of the monochrome signal.

In order to be able to separate the color signal from the monochrome one in the receiver, a fixed frequency sub carrier has to be used, this sub carrier being modulated by the color signal. The color space is three dimensional by the nature of the human vision, so after subtracting the luminance, which is carried by the base signal, the color sub carrier still has to carry a two dimensional signal. Typically the red (R) and the blue (B) information are carried because their signal difference with luminance (R-Y and B-Y) is stronger than of green (G-Y). SECAM differs from the other color systems by the way the R-Y and B-Y signals are carried. First, SECAM uses frequency modulation to encode chrominance information on the sub carrier. Second, instead of transmitting the red and blue information together, it only sends one of them at a time, and uses the information about the other color from the preceding line. It uses a delay line, an analog memory device, for storing one line of color information. This justifies the name Sequential with Memory.
Table 1.3 Characteristics of SECAM format

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>SECAM B,G,H</th>
<th>SECAM D,K,K1,L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lines/Field</td>
<td>625/50</td>
<td>625/50</td>
</tr>
<tr>
<td>Horizontal frequency</td>
<td>15.625 kHz</td>
<td>15.625 kHz</td>
</tr>
<tr>
<td>Vertical frequency</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>Video bandwidth</td>
<td>5 MHz</td>
<td>6 MHz</td>
</tr>
<tr>
<td>Sound carrier</td>
<td>5.5 MHZ</td>
<td>6.5 MHZ</td>
</tr>
</tbody>
</table>

2.3. VIDEO COMPRESSIONS

The compressions involved in the AVI formats falls into four categories

- Resolution: The image in each frame of the video is reduced in size and the number of pixels in the image are reduced. The general zooming algorithms and bi-linear transformation algorithms can be used to manipulate the resolution.

- Frame Rate: Temporal resolution can be defined as number of frames per second (fps). The more fps, the more space required to store the video.

- Pixel Size: The number of bits used to represent an image is determined the quantization levels of the image. More quantization level involves more number of bits. More than a certain limit of quantization, there is no other detail added to the image. In such conditions, the quantization levels may be reduced.

- Color: The image can be represented in 3 different domains (Red, Green, Blue), (Hue, Saturation, Value) and (Cyan, Magenta, Yellow, Constant). These three representations require different amounts bits to picture.

- Image Compression: The high frequency components in an image contribute to the edge details in the image. The JPEG image compression works for removing the high frequency content and restoring the low frequency components.

Figure 2.1 shows the space (MB) occupied on the hard disk when different types of compression algorithms are employed on the image.
Table 1.4 Video Compressions

<table>
<thead>
<tr>
<th>Reduction</th>
<th>Video Format</th>
<th>Size (MB)</th>
<th>Reduce</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolution</td>
<td>640x480, 30 fps, 24-bit</td>
<td>26.37</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>320x240, 30 fps, 24-bit</td>
<td>6.59</td>
<td>25%</td>
</tr>
<tr>
<td></td>
<td>160x120, 30 fps, 24-bit</td>
<td>1.65</td>
<td>6%</td>
</tr>
<tr>
<td>Frame Rate</td>
<td>320x240, 15 fps, 24-bit</td>
<td>3.30</td>
<td>13%</td>
</tr>
<tr>
<td></td>
<td>320x240, 10 fps, 24-bit</td>
<td>2.20</td>
<td>8%</td>
</tr>
<tr>
<td>Pixel Size</td>
<td>320x240, 10 fps, 16-bit</td>
<td>1.46</td>
<td>6%</td>
</tr>
<tr>
<td>Color Format</td>
<td>320x240, 10 fps, 9-bit</td>
<td>0.82</td>
<td>3%</td>
</tr>
<tr>
<td></td>
<td>320x240, 10 fps, 12-bit</td>
<td>1.10</td>
<td>4%</td>
</tr>
<tr>
<td>Compression</td>
<td>320x240, 10 fps, JPEG 90%</td>
<td>0.29</td>
<td>1%</td>
</tr>
<tr>
<td></td>
<td>320x240, 10 fps, JPEG 75%</td>
<td>0.18</td>
<td>1%</td>
</tr>
</tbody>
</table>

Figure 2.1 Video Compressions
An ultrasound video of the movement of the vessel may have any type of video compressed format, but the detail which provides information about the edge is not lost. So, any AVI formatted video file can be used for the EVUS analysis.

### 2.4. VIDEO STRUCTURE IN MATLAB

AVI can be imported to MATLAB workspace using `aviread` command. The video is represented in the format of a data structure, which can read by using the pointer method. The structure of the video is shown in figure 2.2.

![Figure 2.2. Structure of an AVI file in MATLAB](image)

Cdata in the each frame contains the image data. It can be called by accessing using a pointer, `Video(1,1).cdata` or `Video(1,1).colormap`. This image may be a true color image or a RGB indexed image. A true color image doesn’t need any colormap, so the colormap of a truecolor image is NULL. Whereas for a RGB indexed image, a colormap exist. So, the video in the format of DAT has colormap of 64 x 3. These images can be converted to truecolor image using `ind2rgb` command.
CHAPTER 3

TEMPORAL MEDIAN FILTERING

3.1. MEDIAN:

In a random set of data, the median resembles the value which has the highest probability of occurrence in the random set. The median of a finite list of numbers can be found by arranging all the observations from lowest value to highest value and picking the middle one. If there are even number of observations, the median is not unique, so one often takes the mean of the two middle values.

\[
\begin{array}{ccc}
3 & 3 & 4 \\
4 & 87 & 4 \\
4 & 5 & 5 \\
\end{array}
\]

Sorted Pixel values: 3, 3, 4, 4, 4, 5, 5, 87
Median filtered value: 4

Figure 3.1 Median of a 3 x 3 box

Figure 3.2. Median Filtered image using a 3 x 3 box

Considering the matrix shown here, the median value is 4, which is the most common value in a box of 3x3. Statistically speaking, the value 87 makes an abrupt step change in intensity. Instead, these variations can be made smooth by using this median filtering technique.

Films are made up of a series of individual images called frames. When these images are shown rapidly in succession, a viewer has the illusion that motion is occurring. The viewer cannot see the flickering between frames due to an effect known as persistence of vision - whereby the eye retains a visual image for a fraction of a second.
after the source has been removed. Viewers perceive motion due to a psychological effect called beta movement.

A human eye is a perfect sampler and quantizer, and perceives motion at a greater resolution than digital cameras. When the camera moves, it is pretty obvious that the video contains discernible shaken frames. In a continuous video, the frame which has been shakeout has a different characteristic than that of the other frames, as the value 87 occurred in a box of 3.

3.1.1. DEFINITION:

The word ‘Temporal’ means relating to measured time. A median filtering done on frame-by-frame decomposition is temporal median filtering (TMF). Spatial median filtering is done on a box of 3 in a single image, whereas temporal median filtering is done on same pixel of different frames.

The shakeout frames in an ultrasound video are removed. Now, the video is ready for further processing, since none of the frames have deviant characteristics. So, if an algorithm converges for a single frame, it does for all.

3.1.2. APPROACH:

The rate of the change of the subject in the image is a limitation for TMF. If the subject moves slowly, the capture time is more, hence more number of frames can capture the movement. When TMF is done on 5-7 frames, the shakeout frame can be removed at a greater ease. Rather, if the subject shifts faster, the TMF done on 3 frames
may not give a good result. Camera shake and movement of the subject, in the subsequent images, has similar characteristics in the video. However, if there is greater temporal resolution, i.e., more number of frames for the same movement of the subject, the camera shakes can be identified with simplicity. Often camera shakes occur within 2-3 frames of a video, whereas motion of the subject is captured in 7-10 frames.

### 3.2. STEPS INVOLVED

1. Picking the intensity values of (1,1) of frames N to N+5.
2. Arrange these in a 5x1 matrix accordingly.
3. Sort the matrix in the ascending order.
4. Pick out the middle element and assign it to the (1,1) of the N th frame of the output video.
5. For each pixel of the frame repeat steps 2 to 4 until entire image is detailed.
6. Repeat steps 1 to 5 for the N+1 th frame.

![Figure 3.4 Representation of the temporal median filtering](image-url)
CHAPTER 4
IMAGE SEGMENTATION

Segmentation refers to the process of partitioning a digital image into multiple regions (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Scattering from red blood cells (blood noise) increases significantly as the ultrasound frequency is increased above 10 MHz. This reduces the contrast between the vessel wall and the lumen in ultrasound imaging which makes it difficult to localize the vessel wall and plaque.

Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. The result of image segmentation is a set of regions that collectively cover the entire image, or a set of contours extracted from the image. Each of the pixels in a region is similar with respect to some characteristic or computed property, such as color, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristic(s).

A speckle pattern is a random intensity pattern produced by the mutual interference of coherent wave fronts that are subject to phase differences and/or intensity fluctuations. They produce random distributed points on the image. In an ultrasound image, the echo reflected from an obstacle, produces dense patterns and others produce sparse patterns. Speckle has a negative impact on ultrasound imaging.

4.1. STANDARD DEVIATION

In probability and statistics, the Standard Deviation (STD) of a probability distribution or multiset of values is a measure of the spread of its values. It is usually denoted with the letter σ (lower case sigma). It is defined as the root mean square (RMS) deviation of values from their arithmetic mean. The number of values greater than, 0.707 times of the maximum value in a 3x3 box, compared with total number of values in the box gives the standard deviation. Thresholding the STD of a 3x3 box gives an output for less variation in the pixel values.
Figure 4.1 STDs and Means at various boxes in the image
From the figure shown, it can be observed that the Pixel Distribution Curve at different locations of the image is different. Clearly, the STD at the parts of the vessel is less and also at the center of the vessel (blood) is less. So, this decision function (STD) solely cannot determine the required vessel area.

4.2. MEAN

The arithmetic mean is the "standard" average, often simply called the "mean".

\[ \bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i \]

The mean is the arithmetic average of a set of values, or distribution. However, for skewed distributions, the mean is not necessarily the same as the middle value (median), or the most likely (mode). The mean is an average of all characteristics in the box. So, a box containing more number of higher intensity values has a higher mean. Noises like speckle noise, present in the ultrasound sketch, is composed of packets of irregular intensity values which can be eliminated by a mean threshold.

4.3. DECISION FUNCTION

Considering these two decision functions, mean and STD, the vessel area can be extracted with ease. The characteristics of the vessel have greater mean and lesser STD. If there are speckles in the image area then a 3x3 box may not be the right choice. A box of higher dimensions can however solve this problem but makes a square effect of the vessel boundaries. Altering the mean threshold states the thickness of the vessel area.
Chapter 4. Image Segmentation

Figure 4.3 Image Segmentation for various STDs and Means
CHAPTER 5
NOISE REDUCTION

The blood noise in an ultrasound image occurs randomly in different captures. The vessel has a perfect absorption coefficient, and gives a definite structure. The noise occurring in two consecutive frames is never same. A continuous or intermittent noise signal is reflected by scatterers in the fluid, detected, and correlated with a delayed version of itself. One of the ways to eliminate this randomly occurring noise is by an intersection operation. ANDing two consecutive frames, removes the noise but preserves the vessel characteristics. The noise with a low standard deviation can be removed by ANDing two frames, while a noise with higher standard deviations are eliminated by considering more number of frames. However, the limitation for ANDing the number of frames is the movement of the vessel. If more frames are operated, the vessel tissue having faster movement may also be cropped.

Figure 5.1 Intersection of 3 consecutive frames.

The above figure shows that the cropping of the object when an intersection operator is done. It can be inferred that ANDing more frames removes noise at a greater extent but minute details of the vessels are also trimmed.

The noise present in an ultrasound image is just due to the scattering inaccuracy of the echoes. The scattering can never be defined on a perfect theory, as the blood itself moves randomly.
Figure 5.2 shows intersection operation done on 3 consecutive frames and the resulted image does not contain the noise. The corresponding spots circled in the images shows the difference from the output.
CHAPTER 6
CONVEX HULL

In mathematics, the convex hull or convex envelope for a set of points \( X \) in a real vector space \( V \) is the minimal convex set containing \( X \). (Note that \( X \) may be the union of any set of objects made of points). To show this exists, it is necessary to see that every \( X \) is contained in at least one convex set (the whole space \( V \), for example), and any intersection of convex sets containing \( X \) is also a convex set containing \( X \). It is then clear that the convex hull is the intersection of all convex sets containing \( X \), which is an alternative definition. More directly, the convex hull of \( X \) can be described constructively as the set of convex combinations of points from \( X \): that is, the set of points of the form \( \sum_{j=1}^{n} t_j x_j \), where \( n \) is an arbitrary natural number, the numbers \( t_j \) are non-negative and sum to 1, and the points \( x_j \) are in \( X \). It is simple to check that this set satisfies either of the two definitions above.

Imagine that your points are nails sticking into a board. If you stretched a rubber band completely around your points, so that all the points were inside the perimeter of the rubber band, the nails that the rubber band touches are the points of a polygon that is called the convex hull.

6.1. MORPHOLOGICAL OPERATIONS BASED CONVEX HULL

6.1.1. DILATION & EROSION

The expansion of an image in all directions is called dilation. If the rate of dilation is equal in all directions it is called uniform dilation, else non-uniform dilation.
- Choose a 3x3 box in an input image I, say A.
- If the centre pixels of the structure element B and A are both 1’s then C=A∪B, else C=B.
- Convoluted the above 2 steps over the image, resulting in a dilated image of I.

\[
C = A \cup B
\]

Convolute the above 2 steps over the image, resulting in a dilated image of I.

Erosion is similar to dilation, but the dilation is done inwards i.e., eating away the borders. This can be represented as

\[
I \ominus B = [I^c \oplus B]^c
\]

The compliment of input image is dilated using the same structuring element and the result is again complemented.

### 6.1.2. HIT OR MISS TRANSFORMATION

Initially, Hit-Miss Transformation is used in morphological convex hulling for defining the structure of element’s boundary. If the box A is congruent with structuring element B then it called a hit, otherwise it is a miss.

1. Perform Hit or Miss Transformation on I and add it to I.
2. Repeat step 1, until there is no further change
3. Rotate the structuring element in steps of 90°-180°-270°-360° and perform steps 1 & 2.
6.2. QUICK HULL ALGORITHM

Figure 6.3 Morphological Convex Hull operations

Figure 6.4 Quick Hull flowchart
1. At each step we select an edge (shown in light blue), find the associated point that is furthest away (point A) and add it to the hull. Then we delete the selected edge.

2. Add an edge to connect the new point to the remainder of the old hull. The points associated with the deleted edge are reassigned to the new edge if they can see it. (Point B in this case.)

3. Add the second edge to finish connecting the new point to the remainder of the old hull. The remaining points associated with the deleted edge are reassigned to this new edge if they can see it. (In this case there are no such points.) Points that can't see any of the new edges are inside the hull and can be discarded. (In this case, just point C).

4. Select a new edge and repeat.

6.3. GIFT WRAPPING ALGORITHM

This algorithm was introduced by Chand & Kapur, a modified version Jaris’ March algorithm. The gift wrapping algorithm begins with $i=0$ and a point $p_0$ known to be on the convex hull, e.g., the leftmost point, and selects the point $p_{i+1}$ such that all points are to the right of the line $p_i p_{i+1}$. This point may be found on $O(n)$ time by comparing polar angles of all points with respect to point $p_0$ taken for the center of polar coordinates. Letting $i=i+1$, and repeating with until one reaches $p_k=p_0$ again yields the convex hull in $h$ steps. The gift wrapping algorithm is exactly analogous to the process of winding a string (or wrapping paper) around the set of points.
6.4. GRAHAM’S SCAN

The **Graham scan** is a method of computing the convex hull of a given set of points in the plane with time complexity $O(n \log n)$. It is named after Ronald Graham, who published the original algorithm in 1972.

The first step in this algorithm is to find the point with the lowest $y$-coordinate. If there is a tie, the point with the lowest $x$-coordinate out of the tie breaking candidates should be chosen. Call this point $P$. This step takes $O(n)$, where $n$ is the number of points in question.

Next, the set of points must be sorted in increasing order of the angle they and the point $P$ make with the x-axis. Any general-purpose sorting algorithm is appropriate for this. In order to speed up the calculations, it is not actually necessary to calculate the actual angle these points make with the x-axis; instead, it suffices to calculate the tangent of this angle, which can be done with simple integer arithmetic.

The algorithm proceeds by considering each of the points in the sorted array in sequence. For each point, it is determined whether moving from the two previously considered points would increase or decrease the area of the polygon.

Figure 6.7 Gift – wrapping of a set of data
points to this point is a "left turn" or a "right turn". If it is a "right turn", this means that the second-to-last point is not part of the convex hull and should be removed from consideration. This process is continued for as long as the set of the last three points is a "right turn". As soon as a "left turn" is encountered, the algorithm moves on to the next point in the sorted array. (If at any stage the three points are collinear, it is unimportant whether the middle point is removed or not. Removing it would yield a minimal convex hull, but keeping it in does not invalidate it.)

Again, determining whether three points constitute a "left turn" or a "right turn" does not require computing the actual angle between the two line segments, and can actually be achieved by integer arithmetic only. For three points \((x_1, y_1), (x_2, y_2)\) and \((x_3, y_3)\), simply compute the cross product \((x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)\) of the two vectors defined by points \((x_1, y_1), (x_2, y_2)\) and \((x_2, y_2), (x_3, y_3)\). If the result is 0, the points are collinear; if it is positive, the three points constitute a "left turn", otherwise a "right turn".

This process will eventually return to the point at which it started, at which point the algorithm is completed and the array now contains the points on the convex hull in anticlockwise order.

Figure 6.8. Graham's Scan. As one can see, A to B and B to C are counterclockwise, but C to D isn't. The algorithm detects this situation and discards previously chosen segments until the turn taken is counterclockwise (B to D in this case.)
Here is a list of some well-known 2D hull algorithms. For the speed, $n = \#$ points in the input set, and $h = \#$ vertices on the output hull (Note that $h \leq n$, so $nh \leq n^2$).
Comparing convex hull algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Speed</th>
<th>Discovered By</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brute Force</td>
<td>$O(n^4)$</td>
<td>[Anon, the dark ages]</td>
</tr>
<tr>
<td>Gift Wrapping</td>
<td>$O(nh)$</td>
<td>[Chand &amp; Kapur, 1970]</td>
</tr>
<tr>
<td>Graham Scan</td>
<td>$O(n \log n)$</td>
<td>[Graham, 1972]</td>
</tr>
<tr>
<td>Jarvis March</td>
<td>$O(nh)$</td>
<td>[Jarvis, 1973]</td>
</tr>
<tr>
<td>Quick Hull</td>
<td>$O(nh)$</td>
<td>[Eddy, 1977], [Bykat, 1978]</td>
</tr>
<tr>
<td>Divide-and-Conquer</td>
<td>$O(n \log n)$</td>
<td>[Preparata &amp; Hong, 1977]</td>
</tr>
<tr>
<td>Monotone Chain</td>
<td>$O(n \log n)$</td>
<td>[Andrew, 1979]</td>
</tr>
<tr>
<td>Incremental</td>
<td>$O(n \log n)$</td>
<td>[Kallay, 1984]</td>
</tr>
<tr>
<td>Marriage-before-Conquest</td>
<td>$O(n \log h)$</td>
<td>[Kirkpatrick &amp; Seidel, 1986]</td>
</tr>
</tbody>
</table>

6.5. CONVEX HULL IN ULTRASOUND IMAGES

Ultrasound images are formed based on the echo of the obstructions in the body. Scattering is the natural phenomenon of sound, but sound diverges more than as light do. Thus, the image formed may not form a perfect shadow of minute walls of the vessel. A closed contour of the vessel is necessary for further processing. Convex hull is an approach for making it a closed contour.

Figure 6.10 Convex of an Image using Qhull
CHAPTER 7
SKELETONISATION

A skeleton or stick figure of an object can be used to describe its structure and it can done by using medial axis transformation. An intuitive explanation of the medial axis transformation is based on the prairie fire analogy. Consider a rectangle region composed of dry grass on a bare dirt background. If fire were to be started simultaneously on the perimeter of the grass, the fire would proceed to burn toward the centre of the region until all the grass was consumed. For the rectangle, the fire would proceed from each side. As the fire moved simultaneously from the left and top, the fire lines would meet and quench the fire. The quench points or quench lines of a figure are called its medial axis skeleton.

The vessel in the present video dilates and contracts. This movement causes the boundaries to move outwards and inwards but the skeletons are almost same. For alignment of these images in order, skeletons are used.

7.1. PRE PROCESSING
The preprocessing of skeletonisation involves
- Binary thresholding
- Convex hull
- Region filling

7.1.1. BINARY THRESHOLDING

The transfer function of the binary level thresholding is a step, dividing the inputs intensities into two groups with an intensity barrier, resulting in a binary image having a values zero’s (0) and one’s (1). The intensities more than the threshold, (Gray) k, designate a binary value of say one (1) and the others less than gray (k) results the complimenting binary value zero (0). As the skeleton of an image doesn’t hold any distinguishing scale of intensities, binary thresholding is a perfect answer.
7.1.2. CONVEX HULL

The vessel image so found by an ultrasound probe never forms a sealed entity. To make further analysis closed structures are necessary. The convex hull of a vessel gives a polygon which is least possible union of all the points on the vessel tract. The construction of convex hull is more detailed in the previous chapters.

7.1.3. REGION FILLING

Region Filling is a morphological operation used to fill the hollow elements in an image. A skeleton of a ring like structure is also a ring; hence, the blood vessel also forms a loop. The skeletons for further image processing needs pivot points, which can be done only from solid rings but not hollow rings. This step helps for providing solid structures.

- Select a point A inside the hollow element.
- Dilate A using structuring element B, say C
- \( D = A \cap C \)
- When D and A are equal stop the iteration otherwise repeat above steps.

![Figure 7.1 Binary Thresholding (a) Input histogram (b) Output Histogram](image)

<table>
<thead>
<tr>
<th>Structuring element B2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 0</td>
</tr>
<tr>
<td>1 1 1</td>
</tr>
<tr>
<td>0 1 0</td>
</tr>
</tbody>
</table>
7.2. PROCEDURE

The process of finding the skeletons is called skeletonisation. A Skeleton can be defined as follows “If we put fire on all sides of an uniform object and assume that the density of the object is same all along which implies that the rate of burning of the object is constant then the point where any two sides touch in the process of burning constitute a point on the skeleton, thus joining all the points we get a skeleton of the object. For example consider few object listed below with the inner part as their skeleton.

![Figure 7.2 Skeletons of few geometric figures](image)

The mathematical notation for finding the skeleton of an object is as follows,
If S(A) be the skeleton, then,

1) If \( z \) is a point of \( S(A) \) and \( (D)z \) is the largest disk centered at \( z \) and contained in \( A \), one cannot find a large disk (not necessarily centered at \( z \)) containing \( (D)z \) and included in \( A \). The disk \( (D)z \) is called a maximum disk.

2) The disk \( (D)z \) touches the boundary of \( A \) at two or more different places.

The concept of Skeletonisation is merged with the combined aspects of Dilation and Erosion which are the basic morphological operations.
7.3. STEPS INVOLVED

1. Erode the image $A$ using the structuring element $B_1$, say $C$.

\[
\begin{array}{ccc}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{array}
\]

Structuring element $B_1$

2. Open $C$ with the structuring element $B_2$, say $D$.

\[
\begin{array}{ccc}
0 & 1 & 0 \\
1 & 1 & 1 \\
0 & 1 & 0 \\
\end{array}
\]

Structuring element $B_2$

3. Subtract $D$ from $C$, say $K_i$.

4. Repeat steps 1 to 3, till there is no further erosion in $A$, increment $i$ for each step with 1.

5. Add all so formed $K_i$ images, say $S$.

\[
S = \sum_{i=1}^{n} K_i
\]

6. $S$ is the skeleton of $A$.

Mathematically,

\[
S(A) = \sum (S_k(A))
\]

With $S_k(A) = (A \Theta kB) - (A \Theta kB) \circ B$

Where $B$ is a structuring element and $(A$ eroded with $kB)$ indicates $k$ successive erosions of $A$:

\[
(A \Theta kB) = (\ldots (A \Theta B) \Theta B) \ldots) \Theta B
\]

The above lasts till $k$ times and $K$ is the iterative step before $A$ erodes to an empty step. In other words $K = \max \{k \mid (A \Theta kB) \neq \emptyset\}$.
CHAPTER 8
CAMERA CORRECTION USING AFFINE TRANSFORMATION

The movement of the subject in interest, namely vessel, causes distorted frames in the video. These frames obtained after the process of skeletonisation need to be geometrically corrected. Each and every skeleton is oriented in a different direction, so they need to be aligned appropriately. Affine transformations are used to adjust the images basing on the reference points of each frame. Affine transformations include different types of mappings like translation, rotation, shearing, scaling. Given below is a square which underwent the above listed mappings.

Reference points or Pivot points can be calculated by averaging all the location of the points in the skeleton.

![Affine Transformations Diagram](image)

**Figure 8.1 Affine Transformations**

8.1. TRANSLATION

Image translation can be defined as shifting of all the pixels in the image, while preserving their amplitudes. If the old co-ordinates are \((x, y)\) and new co-ordinates are \((x', y')\), \((h, k)\) are the provided co-ordinates then

\[
\begin{align*}
x' &= x + h, \\
y' &= y + k.
\end{align*}
\]

............... (i)
Chapter 8. Camera Correction Using Affine Transformation

Translating a frame accordingly results in the shifting of all pixel co-ordinates to the position mentioned.

8.2. ROTATION

The rotation is based on transformation table. If \((x, \ y)\) are the old co-ordinates and \(\theta\) is the angle through which the image is to be rotated in anti-clockwise then the corresponding new co-ordinates are

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} = \begin{bmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{bmatrix} \begin{bmatrix}
x \\
y
\end{bmatrix}
\]

8.3. SHEARING

Shearing is to make a parallelogram view of the image along single or dual axis. Shearing is of two types. A shear parallel to the \(x\) axis has \(x' = x + ky\) and \(y' = y\)

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} = \begin{bmatrix}
1 & k \\
0 & 1
\end{bmatrix} \begin{bmatrix}
x \\
y
\end{bmatrix}
\]

A shear parallel to the \(y\) axis has \(x' = x\) and \(y' = y + kx\)

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} = \begin{bmatrix}
1 & 0 \\
k & 1
\end{bmatrix} \begin{bmatrix}
x \\
y
\end{bmatrix}
\]

where \((x, \ y)\) and \((x', \ y')\) are the are the input image and sheared image co-ordinates.
Chapter 8. Camera Correction Using Affine Transformation

8.4. PERSPECTIVE PROJECTION

The perspective projection projects points onto the image plane along lines that emanate from a single point, called the center of projection, the effect by which an object has a smaller projection when it is far away from the center of projection and a larger projection when it is closer. The simplest perspective projection uses the origin as the center of projection, and $z = 1$ as the image plane. The functional form of this transformation is then $x' = x / z; \ y' = y / z, z=z$

$$
\begin{bmatrix}
    x_c \\
    y_c \\
    z_c \\
    w_c \\
\end{bmatrix} = \begin{bmatrix}
    1 & 0 & 0 & 0 \\
    0 & 1 & 0 & 0 \\
    0 & 0 & 1 & 0 \\
    0 & 0 & 1 & 0 \\
\end{bmatrix} \begin{bmatrix}
    x \\
    y \\
    z \\
    1 \\
\end{bmatrix}
$$

$$
\begin{bmatrix}
    x' \\
    y' \\
    z' \\
\end{bmatrix} = \begin{bmatrix}
    x_c/w_c \\
    y_c/w_c \\
    z_c/w_c \\
\end{bmatrix}
$$

Figure 8.4 Perspective Projections

The transformation used to align the skeletons is translation. The position of the averaged skeleton point changes with each and every frame. Translation is performed basing on the horizontal and vertical distances between the corresponding pivot points of any 2 consecutive frames.

The pivot points can be calculated by averaging all the co-ordinates of the all points in the skeleton.

**Pivot Point P1 = (x1,y1)**

**Pivot Point P1 = (x2,y2)**

$$(h,k) = (x_1-x_2,y_1-y_2)$$

The input images are translated by $(h, k)$ for all the reference points in each frame, according to the equation (i). Stacking all these frames into a new video resolves camera errors.
CHAPTER 9
POLAR TRANSFORMATION

The position change of all the points on the vessel from the reference point can be more vividly interpreted by transforming it into another domain. The polar coordinate system is a two-dimensional coordinate system in which each point on a plane is determined by an angle and a distance. The polar coordinate system is especially useful in situations where the relationship between two points is most easily expressed in terms of angles and distance; in the more familiar Cartesian or rectangular coordinate system, such a relationship can only be found through trigonometric formulae.

9.1. DEFINITION

As the coordinate system is two-dimensional, each point is determined by two polar coordinates: the radial coordinate and the angular coordinate. The radial coordinate (usually denoted as \( r \)) denotes the point's distance from a central point known as the pole (equivalent to the origin in the Cartesian system). The angular coordinate (also known as the polar angle or the azimuth angle, and usually denoted by \( \theta \) or \( t \)) denotes the positive or anticlockwise (counterclockwise) angle required to reach the point from the 0° ray or polar axis (which is equivalent to the positive x-axis in the Cartesian coordinate plane). A graph plotted with \( r \) (y axis) against \( \theta \) (x axis) gives a curve, which defines the track of the vessel around the reference point.

The two polar coordinates \( r \) and \( \theta \) can be converted to the Cartesian coordinates \( x \) and \( y \) by using the trigonometric functions sine and cosine:

\[
x = r \cdot \cos \theta
\]

\[
y = r \cdot \sin \theta
\]
where

\[ r = \sqrt{x^2 + y^2} \]

\[ \theta = \tan^{-1} \left( \frac{y}{x} \right) \]

9.2. POLAR TRANSFORMATION OF GEOMETRIC FIGURES

Figure 9.2 (a) shows a polar transform of two different geometric figures. A square from its centroid makes a ‘jumping’ waveform, as the distance from the centroid to the midpoint of a side is less than the distance from the corner. The hills in the graph are identical because a square has all equal sides. A ring shown in figure 9.2 (b),
however, forms a line in the image after polar transformation. A circle has all its points equidistant its centroid, i.e., its centre. As the ring in the image is not perfect, the plot is not a straight line.

Each point on the vessel has a particular $\theta$ value. Now, graph containing all the ‘$r$’s of the point from frame 1 to the last is drawn as shown in figure 9.3. This gives the movement of the point for a blood pulse. The slope of the tangent drawn at any point on this curve gives the strain rate at that instant.

![Figure 9.3 Radial Distance plot of a point on the vessel with $\theta = 30$](image)
APPENDIX

READING AND WRITING A VIDEO IN MATLAB

I=aviread('input.avi'); %Reading the Input Video
[m n]=size(I); % 'n' is the total no.of frames in the video
for t=1:n % Accessing each frame
    P=(I(1,t).cdata); % Reading the RGB Image in the frame
    Q=im2double(P); % Converting it into double data type
    for k=1:3
        W(:,:,k)=Q(:,:,k); % Manipulating each pixel of RGB planes
    end
    R=im2uint8(W); % Convert the ouput RGB image to uint8 data type
    S(t)=im2frame(R); % The image is converted to a frame format
end
movie2avi(S,'sampler.avi'); % Compiling it into a video

TEMPORAL MEDIAN FILTERING

%For Temporal median Filtering a video
I=aviread('Input.avi');
[p q]=size(I);
[R C l]=size(I(1,1).cdata);
H=zeros(R,C,l,5);
O=zeros(R,C,l,5);
for r=4:q-2
    for k=1:3
        for f=-3:2
            H(:,:,k,f+4)=(im2double(I(1,(r+f)).cdata(:,:,k))); % Accessing five frames at once
        end
    end
end

Programs

for i=1:R
  for j=1:C
    for s=1:5
      T(s)=H(i,j,k,s);  % Considering corresponding pixel of each frame
    end
    T1=sort(T);       % Sorting these pixels in order
    O(i,j,k,3)=T1(3);
  end
end
end
end
W=im2uint8(O(:,:,3));
S(r-3)=im2frame(W);
end
movie2avi(S,'Temporalmedianfiltered.avi');

IMAGE SEGMENTATION

clear all;clc
V=aviread('Temporalmedianfiltered.avi ');
[p q]=size(V);
for t=1:q
  I=im2double(rgb2gray(V(1,t).cdata));
  [m n]=size(I);
  O=zeros(m,n);
  for i=1:m-3
    for j=1:n-3
      k=1;
      for r=1:3     % Box size
        for c=1:3
          T(k)=I(i+r,j+c);
        end
      end
      % Processing...
    end
  end
end
\[
\begin{align*}
    k &= k + 1; \\
    \text{end} \\
    \text{end} \\
    T &= \text{sort}(T); \\
    \text{if}(\text{std2}(T) < 0.05 \land \text{mean2}(T) > 0.7) \quad \text{\% STD and Mean Threshold Condition} \\
    O(i:i+2,j:j+2) &= I(i:i+2,j:j+2); \\
    \text{end} \\
    \text{end} \\
    \text{end} \\
    W(:,:,1) &= O; \\
    W(:,:,2) &= O; \\
    W(:,:,3) &= O; \\
    S(t) &= \text{im2frame} (\text{im2uint8}(W)); \\
    \text{end} \\
    \text{movie2avi}(S, 'Segmented.avi');
\end{align*}
\]

**MORPHOLOGICAL CONVEX HULL**

clear all; clc;
V = aviread('Segmented.avi');
[x y] = size(V);
for d = 1:y
    I = im2double(rgb2gray(V(1,d).cdata));
    [m n] = size(I);
    I = im2bw(I, 0.1);
    F = [0 0 0; 0 0 0; 1 1 1];
    O = zeros(m, n);
    K = O;
    A = I;
    Q = I;
    s = 1;
    while s < 5
\begin{verbatim}
I=A;
for i=1:m-3
    for j=1:n-3
        sum=0;
        for r=1:3
            for c=1:3
                sum=sum+I(i+r,j+c)*F(r,c);  % Hit or Miss Transformation
            end
        end
        O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|I(i:i+2,j:j+2);
        if sum==3
            O(i+2,j+2)=1;
        end
    end
end
A=I|O;
if isequal(A,I)
    F=rot90(F);
    s=s+1;
    K=K|O;
    O=zeros(m,n);
    I=Q;
    A=I;
end
end
W(:,:,1)=K;
W(:,:,2)=K;
W(:,:,3)=K;
S(d)=im2frame(im2uint8(W));
end
movie2avi(S,'Convexhull.avi');
\end{verbatim}
clear all;clc;
V=aviread('Segmented.avi');
[p q]=size(V);
for t=1:q
    I=im2double(rgb2gray(V(1,t).cdata));
    [m n]=size(I);
    xarray=[];
    yarray=[];
    for i=1:m
        for j=1:n
            if I(i,j)==1
                xarray=[xarray i];
                yarray=[yarray j];
            end
        end
    end
    k=convhull(xarray,yarray);
    p=size(k);
    O=zeros(m,n);
    for i=1:p
        O(xarray(k(i)),yarray(k(i)))=1;
    end
    W(:,:,1)=O;
    W(:,:,2)=O;
    W(:,:,3)=O;
    S(t)=im2frame(im2uint8(W));
end
movie2avi(S,'Convexhull.avi');
Programs

REGION FILLING

clear;clc;
V=aviread('Convexhull.avi');
[q x]=size(V);
for a=1:x
J=im2double(rgb2gray(V(1,a).cdata));
I=zeros(size(J));
I(220,310)=1;
A=1-J;
K=J;
while (1-isequal(K,I))
    K=I;
    F=[0 1 0;1 1 1;0 1 0];
    [m n]=size(I);
    O=zeros(m,n);
    for i=1:m-3
        for j=1:n-3
            if F(2,2)&I(i+1,j+1)==1
                O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|(F|I(i:i+2,j:j+2));
            else
                O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|I(i:i+2,j:j+2);
            end
        end
    end
    I=A&O;
end
O=J+I;
W(:,:,1)=O;
W(:,:,2)=O;
W(:,:,3)=O;
S(a)=im2frame(im2uint8(W));
end
movie2avi(S,'Regionfilled.avi');

SKELETONISATION USING IN BUILT MATLAB FUNCTION

clear all;clc;
I=aviread('Regionfilled.avi');
[p q]=size(I);
for y=1:q
    P=(im2double(rgb2gray(I(1,y).cdata)));
    P=im2bw(P,0.5);
    P=bwmorph(P,'skel',Inf);
    O=1*P;
    W(:,:,1)=O;
    W(:,:,2)=O;
    W(:,:,3)=O;
    R=im2uint8(W);
    S(y)=im2frame(R);
end
movie2avi(S,'Skeletons.avi');

SKELETONISATION USING MORPHOLOGICAL OPERATORS

clear all;clc;
P=aviread('Regionfilled.avi');
[p q]=size(P);
for y=1:q
    J=(im2double(rgb2gray(P(1,y).cdata)));
    O1=zeros(size(J));
    for q=1:30
        for a=1:q
            %Erosion
            I=~J;
            F=[1 1 1;1 1 1;1 1 1];
            [m n]=size(I);
            O=zeros(m,n);
            for i=1:m-3
                for j=1:n-3
                    if F(2,2)&I(i+1,j+1)==1
                        O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|(F|I(i:i+2,j:j+2));
                    else
                        O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|I(i:i+2,j:j+2);
                    end
                end
            end
            J=~O;
        end
    end
    end
    %Opening
    I=~J;
    F=[0 1 0;1 1 1;0 1 0];
    [m n]=size(I);
    O=zeros(m,n);
    for i=1:m-3
        for j=1:n-3
            if F(2,2)&I(i+1,j+1)==1
                O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|(F|I(i:i+2,j:j+2));
            else
                O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|I(i:i+2,j:j+2);
            end
        end
    end
end
O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|(F|l(i:i+2,j:j+2));
else
O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|l(i:i+2,j:j+2);
end
end
end
I=-O;
O=zeros(m,n);
for i=1:m-3
    for j=1:n-3
        if F(2,2)&I(i+1,j+1)==1
            O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|(F|l(i:i+2,j:j+2));
        else
            O(i:i+2,j:j+2)=O(i:i+2,j:j+2)|l(i:i+2,j:j+2);
        end
    end
end
O1=O1+J-O;
end

W(:,:,1)=O1;
W(:,:,2)=O1;
W(:,:,3)=O1;
R=im2uint8(W);
S(y)=im2frame(R);
end
movie2avi(S,'Skeletons.avi');
DETERMINING THE PIVOT POINTS

clear all;clc;
I=aviread('Skeletons.avi');
[p q]=size(I);
for y=1:q
    P=(im2double(rgb2gray(I(1,y).cdata)));
    [m n]=size(P);
    x=0;
    a=0;
    k=0;
    for i=1:m
        for j=1:n
            if P(i,j)==1
                x=x+i;
                a=a+j;
                k=k+1;
            end
        end
    end
    x=round(x/k);
    a=round(a/k);
    O=zeros(m,n);
    O(x,a)=1;
    W(:,:,1)=O;
    W(:,:,2)=O;
    W(:,:,3)=O;
    R=im2uint8(W);
    S(y)=im2frame(R);
end
movie2avi(S,'Pivotpoints.avi');
CAMERA CORRECTION USING AFFINE TRANSFORMATION

clear all;clc;
I=aviread('Input.avi');
J=aviread('Pivotpoints.avi');
[p q]=size(J);
for y=1:q
    P=(im2double(rgb2gray(I(1,y).cdata)));
    Q=(im2double(rgb2gray(J(1,y).cdata)));
    [m n]=size(Q);
    for i=1:m
        for j=1:n
            if Q(i,j)>0.05
                x=i;
                a=j;
            end
        end
    end
    s=translate(strel(1),[round(m/2)-x round(n/2)-a]);
    O=imdilate(P,s);
    W(:,:,1)=O;
    W(:,:,2)=O;
    W(:,:,3)=O;
    R=im2uint8(W);
    S(y)=im2frame(R);
end
movie2avi(S,'Cameracorrected.avi');
V=aviread('Cameracorrected.avi');
[p q]=size(V);
figure;
hold on
for a=1:q
    I=im2double(rgb2gray(V(1,a).cdata));
    [m n]=size(I);
    xarray=[];
    yarray=[];
    for i=1:m
        for j=1:n
            if I(i,j)==1
                xarray=[xarray i];
                yarray=[yarray j];
            end
        end
    end
    x=xarray-round((max(xarray)+min(xarray))/2);
    y=yarray-round((max(yarray)+min(yarray))/2);
    [th r]=cart2pol(x,y);
    th=1+round(th*100);
    r=1+round(r);
    th=1+th-min(th);
    O=zeros(max(r),max(th));
    [m n]=size(r);
    for i=1:n
        O(r(i),th(i))=1;
    end
    W=zeros(max(r),max(th),3);
Programs

W(:,:,1)=O;
W(:,:,2)=O;
W(:,:,3)=O;
S(a)=im2frame(im2uint8(W));
end
movie2avi(S,'Polartransformed.avi');

FINAL PLOTTING

V=aviread('Cameracorrected.avi');
[p q]=size(V);
Theta=input('Angle:');
Theta=deg2rad(Theta);
Theta=round(Theta*100);
final=[];
for a=1:q
    I=im2double(rgb2gray(V(1,a).cdata));
    [m n]=size(I);
    xarray=[];
    yarray=[];
    for i=1:m
        for j=1:n
            if I(i,j)==1
                xarray=[xarray i];
                yarray=[yarray j];
            end
        end
    end
    x=xarray-round((max(xarray)+min(xarray))/2);
    y=yarray-round((max(yarray)+min(yarray))/2);
    [th r]=cart2pol(x,y);
th=round(th*100);
[x y]=size(th);
for i=1:y
    if th(i)==Theta
        k=i;
        break;
    end
end
final=[final r(k)];
end
plot(final)
REFERENCES


