CARDIAC MOTION ANALYSIS IN MRI
FOR CLASSIFICATION

by

129322

Dilek Göksel

B.Sc. in Physics Engineering İstanbul Technical University, 1999

Submitted to the Institute of Biomedical Engineering

in partial fulfillment of the requirements

for the degree of
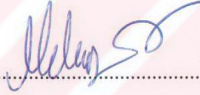
Master of Science

in

Biomedical Engineering

125322

Boğaziçi University

January, 2002

CARDIAC MOTION ANALYSIS IN MRI
FOR CLASSIFICATION

APPROVED BY:

Assoc.Prof. Mehmed Özkan ....
(Thesis Supervisor)

Assoc.Prof. Ahmet Ademoğlu ....

Assist.Prof. Cengizhan Öztürk ....

DATE OF APPROVAL: January 17, 2002

ACKNOWLEDGMENTS

CARDIAC MOTION ANALYSIS IN MRI
FOR CLASSIFICATION

ABSTRACT

Although several techniques exist for the analysis of cardiac tagged MR images, a rapid screening tool do not yet exits. Our proposed technique tries to perform rapid classification to diagnose the abnormalities in human left ventricle and the final aim of this study is to identify the investigated myocardium in the analyzed tagged MR images as pathological and non.

In this thesis, images are first analyzed using harmonic phase (*HARP*) analysis and synthetic tags are computed over the myocardium. The data is normalized to perform a comparison between different myocardiums having various tag lines and time frames. The aim of the normalization is to eliminate the shift, scale and rotation variance. Cubic curves are fitted to the normalized tags and curve parameters are compared at various regions of the myocardium. In this initial study, the curve parameters are examined with probability density function between normal and diseased hearts, such as left ventricles with dilated cardiomyopathy (DCM) and infarcted regions.

Finally, the confusion matrix is evaluated to examine the correctness of the segmentation algorithm. This method could be a very fast and automatic screening tool for identifying diseased locations in tagged MRI.

Keywords: tagged MRI, HARP, cardiac motion analysis

KARDİYAK MR GÖRÜNTÜLERİNDE SINIFLANDIRMA AMAÇLI HAREKET ANALİZİ

ÖZET

Manyetik işaretlenmiş (*tagged*) kardiyak MR görüntülerini işlemek için pek çok teknik var olsa da hızlı bir analiz hala sağlanamamıştır. Öneridiğimiz teknik ile insan sol karıncığında tetkik yapılabilmesi için süratli bir sınıflandırma gerçekleştirilmektedir. Çalışmamızın ana hedefini manyetik işaretlenmiş kalp MR görüntülerinde hastalık olup olmadığının tespit edilebilmesi oluşturmaktadır.

Tez çalışmamızda öncelikle kardiyak görüntülerde harmonik faz (*HARP*) analizi uygunlanmış ve myokardiyum üzerinde sentetik manyetik işaretler hesaplanmıştır. Bulunan manyetik işaret noktaları, manyetik işaretleri ve zaman dilimleri farklı çekilmiş myokardiyum görüntüleri ile karşılaştırabilmek için normalize edilmiştir. Normalizasyon işlemi kaydırma, oranlama ve döndürme olarak üç adım için gerçekleştirildi. Normalize edilen manyetik işaret noktaları, kübik eğrilere yerleştirilerek, myokardiyumun belli bölgelerinde eğri parametreleri karşılaştırıldı. Kübik eğri parametreleri olasılık yoğunluk fonksiyonuna konularak sağlıklı ve hastalıklı kalp incelemesi gerçekleştirilmiştir.

Sonuçlar, konfüzyon matrisine yazılarak segmentasyon algoritmasının doğruluğuna bakılmıştır. Uygulanan metod manyetik işaretlenmiş MR görüntülerinde, hastalık teşhisi için çok hızlı ve otomatik bir analiz yöntemi olarak görünmektedir.

Anahtar Sözcükler: manyetik işaretlenmiş MRG, HARP, kalp hareketi analizi

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

MRI	Magnetic Resonance Imaging
CT	Computed Tomography
HARP	Harmonic Phase
LV	Left Ventricle
LA	Left Atrium
RV	Right Ventricle
RA	Right Atrium
DCM	Dilated Cardiomyopathy
ECG	Electrocardiogram
CSPAMM	Complementary Spatial Modulation of the Magnetization
RF	Radio Frequency
TR	Repetition Time
FFT	Fast Fourier Transform
SNR	Signal to Noise Ratio

1. INTRODUCTION

1.1 Motivation and Objectives

A fundamental problem in image processing and computer vision is the computation of a deforming object's motion from image sequence data. One deforming object, *the heart,* has received a great deal of attention in recent years because of its complex motion and the fact that analysis of cardiac motion can be used to diagnose heart muscle damage caused by a heart attack. Characterization of myocardial deformation during systolic contraction is necessary to understand the physiology of the normal heart and the effects of cardiovascular disease. Image sequences of the heart can be acquired with several modalities including ultrasound, computed tomography (CT), and magnetic resonance (MR) imaging. MR in particular has shown great promise for imaging cardiac motion because of a technique called "tagging".

In cardiac tagging, tag deformation pattern provides useful qualitative and quantitative information about the functional properties of underlying myocardium. Tagged images appear as a spatially encoded pattern that moves with the heart tissue as it moves through the heart cycle. Usually tags are imposed at end-diastole and subsequent images are acquired during systole, revealing the contraction within the wall muscle. The images are usually acquired over several heartbeats, with the assumption that the heart motion is periodic. In this case, breathing is a problem and one of the acceptable solutions is breath holding which is not preferred for cardiac patient.

The advantage of imposing the tags is that they can be identified in the acquired images and they depict the motion inside the wall muscle. Moreover, the regularity of the tags' shape enables measuring the motion quantitatively, i.e. producing numerical measurements indicating the regional function. So far, tagging has been successful in producing reliable regional measurements that would become a gold standard.

This study is concerned with estimating heart motion from tagged MR image sequences, and the final aim is to perform rapid classification of tagged cardiac MR images as normal and abnormal.

The main contribution is to describe an image analysis approach for tagged MR images of the heart. The cardiac left ventricle is one of the most crucial and thoroughly studied structures in the human body. This thesis will provide the basis for the estimation of complex motion parameters of the left ventricle (LV). This information of the tag deformation pattern can be utilized for diagnosis of cardiac pathologies.

1.2 Outline

In Chapter 2, the anatomy of the human heart, heart imaging technique, MR tagging and harmonic phase (HARP) imaging are explained.

Chapter 3 gives the detailed information about the method used in this study such as the image acquisition and image processing techniques like contouring and tag detection, motion tracking, normalization of the selected LV images, and the confusion matrix generation.

Chapter 4 includes the results of the proposed technique for training and test dataset, and finally, in chapter 5 the normal and pathological human heart left ventricles are evaluated using confusion matrices.

2. BACKGROUND

2.1 Cardiac Anatomy

The heart is composed of four chambers as shown Figure 2.1: the right atrium (RA), left atrium (LA), right ventricle (RV) and left ventricle (LV). The atria and ventricles are surrounded by muscle tissue called the myocardium and together form a pump that moves blood throughout the body. Contraction and relaxation of the muscle fibers in the myocardium cause the pumping action of the heart. The gray region in the short-axis cross-section of the heart in Figure 2.2(b) represents the myocardium. The inner surface of the myocardium is called the *endocardium*, and the outer surface is called the *epicardium*.
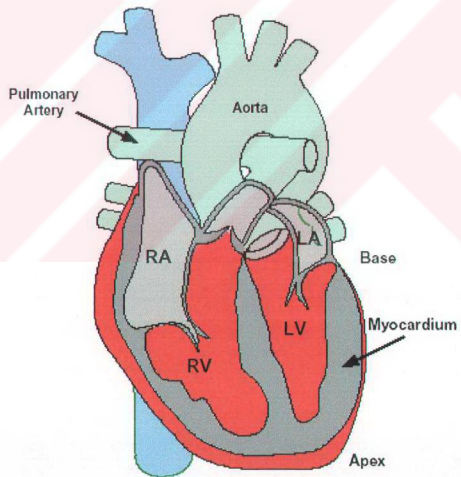
Figure 2.1 Long-axis cross-section of the heart [1].

2.2 Cardiac Physiology

The heart cycle is composed of systole and diastole phases. At the end of diastole, the atria are fully contracted and the ventricles are filled with blood. During systole the ventricles contract and pump blood into the pulmonary artery from the right ventricle and the aorta from the left ventricle. At the same time, the atria fill with blood from the veins into the right atria and the lungs into the left atria. At end-systole, the ventricles are fully contracted and the atria are filled with blood. During diastole the atria contract and fill the ventricles with blood.
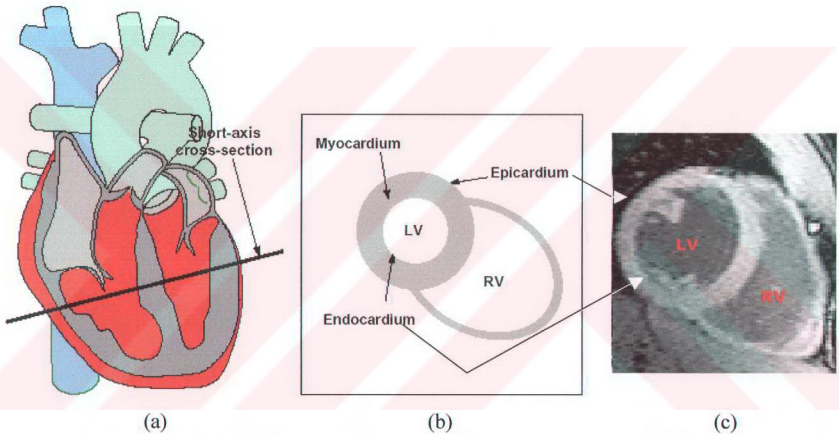
Figure 2.2 (a) Short-axis cross-section of the heart. (b) Location of the LV wall and RV wall in a short-axis image [1]. (c) Original short-axis MR image.

The systolic contraction of the heart is most easily observed in the LV. The contraction of the LV wall is characterized by circumferential shortening and radial thickening of the myocardium. The LV wall is significantly more muscular than the RV, because the blood pressure in the aorta is roughly four times the pressure in the pulmonary artery, and as a result the LV must produce four times more pressure than the RV [1,2]. The LV must generate high pressure to overcome the resistance of the systemic circulation. It typically contains the great majority of cardiac muscle configured in a roughly cylindrical shape. By contrast, the right ventricle (RV), only facing the low resistance of the capillary beds in the lungs, typically has a thin myocardium and a flattened shape that

curves around the LV. Another result of this pressure difference is that any damage to the LV wall will have a significant effect on cardiac performance. Most research efforts are accordingly focused on LV performance.

Healthy cardiac contraction generally causes concentric sections of the myocardial wall to move inward, thereby decreasing the volume of the respective cardiac chamber and forcing blood out through the appropriate valve. Impaired or dead myocardial tissue may still move inward by being dragged passively by neighboring healthy tissue, but this occurs without the normal increase in wall thickness. Thus, as a pump the heart may be analyzed via imaging data by determining the position of the valves and the motion and thickness of the myocardium.

The pathological data used in this thesis work are images of human LV of infarcted and dilated cardiomyopathy (DCM) patients. Cardiomyopathy means heart (cardio) muscle disease (myopathy). The cardiomyopathies are best classified according to their anatomic and pathophysiologic types as dilated, hypertrophic, or restrictive. The cause or causes may or may be not known in each category [3]. "DCM is characterized by an increase in LV or biventricular internal dimensions without an appropriate increase in septal and free wall thicknesses. The essential physiologic impairment is in systolic function (depressed contractility). Ventricular volumes increase as ejection fractions fall, and the ventricles progressively dilate, especially the left. Stroke volume is initially, maintained despite depressed ejection fraction, and compensatory tachycardia may help maintain cardiac output [3].

One of the most common types of damage to the LV wall occurs during a heart attack when one of the coronary arteries, which are responsible for supplying the LV wall with blood, becomes occluded and deprives a region of heart wall of the blood flow. That deprived region is called ischemic. If the region is deprived of blood flow long enough, the muscle tissue dies, and the dead region is called an infarct. Part of the diagnosis and treatment of the heart attacks is identifying these regions of ischemia and infarction.

There are two main characteristics of ischemic and infarcted tissue that can be used to distinguish them from healthy tissue. First infarcted and ischemic regions exhibit reduced blood flow (perfusion) [4]. Second, infarcted and ischemic regions exhibit reduced contraction during systole [5]. Perfusion techniques that try to identify reduced blood flow image the blood flow into the myocardium using an injection of a radioactive tracer [6] or contrast agent [4] into the blood stream. Diagnostic techniques trying to identify regions of reduced contraction use cardiac motion analysis [5]. Both perfusion and motion analysis techniques are in clinical use today. Because of the required drug injection, perfusion techniques are invasive. Many motion analysis techniques, however, have the advantage of being non-invasive.

2.3 MR Tagging

New magnetic resonance imaging techniques allow assessing non-invasively the 2 and 3 dimensional (3-D) motion of the human heart during the cardiac cycle. Prior to an electrocardiogram (ECG) triggered multi-heart phase magnetic resonance imaging procedure, the muscle tissue of the myocardium is labeled by a spatially periodic modulation of the magnetization. Then 12 to 20-heart phase images are acquired with a temporal resolution of 35 ms. In these images, the periodic grid of modulated spins appears as a grid pattern of dark stripes, which are fixed with respect to the muscle tissue (Figure 2.3).
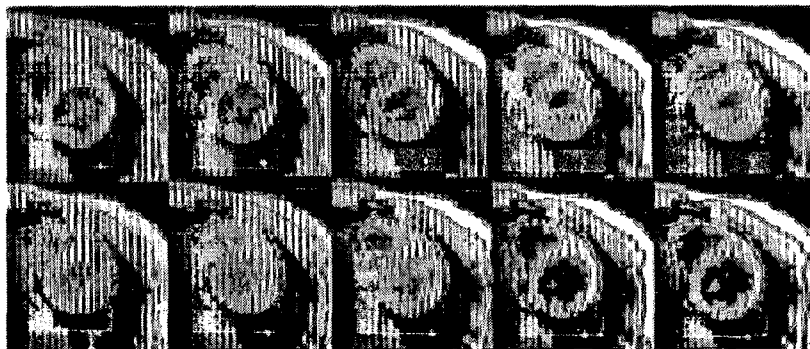
Figure 2.3 A sequence of images from a tagging protocol designed to measure myocardial deformation over the entire heart cycle. The top row shows the deformation of tags that were place in the myocardium at end-diastole. The bottom row shows a sequence of images after the tags have been placed in the heart at end-systole.

Measurement of heart wall motion during contraction is a fundamental challenge in the study of cardiac mechanics. Motion measurement has the potential to be used as a diagnostic tool for assessment of heart disease and as a theoretical tool for physiological analysis. From the displacement, the rotation, and the distortion of this pattern the heart wall motion can be derived. For the identification of the grid and the quantification of the motion, a computer supported procedure is applied which ends up by the calculation and the visualization of the vector plots indicating the motion of any arbitrary point of the myocardium. However conventional tagging techniques suffer from a rapid fading of the grid so that only the contraction phase may be assessed. Furthermore the through plane motion of the myocardium is neglected. To overcome these shortcomings new and more elaborate slice following tagging techniques (CSPAMM; complementary spatial modulation of the magnetization) have been developed which allow the motion analysis throughout the entire cardiac cycle [7]. From this data parameters such as ejection fraction, local radial displacement and rotation angle, local shear rates between epi- and endocardium, can be calculated.

2.3.1 Introduction to Tagging

The technique for tracking the motion of the left ventricle of the heart during contraction uses a sequence of magnetic resonance images and an established imaging method known as "tagging".

MRI tagging means 'marking' the myocardium non-invasively. In tagging, spatially encoded magnetic saturation planes, *tags*, are created within tissues. The creation of tags consists in the application of radio frequency pulses in plane(s) perpendicular to the imaging plane, prior to the application of the radio frequency pulses required for imaging [8]. These act as temporary markers and move with the tissue. In cardiac tagging, tag deformation pattern provides useful qualitative and quantitative information about the functional properties of underlying myocardium.

MR tags are produced by a radio frequency (RF) pulse refers to as the "tagging pulse" applied in addition with pulsed magnetic field gradients. These gradient pulses are triggered by the rising edge of the R-wave in the patient's ECG. The resulting perturbating spins decay at the rate of the longitudinal relaxation of the tissue. At a prescribed delay time after the tags are generated, images are taken using standard spin-echo methods. The tags appear in the image as a dark pattern that moves with the tissue. The regions where the magnetization of the hydrogen nuclei has been perturbed prior to imaging and therefore produce a signal difference with non-tagged regions for a time proportional to T1 are called *myocardial tags*. Since the tags result from perturbations of the magnetization of the tissue itself, the deformation of the tags accurately reflects the motion of the underlying tissue [9].

The movement of the heart through short axis image planes is known as cardiac through-plane motion (Figure 2.3). To study the myocardial motion during systolic contraction, tags are generated during end-diastole by using ECG gating for synchronization, and images are acquired at several times during systolic contraction. The pulses are used to saturate parallel planes of magnetization perpendicular to the imaging plane. At the intersection of the tag plane and imaging plane a dark line appears in the image. At end-diastole the lines are parallel; as the heart contracts the deformation of the lines shows the underlying deformation of the myocardium. The measured deformation of a single tag plane contains only unidirectional information of the past motion. In order to track the motion of a

cardiac material point, this sparse, single dimensional data has to be combined with similar information gathered from other tag sets and all time frames. So to obtain myocardial deformation information over the entire LV, many images containing several MR tags must be acquired.

Although "tag lines" is the normal term for these induced features, tags generally consist of sheets of dark tissue. Tag lines are the intersection of these "tag surfaces" (deformed tag plane) with image planes. This concept is illustrated in the next image, which shows a single tag surface intersecting a set of images (Figure 2.3). The tag surface contributes one tag line to each of the image planes. A standard tagged MRI data set consists of many such tag surfaces arranged in a regular 3-D grid (Figure 2.5). The contracting heart deforms this regular grid and from such deformation, motion information is extracted (Figure 2.4).
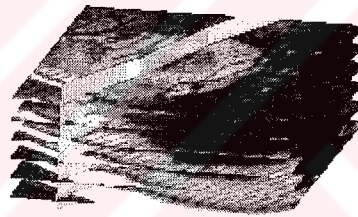
Figure 2.4 Stack of short axis image planes intersected by a single tag surface [10].
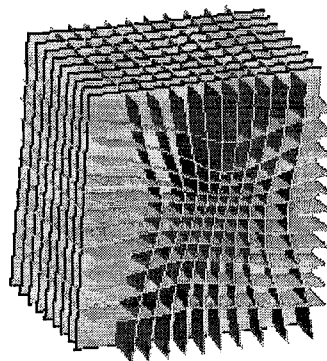
Figure 2.5 Tag surfaces viewed as a regular material grid [10].

Past approaches for measuring heart motion can be divided into two types. One type tracks the movement of material markers, which are either implanted or naturally occurring landmarks within the heart [7,9-11]. The second type of approach uses tagged magnetic resonance imaging techniques to estimate the displacement field for all points within the heart [8,12,13].

Grid reconstruction must be performed because the position of the tag surfaces is only known within image planes. Between image planes, the tag surfaces are estimated using cubic splines, which individually interpolate each surface. The complete reconstructed tag lines are the combination of the cubic spline surfaces.

A comparison of standard cardiac MR images with tagged MR images in the same heart is shown in Figure 2.6. The images on the left are at end-diastole, which is the phase of the heart cycle in which the ventricular cavity has achieved maximum filling. The images on the left are at end-systole, the phase of the heart cycle when the maximum amount of blood has been ejected and the muscle is at maximum contraction.
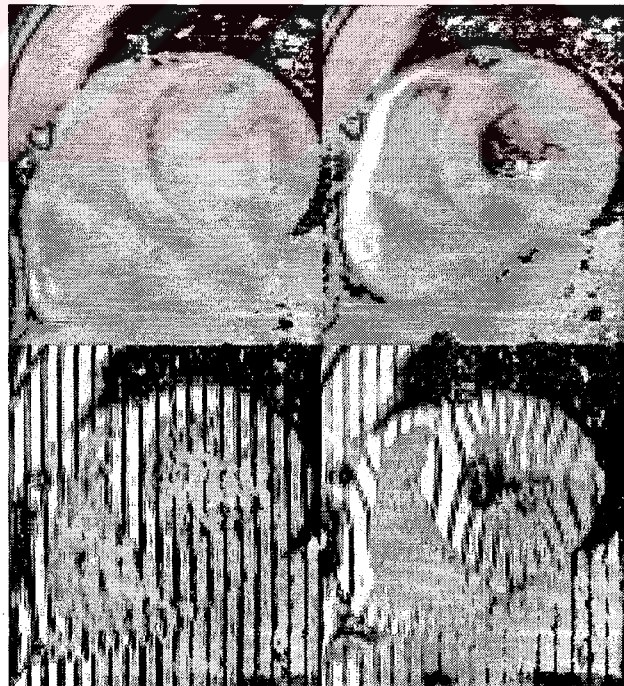
Figure 2.6 Non-tagged and tagged healthy human LV.

2.3.2 Cardiac Tagging Techniques

Cardiac tagging uses a very simple principle. A saturation pattern is placed in the imaging volume and the volume is imaged after some time delay; the change in shape of the saturation pattern in the image reflects the change in shape of the underlying body containing the saturation pattern. The principle of "tagging" spins was first demonstrated by Zerhouni et. al. that the same principle could be used to visually mark tissue with tagged magnetization to measure the more complex deformations of the heart [14]. Axel and Dougherty subsequently proposed a very efficient scheme for generating parallel planes of saturation throughout the entire imaging volume [15].

The process of cardiac tagging can be investigated in three stages:

(1) a saturation pattern is placed in the myocardial tissue with spatially selective RF pulses,

(2) a sequence of MR images is obtained in which the motion of the saturation pattern can be observed,

(3) the motion of the saturation pattern is used to solve for the motion of the myocardium.

The theory of k-space excitation is the simplest method for designing saturation patterns for myocardial tagging [16,17]. For example, a sequence of non-selective RF pulses separated by gradient pulses and the resulting tagging pattern are shown in Figure 2.7. First, the RF/gradient tagging pulse is transformed to show the components in excitation k-space. From there, a simple Fourier transform is used to predict the basic pattern of M_z saturation from the tagging pulse, as shown in the tagging pattern box in Figure 2.7. Using highly crafted RF pulses for tag definition is not an optimal use of imaging time, because the precision of tag detection is not a sensitive function of tag shape.
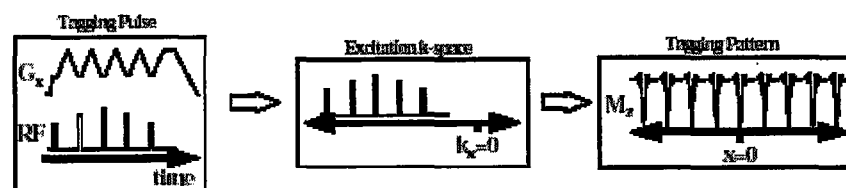
Figure 2.7 The excitation k-space description of the generation of a tagging pattern [17].

Therefore, some of the many tagging patterns proposed for imaging myocardial deformation are: starburst radial tags, parallel line patterns, tagging grids, striped radial tags, sinc modulated comb functions, and contrast enhanced difference patterns [17].

Consideration of the rate and extent of the motion must be taken into account when designing the appropriate tagging pattern and imaging protocol. Because of the simple nature of the saturation pattern virtually any imaging sequence can have tagging pulses added to it. Figure 2.8 shows a sketch of the timing diagram for a tagged segmented k-space sequence [18].
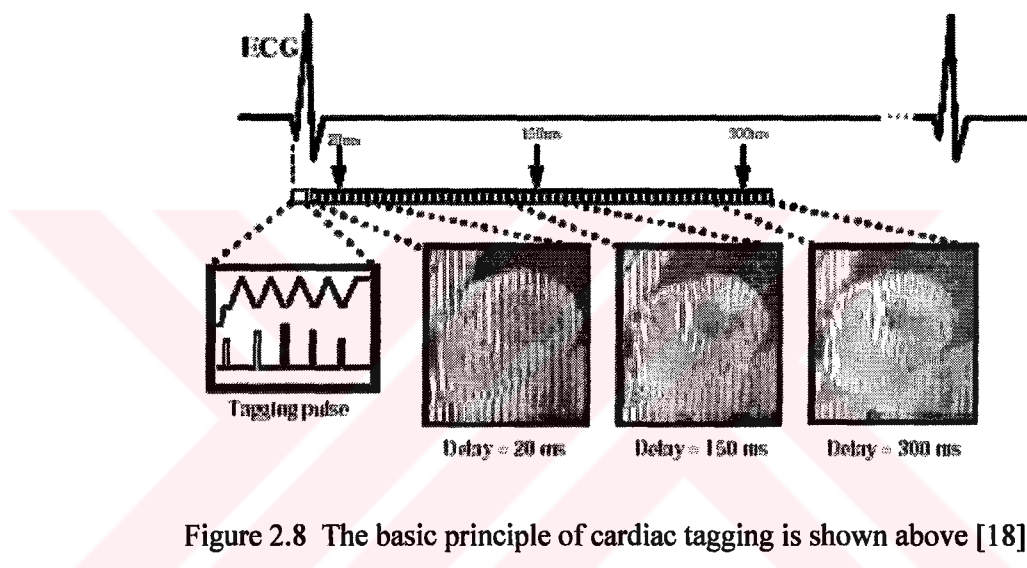
Figure 2.8 The basic principle of cardiac tagging is shown above [18].

First, a saturation pulse illustrated in the bold box on the left, is played after a trigger signal from the ECG. In this example, movie frames of the cardiac cycle are produced by segmented k-space imaging. The shaded boxes show the 7 TR intervals that are grouped together to form one movie frame. The delay from the ECG trigger of the movie frame is given as the center of the 7 TR image acquisition window [18].

Tagged images provide an immediate visualization of myocardial wall motion in the heart. However, in order to make quantitative estimates of the extent and severity of the wall motion abnormality, the full displacement field from the tagging data must be reconstructed. The displacement d of the tag point, which is measured at a given time at the tag point T, gives only one component of the past trajectory, so it cannot be mapped back uniquely. For example, point T might originate from $p1$, or from $p2$ (Figure 2.9).
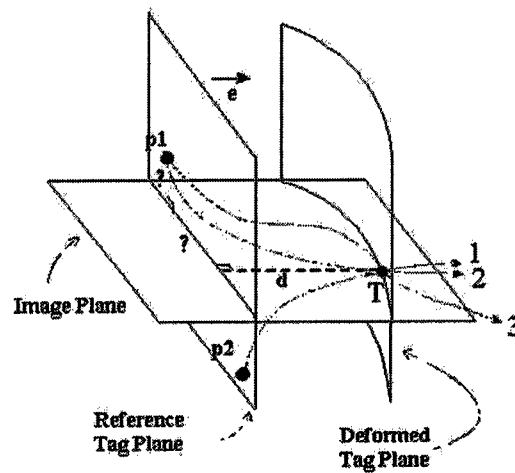
Figure 2.9 Displacement field measurement from planar tagged images [19].

Only by incorporating the tag displacement information sampled from other tag planes, the correct matching point *p1* at the undeformed state can be found. Using all tags at a given time frame, we can locate point *p1*, but we cannot be sure of its trajectory (the lines numbered as 2 and 3 are two of the many possible paths). Using the matching points at every time frame, a final forward motion field and correct trajectory can be calculated.

2.3.3 The Challenges of Tagged MRI

The challenges of tagged MRI in the field of imaging can be divided into four categories:

1. *Image Acquisition:* Sequences of tagged MR images are obtained noninvasively using a "pulse sequence" that takes advantage of the physics behind magnetic resonance. Designing the pulse sequence or analyzing the resulting images requires modeling of the underlying physics.

2. *Data Extraction:* Once a tagged MR image has been obtained, relevant information must be extracted from each image. For example, the boundaries of anatomical objects may be segmented or the location of tag lines may be identified.

3. *Data Fusion:* Next, the information extracted from one or more images must be combined to calculate meaningful parameters describing cardiac function. For example, the motion of tag lines over time can be used to estimate the buildup of strain during contraction of cardiac tissue. For a single heart, this step typically involves hundreds of images arrayed in time and spatial position.

4. *Display and Use:* Finally, the functional parameters must be put to use. In a clinical setting, these parameters, which often compose a 3-D field, must be displayed in a manner that allows physicians to make diagnoses.

2.4 Harmonic Phase Imaging

Tagging has been successful in producing reliable regional measurements that may become the gold standard. Unfortunately, such success has not been translated into a clinical tool for assessing the cardiac function. In the clinical arena, speed, automation, and costs are determinant factors for choosing tagging, and the most successful methods that have been developed so far to deal with tags lack these factors. Harmonic phase magnetic resonance imaging (HARP-MRI) is a promising new method that overcomes the drawbacks of the other existing methods and brings tagging closer to the clinical applications [20,21]. It is based on a different perception and understanding of the tagging process. Moreover, HARP-MRI is not limited to an image processing tool, but it paves the way to new imaging techniques based on its principles [22,23].

By using tagging, energy concentrations occur on the image spectrum (k-space), which HARP technique utilizes. Setting band-pass filters on the first harmonic peak of the spectrum, the first harmonic is isolated and with inverse Fourier transformation a complex HARP image consisting of one amplitude and one phase image data is created.

Figure 2.10 shows the overview of HARP for a short axis, tagged MR image of a healthy human heart at end-systole. The left ventricle appears as an annulus at the center of the image, and the tag lines, which are straight at end-diastole, are bent due to the heart contraction. The magnitude of the Fourier transform of the original image of Figure 2. 10 (a) is shown in Figure 2.10(b). These clusters in Figure 2.10(b) are produced by the tagging process. There are five spectral peaks; the one at the center is the *DC spectral peak* and the other peaks above and below are called the harmonic spectral peaks. The *DC harmonic* contains no tagging information. The outcoming harmonic peaks are directly dependent of the applied RF pulse and the created tagging pattern (Figure 2.11).

The tagging pattern is periodic, at least when the tags are imposed and before motion; thus, it can be expanded using the Fourier series into summation of sinusoids or harmonics. Each of these harmonics when multiplied by the untagged tissue produces a harmonic peak shifted in the spectrum with a frequency proportional to the tagging frequency. For the other higher harmonics, their shifts are multiples of the tagging frequency. The shifting is in direction orthogonal to the tag orientation.

It's useful, therefore, to define the *tagging vector* as the vector orthogonal to the tag lines and its magnitude is the reciprocal of the tags period. Because of this harmonic expansion, any tagged MR image can be written as a summation of images. Each of the terms in the summation is a complex image corresponding to single harmonic peak. These complex images are called the *harmonic images*.
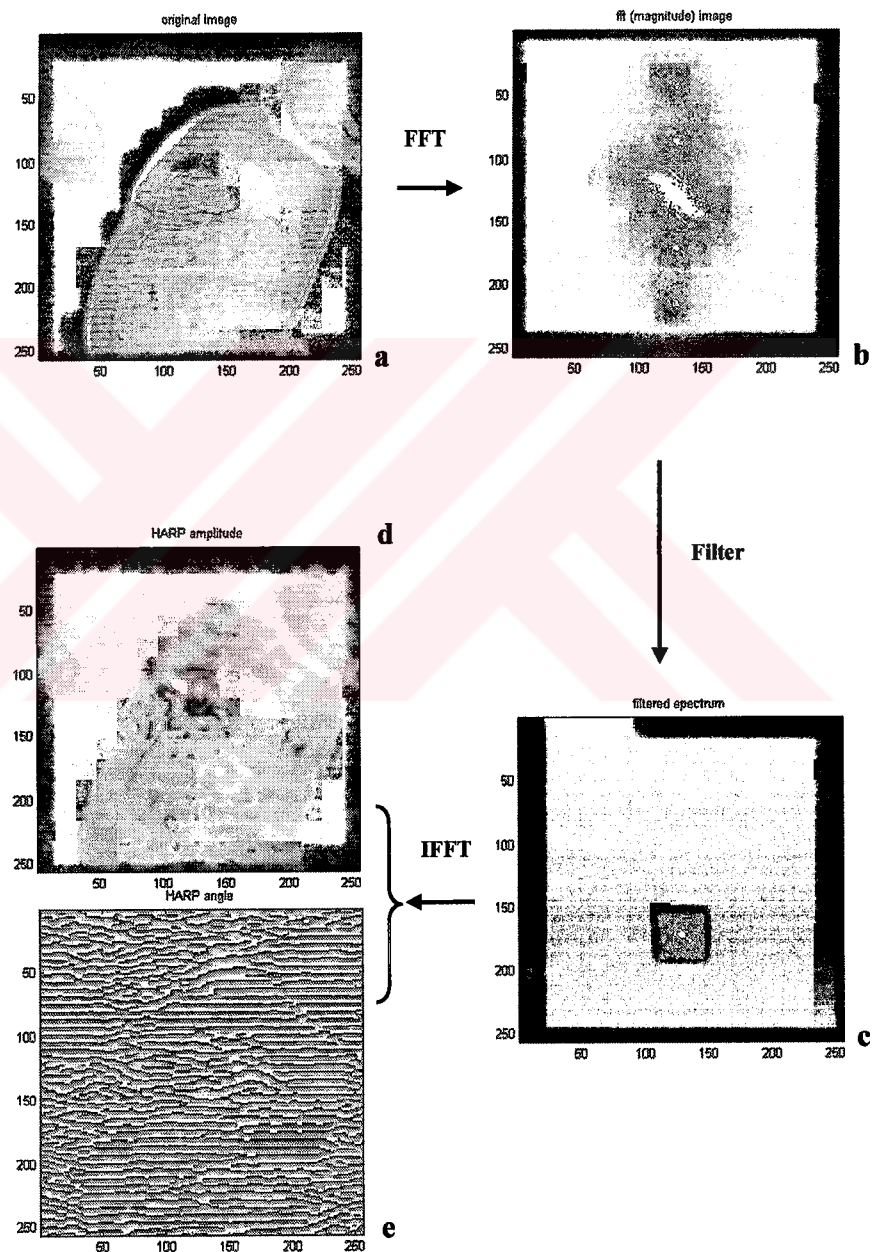
Figure 2.10 (a) A tagged LV image. (b) It's Fourier transform showing the magnitude. (c) Filtered spectrum. (d) Magnitude and (e) Phase of the complex image.
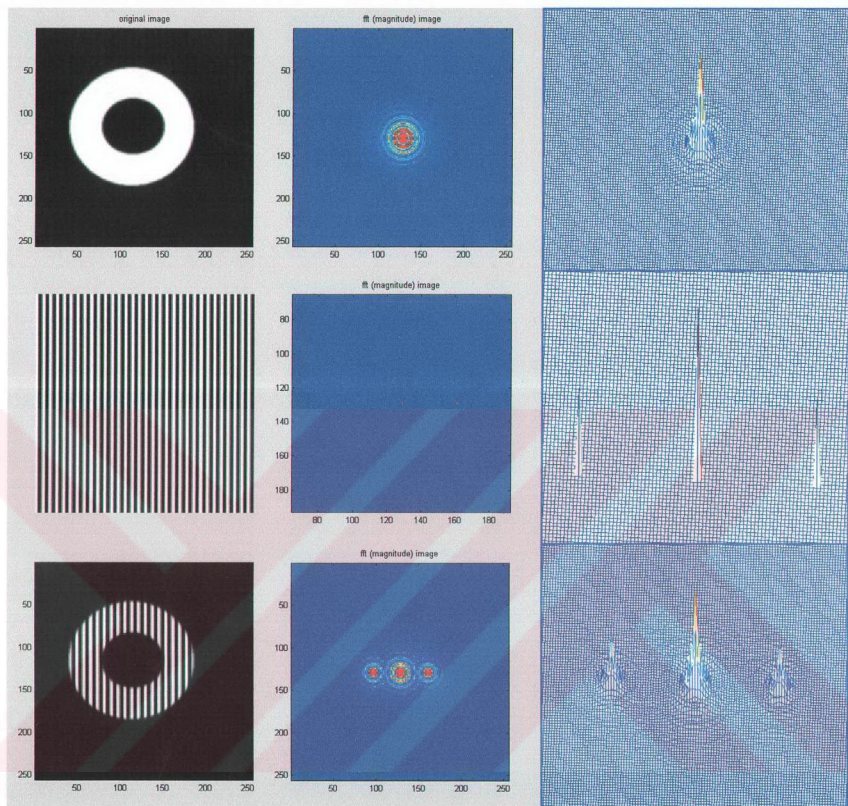
Figure 2.11 Getting the harmonic peaks after the creation and application of tagging pattern: On the left side column, from top to bottom; a synthetic LV image, a tag pattern, and the final tagged LV image is to be seen. The Fourier transforms of each image is placed on its opposite column. The FFT of the final image is the convolution of the other represented FFT's. According to the applied sinusoidal tagging pulse, the amount of the harmonic peaks is ascertained. The amplitude modulation of the underlying signal intensity is described by a pattern of cosines. Because a cosine has two symmetric spectral peaks in Fourier space (right column middle graph in upper figure), a 1-D SPAMM pattern generated with N RF pulses has $2N-1$ spectral peaks.

Assuming that the harmonic images are well separated, it is possibly to extract the harmonic image from the summation using a bandpass filter. In fact, the region shown in Figure 2. 10 (c) represents the passregion of a bandpass filter. In digital image processing, filtering in the spatial domain is widely used. In the spatial domain, an image $i(x,y)$ has to undergo a complex two-dimensional convolution with a filter $h(x,y)$ to yield a new, filtered image $j(x,y)$.

$$j(x,y) = i(x,y)*h(x,y) \tag{2.1}$$

$$J(u,v) = I(u,v)H(u,v) \tag{2.2}$$

The convolution, denoted as (*), is a combination of many arithmetic operations to be performed between a convolution kernel (the filter) and each image pixel as the kernel moves over the image.

Filtering in the frequency domain, after Fourier transformation of the image, is both versatile and straightforward [24]. In the frequency domain, the complex convolution between image and filter is reduced to a simple multiplication given in the Equation 2.1 of their corresponding Fourier transforms (Figure 2.11).

When zeroing everything outside the square and compute the inverse Fourier transform, a complex image is produced. We can decompose the complex image into *magnitude* and *phase*. HARP imaging isolates with a bandpass filter the k-th spectral peak centered at frequency w_k - typically the lowest harmonic frequency in a certain tag direction.

The inverse Fourier transform of the bandpass region yields a complex harmonic image $I(x,y)$ (Figure 2. 10.d-e):

$$D_k(y)e^{j\phi_k(y)} = IFFT\{Filter\{FFT[I(x,y)]\}\} \tag{2.3}$$

where D_k is called the *harmonic magnitude image* and ϕ_k is called the *harmonic phase image*. The harmonic magnitude image $D_k(y, t)$ reflects both the changes in geometry of the heart and the image intensity changes caused by tag fading. The motion causes a spreading of the energy around the spectral peak. In short axis images with tag planes

parallel to the long axis, the spreading is reasonably localized and it is possible to design a bandpass filter whose band-pass region isolates only a single spectral peak including most of the effects of phase modulation [20].

The magnitude of the harmonic image is called the *harmonic magnitude*, to distinguish it from the usual magnitude MR image. Figure 2.10 (d) shows the magnitude image of the harmonic image corresponding to the filtered peak in Figure 2.10 (c). This harmonic magnitude looks similar to the tagged image in Figure 2.10 (a) with two observations: The tag lines are absent, and the image is blurred. The latter is due to the filtering process, which reduces the resolution of the image. Despite this loss in resolution, the harmonic magnitude can be used for segmentation. Multiplying the binary images of the left ventricles with the phase images masking is done, and the results for one healthy human data are shown in Figure 2.12. This mask is useful for visualization and contains tagged and masked the cine MR images of a healthy human LV. The end-diastolic phase until end-systole is seen clearly in the Figure 2.12.
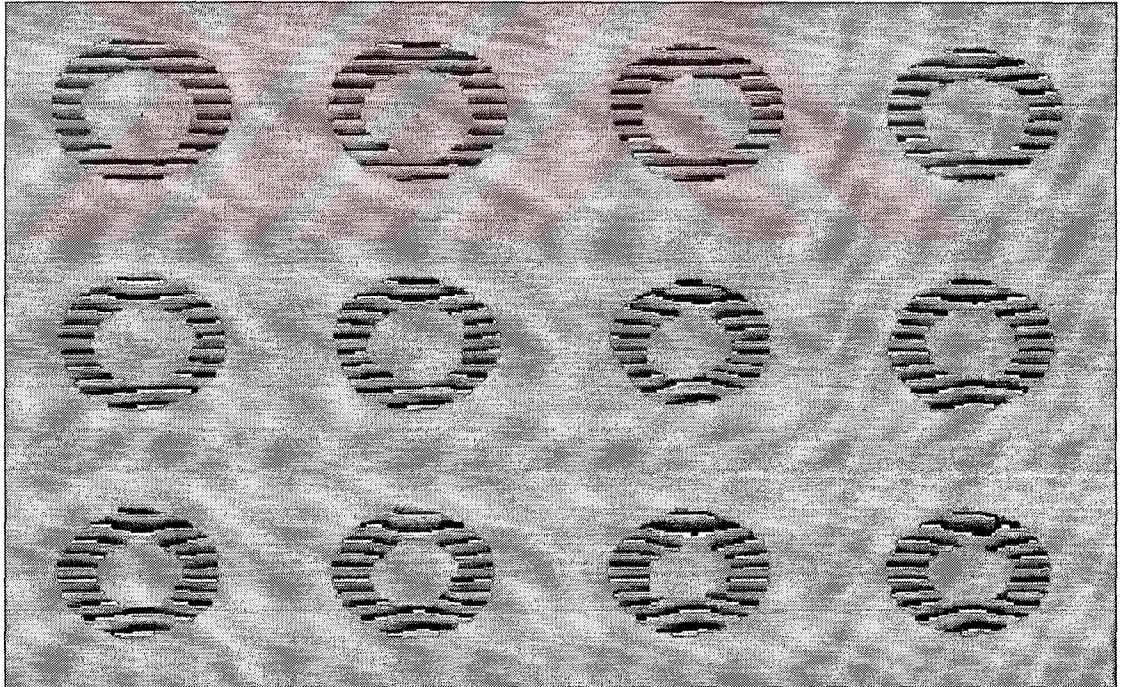
Figure 2.12 Masked LV with tags while systole.

The other component of the harmonic image that is called the *harmonic phase*, or simply *HARP*, can only be computed between the -180 to 180 degrees. The measured phase is wrapped version of the phase. The harmonic phase ϕ_k of $I(x,y)$ is an image given by:

$$\phi_k(y) = \omega_k \, p(x,y) \tag{2.4}$$

where ω_k refers to as the locations of the spectral peaks and $p(x,y)$ is representing the denoted points. A HARP image is defined as the calculated phase of the complex image $I(x,y)$ [21,25]:

$$a_k(x,y) = \angle I_k(x,y) \tag{2.5}$$

where

$$\angle I_k(x,y) = \begin{cases} \tan^{-1} \dfrac{\operatorname{Im}\{I_k\}}{\operatorname{Re}\{I_k\}}, & \operatorname{Re}\{I_k\} \geq 0 \\[4mm] \pi + \tan^{-1} \dfrac{\operatorname{Im}\{I_k\}}{\operatorname{Re}\{I_k\}}, & otherwise \end{cases} \tag{2.6}$$

Because of the inverse tangent operator, a HARP image is principal value of its corresponding phase image, and is restricted to be in the range $[-\pi,+\pi)$ [21,23,26]. Formally a HARP image is related to the phase image as follows:

$$a_k(x,y) = W(\phi_k(x,y)) \tag{2.7}$$

where the nonlinear wrapping function $W(\phi)$ is given by:

$$W(\phi) = \operatorname{mod}(\phi + \pi, 2\pi) - \pi \tag{2.8}$$

Figure 2.13(b) shows the phase of the image in Figure 2.13(a) masked using the harmonic magnitude, which we have described earlier. The sharp features that we see are due to the change of value from 180 to -180 degrees caused by wrapping (Equation 2.8). A keen observation of the phase shows that the sharp features bend similar to the tag lines in the original image. This observation indicates that the tag lines are represented in the phase of the harmonic image. It can be shown that the phase actually depends on the underlying motion as well as the tagging parameters. In fact, the motion is not only represented at the sharp features in the HARP image, but at other points as well.
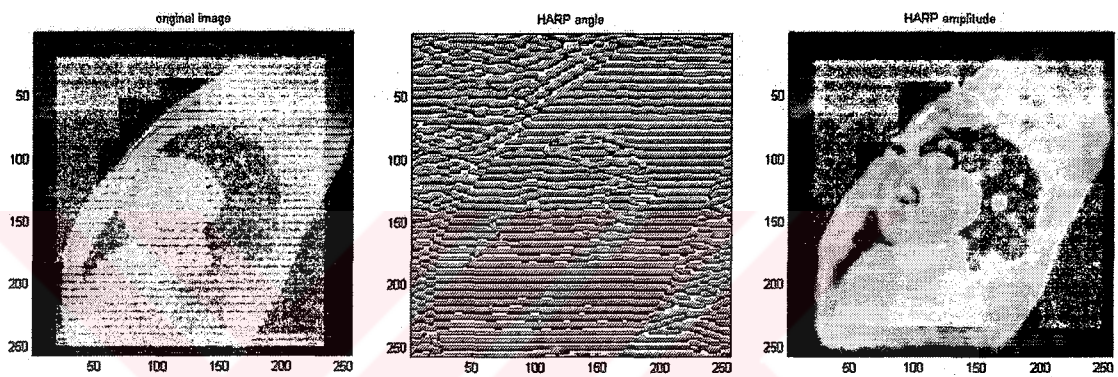
Figure 2.13 (a) Short axis tagged healthy human LV, (b) Harp angle, (c) amplitude images.
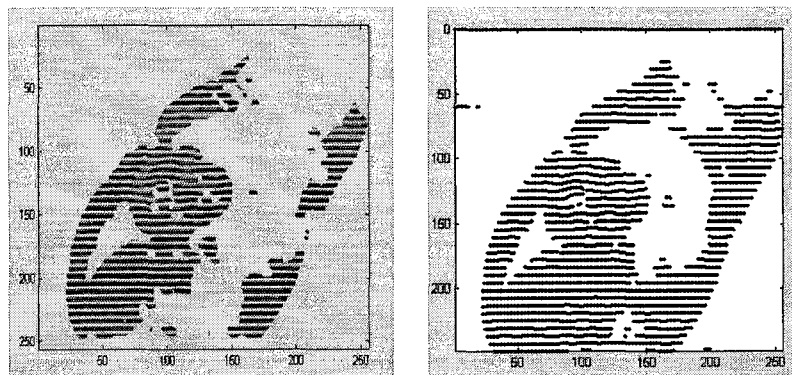
Figure 2.14 (a) Masked Image of the original MR image in Figure 2.13(a).
(b) Calculated tags representation after analysis.

Mathematically, it is shown that the phase of the harmonic images is linearly dependent on the motion direction orthogonal to the tags (in the direction of the tagging vector) (Equation 2.4), and distorted by a nonlinear wrapping function (Equation 2.8). The tag lines in the phase image (Figure 2.13b) correspond to the transition in angle from $+\pi$ to $-\pi$ caused by the wrapping. Since these lines define well the apparent movement of the tag pattern itself, they demonstrate the fact that the tag pattern phase is a material tissue property, remaining constant despite fading and intensity variations of the tag pattern.

The spectrum of the harmonic image is definitely spread over the whole spatial frequencies. It can be argued that most of the spectrum will concentrated around the corresponding harmonic peak shown earlier. The size of the spectral peak can be considered containing most of the energy of the harmonic peak (for example 90%). The size of this region and location is definitely going to be affected by the underlying motion of the heart. The contraction of the heart causes increase in the size of the area (spreading), with its maximum at end-systole (where maximum tags bending occur). The spreading of the spectrum can be explained by the changes in the local spatial frequency of the tags due to strain. For regions where contraction is orthogonal to the tag lines, a certain increase in tagging frequency occurs in proportion to the strain. On the contrary, a stretching in the tissue causes decrease in the local frequency. Since the spectrum is similar to the distribution of these local frequencies, its distribution is bound by the maximum changes in the local frequencies, which is dependent on the maximum local strains [22].

In case of the heart, the local strain is bound by the local contractility of the fibers. Thus, the expected size of the harmonic is expected to be limited. Doing some simulations it's assumed that the radius of the harmonic is of the order of the maximum strain and the tagging frequency. "Literally, if the maximum shortening in the case of the heart is 40%, we assume that the radius of the harmonic peak and the filter is 40% of the tagging frequency" [22].

The harmonic images intensity fades in accordance with the tags fading, thus reducing the SNR of the harmonic image. A spill over from other harmonics (interference) into the filter passregion is another artifact. The design of the filter, and the imaging pulse sequence in the future, should address these factors to minimize its effects.

3. METHODS

3.1 Cardiac Motion Estimation

The main objective in cardiac motion analysis is estimating measures of local LV wall contraction in order to identify pathological regions like ischemia, infarction, and DCM, which is not a local pathology. These measures are typically derived from a local motion description estimated from the time sequences of cross-sectional, in our case short axis LV images.

The primary difficulty in estimating local cardiac motion from a standard MR image sequence is the inability to observe motion in a direction orthogonal to the image intensity gradient, which means the motion cannot be determined in the tangential direction. Motion in the radial direction can be determined from the position of the endocardial boundary in each frame.

The motion of a tag pattern provides indirect information about the underlying tissue motion. An inverse problem that is known in the image processing and computer vision literature as the *motion estimation problem* must be solved to obtain a quantitative estimate of LV motion from the motion of the tag pattern. There are two basic approaches to the motion estimation problem: optical flow and feature-based methods [1,23]. Optical flow methods estimate a dense velocity or displacement field from the intensity changes in the image sequence. Feature-based methods identify a relatively sparse set of points, lines, planes, etc. in each image in the sequence. An inter-frame correspondence is established between these features and, along with constraints such as rigid-body motion or motion smoothness, this is used to estimate a set of motion parameters.

Once the features have been identified and tracked, some type of a priori motion constraint is used to estimate motion parameters of the entire object from the motion of the features. Examples of such motion constraints include isometric (length and angle preserving) motion and homothetic (uniform expansion or contraction) motion. The complex motion of the LV wall does not fit any of the constraints' categories and necessitates the use of a more general motion constraint.

3.2 Reliability of HARP Technique

Before analyzing the human cardiac left ventricles with the proposed technique, the reliability of HARP method is tested. The analyzing method is proved by the synthetic images shown in Figures 3.1, 3.2(b), and 3.3. All steps of the technique are represented in next figures.
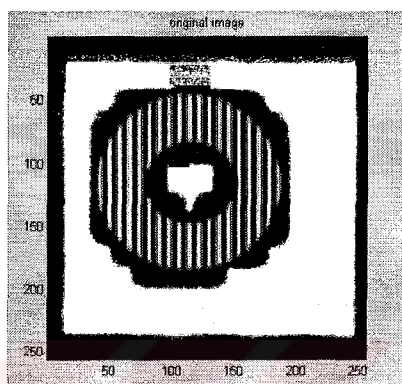
Figure 3.1 Short axis tagged synthetic LV.
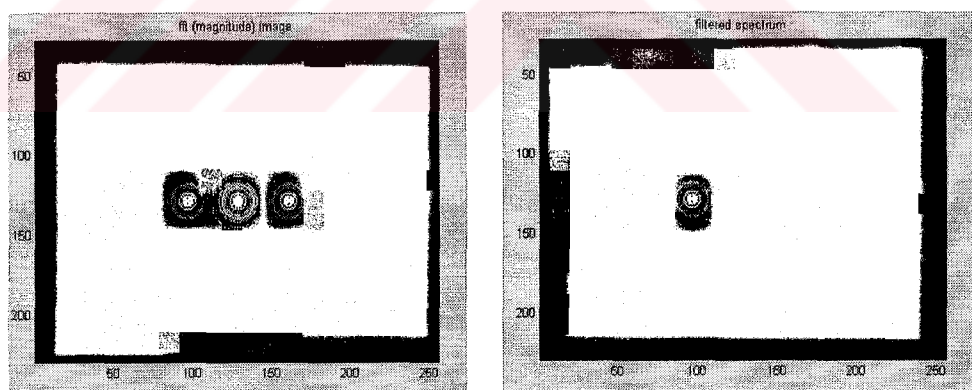
Figure 3.2 (a) Harmonic peaks of the original synthetic image. (b) The energy spectrum after bandpass filtering.

After applying the Fourier transform, the energy spectrum in Figure 2.14(a) is achieved. The first harmonic peak is being filtered by the bandpass filter (Figure 2.14b). As a result of HARP technique, the complex image is found, and it's phase and magnitude images are represented in the Figure 2.15.
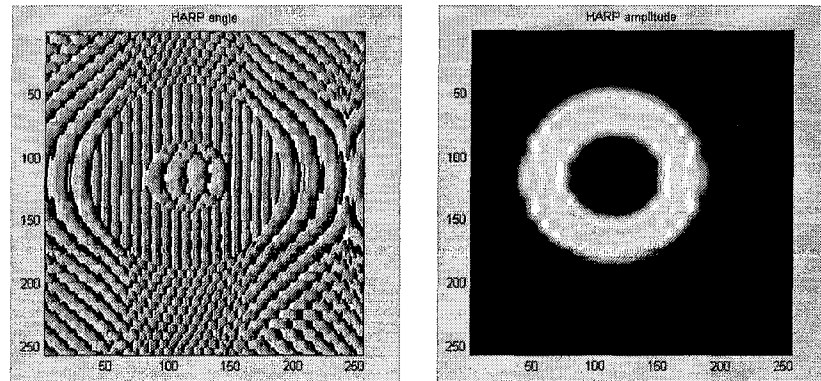
Figure 3.3 HARP angle and magnitude images of the synthetic LV.
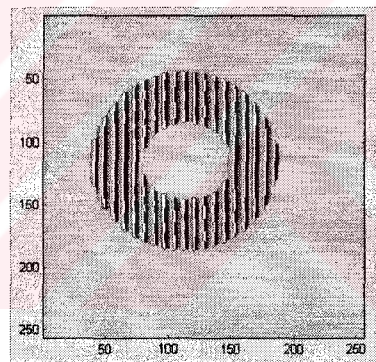
Figure 3.4 Contoured and masked synthetic LV.
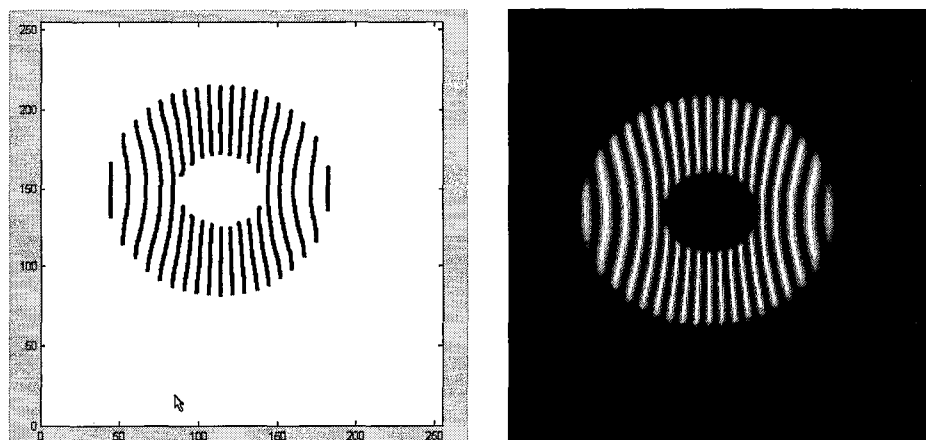
Figure 3.5 (a) Calculated tags of a deformed (b) synthetic LV.

The harmonic phase derived tag detection error in synthetic data is acceptable. There appear to be only minor differences between HARP isocontours and tag contours estimated using Findtags. The positions of MR tags placed on synthetic images were found with an error of 0.562 ± 0.016 pixel [27]. This was shown to be robust at the expected noise levels. For the tag lines on a healthy human myocardium, the calculated locations were compared to tag points that are found using Findtags. For various frames in the cardiac cycle these errors were: 0.14 ± 0.12; 0.18 ± 0.17; 0.24 ± 0.22; 0.29 ± 0.33 pixel. Although the same examination is done for noise applied synthetic LV representations, and the tag lines detected by our algorithm corresponds the tags found by Findtags (Figure 3.6). The noise ratio is defined as the ratio of the noise standard deviation to the maximum image magnitude (the reciprocal of the SNR is used so that the zero-noise case can be plotted). It is typical that the performance degrades gradually as the noise increases.
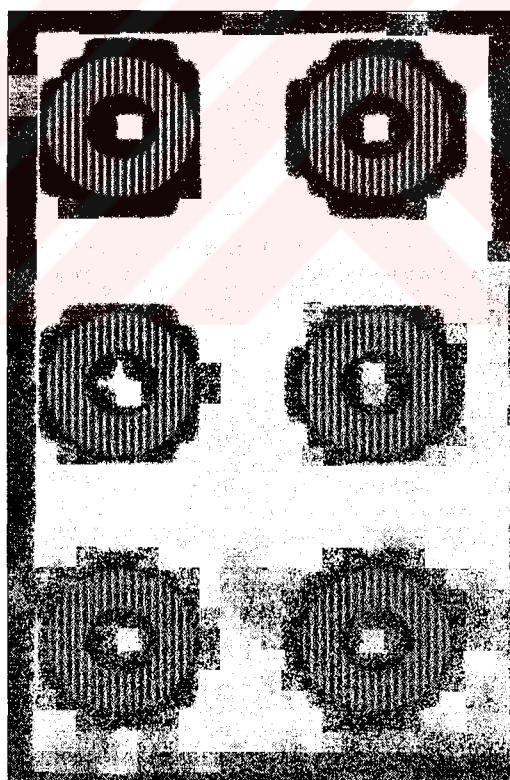
Figure 3.6 Noise applied synthetic left ventricles.

3.3 Image Processing

3.3.1 Contouring and Tag Detection

The first step in any feature-based method is to identify a set of features in each image to establish an interframe correspondence between them. In the MR tagged cardiac motion estimation problem, the features of interest are tag points and their interpolated tag lines on the tag planes. The endocardial and epicardial contours of the LV may also be identified. These LV contours are used for segmentation. Guttman, et al. proposed a semi-automated algorithm, *Findtags*, for detecting and tracking endocardial and epicardial contours and tag lines from planar-tagged MR images based on differential edge detection and template matching techniques [28-30]. In Figure 3.7, the tag lines estimation of the Findtags software program is seen.
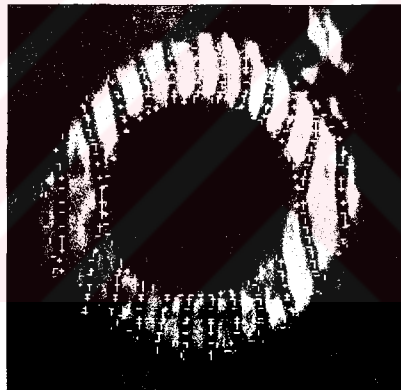
Figure 3.7 The Findtags interface. A short axis tags for a specific slice.
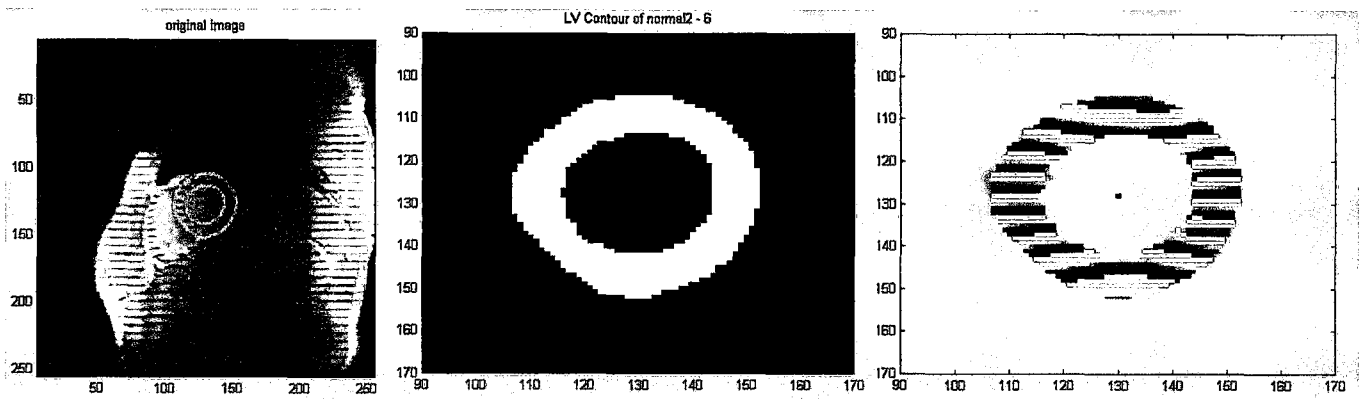
Figure 3.8 The manually applied and by Findtags interpolated LV and RV contour is set on the original image (a) in blue color. (b) The contour is represented in binary image. (c) Contoured ventricles masked with the angle image.

Using data from the normal and abnormal human subject, Findtags is used to detect the contours (endo and epicardium) in images (Figure 3.8). The estimated and stored contour data for each data set is used to mask the angle image, and the relating HARP image is get to estimate the tag lines.

The analysis is continued by detecting the tag locations, which corresponds to phase discontinuities. Theory predicts that tag brightness minima should be located at a phase angle of π radians, and therefore π isocontours of HARP images should be very close to the tag points identified by Findtags [31]. There appear to be only minor differences between HARP isocontours and tag contours estimated using Findtags [27]. By analyzing the HARP phase images, there are several practical problems with the direct use of the Equation 2.7. The phase image ϕ is not available, and its wrapped version a must be used in its place. It's clear from Equation 2.7 that the gradient of a_k is the same as that of ϕ_k except at a wrapping artifact, points of discontinuity (Figure 3.8c), where the gradient magnitude is theoretically infinite and practically very large [20,21,23]. Adding π to a_k the rewrapping shifts the wrapping artifact by ½ spatial period, leaving the gradient of this result equal to that of ϕ_k wherever the original wrapping artifact occurred. In other words, the HARP angle isocontour value a_k is set to π so that the generated tag lines coincided with the tag lines in the original image. So the determining of the tag lines is converted to the problem of tag localization to zero-crossing detection at each row/column using $\pi/2$-shifted harmonic phase image [21]. The zero-crossings that define the tag points are calculated for each time frame of healthy and diseased-tagged MR images (Figure 3.9). Points detected on each tag line are registered in space and time.
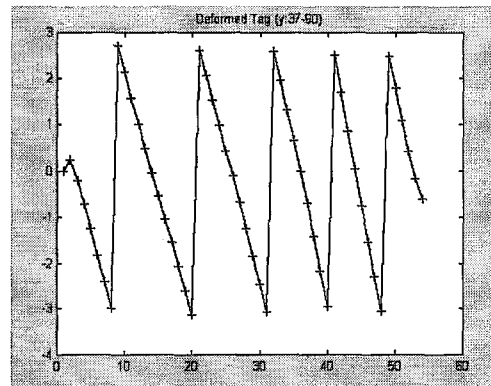
Figure 3.9 Representation of $\pi/2$-shifted HARP image to estimate the zerocrossings.

3.3.2 Motion Tracking

The motion tracking algorithm used and adapted in this thesis study is based on the cubic splines analysis. It was used to reconstruct a surface from a scattered set of data samples. The method is geometry dependent; therefore shift, scale and rotation normalization is applied to every data.

Several reasons for choosing cubic spline interpolation method were: That it has a very tractable, easy-to-code form. When the heart contracts, the thin, essential planar, surface of tagged tissue is subjected to deforming stress. Also, the interpolation can be used with the irregular sampling patterns of tag line data. It does not require boundary conditions, and at last, it has the desirable characteristic of being continuously differentiable.

3.3.3 Cubic Spline Field Fitting

A problem with polynomial interpolation is that higher-order polynomials sometimes produce undesirable fluctuations when the polynomials are forced to fit the data exactly. Small errors in the data can then have undesirable effects on interpolated values. The spline provides a technique for obtaining a smoother interpolation formula. A cubic spline $s(x)$ is constructed for each interval between data points by determining the four polynomial coefficients as follows. One of the requirements is that the endpoints of the polynomial match the data (N is the number of spline fitting points):

$$s(x_i) = f(x_i) \qquad i=1, .., N \qquad (3.1)$$

The two other constraints arise from the requirement that the first and second derivatives be the same as in adjoining intervals. (These constraints only provide two constraints, because they are shared with the nearby data intervals). It is conventional to specify that the second derivatives vanish at the endpoints of the data set. This then specifies a set of simultaneous equations to be solved for the interpolating function [32]. Computer routines are readily available to perform these interpolations.

The estimation of the value of the cubic curve coefficients throughout a particular region of interest in our case the LV is done by the given discrete samples of that parameter (tag points) in and around that region. The calculated tag points are fitted to cubic-splines to find out the coefficients for each tag line (l) in every frame (t);

$$F_{l,t}(x) = A_{l,t}x^3 + B_{l,t}x^2 + C_{l,t}x + D_{l,t} \qquad (3.2)$$

where $F(x)$ defines the tag line function with A, B, C and D the cubic curve coefficients. In displacement field-fitting, the parameter of interest is the 3D displacement vector and the samples are the values of 1D displacement measured at points on tags in the heart wall. Although field-fitting is generally applicable to any motion detection method, it has been applied here to the analysis of the cubic curve coefficients of displacement measurements from parallel tagged MR images. This type of data is depicted in Figure 3.10, which diagrams a short axis image and a deformed set of tags. The calculated tag points and its interpolated line are represented in this figure.
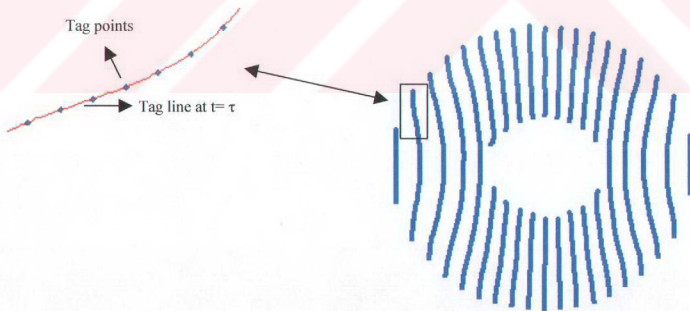
Figure 3.10 Depiction of a short image at some time after initial tagging. The inset depicts an enlarged view of one deformed tag line after detection of tag points at 1 mm intervals.

3.4 Normalization of the Calculated Tag Lines

To be able to make a classification of the data set according to analysis of the normal tagged LV images, the contoured left ventricles images should be normalized. To adjust the coordinates of the tag points to shift, scale and rotation invariance, following steps are fulfilled:

First, the origin of every investigated LV is fixed by detecting the edges of the relating LV contour in binary image (Figure 3.8(b)). Edging the endo- and epicardium contours of the LV, the coordinates of its origin is ascertained. Every LV is set then to a new coordinate system, where the origin of the LV contour is its coordinate system's origin (Figure 3.11). Next, the straight line for each calculated tag line is estimated by one dimensional curve fitting (Figure 3.11a). So the angle between the fitted straight line and the investigated tag line is found. The LV is rotated then for that angle value. For each LV mostly straight lying coordinates are found in this way (Figure 3.11c).
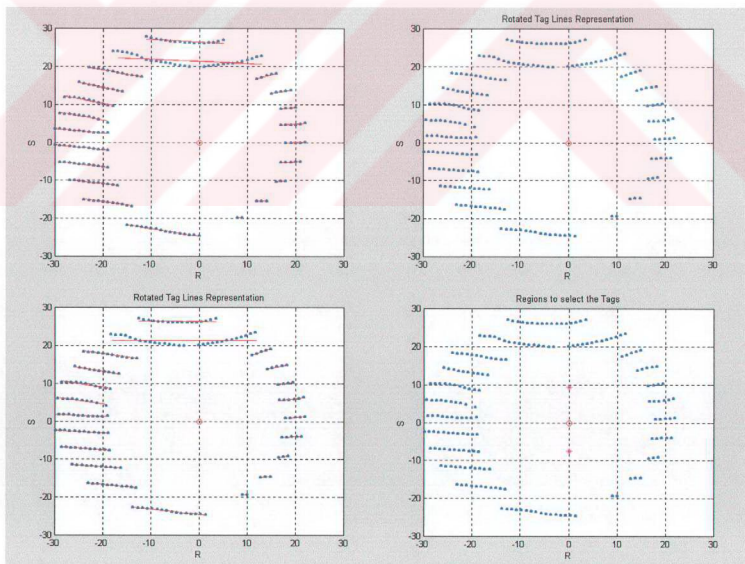
Figure 3.11 Normalization steps: Calculated origin. Coordinate system's origin is on the center of the LV, (c) Red lines represents the straight lines to find the rotation angle, (d) Rotated and segmented LV contour.

To calculate the normalization factor of each slice, the first and the last tag lines in the HARP image is found and the distance Δy between these two tag lines is calculated. The y-coordinates of the tags are divided by that value. Also, for every tag line, the x-coordinates are normalized for the value calculated by each tag line's own Δx, which lies in interval [-0.5, 0.5]. Asymmetric scaling is done by that way. Before setting the tags into the cubic curve fitting, the origin of the coordinate system is carried to the middle of the investigated tag line. The reason to do this normalizing is to examine every tag line independent of any other tag, and the coordinate system. So every single tag line is examined on its own, and a classification may be allowed according to these procedures. For the classification, the LV contour is set to three regions of interests (Figure 3.11d). The segmented regions will lead to compare the tag lines in the same regions of interest even if the LV is tagged much or less in another examined data.

3.5 Application of Probability Density Function

To ascertain the data whether it is pathological or healthy, probability density function for multivariate normal distribution is used. We worked on the end-diastolic and end-systolic time frames of every examined image sequence, which corresponds to the relaxed and maximum contracted phases of the LV. First, we selected totally 8 tag lines from the images sequences that are shown as pink dots in Figure 3.5d. For each time frame, and each tag line of this time frame, the coefficients calculated by fitting the tag points to the cubic spline, are stored in matrixes. So to examine the coefficients separately by using Equation 3.3, the column of the selected tag's coefficient is found, and set into the probability density function, where the mean value is calculated from the data of the training set.

$$f_{X_1 X_2}(x_1, x_2) = \frac{1}{\sqrt{(2\pi)^n \det(C_{xx})}} e^{-(1/2)(x-\mu_x)' C_{xx}^{-1}(x-\mu_x)} \tag{3.3}$$

where x and μ_x are each 8 by 1 vectors with components x_k and $E\{X_k\}$, respectively, for $0 \le k \le 1$. There are 8 elements of the examined data x, since 8 tag lines are selected from each sample in end-diastolic and end-systolic time frames. The four curve parameters are investigated separately. As a result of the parameter analysis, it's seen that the motion difference affects highly the first coefficient of the curves. Therefore x is an 8 by 1 vector containing the A' s of the tested tag line, and C_{xx} is the associated 8 by 8 covariance matrix. The symbol $\det(C_{xx})$ denotes the determinant of the covariance matrix [33].

3.6 Identification of Data by Confusion Matrix

The data includes 5 normal, 4 DCM, and 5 human infarct tagged MR images. The data is shared into training and test dataset. So the training data has 3 normal, 2 DCM, and 3 infarct subject, where the test dataset is the rest with 2 healthy, 2 DCM, and 3 infarct

image set. To differentiate normal and diseased myocardium using their curve parameters the test data is examined. The evaluation of the reliability of the algorithm is tested with the training dataset. The training and test data analysis results are arranged in confusion matrices as shown in Table 3.1 and Table 3.2. The dataset from which the test tag lines are chosen are placed at the first column of these tables. The abbreviations N, D1, and D2 on the table stand for normal, DCM and infarcted data. The first row shows the distribution of segmentation among three tissues. For example, in the third row, 2 test data compared among 2 timeframes as mentioned before while end-diastole and end-systole and 8 selected tag lines for each are classified as 2 diseased of second type (D2). The training normal (N) data is also classified as normal with 100% success.

Table 3.1
Classification results of 7 training dataset.

	End-Diastole	Result of identification		
		N	D1	D2
Samples from which the tags are chosen	N	3	0	0
	D1	0	2	0
	D2	0	0	2

Table 3.2
Classification results of 7 test dataset.

	Mid-Systole	Result of identification		
		N	D1	D2
Samples from which the tags are chosen	N	2	0	0
	D1	1	1	0
	D2	1	0	2

On the Table 3.2, the results of test data analysis are expressed. The classification is fully successful for the healthy data. On the other hand, in each diseased dataset one sample fails to be identified correctly.

4. RESULTS AND DISCUSSION

In this study, the motion analysis of the human left ventricle is achieved to identify the dataset whether there is pathology in the LV or not. The next paragraphs introduce the explanation of the results for in vitro and in vivo analysis of the cardiac images.

Before analyzing the human data, the HARP method is proved to be correct by using the Findtags software as gold standard as mentioned in Chapter 3.2. The calculated tag lines' error for synthetic LV is 0.562 ± 0.016 pixel. For various frames in the real cardiac cycle these errors were: 0.14 ± 0.12; 0.18 ± 0.17; 0.24 ± 0.22; 0.29 ± 0.33 pixel [27,34,35].

To examine the test data coefficients, which consist of 3 normal, and 4 pathological data with 2 DCM, and 2 infarction, the mean values of the normal and diseased data coefficients of the training dataset are found, and placed in the probability density function in Equation 3.3. The test data includes 2 normal, and 4 pathological data with 2 DCM, and 3 infarcted LV. At the beginning of the test we admit the test data as unknown dataset. To compare and identify the results of the probability density function for training and test data, confusion matrix is created. Table 3.1 and Table 3.2 give the outcome of the identification algorithm. One sample in each diseased dataset failed to be identified correctly. So 1 of 2 DCM, and 1 of 3 infarcted data are classified as healthy data in the analysis. The reason for this failure might be the selection of the tags. The motion information of the underlying tissue is on every tag applied on the myocardium, but in diseased data the fact changes, because of the pathology on the LV wall the tags are not well contracted even in mid-systole. Therefore the information from such a tag could be defined as normal behavior of the LV and identified as healthy tissue. But because the first and last tags on most LV fade while mid-systole, we were not able to add them to the analyzed tags; they are left out on purpose.

The time frames and tag line number of the LV sequences of the data are different. To overcome this difference, the end-diastolic and end-systolic time frames are selected. The tag lines to examine are chosen from the 3 region of interest for each image sequence. The aim of selecting the mentioned two time frames is that the motion variation is known, and similar conditions are expected in between the datasets. In fact, in examining the

diseased data, the coefficients' values are not in a similar partition. The coefficients A, B, C and D of the abnormal dataset are different then normal-to-normal data comparison (Figure 4.1). That's expected because the DCM and infarct myocardium differs in systolic motion compared to the healthy myocardium. Therefore according to the variances in the four coefficients, the classification can be done successfully.

The curve-fit to the tag line passing through the mid-myocardium at the top portion of the LV for each data set at end-systole is performed to get the coefficients A, B, C and D. For the selected coefficient (A), the changes between normal and diseased, and relaxed and contracted time frames are presented in Figure 4.1 and Figure 4.2. The changes for the 8 tag lines can be seen on the figures. Because of the contraction while systole, the blue (healthy) coefficient A changes, which is expected. The green and pink colored diseased data curve coefficient doesn't differ much and its variance between end-diastole and end-systole is not the value of healthy sample coefficient's variance [36].
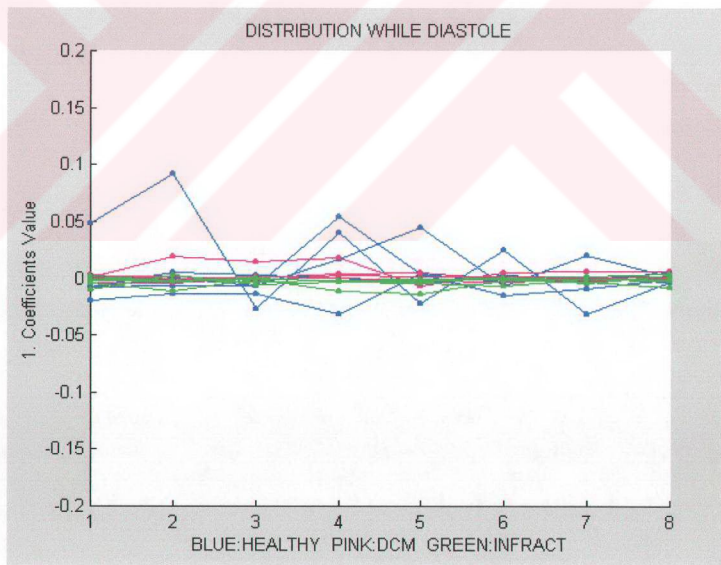
Figure 4.1   Normalized LV curve coefficient A at end-diastole; Blue lines represent the healthy data, and the pathological data is represented in green (infarcted) and pink (DCM) color. The x-axis represents the selected 8 taglines, and the y-axis is the first curve coefficient A.
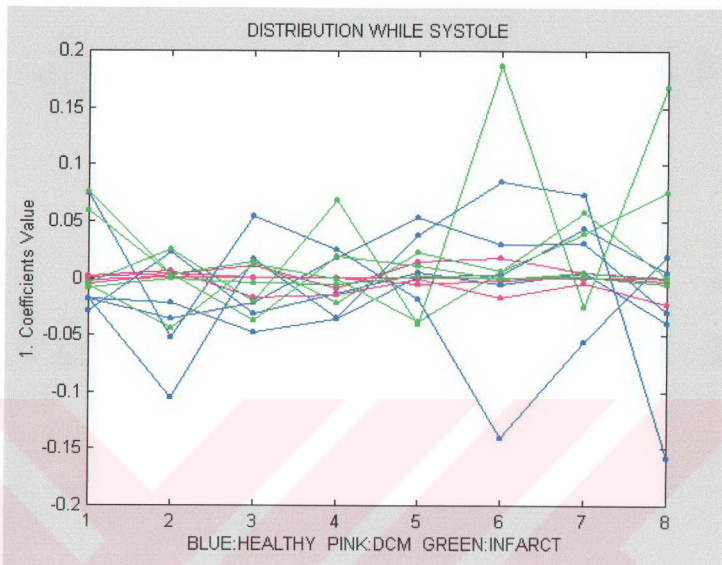
Figure 4.2 Normalized LV curve coefficient A at end-systole; Blue lines represent the healthy data, and the pathological data is expressed in green (infarcted) and pink (DCM) color. The x-axis represents the selected 8 taglines, and the y-axis is the first curve coefficient A.

The tag points for the end-diastolic and end-systolic frame of a healthy cardiac systole are displayed in Figure 4.3. Figures 4.4 shows the initial analysis for the first tag line of Figure 4.3. In most cases, the simple cubic polynomial curve-fit is adequate.
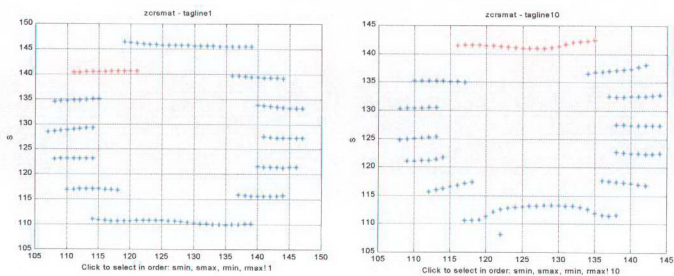
Figure 4.3 Calculated tag points for healthy human heart at end-diastole (left) and (right) end-systole.
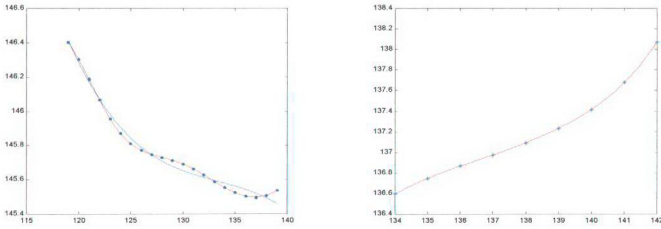
Figure 4.4 Cubic spline and polynomial data interpolation of the first tag line of tag points represented in Figure 4.2.

As mentioned, the diseased data consists of human dilated (DCM) and infarcted cardiac MR images. The analyzed tag points for the diseased data in end-systolic time frame, where the contraction must be maximum, are displayed in Figure 4.5 for DCM and in Figure 4.6 for infarcted LV. The curve-fit results for the tag points appear sometimes in an exaggerated curvature. That's because of zooming on the tag and unequal scaling of both axes.
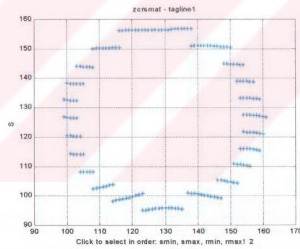
Figure 4.5 Calculated tag points of a DCM image while end-systole.
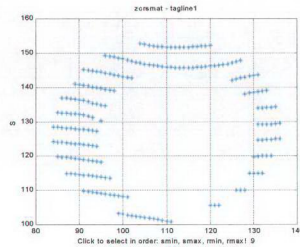
Figure 4.6 Determined tag points of an infarct data while contraction.

The expected changes show up in the results. As represented in Figure 4.5 and 4.6, the DCM and infarct myocardium differs in systolic motion compared to the healthy myocardium (Figure 4.3). Our method has been a verification of the facts.

In this study, we had limited number of images on the dataset. To deliver an automatic classification for the diseased images, the calculation must be done for many MR tagged cardiac images.

5. CONCLUSION AND FUTURE DIRECTIONS

The aim of this research was to rapidly classify the tagged human cardiac MR images, whether they are normal or abnormal.

To evaluate the correctness of the segmentation algorithm the confusion matrix method is used. Although, the sample number was not sufficient, the whole 14 tagged human cardiac images are divided into test and training datasets.

In our approach, we used the cubic curve fit and we found the polynomial coefficients of the tag line at the end systolic time frame, where the tag displacement is high compared to the first time frame. All of the examined dataset of diseased samples differs in cubic curve coefficients that have been normalized and calculated before classification process. The four coefficients of the training and test dataset are calculated, and compared to identify whether the examined data is healthy or diseased. The significantly differences between the dispersion of the coefficients lead to classify the datasets.

In this work, we had limited number of images on the dataset. To deliver an automatic classification for the diseased images, the calculation must be done for many MR tagged cardiac images. Our future work consists in cluster analysis to identify unknown datasets upon their coefficient variation.

APPENDIX

LISTING OF SOFTWARE

Matlab Code for Calculating the HARP Images: normal7.m

```
%
% code for running the whole time frame (12 slice)
% HARP for images HFC7.006.I.069-080!!!
%
I=zeros(256,256,12);

for k=69:80
    t=k-68;
I(:,:,t)=getsigna3(['C:\WINDOWS\Desktop\imgeler\normal7\I.1',int2str(k)]);
    beta=0.5;
    figure;imagesc(I(:,:,t));axis image;colormap gray;title(['Original
Image' int2str(t)]);brighten(beta);
    pause
end;


%
% show the original image
%
%for t=1:12
%figure; imagesc(I(:,:,t)); axis image; colormap gray;
%title('original image')
%end;

if 0  % use if you notice aliasing
    I2 = repmat(hanning(256)',256,1) .* I;
else,
    I2 = I;
end;


%
% fft of the image
%
for t=1:12;
    fI(:,:,t) = fft2(I2(:,:,t));
    fIs(:,:,t) = fftshift(fI(:,:,t));
    absfIs(:,:,t) = abs(fIs(:,:,t));

    %%phase image is figured%%

    %figure; imagesc(angle(fIs(:,:,t)));
    %colormap gray; axis image;
    %title('fft (phase) image');
    %
    if t==1
        %beta=0.5;
        figure;imagesc(absfIs(:,:,t),[0 80000]);
    %brighten(beta)  % this scale can be adjusted to view better
        colormap gray;
        axis image;
        title(['fft (magnitude) image' int2str(t)]);
        set(gcf,'Tag','fftimage');
    end;
```

```matlab
    if t==1
        fig=findobj('Tag','fftimage');
        figure(fig);
        [Xc,Yc] = ginput(1);
        Xc = round(Xc);
        Yc = round(Yc);
        %
        % make the edges of filter round
        % size of the filter should be determined automatically in the
final progam
        %
        sigma = 8;
        wsize = 31;
        midp = (wsize+sigma-1)/2;
        filt1 = myfilter(wsize,sigma);
    end;
end;
%
% select only the first harmonic
%
if 1
    filt2 = zeros(256);
    filt2((Yc-midp):(Yc+midp),(Xc-midp):(Xc+midp)) = filt1;
    for t=1:12;
        fIs2(:,:,t) = filt2 .* fIs(:,:,t);
    end;
else,
    fIs2 = fIs((Yc-midp):(Yc+midp),(Xc-midp):(Xc+midp));
end;
%%%
%%%filtered spectrum is figured for 12 time frames
%%%
%for t=1:12;
%figure;imagesc(abs(fIs2(:,:,t)),[0 100000]);
%colormap gray; axis image;
%title(['filtered spectrum' int2str(t)]);
%end;
%pause
%
% inverse fft of the filtered amplitude image
%
for t=1:12;
    I2(:,:,t) = ifft2(fftshift(fIs2(:,:,t)),256,256);
    I2abs(:,:,t) = abs(I2(:,:,t));
    I2ang(:,:,t) = angle(I2(:,:,t));

    if 0,
        sel = I2abs(:,:,t)<10;
        I2abs(sel) = 0;
        I2ang(sel) = 0;
    end;
end;
if 0
    imwrite(uint8(I2abs(:,:,t)),'tags.tif','compression','none');
end;

%figure; imagesc(angle(I2(:,:,t)));
%axis image; colormap gray;
%title(['HARP angle of ' int2str(t)]);
%end;
```

```
save(['LV_049_060' int2str(t)],'I2abs','I2ang');
% title(['HARP amplitude' int2str(t)]);
imwrite(uint8(I2abs(:,:,t)),'tags.tif','compression','none');


%
% ROI for the heart
%
for t=1:12;
    xroic = 128;
    yroic = 128;
    roisize = 100;

    figure;
    subplot(1,3,1);
    imagesc(I((xroic-roisize) :(xroic+roisize), (yroic-roisize)
:(yroic+roisize),t ));
    axis image; colormap gray;
    title(['Orig image' int2str(t)]);

    subplot(1,3,2);
    imagesc(I2ang((xroic-roisize) :(xroic+roisize), (yroic-roisize)
:(yroic+roisize),t ));
    axis image; colormap gray;
    title(['HARP angle' int2str(t)]);

    subplot(1,3,3);
    imagesc(I2abs((xroic-roisize) :(xroic+roisize), (yroic-roisize)
:(yroic+roisize), t));
    axis image; colormap gray;
    title(['HARP amplitude' int2str(t)]);
    pause

end;
clear k t fIs2 fI absfIs sigma midp wsize
clear roisize filt1 filt2 Xc Yc xroic yroic fIs fig
```

Matlab Code for Contouring the LV: poi_normal7.m

```
%
% normal7: HFC7.006.I.069-080 Points files of the sequence to be read
%

poi_normal7=readpoints('C:\FTP Data\Points\hfc7\','normal7','006');
p_normal7=poi_normal7;
p_normal7(9)=poi_normal7(1);
p_normal7(10)=poi_normal7(2);
p_normal7(11)=poi_normal7(3);
p_normal7(12)=poi_normal7(4);
p_normal7(2)=poi_normal7(5);
p_normal7(3)=poi_normal7(6);
p_normal7(4)=poi_normal7(7);
p_normal7(5)=poi_normal7(8);
p_normal7(6)=poi_normal7(9);
p_normal7(7)=poi_normal7(10);
p_normal7(8)=poi_normal7(11);
p_normal7(1)=poi_normal7(12);
```

```
save p_normal7.mat p_normal7;
clear poi_normal7;
for r=1:12;
    inner_c(:,:,r)=p_normal7(1,r).inner_contour;
    outer_c(:,:,r)=p_normal7(1,r).outer_contour;
    in_c(:,:,r)=inner_c(:,:,r)/1.4063; %mm_per_pixel:1.4063 !!!
    out_c(:,:,r)=outer_c(:,:,r)/1.4063;
end;

% endo and epicardium contours are calculated.

for r=1:12;
    [bw1(:,:,r),x1(:,:,r),y1(:,:,r)]= roipoly
(I(:,:,r),in_c(1,:,r),in_c(2,:,r));
    [bw2(:,:,r),x2(:,:,r),y2(:,:,r)]= roipoly
(I(:,:,r),out_c(1,:,r),out_c(2,:,r));
    %figure;imagesc(bw1(:,:,r));colormap(gray);hold on;imagesc(bw2(:,:,r))
    %1:white, 0:black
    bw(:,:,r)=xor(bw1(:,:,r),bw2(:,:,r));
    figure;imagesc(bw(:,:,r));colormap(gray)
    title(['LV Contour of normal7 - ' int2str(r)]);
    pause
end;
clear r in_c out_c x1 y1 x2 y2 inner_c outer_c;
```

Matlab Code for Finding the Edges of the LV: ec.m

```
sb=size(bw,3);
MEC=zeros(2,2,sb);
Rx=zeros(sb,1);Ry=zeros(sb,1);
for p=1:sb
    ec(:,:,p)=edge(bw1(:,:,p));
    % figure;imshow(ec(:,:,p));title(['Binary Image Contour' int2str(p)]);
    % pause
    [ecx ecy]=find(ec(:,:,p)); % edge coordinates
    mxcx=max(ecx);mncx=min(ecx); % min&max of the edge coordinates
    mxcy=max(ecy);mncy=min(ecy);

    eval(['mncx' int2str(p) ' =mncx;']);eval(['mxcx' int2str(p) '
=mxcx;']);
    eval(['mncy' int2str(p) ' =mncy;']);eval(['mxcy' int2str(p) '
=mxcy;']);
    eval(['MEC(:,:,p)=[' 'mncx' int2str(p) ',' 'mxcx' int2str(p) ';'
'mncy' int2str(p) ',' 'mxcy' int2str(p) '];']);
    eval(['Rx' int2str(p) '= mean(MEC(1,:,p))' ';' 'Ry' int2str(p)
'=mean(MEC(2,:,p));']);
    eval(['Rx(p)=' 'Rx' int2str(p) ';' 'Ry(p)=' 'Ry' int2str(p) ';']);
    %ec=~ec;%0:white; 1:black
    %figure;imshow(ec(:,:,p));title(['Binary Image Contour & Center '
int2str(p)]);
    %hold on;plot(Rx(p),Ry(p),'.r');
    %pause
    eval(['clear mncx mxcx mncy mxcy;']);
    eval(['clear mncx' int2str(p) ' mxcx' int2str(p) ]);
    eval(['clear mncy' int2str(p) ' mxcy' int2str(p) ]);
```

```
end;
for p=1:sb;
    ALL(p,:)=[Rx(p),Ry(p)]; % Center coordinates in matrix ALL
end;


%bw2=~bw2;
for r=1:sb;
    biw(:,:,r)=bw(:,:,r); %bw(:,:,r) was calculated in poi_norm7!
    [bwx bwy]=find(biw(:,:,r));
    mat=[bwx bwy];
    sm=size(mat);
    smm=sm(1)+1;
    new=zeros(smm,sm(2));
    ek(r,:)=ALL(r,:);
    for de=1:sm(1)
        new(de,:)=mat(de,:);
        new(smm,:)=ek(r,:);
    end;
    eval(['coord' int2str(r) ' =new;']);
end;
for p=1:sb
    eval(['cpix' int2str(p) '=zeros(256,256);']);
    eval(['s' '=' 'size' '(' 'coord' int2str(p) ');']);
    I22= I2ang(:,:,p)-pi;
    I22(I22<(-pi))= I22(I22<(-pi)) + 2*pi;
    for t=1:s
        eval(['cx' int2str(p) '=' 'coord' int2str(p) '(t,1)' ';']);
        eval(['cy' int2str(p) '=' 'coord' int2str(p) '(t,2)' ';']);
        eval(['cpix' int2str(p) '(' 'cx' int2str(p) ',' 'cy' int2str(p) ')'
'=' 'I22' '(' 'cx' int2str(p) ',' 'cy' int2str(p) ');']);
    end;
    %figure;imagesc(['cpix' int2str(p)]);colormap(gray);title(['Masked
Phase ' int2str(p)]);
    %pause
    eval(['clear cx' int2str(p) ' coord' int2str(p)]);
    eval(['clear cy' int2str(p) ' Rx' int2str(p) ' Ry' int2str(p)]);

end;
clear t s p r ec ecy ecx de I22 sb mat new ek sm smm biw Rx Ry;
```

Matlab Code to Calculate the Zero-Crossings: zcrsmat_normal7.m

```
%%
% zcrsmat1-12 for HFC7 will be found.
%%

init=1;
fin=256;
TF=12;
M=zeros(256,256,TF);
M(:,:,1)=cpix1';
M(:,:,2)=cpix2';
M(:,:,3)=cpix3';
M(:,:,4)=cpix4';
M(:,:,5)=cpix5';
M(:,:,6)=cpix6';
```

```
M(:,:,7)=cpix7';
M(:,:,8)=cpix8';
M(:,:,9)=cpix9';
M(:,:,10)=cpix10';
M(:,:,11)=cpix11';
M(:,:,12)=cpix12';
zcrsmat_seri_genel= zeros(256,256,TF);
for mm=1:TF;
    for i=init:fin
        row=M(i,:,mm);
        a=find(row);
        sa=size(a);
        sacol=sa(2);
        for j=1:sacol
            newrow(j)=row(a(j));
        end

        %else
        newrow =row;
        sacol = length(newrow)-1;
        %end

        %ind=zeros(1,sacol); % created to store the (-) values after (+)'s.
        dd=1;
        clear h hh;
        for d=2:sacol
            if newrow(d-1)>0 & newrow(d)>=0 & 0>newrow(d+1),
                h(:,dd)=[newrow(d) newrow(d+1)]';
                ind(dd)=d;
                dd=dd+1;
                %elseif newrow(d)==0,
                %   h(:,dd)=[0 0]';
                %    ind(dd)=d;
                %    dd=dd+1;
            end
        end

        if dd>1
            hh=h;
            is = size(hh,2);
            fitp=zeros(2,is);

            xo=[-1,1]';

            for g=1:is,
                fitp(:,g)=(hh(:,g)'*inv([ind(g) ind(g)+1; 1 1]))';
            end

            zcrs=(-fitp(2,:)./fitp(1,:))-1;

            zcrsmat_seri_genel(i,1:length(zcrs),mm)=zcrs;
        end;
    end;
end;
clear M fitp dd hh h g ind mm is zcrs newrow sacol a row init fin d i j p
sa xo
```

Matlab Code for Calculating the Tag Points upon Normalization:
taglines_n7_dnm.m

```
zcrsmat=zcrsmat_normal7;
TL=12;
for gs=9; % gs=number of time frames
    for bjk=1:TL; % bjk= number of tag lines
        [R,C,S]=find(zcrsmat(:,:,gs)); % ss9=163x1
        for h=1:size(R,1)
            R(h)=R(h)-mean(ALL(:,2));S(h)=S(h)-mean(ALL(:,1));
        end;
        plot(R,S,'.');hold on;plot(0,0,'ro');xlabel('R');ylabel('S');grid on
        xlabel(['Sırasıyla smin, smax, rmin, rmax için tıklayınız ! '
int2str(gs)])
        hold on;title(['zcrsmat - tagline' int2str(bjk)]);
        gin=ginput(4);
        smin=round(gin(1,2));
        smax=round(gin(2,2));
        rmin=round(gin(3,1));
        rmax=round(gin(4,1));

        intrv=find(S>=smin & S<=smax & R>=rmin & R<=rmax);

        snkt=S(intrv);rnkt=R(intrv);
        eval(['snkt' int2str(bjk) '=snkt;rnkt' int2str(bjk) '=rnkt;']);
        eval(['[' 'a' int2str(bjk) ',' 'd' int2str(bjk) ']' '=' 'polyfit'
'(rnkt,snkt,1);']); %snkt original points
eval(['LN' int2str(bjk) '=dilekfun' '(' 'a' int2str(bjk) ',rnkt' ');']);
%LN points of the best fitting smooth line
        clear rnkt snkt;
    end;
end;

figure;plot(R,S,'.');hold on;plot(0,0,'ro');xlabel('R');ylabel('S');grid
on
hold on;plot(rnkt1,LN1,'r-')
hold on;plot(rnkt2,LN2,'r-')
hold on;plot(rnkt3,LN3,'r-')
hold on;plot(rnkt4,LN4,'r-')
hold on;plot(rnkt5,LN5,'r-')
hold on;plot(rnkt6,LN6,'r-')
hold on;plot(rnkt7,LN7,'r-')
hold on;plot(rnkt8,LN8,'r-')
hold on;plot(rnkt9,LN9,'r-')
hold on;plot(rnkt10,LN10,'r-')
hold on;plot(rnkt11,LN11,'r-')
hold on;plot(rnkt12,LN12,'r-')
axis([-30 30 -30 30]);
%%
angles
%%
% polar coordinates to rotate the coordinate system for difference angle
%%
% ang_var>0 --> subtract ang_var
for bjk=1:TL
    eval(['SL=size(LN' int2str(bjk) ',1);']);
    eval(['Line' int2str(bjk) '=zeros(SL,2);']);
    %for main points:snkt
    eval(['[TH' int2str(bjk) ',' 'radius' int2str(bjk) ']=cart2pol('
'rnkt' int2str(bjk) ',snkt' int2str(bjk) ');']);
```

```
    eval(['TH' int2str(bjk) '=TH' int2str(bjk) '-(ang_var);']);
    eval(['[LN_x' int2str(bjk) ',' 'LN_y' int2str(bjk) ']=pol2cart(TH'
int2str(bjk) ',radius' int2str(bjk) ');']);
    eval(['Line' int2str(bjk) '(:,:)=[LN_x' int2str(bjk) ',LN_y'
int2str(bjk) '];']);
    %fitted lines after shifting
    %smooth Lines: LN
    eval(['[LTH' int2str(bjk) ',' 'Lradius' int2str(bjk) ']=cart2pol('
'rnkt' int2str(bjk) ',LN' int2str(bjk) ');']);
    eval(['LTH' int2str(bjk) '=LTH' int2str(bjk) '-(ang_var);']);
    eval(['[LLN_x' int2str(bjk) ',' 'LLN_y' int2str(bjk) ']=pol2cart(LTH'
int2str(bjk) ',Lradius' int2str(bjk) ');']);
    eval(['LLine' int2str(bjk) '(:,:)=[LLN_x' int2str(bjk) ',LLN_y'
int2str(bjk) '];']);
    clear SL
end;
%shifted tag points representation:
figure;plot(Line1(:,1),Line1(:,2),'b.')
hold on;plot(Line2(:,1),Line2(:,2),'b.')
hold on;plot(Line3(:,1),Line3(:,2),'b.')
hold on;plot(Line4(:,1),Line4(:,2),'b.')
hold on;plot(Line5(:,1),Line5(:,2),'b.')
hold on;plot(Line6(:,1),Line6(:,2),'b.')
hold on;plot(Line7(:,1),Line7(:,2),'b.')
hold on;plot(Line8(:,1),Line8(:,2),'b.')
hold on;plot(Line9(:,1),Line9(:,2),'b.')
hold on;plot(Line10(:,1),Line10(:,2),'b.')
hold on;plot(Line11(:,1),Line11(:,2),'b.')
hold on;plot(Line12(:,1),Line12(:,2),'b.')
plot(0,0,'ro');xlabel('R');ylabel('S');grid on
title('Rotated Tag Lines Representation')
axis([-30 30 -30 30]);

%%
%LINES FITTED TO THE SHIFTED TAGPOINTS!
%%
hold on;plot(LLine1(:,1),LLine1(:,2),'r-')
hold on;plot(LLine2(:,1),LLine2(:,2),'r-')
hold on;plot(LLine3(:,1),LLine3(:,2),'r-')
hold on;plot(LLine4(:,1),LLine4(:,2),'r-')
hold on;plot(LLine5(:,1),LLine5(:,2),'r-')
hold on;plot(LLine6(:,1),LLine6(:,2),'r-')
hold on;plot(LLine7(:,1),LLine7(:,2),'r-')
hold on;plot(LLine8(:,1),LLine8(:,2),'r-')
hold on;plot(LLine9(:,1),LLine9(:,2),'r-')
hold on;plot(LLine10(:,1),LLine10(:,2),'r-')
hold on;plot(LLine11(:,1),LLine11(:,2),'r-')
hold on;plot(LLine12(:,1),LLine12(:,2),'r-')

%% points defining the regions (code : cross)
% axes crossing of the first & last tag lines

Line1(:,:)=[LN_x1,LN_y1];
[m1,c1]=polyfit(LN_x1,LN_y1,1);
Pust=m1(2);

eval(['Line' int2str(TL),'(:,:)=[LN_x'
int2str(TL),',LN_y',int2str(TL),'];']);
eval(['[m2,c2]=polyfit(LN_x',int2str(TL),',LN_y',int2str(TL),',1);']);
Palt=m2(2);
```

```
P=[Pust Palt];
Abstand=(Pust+abs(Palt))/3;norm_fac=Abstand*3;
P1=Pust-Abstand;P2=P1-Abstand;
%P3=P2-Abstand;%P3=Palt verification: % variance=Palt-P3=zero!
figure;plot(Line1(:,1),Line1(:,2),'b.');
hold on;plot(Line2(:,1),Line2(:,2),'b.');
hold on;plot(Line3(:,1),Line3(:,2),'b.');
hold on;plot(Line4(:,1),Line4(:,2),'b.');
hold on;plot(Line5(:,1),Line5(:,2),'b.');
hold on;plot(Line6(:,1),Line6(:,2),'b.');
hold on;plot(Line7(:,1),Line7(:,2),'b.');
hold on;plot(Line8(:,1),Line8(:,2),'b.');
hold on;plot(Line9(:,1),Line9(:,2),'b.');
hold on;plot(Line10(:,1),Line10(:,2),'b.');
hold on;plot(Line11(:,1),Line11(:,2),'b.');
hold on;plot(Line12(:,1),Line12(:,2),'b.');
plot(0,0,'ro');xlabel('R');ylabel('S');grid on
hold on;plot(0,P1,'m*')
hold on;plot(0,P2,'m*');%axis([-30 30 -30 30]);
title('Regions to select the Tags')

for bjk=1:TL
    eval(['middle' int2str(bjk) '=(max(LN_x' int2str(bjk) ')+min(LN_x'
int2str(bjk) '))/2;']);
    eval(['newx' int2str(bjk) '=LN_x' int2str(bjk) '-middle' int2str(bjk)
';']);
    eval(['ratio' int2str(bjk) '=(max(LN_x' int2str(bjk) ')-min(LN_x'
int2str(bjk) '));']);
    eval(['newx' int2str(bjk) '=newx' int2str(bjk) './ratio' int2str(bjk)
';']);

    eval(['[m' int2str(bjk) ',c' int2str(bjk) ']=polyfit(LLN_x'
int2str(bjk) ',LLN_y' int2str(bjk) ',1);']);
    eval(['pay' int2str(bjk) '=m' int2str(bjk) '(2);']);
    eval(['newyy' int2str(bjk) '=LN_y' int2str(bjk) '-pay' int2str(bjk)
';']);
    eval(['newy' int2str(bjk) '=newyy' int2str(bjk) './norm_fac;']);

    eval(['newLine',int2str(bjk),'(:,:)=[newx'
int2str(bjk),',newy',int2str(bjk),'];']);
end;

for bjk=1:TL
    eval(['meanratio(bjk)=[ratio' int2str(bjk) '];']);
end;
meanratio=mean(meanratio);

figure;plot(newLine1(:,1),newLine1(:,2),'g.');hold
on;plot(newLine2(:,1),newLine2(:,2),'m.');
hold on;plot(newLine3(:,1),newLine3(:,2),'b.');hold
on;plot(newLine4(:,1),newLine4(:,2),'b.');
hold on;plot(newLine5(:,1),newLine5(:,2),'b.');hold
on;plot(newLine6(:,1),newLine6(:,2),'b.');
hold on;plot(newLine7(:,1),newLine7(:,2),'b.');hold
on;plot(newLine8(:,1),newLine8(:,2),'b.');
hold on;plot(newLine9(:,1),newLine9(:,2),'b.');hold
on;plot(newLine10(:,1),newLine10(:,2),'b.');
hold on;plot(newLine11(:,1),newLine11(:,2),'b.');
hold on;plot(newLine12(:,1),newLine12(:,2),'b.');
title('Rotated Tags devided by the Normalization Factor')
```

Matlab Code for Finding the Angle for Rotation Normalization: angles.m

```
%%
% code to calculate the angles between taglines and x-axis
%%

slope=zeros(TL,1); %slopes of every tag line in a single matrix.
for ft=1:TL
    eval(['slope(ft,1)=a' int2str(ft) '(1);']);
end;
m2=0; % x-axis : y=0
for ft=1:TL
    phi(ft,1)=aci(slope(ft),m2);
    ph=phi(ft);
    phi_degree(ft,1)=degree(ph);
end;
aci_sapmasi=mean(phi_degree)
ang_var=mean(phi)
```

Matlab Code to Fit the Normalized Tagpoints into Cubic Spline: pp_n7N_dnm.m

```
%%
% All tags' coefficients are calculated for the 1.time frame
%%
figure;plot(newLine1(:,1),newLine1(:,2),'g.');hold
on;plot(newLine2(:,1),newLine2(:,2),'m.');
hold on;plot(newLine3(:,1),newLine3(:,2),'b.');hold
on;plot(newLine4(:,1),newLine4(:,2),'b.');
hold on;plot(newLine5(:,1),newLine5(:,2),'b.');hold
on;plot(newLine6(:,1),newLine6(:,2),'b.');
hold on;plot(newLine7(:,1),newLine7(:,2),'b.');hold
on;plot(newLine8(:,1),newLine8(:,2),'b.');
hold on;plot(newLine9(:,1),newLine9(:,2),'b.');hold
on;plot(newLine10(:,1),newLine10(:,2),'b.');
hold on;plot(newLine11(:,1),newLine11(:,2),'b.');
hold on;plot(newLine12(:,1),newLine12(:,2),'b.');
title('Rotated Tags devided by the Normalization Factor')

TL=12;%tag line
TF=12;%time frame
pp_n7N_dnm22=zeros(TL,4,1);
%load zcrsmat_normal1.mat
for gs=1%:TF; %gs=number of time frames
    for bjk=1:TL;%bjk= number of tag lines
        eval(['newLine=newLine' int2str(bjk) ';']);
        %figure;plot(nlnewLine1(:,1),nlnewLine1(:,2),'g.');hold
on;plot(nlnewLine2(:,1),nlnewLine2(:,2),'m.');
        %plot(0,0,'ro');xlabel('R');ylabel('S');
        grid on
        title(['Rotated Tag Lines Representation' int2str(bjk)]);
        if bjk==1 & gs==1
            eval(['[a_first,d_first] = polyfit(newLine' int2str(bjk)
'(:,1),newLine' int2str(bjk) '(:,2),1);']);
        elseif bjk==TL & gs==1
            eval(['[a_last,d_last] = polyfit(newLine' int2str(bjk)
'(:,1),newLine' int2str(bjk) '(:,2),1);']);
```

```
            eval(['LN_f=dilekfun(a_first,newLine' int2str(bjk) '(:,1));']);
            eval(['LN_l=dilekfun(a_last,newLine' int2str(bjk) '(:,1));']);
        end
        eval(['LN_x=newLine' int2str(bjk) '(:,1);']);
        eval(['LN_y=newLine' int2str(bjk) '(:,2);']);
        prods=spline(LN_x,LN_y,min(LN_x):.01:max(LN_x));
        [pp,ss]=polyfit(min(LN_x):.01:max(LN_x),prods,3);
        pp_n7N_dnm22(bjk,:,gs)=pp;
        des=0;
        for er=min(LN_x):.01:max(LN_x)
            des=des+1;
            denek(des)=pp(1)*er^3+pp(2)*er^2+pp(3)*er+pp(4);
        end
        x=[min(LN_x):.1:max(LN_x)];
        y=pp(1)*x.^3+pp(2)*x.^2+pp(3)*x+pp(4);
        %figure;plot(x,y,'r.'); title('Coefficients fitted on the array')
        %hold on;plot(newLine(:,1),newLine(:,2),'g.')

        hold on;plot(min(LN_x):.01:max(LN_x),prods,'b-');
        hold on;plot(min(LN_x):.01:max(LN_x),denek,'r-');
        title('Blue:Spline fitted points; Red:Calculated Coefficients on
Curve')

        clear denek prods LN_x LN_y;

        pause
    end;
end;
save pp_n7N_dnm22.mat pp_n7N_dnm22
```

Matlab Code to Segment the Myocardium into Region of Interests: cross.m

```
% axes crossing of the first & last tag lines

Line1(:,:)=[LN_x1,LN_y1];
[m1,c1]=polyfit(LN_x1,LN_y1,1);
Pust=m1(2);

eval(['Line' int2str(TL),'(:,:)=[LN_x'
int2str(TL),',LN_y',int2str(TL),'];']);
eval(['[m2,c2]=polyfit(LN_x',int2str(TL),',LN_y',int2str(TL),',1);']);
Palt=m2(2);
P=[Pust Palt];
Abstand=(Pust+abs(Palt))/3;
norm_fac=Abstand*3;
P1=Pust-Abstand;
P2=P1-Abstand;
figure;plot(Line1(:,1),Line1(:,2),'b.');
hold on;plot(Line2(:,1),Line2(:,2),'b.');
hold on;plot(Line3(:,1),Line3(:,2),'b.');
hold on;plot(Line4(:,1),Line4(:,2),'b.');
hold on;plot(Line5(:,1),Line5(:,2),'b.');
hold on;plot(Line6(:,1),Line6(:,2),'b.');
hold on;plot(Line7(:,1),Line7(:,2),'b.');
hold on;plot(Line8(:,1),Line8(:,2),'b.');
hold on;plot(Line9(:,1),Line9(:,2),'b.');
```

```
hold on;plot(Line10(:,1),Line10(:,2),'b.');
hold on;plot(Line11(:,1),Line11(:,2),'b.');
hold on;plot(Line12(:,1),Line12(:,2),'b.');
hold on;plot(Line13(:,1),Line13(:,2),'b.');
hold on;plot(Line14(:,1),Line14(:,2),'b.');
hold on;plot(Line15(:,1),Line15(:,2),'b.');
hold on;plot(Line16(:,1),Line16(:,2),'b.')

plot(0,0,'ro');xlabel('R');ylabel('S');grid on

hold on;plot(0,P1,'m*')
hold on;plot(0,P2,'m*');
title('Regions to select the Tags')
for bjk=1:TL
    eval(['Li' int2str(bjk) '=Line' int2str(bjk) '/norm_fac;']);
end;
figure;plot(Li1(:,1),Li1(:,2),'b.');hold on;plot(Li2(:,1),Li2(:,2),'b.')
hold on;plot(Li3(:,1),Li3(:,2),'b.');hold
on;plot(Li4(:,1),Li4(:,2),'b.');
hold on;plot(Li5(:,1),Li5(:,2),'b.');hold
on;plot(Li6(:,1),Li6(:,2),'b.');
hold on;plot(Li7(:,1),Li7(:,2),'b.');hold on;plot(Li8(:,1),Li8(:,2),'b.')
hold on;plot(Li9(:,1),Li9(:,2),'b.');hold
on;plot(Li10(:,1),Li10(:,2),'b.')
hold on;plot(Li11(:,1),Li11(:,2),'b.');hold
on;plot(Li12(:,1),Li12(:,2),'b.')
hold on;plot(Li13(:,1),Li13(:,2),'b.');hold
on;plot(Li14(:,1),Li14(:,2),'b.')
hold on;plot(Li15(:,1),Li15(:,2),'b.');hold
on;plot(Li16(:,1),Li16(:,2),'b.');
P1N=(Pust-Abstand)/norm_fac;
P2N=(P1-Abstand)/norm_fac;
hold on;plot(0,P1N,'m*')
hold on;plot(0,P2N,'m*');
title('Rotated Tags devided by the Normalization Factor')
```

Matlab Code to Select Fix Tags from each Region of Interest: selected_tags_n3R.m

```
for bjk=1:TL
    eval(['ortLN_y' int2str(bjk) '=mean(LN_y' int2str(bjk) ');']);
    eval(['ortLineY(bjk,1)=[ortLN_y' int2str(bjk) '];']);
    if ortLineY(bjk)>=P1
        LY_up(bjk,1)=ortLineY(bjk);
        in=size(LY_up,1);
        ind_up=[1:in]';
    elseif ortLineY(bjk)>P2 & ortLineY(bjk)<P1
        LY_mid(bjk,1)=ortLineY(bjk);
        inm=size(LY_mid,1);
        ind_mid=[1:inm]';
    else ortLineY(bjk)<=P2;
        LY_bot(bjk,1)=ortLineY(bjk);
        inb=size(LY_bot,1);
        ind_bot=[1:inb]';

    end
end;
```

```
LineY_up=[ind_up,LY_up];
[Y,up] = min(LineY_up); n3R_up=up(1,2)

LineY_mid=[ind_mid,LY_mid];
n3R_md_min=min(find(LY_mid));
n3R_md_max=max(find(LY_mid))
LineY_bot=[ind_bot,LY_bot];
n3R_bt_max=max(find(LY_bot))
n3R_bt_min=min(find(LY_bot))


save n3R_up.mat n3R_up;save n3R_md_max.mat n3R_md_max;
save n3R_md_min.mat n3R_md_min;
save n3R_bt_max.mat n3R_bt_max;save n3R_bt_min.mat n3R_bt_min;
```

Matlab Code to Load the Calculated Curve Coefficients & Selected Tags: loading.m

```
%% HEALTY(BLUE) vs. UNHEALTHY(PINK) DISPLAY :
%%  uploading the coefficients

% NORMAL DATA
load pp_n1N_dnm22.mat;load pp_n1R_dnm22.mat;
load pp_n2N_dnm22.mat;load pp_n2R_dnm22.mat;
load pp_n3N_dnm22.mat;load pp_n3R_dnm22.mat;
load pp_n4N_dnm22.mat;load pp_n4R_dnm22.mat;
load pp_n7N_dnm22.mat;load pp_n7R_dnm22.mat;


% DCM
load pp_bivN_dnm22.mat;load pp_bivR_dnm22.mat;
load pp_padN_dnm22.mat;load pp_padR_dnm22.mat;
load pp_shiN_dnm22.mat;load pp_shiR_dnm22.mat;
load pp_tay_dnm22.mat;load pp_tayR_dnm22.mat;


% INFARCT
load pp_ab3_dnm22.mat;load pp_ab3R_dnm22.mat;
load pp_ab6_dnm22.mat;load pp_ab6R_dnm22.mat;
load pp_ab11_dnm22.mat;load pp_ab11R_dnm22.mat;
load pp_ab12_dnm22.mat;load pp_ab12R_dnm22.mat;
load pp_ab14_dnm22.mat;load pp_ab14R_dnm22.mat;


pp1=pp_n1N_dnm22;  pp1R=pp_n1R_dnm22;
pp2=pp_n2N_dnm22;  pp2R=pp_n2R_dnm22;
pp3=pp_n3N_dnm22;  pp3R=pp_n3R_dnm22;
pp4=pp_n4N_dnm22;  pp4R=pp_n4R_dnm22;
pp5=pp_n7N_dnm22;  pp5R=pp_n7R_dnm22;
pp6=pp_bivN_dnm22;pp6R=pp_bivR_dnm22;
pp7=pp_padN_dnm22;pp7R=pp_padR_dnm22;
pp8=pp_shiN_dnm22;pp8R=pp_shiR_dnm22;
pp9=pp_tay_dnm22;  pp9R=pp_tayR_dnm22;
pp10=pp_ab3_dnm22;pp10R=pp_ab3R_dnm22;
pp11=pp_ab6_dnm22;pp11R=pp_ab6R_dnm22;
pp12=pp_ab11_dnm22;pp12R=pp_ab11R_dnm22;
pp13=pp_ab12_dnm22;pp13R=pp_ab12R_dnm22;
pp14=pp_ab14_dnm22;pp14R=pp_ab14R_dnm22;
```

```
%% related sections' selected tags for every data set:

% 1A) normal - diastole
load n11N_up1.mat;load n11N_md_min1.mat;load n11N_md_max1.mat;load
n11N_bt_min1.mat;%load n11N_bt_max1.mat;
load n11N_up2.mat;load n11N_md_min2.mat;load n11N_md_max2.mat;load
n11N_bt_min2.mat;%load n11N_bt_max2.mat;
load n21N_up1.mat;load n21N_md_min1.mat;load n21N_md_max1.mat;load
n21N_bt_min1.mat;%load n21N_bt_max1.mat;
load n21N_up2.mat;load n21N_md_min2.mat;load n21N_md_max2.mat;load
n21N_bt_min2.mat;%load n21N_bt_max2.mat;
load n31N_up1.mat;load n31N_md_min1.mat;load n31N_md_max1.mat;load
n31N_bt_min1.mat;%load n31N_bt_max1.mat;
load n31N_up2.mat;load n31N_md_min2.mat;load n31N_md_max2.mat;load
n31N_bt_min2.mat;%load n31N_bt_max2.mat;
load n41N_up1.mat;load n41N_md_min1.mat;load n41N_md_max1.mat;load
n41N_bt_min1.mat;%load n41N_bt_max1.mat;
load n41N_up2.mat;load n41N_md_min2.mat;load n41N_md_max2.mat;load
n41N_bt_min2.mat;%load n41N_bt_max2.mat;
load n71N_up1.mat;load n71N_md_min1.mat;load n71N_md_max1.mat;load
n71N_bt_min1.mat;%load n71N_bt_max1.mat;
load n71N_up2.mat;load n71N_md_min2.mat;load n71N_md_max2.mat;load
n71N_bt_min2.mat;%load n71N_bt_max2.mat


% 1B) normal - systole
load n1RN_up1.mat;load n1RN_md_min1.mat;load n1RN_md_max1.mat;load
n1RN_bt_min1.mat;%load n1RN_bt_max1.mat;
load n1RN_up2.mat;load n1RN_md_min2.mat;load n1RN_md_max2.mat;load
n1RN_bt_min2.mat;%load n1RN_bt_max2.mat;
load n21R_up1.mat;load n21R_md_min1.mat;load n21R_md_max1.mat;load
n21R_bt_min1.mat;%load n21R_bt_max1.mat;
load n21R_up2.mat;load n21R_md_min2.mat;load n21R_md_max2.mat;load
n21R_bt_min2.mat;%load n21R_bt_max2.mat;
load n31R_up1.mat;load n31R_md_min1.mat;load n31R_md_max1.mat;load
n31R_bt_min1.mat;%load n31R_bt_max1.mat;
load n31R_up2.mat;load n31R_md_min2.mat;load n31R_md_max2.mat;load
n31R_bt_min2.mat;%load n31R_bt_max2.mat;
load n4RN_up1.mat;load n4RN_md_min1.mat;load n4RN_md_max1.mat;load
n4RN_bt_min1.mat;%load n4RN_bt_max1.mat;
load n4RN_up2.mat;load n4RN_md_min2.mat;load n4RN_md_max2.mat;load
n4RN_bt_min2.mat;%load n4RN_bt_max2.mat;
load n7RN_up1.mat;load n7RN_md_min1.mat;load n7RN_md_max1.mat;load
n7RN_bt_min1.mat;%load n7RN_bt_max1.mat;
load n7RN_up2.mat;load n7RN_md_min2.mat;load n7RN_md_max2.mat;load
n7RN_bt_min2.mat;%load n7RN_bt_max2.mat;


% 2A) dcm - diastole
load biv1N_up1.mat;load biv1N_bt_min1.mat;load biv1N_md_max1.mat;load
biv1N_md_min1.mat;%load biv1N_bt_max1.mat;
load biv1N_up2.mat;load biv1N_bt_min2.mat;load biv1N_md_max2.mat;load
biv1N_md_min2.mat;%load biv1N_bt_max2.mat;
load pad1N_up1.mat;load pad1N_bt_min1.mat;load pad1N_md_max1.mat;load
pad1N_md_min1.mat;%load pad1N_bt_max1.mat;
load pad1N_up2.mat;load pad1N_bt_min2.mat;load pad1N_md_max2.mat;load
pad1N_md_min2.mat;%load pad1N_bt_max2.mat;
load shi1N_up1.mat;load shi1N_bt_min1.mat;load shi1N_md_max1.mat;load
shi1N_md_min1.mat;%load shi1N_bt_max1.mat;
load shi1N_up2.mat;load shi1N_bt_min2.mat;load shi1N_md_max2.mat;load
shi1N_md_min2.mat;%load shi1N_bt_max2.mat;
```

```
load tay1N_up1.mat;load tay1N_bt_min1.mat;load tay1N_md_max1.mat;load
tay1N_md_min1.mat;%load tay1N_bt_max1.mat;
load tay1N_up2.mat;load tay1N_bt_min2.mat;load tay1N_md_max2.mat;load
tay1N_md_min2.mat;%load tay1N_bt_max2.mat;

% 2B) dcm - systole
load bivRN_up1.mat;load bivRN_bt_min1.mat;load bivRN_md_min1.mat;load
bivRN_md_max1.mat;%load bivRN_bt_max1.mat;;
load bivRN_up2.mat;load bivRN_bt_min2.mat;load bivRN_md_min2.mat;load
bivRN_md_max2.mat;%load bivRN_bt_max2.mat;;
load padRN_up1.mat;load padRN_bt_min1.mat;load padRN_md_min1.mat;load
padRN_md_max1.mat;%load padRN_bt_max1.mat;;
load padRN_up2.mat;load padRN_bt_min2.mat;load padRN_md_min2.mat;load
padRN_md_max2.mat;%load padRN_bt_max2.mat;;
load shiRN_up1.mat;load shiRN_bt_min1.mat;load shiRN_md_min1.mat;load
shiRN_md_max1.mat;%load shiRN_bt_max1.mat;;
load shiRN_up2.mat;load shiRN_bt_min2.mat;load shiRN_md_min2.mat;load
shiRN_md_max2.mat;%load shiRN_bt_max2.mat;;
load tayRN_up1.mat;load tayRN_bt_min1.mat;load tayRN_md_min1.mat;load
tayRN_md_max1.mat;%load tayRN_bt_max1.mat;;
load tayRN_up2.mat;load tayRN_bt_min2.mat;load tayRN_md_min2.mat;load
tayRN_md_max2.mat;%load tayRN_bt_max2.mat;;

% 3A) infarct - diastole
load ab31_up1.mat;load ab31_md_min1.mat;load ab31_md_max1.mat;load
ab31_bt_min1.mat;%load ab31_bt_max1.mat;
load ab31_up2.mat;load ab31_md_min2.mat;load ab31_md_max2.mat;load
ab31_bt_min2.mat;%load ab31_bt_max2.mat;
load ab61_up1.mat;load ab61_md_min1.mat;load ab61_md_max1.mat;load
ab61_bt_min1.mat;%load ab61_bt_max1.mat;
load ab61_up2.mat;load ab61_md_min2.mat;load ab61_md_max2.mat;load
ab61_bt_min2.mat;%load ab61_bt_max2.mat;
load ab111_up1.mat;load ab111_md_min1.mat;load ab111_md_max1.mat;load
ab111_bt_min1.mat;%load ab111_bt_max1.mat;
load ab111_up2.mat;load ab111_md_min2.mat;load ab111_md_max2.mat;load
ab111_bt_min2.mat;%load ab111_bt_max2.mat;
load ab121_up1.mat;load ab121_md_min1.mat;load ab121_md_max1.mat;load
ab121_bt_min1.mat;%load ab121_bt_max1.mat;
load ab121_up2.mat;load ab121_md_min2.mat;load ab121_md_max2.mat;load
ab121_bt_min2.mat;%load ab121_bt_max2.mat;
load ab141_up1.mat;load ab141_md_min1.mat;load ab141_md_max1.mat;load
ab141_bt_min1.mat;%load ab141_bt_max1.mat;
load ab141_up2.mat;load ab141_md_min2.mat;load ab141_md_max2.mat;load
ab141_bt_min2.mat;%load ab141_bt_max2.mat;

% 3B) infarct - systole
load ab3R_up1.mat;load ab3R_md_min1.mat;load ab3R_md_max1.mat;load
ab3R_bt_min1.mat;%load ab3R_bt_max1.mat;
load ab3R_up2.mat;load ab3R_md_min2.mat;load ab3R_md_max2.mat;load
ab3R_bt_min2.mat;%load ab3R_bt_max2.mat;
load ab6R_up1.mat;load ab6R_md_min1.mat;load ab6R_md_max1.mat;load
ab6R_bt_min1.mat;%load ab6R_bt_max1.mat;
load ab6R_up2.mat;load ab6R_md_min2.mat;load ab6R_md_max2.mat;load
ab6R_bt_min2.mat;%load ab6R_bt_max2.mat;
load ab11R_up1.mat;load ab11R_md_min1.mat;load ab11R_md_max1.mat;load
ab11R_bt_min1.mat;%load ab11R_bt_max1.mat;
load ab11R_up2.mat;load ab11R_md_min2.mat;load ab11R_md_max2.mat;load
ab11R_bt_min2.mat;%load ab11R_bt_max2.mat;
load ab12R_up1.mat;load ab12R_md_min1.mat;load ab12R_md_max1.mat;load
ab12R_bt_min1.mat;%load ab12R_bt_max1.mat;
```

```
load ab12R_up2.mat;load ab12R_md_min2.mat;load ab12R_md_max2.mat;load
ab12R_bt_min2.mat;%load ab12R_bt_max2.mat;
load ab14R_up1.mat;load ab14R_md_min1.mat;load ab14R_md_max1.mat;load
ab14R_bt_min1.mat;%load ab14R_bt_max1.mat;
load ab14R_up2.mat;load ab14R_md_min2.mat;load ab14R_md_max2.mat;load
ab14R_bt_min2.mat;%load ab14R_bt_max2.mat;

% analyzed time frame is the first tf : enddiastole
% WRITE & SORT SELECTED TAGS IN A MATRIX

% NORMAL & DIASTOLE
tg_n1N=[n11N_up1,n11N_up2,n11N_md_min1,n11N_md_min2,n11N_md_max1,n11N_md_
max2,n11N_bt_min1,n11N_bt_min2]; %tg:detected tag line
tg_n1N=sort(tg_n1N);
tg_n2N=[n21N_up1,n21N_up2,n21N_md_min1,n21N_md_min2,n21N_md_max1,n21N_md_
max2,n21N_bt_min1,n21N_bt_min2];
tg_n2N=sort(tg_n1N);
tg_n3N=[n31N_up1,n31N_up2,n31N_md_min1,n31N_md_min2,n31N_md_max1,n31N_md_
max2,n31N_bt_min1,n31N_bt_min2];
tg_n3N=sort(tg_n3N);
tg_n4N=[n41N_up1,n41N_up2,n41N_md_min1,n41N_md_min2,n41N_md_max1,n41N_md_
max2,n41N_bt_min1,n41N_bt_min2];
tg_n4N=sort(tg_n4N);
tg_n7N=[n71N_up1,n71N_up2,n71N_md_min1,n71N_md_min2,n71N_md_max1,n71N_md_
max2,n71N_bt_min1,n71N_bt_min2];
tg_n7N=sort(tg_n7N);

% NORMAL & SYSTOLE
tg_n1R=[n1RN_up1,n1RN_up2,n1RN_md_min1,n1RN_md_min2,n1RN_md_max1,n1RN_md_
max2,n1RN_bt_min1,n1RN_bt_min2];
tg_n1R=sort(tg_n1R);
tg_n2R=[n21R_up1,n21R_up2,n21R_md_min1,n21R_md_min2,n21R_md_max1,n21R_md_
max2,n21R_bt_min1,n21R_bt_min2];
tg_n2R=sort(tg_n1R);
tg_n3R=[n31R_up1,n31R_up2,n31R_md_min1,n31R_md_min2,n31R_md_max1,n31R_md_
max2,n31R_bt_min1,n31R_bt_min2];
tg_n3R=sort(tg_n3R);
tg_n4R=[n4RN_up1,n4RN_up2,n4RN_md_min1,n4RN_md_min2,n4RN_md_max1,n4RN_md_
max2,n4RN_bt_min1,n4RN_bt_min2];
tg_n4R=sort(tg_n4R);tg_n4R=tg_n4R;
tg_n7R=[n7RN_up1,n7RN_up2,n7RN_md_min1,n7RN_md_min2,n7RN_md_max1,n7RN_md_
max2,n7RN_bt_min1,n7RN_bt_min2];
tg_n7R=sort(tg_n7R);

% DCM & DIASTOLE
tg_biv1N=[biv1N_up1,biv1N_up2,biv1N_bt_min1,biv1N_bt_min2,biv1N_md_max1,
biv1N_md_max2,biv1N_md_min1,biv1N_md_min2];
tg_biv1N=sort(tg_biv1N);
tg_pad1N=[pad1N_up1,pad1N_up2,pad1N_bt_min1,pad1N_bt_min2,pad1N_md_max1,
pad1N_md_max2,pad1N_md_min1,pad1N_md_min2];
tg_pad1N=sort(tg_pad1N);
tg_shi1N=[shi1N_up1,shi1N_up2,shi1N_bt_min1,shi1N_bt_min2,shi1N_md_max1,
shi1N_md_max2,shi1N_md_min1,shi1N_md_min2];
tg_shi1N=sort(tg_shi1N);
tg_tay1N=[tay1N_up1,tay1N_up2,tay1N_bt_min1,tay1N_bt_min2,tay1N_md_max1,
tay1N_md_max2,tay1N_md_min1,tay1N_md_min2];
tg_tay1N=sort(tg_tay1N);
```

```
%DCM & SYSTOLE
tg_bivR=[bivRN_up1,bivRN_up2,bivRN_bt_min1,bivRN_bt_min2,bivRN_md_max1,
bivRN_md_max2,bivRN_md_min1,bivRN_md_min2];
tg_padR=[padRN_up1,padRN_up2,padRN_bt_min1,padRN_bt_min2,padRN_md_max1,
padRN_md_max2,padRN_md_min1,padRN_md_min2];
tg_padR=sort(tg_padR);
tg_shiR=[shiRN_up1,shiRN_up2,shiRN_bt_min1,shiRN_bt_min2,shiRN_md_max1,
shiRN_md_max2,shiRN_md_min1,shiRN_md_min2];
tg_shiR=sort(tg_shiR);
tg_tayR=[tayRN_up1,tayRN_up2,tayRN_bt_min1,tayRN_bt_min2,tayRN_md_max1,
tayRN_md_max2,tayRN_md_min1,tayRN_md_min2];
tg_tayR=sort(tg_tayR);

% INFARCT & DIASTOLE
tg_ab3N=[ab31_up1,ab31_md_min1,ab31_md_max1,ab31_bt_min1,ab31_up2,ab31_md
_min2,ab31_md_max2,ab31_bt_min2];
tg_ab3N=sort(tg_ab3N);
tg_ab6N=[ab61_up1,ab61_md_min1,ab61_md_min2,ab61_md_max1,ab61_bt_min1,
ab61_up2,ab61_md_max2,ab61_bt_min2];
tg_ab6N=sort(tg_ab6N);
tg_ab11N=[ab111_up1,ab111_md_min1,ab111_md_max1,ab111_bt_min1,ab111_up2,
ab111_md_min2,ab111_md_max2,ab111_bt_min2];
tg_ab11N=sort(tg_ab11N);
tg_ab12N=[ab121_up1,ab121_md_min1,ab121_md_max1,ab121_bt_min1,ab121_up2,
ab121_md_min2,ab121_md_max2,ab121_bt_min2];
tg_ab12N=sort(tg_ab12N);
tg_ab14N=[ab141_up1,ab141_md_min1,ab141_md_max1,ab141_bt_min1,ab141_up2,
ab141_md_min2,ab141_md_max2,ab141_bt_min2];
tg_ab14N=sort(tg_ab14N);

% INFARCT & SYSTOLE
tg_ab3R=[ab3R_up1,ab3R_md_min1,ab3R_md_max1,ab3R_bt_min1,ab3R_up2,ab3R_md
_min2,ab3R_md_max2,ab3R_bt_min2];
tg_ab3R=sort(tg_ab3R);
tg_ab6R=[ab6R_up1,ab6R_md_min1,ab6R_md_max1,ab6R_bt_min1,ab6R_up2,ab6R_md
_min2,ab6R_md_max2,ab6R_bt_min2];
tg_ab6R=sort(tg_ab6R);
tg_ab11R=[ab11R_up1,ab11R_md_min1,ab11R_md_max1,ab11R_bt_min1,ab11R_up2,
ab11R_md_min2,ab11R_md_max2,ab11R_bt_min2];
tg_ab11R=sort(tg_ab11R);
tg_ab12R=[ab12R_up1,ab12R_md_min1,ab12R_md_max1,ab12R_bt_min1,ab12R_up2,
ab12R_md_min2,ab12R_md_max2,ab12R_bt_min2];
tg_ab12R=sort(tg_ab12R);
tg_ab14R=[ab14R_up1,ab14R_md_min1,ab14R_md_max1,ab14R_bt_min1,ab14R_up2,
ab14R_md_min2,ab14R_md_max2,ab14R_bt_min2];
tg_ab14R=sort(tg_ab14R);
sz_tg=size(tg_ab14R,2);
```

Matlab Code to Represent the Calculated Curve Coefficients: pink.m

```
loading
for cf=1:4;
figure;hold on;plot(pp1(tg_n1N,cf),'b.-');hold
on;plot(pp2(tg_n2N,cf),'b.-');
hold on;plot(pp3(tg_n3N,cf),'b.-');hold on;plot(pp4(tg_n4N,cf),'b.-');
hold on;plot(pp5(tg_n7N,cf),'b.-');
hold on;plot(pp6(tg_biv1N,cf),'m.-');plot(pp7(tg_pad1N,cf),'m.-')
hold on;plot(pp8(tg_shi1N,cf),'m.-');hold on;plot(pp9(tg_tay1N,cf),'m.-')
hold on;plot(pp10(tg_ab3N,cf),'g.-');hold on;plot(pp11(tg_ab6N,cf),'g.-')
hold on;plot(pp12(tg_ab11N,cf),'g.-');hold on;plot(pp13(tg_ab12N,cf),'g.-
');
hold on;plot(pp14(tg_ab14N,cf),'g.-');axis([0 sz_tg -0.6 0.8]);

ylabel([int2str(cf) '. Coefficients Value']);
xlabel('BLUE:HEALTHY  PINK:DCM  GREEN:INFRACT');
title('DISTRIBUTION WHILE DIASTOLE');
% ylabel('First Coefficients (A) Values');
% xlabel('BLUE:HEALTHY  PINK:DCM  GREEN:INFRACT');

figure;plot(pp1R(tg_n1R,cf),'b.-');hold on;plot(pp2R(tg_n2R,cf),'b.-');
hold on;plot(pp3R(tg_n3R,cf),'b.-');hold on;plot(pp4R(tg_n4R,cf),'b.-');
hold on;plot(pp5R(tg_n7R,cf),'b.-');
hold on;plot(pp6R(tg_bivR,cf),'m.-');plot(pp7R(tg_padR,cf),'m.-')
hold on;plot(pp8R(tg_shiR,cf),'m.-');hold on;plot(pp9R(tg_tayR,cf),'m.-')
hold on;plot(pp10R(tg_ab3R,cf),'g.-');
hold on;plot(pp11R(tg_ab6R,cf),'g.-');
hold on;plot(pp12R(tg_ab11R,cf),'g.-');
hold on;plot(pp13R(tg_ab12R,cf),'g.-');
hold on;plot(pp14R(tg_ab14R,cf),'g.-');axis([0 sz_tg -0.6 0.8]);
ylabel([int2str(cf) '. Coefficients Value']);
xlabel('BLUE:HEALTHY  PINK:DCM  GREEN:INFARCT');
title('DISTRIBUTION WHILE SYSTOLE')

figure;
subplot(1,2,1);
plot(pp1(tg_n1N,cf),'b.-');hold on;plot(pp2(tg_n2N,cf),'b.-');
hold on;plot(pp3(tg_n3N,cf),'b.-');hold on;plot(pp4(tg_n4N,cf),'b.-');
hold on;plot(pp5(tg_n7N,cf),'b.-');
hold on;plot(pp6(tg_biv1N,cf),'m.-');plot(pp7(tg_pad1N,cf),'m.-')
hold on;plot(pp8(tg_shi1N,cf),'m.-');hold on;plot(pp9(tg_tay1N,cf),'m.-')
hold on;plot(pp10(tg_ab3N,cf),'g.-');hold on;plot(pp11(tg_ab6N,cf),'g.-')
hold on;plot(pp12(tg_ab11N,cf),'g.-');
hold on;plot(pp13(tg_ab12N,cf),'g.-');
hold on;plot(pp14(tg_ab14N,cf),'g.-');axis([0 sz_tg -0.6 0.8]);
ylabel([int2str(cf) '. Coefficients Value']);
xlabel('    BLUE:HEALTHY  PINK:DCM  GREEN:INFARCT');
title('DISTRIBUTION WHILE DIASTOLE')

subplot(1,2,2);plot(pp1R(tg_n1R,cf),'b.-');hold
on;plot(pp2R(tg_n2R,cf),'b.-');
hold on;plot(pp3R(tg_n3R,cf),'b.-');hold on;plot(pp4R(tg_n4R,cf),'b.-');
hold on;plot(pp5R(tg_n7R,cf),'b.-');
hold on;plot(pp6R(tg_bivR,cf),'m.-');plot(pp7R(tg_padR,cf),'m.-')
hold on;plot(pp8R(tg_shiR,cf),'m.-');hold on;plot(pp9R(tg_tayR,cf),'m.-')
hold on;plot(pp10R(tg_ab3R,cf),'g.-');
hold on;plot(pp11R(tg_ab6R,cf),'g.-');
hold on;plot(pp12R(tg_ab11R,cf),'g.-');
```

```
hold on;plot(pp13R(tg_ab12R,cf),'g.-');
hold on;plot(pp14R(tg_ab14R,cf),'g.-');axis([0 sz_tg -0.6 0.8]);
%ylabel('First Coefficients (A) Values');xlabel('BLUE:HEALTHY  PINK:DCM
GREEN:INFARCT');
title('DISTRIBUTION WHILE SYSTOLE')


%%
%% NORMAL at systole & diastole
%%
figure;hold on;plot(pp1(tg_n1N,cf),'k.-');hold
on;plot(pp2(tg_n2N,cf),'k.-');
hold on;plot(pp3(tg_n3N,cf),'k.-');hold on;plot(pp4(tg_n4N,cf),'k.-');
hold on;plot(pp5(tg_n7N,cf),'k.-');
hold on;plot(pp1R(tg_n1R,cf),'r.-');hold on;plot(pp2R(tg_n2R,cf),'r.-');
hold on;plot(pp3R(tg_n3R,cf),'r.-');hold on;plot(pp4R(tg_n4R,cf),'r.-');
hold on;plot(pp5R(tg_n7R,cf),'r.-')
ylabel([int2str(cf) '. Coefficients Value']);
xlabel('BLACK:HEALTHY DIASTOLE    RED:HEALTHY SYSTOLE');
title('CHANGES WITHIN HEALTY (A)s')


figure;
subplot(2,3,1);
plot(pp1(tg_n1N,cf),'k.-');hold on;plot(pp1R(tg_n1R,cf),'r.-');

subplot(2,3,2);
plot(pp2(tg_n2N,cf),'k.-');hold on;plot(pp2R(tg_n2R,cf),'r.-');
title('CHANGES WITHIN HEALTY DATA')

subplot(2,3,3);
hold on;plot(pp3(tg_n3N,cf),'k.-'); hold on;plot(pp3R(tg_n3R,cf),'r.-');

subplot(2,3,4);plot(pp4(tg_n4N,cf),'k.-');hold
on;plot(pp4R(tg_n4R,cf),'r.-');
ylabel([int2str(cf) '. Coefficients Value']);

subplot(2,3,5);
hold on;plot(pp5(tg_n7N,cf),'k.-');hold on;plot(pp5R(tg_n7R,cf),'r.-')
xlabel('BLACK:HEALTHY DIASTOLE    RED:HEALTHY SYSTOLE');

subplot(2,3,6);
plot(pp1(tg_n1N,cf),'k.-');hold on;plot(pp2(tg_n2N,cf),'k.-');
hold on;plot(pp3(tg_n3N,cf),'k.-');hold on;plot(pp4(tg_n4N,cf),'k.-');
hold on;plot(pp5(tg_n7N,cf),'k.-');
hold on;plot(pp1R(tg_n1R,cf),'r.-');hold on;plot(pp2R(tg_n2R,cf),'r.-');
hold on;plot(pp3R(tg_n3R,cf),'r.-');hold on;plot(pp4R(tg_n4R,cf),'r.-');
hold on;plot(pp5R(tg_n7R,cf),'r.-');


%%
%% DCM at systole & diastole
%%
figure;hold on;plot(pp6(tg_biv1N,cf),'k<-');plot(pp7(tg_pad1N,cf),'k<-');
hold on;plot(pp8(tg_shi1N,cf),'k<-');
hold on;plot(pp9(tg_tay1N,cf),'k<-');
hold on;plot(pp6R(tg_bivR,cf),'r>-');
hold on;plot(pp7R(tg_padR,cf),'r>-');
hold on;plot(pp8R(tg_shiR,cf),'r>-');hold on;plot(pp9R(tg_tayR,cf),'r>-')
ylabel([int2str(cf) '. Coefficients Value']);
xlabel('BLACK:DCM DIASTOLE   RED:DCM SYSTOLE');
```

```
title('CHANGES WITHIN DCM DATA')

figure;subplot(2,3,1);
plot(pp6(tg_biv1N,cf),'k.-');hold on;plot(pp6R(tg_bivR,cf),'r.-');
subplot(2,3,2);
plot(pp7(tg_pad1N,cf),'k.-');hold on;plot(pp7R(tg_padR,cf),'r.-');
title('CHANGES WITHIN DCM COEFFICIENTS')
subplot(2,3,3);
plot(pp8(tg_shi1N,cf),'k.-');hold on;plot(pp8R(tg_shiR,cf),'r.-');
subplot(2,3,4);
plot(pp9(tg_tay1N,cf),'k.-');hold on;plot(pp9R(tg_tayR,cf),'r.-');
subplot(2,3,5);
plot(pp6(tg_biv1N,cf),'k.-');plot(pp7(tg_pad1N,cf),'k.-');
hold on;plot(pp8(tg_shi1N,cf),'k.-');hold on;plot(pp9(tg_tay1N,cf),'k.-')
hold on;plot(pp6R(tg_bivR,cf),'r.-');hold on;plot(pp7R(tg_padR,cf),'r.-')
hold on;plot(pp8R(tg_shiR,cf),'r.-');hold on;plot(pp9R(tg_tayR,cf),'r.-')
xlabel('BLACK:DCM DIASTOLE   RED:DCM SYSTOLE');

%%
%% INFARCT at systole & diastole
%%
figure;plot(pp10(tg_ab3N,cf),'kx-');hold on;plot(pp11(tg_ab6N,cf),'kx-')
hold on;plot(pp12(tg_ab11N,cf),'kx-');
hold on;plot(pp13(tg_ab12N,cf),'kx-');
hold on;plot(pp14(tg_ab14N,cf),'kx-');
hold on;plot(pp10R(tg_ab3R,cf),'rx-');
hold on;plot(pp11R(tg_ab6R,cf),'rx-');
hold on;plot(pp12R(tg_ab11R,cf),'rx-');
hold on;plot(pp13R(tg_ab12R,cf),'rx-');
hold on;plot(pp14R(tg_ab14R,cf),'rx-');

ylabel([int2str(cf) '. Coefficients Value']);
xlabel('BLACK:INFARCT DIASTOLE   RED:INFARCT SYSTOLE');
title('CHANGES WITHIN INFARCT (A)s')

subplot(2,3,1);
plot(pp10(tg_ab3N,cf),'kx-');hold on;plot(pp10R(tg_ab3R,cf),'rx-');
subplot(2,3,2);
plot(pp11(tg_ab3N,cf),'kx-');hold on;plot(pp11R(tg_ab3R,cf),'rx-');
title('CHANGES WITHIN INFARCT DATA')
subplot(2,3,3);
plot(pp12(tg_ab3N,cf),'kx-');hold on;plot(pp12R(tg_ab3R,cf),'rx-');
subplot(2,3,4);
plot(pp13(tg_ab3N,cf),'kx-');hold on;plot(pp13R(tg_ab3R,cf),'rx-');
subplot(2,3,5);
plot(pp14(tg_ab3N,cf),'kx-');hold on;plot(pp14R(tg_ab3R,cf),'rx-');
xlabel('BLACK:INFARCT DIASTOLE   RED:INFARCT SYSTOLE');
subplot(2,3,6);
plot(pp10(tg_ab3N,cf),'k.-');hold on;plot(pp11(tg_ab6N,cf),'k.-')
hold on;plot(pp12(tg_ab11N,cf),'k.-');
hold on;plot(pp13(tg_ab12N,cf),'k.-');
hold on;plot(pp14(tg_ab14N,cf),'k.-');
hold on;plot(pp10R(tg_ab3R,cf),'r.-');
hold on;plot(pp11R(tg_ab6R,cf),'r.-');
hold on;plot(pp12R(tg_ab11R,cf),'r.-');
hold on;plot(pp13R(tg_ab12R,cf),'r.-');
hold on;plot(pp14R(tg_ab14R,cf),'r.-');
pause
close all
end;
```

Matlab Code for Calculating Standard Deviation & Mean Values: st_dv.m

```
loading
for cf=1:4
    sigmaN1(:,cf)=std(pp1(tg_n1N,cf),1);
    sigmaN2(:,cf)=std(pp2(tg_n2N,cf),1);
    sigmaN3(:,cf)=std(pp3(tg_n3N,cf),1);
        sigmaN4(:,cf)=std(pp4(tg_n4N,cf),1);%test
        sigmaN5(:,cf)=std(pp5(tg_n7N,cf),1);%test
    sigmadcm1(:,cf)=std(pp6(tg_biv1N,cf),1);
    sigmadcm2(:,cf)=std(pp7(tg_pad1N,cf),1);
        sigmadcm3(:,cf)=std(pp8(tg_shi1N,cf),1);%test
        sigmadcm4(:,cf)=std(pp9(tg_tay1N,cf),1);%test
    sigmaI1(:,cf)=std(pp10(tg_ab3N,cf),1);
    sigmaI2(:,cf)=std(pp11(tg_ab6N,cf),1);
        sigmaI3(:,cf)=std(pp12(tg_ab11N,cf),1);%test
        sigmaI4(:,cf)=std(pp13(tg_ab12N,cf),1);%test
        sigmaI5(:,cf)=std(pp14(tg_ab14N,cf),1);%test
end;

sgmNN=[sigmaN1;sigmaN2;sigmaN3];
sgmN=orta(sgmNN) % Normal Training Data st_dev. Of (A,B,C,D)!!!

sgmDD=[sigmadcm1;sigmadcm2];
sgmDCM=orta(sgmDD)

sgmII=[sigmaI1;sigmaI2];
sgmINF=orta(sgmII)

sgmDis=[sgmDCM;sgmINF];
sgmD=orta(sgmDis) %sigma diseased!!!

clear sgmNN sgmDD sgmII sgmN sgmDCM sgmINF


for cf=1:4
    meanN(:,cf)=mean([pp1(tg_n1N,cf),pp2(tg_n2N,cf),pp3(tg_n3N,cf)],2);
    meanDCM(:,cf)=mean([pp6(tg_biv1N,cf),pp7(tg_pad1N,cf)],2);
    meanINF(:,cf)=mean([pp10(tg_ab3N,cf),pp11(tg_ab6N,cf)],2);
    meanD(:,cf)=mean([meanDCM(:,cf),meanINF(:,cf)],2);
end;

for cf=1:4
    cik=pdf_dg(pp8(tg_shi1N,cf),meanN(:,cf))
end;

for cf=1:4
    meanN1(:,cf)=mean(pp1(tg_n1N,cf),1);
    meanN2(:,cf)=mean(pp2(tg_n2N,cf),1);
    meanN3(:,cf)=mean(pp3(tg_n3N,cf),1);
    meandcm1(:,cf)=mean(pp6(tg_biv1N,cf),1);
    meandcm2(:,cf)=mean(pp7(tg_pad1N,cf),1);
    meanI1(:,cf)=mean(pp10(tg_ab3N,cf),1);
    meanI2(:,cf)=mean(pp11(tg_ab6N,cf),1);

end;
```

Matlab Code to Compare & Identify if the Coefficients Diseased or Not: final.m

```
loading
% STD FOR TRAIN_SET FOR EVERY SELECTED TAGLINE ( "8" PER EACH TIMEFRAME)
for cf=1:4
    st_devN(cf,:)=std([pp1(tg_n1N,cf),pp2(tg_n2N,cf),pp3(tg_n3N,cf)]');
    st_devDCM(cf,:)=std([pp6(tg_biv1N,cf),pp7(tg_pad1N,cf)]');
    st_devINF(cf,:)=std([pp10(tg_ab3N,cf),pp11(tg_ab6N,cf)]');


st_devN_R(cf,:)=std([pp1R(tg_n1R,cf),pp2R(tg_n2R,cf),pp3R(tg_n3R,cf)]');
    st_devDCM_R(cf,:)=std([pp6R(tg_bivR,cf),pp7R(tg_padR,cf)]');
    st_devINF_R(cf,:)=std([pp10R(tg_ab3R,cf),pp11R(tg_ab6R,cf)]');
end;

st_devN=st_devN';st_devDCM=st_devDCM';st_devINF=st_devINF';
st_devN_r=st_devN_R';st_devDCM_R=st_devDCM_R';st_devINF_R=st_devINF_R';

for cf=1:4
    st_devD(:,cf)=mean([st_devDCM(:,cf),st_devINF(:,cf)],2);
    st_devD_R(:,cf)=mean([st_devDCM_R(:,cf),st_devINF_R(:,cf)],2);
end;

% STD FOR TRAIN_SET FOR EACH COEFFICIENT (A,B,C,D)
for cf=1:4
    sigmaN1(:,cf)=std(pp1(tg_n1N,cf),1);
    sigmaN2(:,cf)=std(pp2(tg_n2N,cf),1);
    sigmaN3(:,cf)=std(pp3(tg_n3N,cf),1);
    sigmaN4(:,cf)=std(pp4(tg_n4N,cf),1);%test
    sigmaN5(:,cf)=std(pp5(tg_n7N,cf),1);%test
    sigmadcm1(:,cf)=std(pp6(tg_biv1N,cf),1);
    sigmadcm2(:,cf)=std(pp7(tg_pad1N,cf),1);
    sigmadcm3(:,cf)=std(pp8(tg_shi1N,cf),1);%test
    sigmadcm4(:,cf)=std(pp9(tg_tay1N,cf),1);%test
    sigmaI1(:,cf)=std(pp10(tg_ab3N,cf),1);
    sigmaI2(:,cf)=std(pp11(tg_ab6N,cf),1);
    sigmaI3(:,cf)=std(pp12(tg_ab11N,cf),1);%test
    sigmaI4(:,cf)=std(pp13(tg_ab12N,cf),1);%test
    sigmaI5(:,cf)=std(pp14(tg_ab14N,cf),1);%test

    sigmaN1R(:,cf)=std(pp1R(tg_n1R,cf),1);
    sigmaN2R(:,cf)=std(pp2R(tg_n2R,cf),1);
    sigmaN3R(:,cf)=std(pp3R(tg_n3R,cf),1);
    sigmaN4R(:,cf)=std(pp4R(tg_n4R,cf),1);%test
    sigmaN5R(:,cf)=std(pp5R(tg_n7R,cf),1);%test
    sigmadcm1R(:,cf)=std(pp6R(tg_bivR,cf),1);
    sigmadcm2R(:,cf)=std(pp7R(tg_padR,cf),1);
    sigmadcm3R(:,cf)=std(pp8R(tg_shiR,cf),1);%test
    sigmadcm4R(:,cf)=std(pp9R(tg_tayR,cf),1);%test
    sigmaI1R(:,cf)=std(pp10R(tg_ab3R,cf),1);
    sigmaI2R(:,cf)=std(pp11R(tg_ab6R,cf),1);
    sigmaI3R(:,cf)=std(pp12R(tg_ab11R,cf),1);%test
    sigmaI4R(:,cf)=std(pp13R(tg_ab12R,cf),1);%test
    sigmaI5R(:,cf)=std(pp14R(tg_ab14R,cf),1);%test
end;
```

```
sgmNN=[sigmaN1;sigmaN2;sigmaN3];
sgmN=orta(sgmNN) % Normal Train. Patients' sigmaA,B,C,D values

sgmDD=[sigmadcm1;sigmadcm2];
sgmDCM=orta(sgmDD)

sgmII=[sigmaI1;sigmaI2];
sgmINF=orta(sgmII)

sgmDis=[sgmDCM;sgmINF];
sgmD=orta(sgmDis) %sigma diseased.


sgmNNR=[sigmaN1R;sigmaN2R;sigmaN3R];
sgmN_R=orta(sgmNNR)
sgmDDR=[sigmadcm1R;sigmadcm2R];
sgmDCM_R=orta(sgmDDR)
sgmIIR=[sigmaI1R;sigmaI2R];
sgmINF_R=orta(sgmIIR)
sgmDisR=[sgmDCM_R;sgmINF_R];
sgmD_R=orta(sgmDisR)

clear sgmNN sgmNNR sgmDD sgmDDR sgmII sgmIIR sgmN sgmDCM sgmINF sgmDisR
sgmDis

for cf=1:4
    meanN(:,cf)=mean([pp1(tg_n1N,cf),pp2(tg_n2N,cf),pp3(tg_n3N,cf)],2);
    meanDCM(:,cf)=mean([pp6(tg_biv1N,cf),pp7(tg_pad1N,cf)],2);
    meanINF(:,cf)=mean([pp10(tg_ab3N,cf),pp11(tg_ab6N,cf)],2);
    meanD(:,cf)=mean([meanDCM(:,cf),meanINF(:,cf)],2);


meanN_R(:,cf)=mean([pp1R(tg_n1R,cf),pp2R(tg_n2R,cf),pp3R(tg_n3R,cf)],2);
    meanDCM_R(:,cf)=mean([pp6R(tg_bivR,cf),pp7R(tg_padR,cf)],2);
    meanINF_R(:,cf)=mean([pp10R(tg_ab3R,cf),pp11R(tg_ab6R,cf)],2);
    meanD_R(:,cf)=mean([meanDCM_R(:,cf),meanINF_R(:,cf)],2);
end;

for cf=1:4
    test1N=pdf_dg(pp4(tg_n4N,cf),meanN(:,cf));
    test1D=pdf_dg(pp4(tg_n4N,cf),meanD(:,cf));

    test2N=pdf_dg(pp5(tg_n7N,cf),meanN(:,cf));
    test2D=pdf_dg(pp5(tg_n7N,cf),meanD(:,cf));

    test3N=pdf_dg(pp8(tg_shi1N,cf),meanN(:,cf));
    test3D=pdf_dg(pp8(tg_shi1N,cf),meanD(:,cf));

    test4N=pdf_dg(pp9(tg_tay1N,cf),meanN(:,cf));
    test4D=pdf_dg(pp9(tg_tay1N,cf),meanD(:,cf));

    test5N=pdf_dg(pp12(tg_ab11N,cf),meanN(:,cf));
    test5D=pdf_dg(pp12(tg_ab11N,cf),meanD(:,cf));

    test6N=pdf_dg(pp13(tg_ab12N,cf),meanN(:,cf));
    test6D=pdf_dg(pp13(tg_ab12N,cf),meanD(:,cf));

    test7N=pdf_dg(pp14(tg_ab14N,cf),meanN(:,cf));
    test7D=pdf_dg(pp14(tg_ab14N,cf),meanD(:,cf));
    %
```

```
        test1DCM=pdf_dg(pp4(tg_n4N,cf),meanDCM(:,cf));
        test2DCM=pdf_dg(pp5(tg_n7N,cf),meanDCM(:,cf));
        test3DCM=pdf_dg(pp8(tg_shi1N,cf),meanDCM(:,cf));
        test4DCM=pdf_dg(pp9(tg_tay1N,cf),meanDCM(:,cf));
        test5DCM=pdf_dg(pp12(tg_ab11N,cf),meanDCM(:,cf));
        test6DCM=pdf_dg(pp13(tg_ab12N,cf),meanDCM(:,cf));
        test7DCM=pdf_dg(pp14(tg_ab14N,cf),meanDCM(:,cf));
        %
        test1DCMR=pdf_dg(pp4R(tg_n4R,cf),meanDCM_R(:,cf));
        test2DCMR=pdf_dg(pp5R(tg_n7R,cf),meanDCM_R(:,cf));
        test3DCMR=pdf_dg(pp8R(tg_shiR,cf),meanDCM_R(:,cf));
        test4DCMR=pdf_dg(pp9R(tg_tayR,cf),meanDCM_R(:,cf));
        test5DCMR=pdf_dg(pp12R(tg_ab11R,cf),meanDCM_R(:,cf));
        test6DCMR=pdf_dg(pp13R(tg_ab12R,cf),meanDCM_R(:,cf));
        test7DCMR=pdf_dg(pp14R(tg_ab14R,cf),meanDCM_R(:,cf));
        %
        test1NR=pdf_dg(pp4R(tg_n4R,cf),meanN_R(:,cf));
        test1DR=pdf_dg(pp4R(tg_n4R,cf),meanD_R(:,cf));
        test2NR=pdf_dg(pp5R(tg_n7R,cf),meanN_R(:,cf));
        test2DR=pdf_dg(pp5R(tg_n7R,cf),meanD_R(:,cf));
        test3NR=pdf_dg(pp8R(tg_shiR,cf),meanN_R(:,cf));
        test3DR=pdf_dg(pp8R(tg_shiR,cf),meanD_R(:,cf));
        test4NR=pdf_dg(pp9R(tg_tayR,cf),meanN_R(:,cf));
        test4DR=pdf_dg(pp9R(tg_tayR,cf),meanD_R(:,cf));
        test5NR=pdf_dg(pp12R(tg_ab11R,cf),meanN_R(:,cf));
        test5DR=pdf_dg(pp12R(tg_ab11R,cf),meanD_R(:,cf));
        test6NR=pdf_dg(pp13R(tg_ab12R,cf),meanN_R(:,cf));
        test6DR=pdf_dg(pp13R(tg_ab12R,cf),meanD_R(:,cf));
        test7NR=pdf_dg(pp14R(tg_ab14R,cf),meanN_R(:,cf));
        test7DR=pdf_dg(pp14R(tg_ab14R,cf),meanD_R(:,cf));
        %
        test1INF=pdf_dg(pp4(tg_n4N,cf),meanINF(:,cf));
        test2INF=pdf_dg(pp5(tg_n7N,cf),meanINF(:,cf));
        test3INF=pdf_dg(pp8(tg_shi1N,cf),meanINF(:,cf));
        test4INF=pdf_dg(pp9(tg_tay1N,cf),meanINF(:,cf));
        test5INF=pdf_dg(pp12(tg_ab11N,cf),meanINF(:,cf));
        test6INF=pdf_dg(pp13(tg_ab12N,cf),meanINF(:,cf));
        test7INF=pdf_dg(pp14(tg_ab14N,cf),meanINF(:,cf));
        %
        test1INFR=pdf_dg(pp4R(tg_n4R,cf),meanINF_R(:,cf));
        test2INFR=pdf_dg(pp5R(tg_n7R,cf),meanINF_R(:,cf));
        test3INFR=pdf_dg(pp8R(tg_shiR,cf),meanINF_R(:,cf));
        test4INFR=pdf_dg(pp9R(tg_tayR,cf),meanINF_R(:,cf));
        test5INFR=pdf_dg(pp12R(tg_ab11R,cf),meanINF_R(:,cf));
        test6INFR=pdf_dg(pp13R(tg_ab12R,cf),meanINF_R(:,cf));
        test7INFR=pdf_dg(pp14R(tg_ab14R,cf),meanINF_R(:,cf));

end;

for ff=1:7
    if ff==3 | ff==4,
        eval(['if abs(test' int2str(ff) 'N)<abs(test' int2str(ff)
'DCM),Test' int2str(ff) '_NORMAL=test' int2str(ff) 'N, elseif abs(test'
int2str(ff) 'N)>abs(test' int2str(ff) 'DCM),Test' int2str(ff) '_DCM=test'
int2str(ff) 'DCM,end'])
        eval(['if abs(test' int2str(ff) 'NR)<abs(test' int2str(ff)
'DCMR),Test' int2str(ff) 'R_NORMAL=test' int2str(ff) 'NR, elseif
abs(test' int2str(ff) 'NR)>abs(test' int2str(ff) 'DCMR),TestR'
int2str(ff) '_DCM=test' int2str(ff) 'DCMR,end'])
    elseif ff==5 | ff==6 | ff==7,
```

```
    eval(['if abs(test' int2str(ff) 'N)<abs(test' int2str(ff) 'INF),Test'
int2str(ff) '_NORMAL=test' int2str(ff) 'N, elseif abs(test' int2str(ff)
'N)>abs(test' int2str(ff) 'INF),Test' int2str(ff) '_INFARCTED=test'
int2str(ff) 'INF,end'])
    eval(['if abs(test' int2str(ff) 'NR)<abs(test' int2str(ff)
'INFR),Test' int2str(ff) 'R_NORMAL=test' int2str(ff) 'NR, elseif
abs(test' int2str(ff) 'NR)>abs(test' int2str(ff) 'INFR),TestR'
int2str(ff) '_INFARCTED=test' int2str(ff) 'INFR,end'])
else
    eval(['if abs(test' int2str(ff) 'N)<abs(test' int2str(ff) 'D),Test'
int2str(ff) '_NORMAL=test' int2str(ff) 'N, elseif abs(test' int2str(ff)
'N)>abs(test' int2str(ff) 'D),Test' int2str(ff) '_DISEASED=test'
int2str(ff) 'D,end'])
    eval(['if abs(test' int2str(ff) 'NR)<abs(test' int2str(ff) 'DR),Test'
int2str(ff) 'R_NORMAL=test' int2str(ff) 'NR, elseif abs(test' int2str(ff)
'NR)>abs(test' int2str(ff) 'DR),TestR' int2str(ff) '_DISEASED=test'
int2str(ff) 'DR,end'])
end
end
```

Matlab Code to Calculate the Probability Density Function: pdf_dg.m

```
function [y]=pdf_dg(test_data,m_data);

covar=cov_dilek([m_data,test_data]);%8x1
const1=(2*pi)^2;
const2=det(covar);
const3=const1*const2;
const4=sqrt(abs(const3));
const5=1/const4;

const6=test_data-m_data;
const7=(inv(covar));
const8=exp((-0.5)*(const6')*(const7)*(const6));
y=const5*const8;
```

REFERENCES

1. Denney, S.T., "Stochastic Estimation of Deformable Motion from MR Tagged Cardiac Images," Ph.D. Dissertation, Johns Hopkins University, Baltimore, Maryland, 1995.

2. Gray, H., *Gray's Anatomy*, C.Livingstone Press 37th Edition, New York, 1989.

3. Wyngaarden, J.B., and L.H. Smith, *Cecil Textbook of Medicine*, Saunders Company, New York, 1988.

4. de Ross, A., E. Wall, *et al.*, "Magnetic Resonance Imaging in the Diagnosis and Evaluation of Myocardial Infarct," *Mag. Res. Quarterly*, Vol. 7(3). pp. 191-207, 1991.

5. Reichek, R., "Magnetic Resonance Imaging for Assessment of Myocardial Infarction," *Mag. Res. Quarterly*, Vol. 7(4), pp. 255-274, 1991.

6. Dilsizian, V., and R.O. Bonow, "Current diagnostic techniques of assessing myocardial viability in patients with hibernating and stunned myocardium," *Circulation*, Vol.87(1), pp. 1-20, January 1993.

7. Moore, C., E.R. McVeigh, and E.A. Zerhouni, "Three-dimensional Systolic Strain Patterns in the Normal Human Left Ventricle: Characterization with Tagged MR Imaging," *Radiology*, Vol. 214, pp. 453-466, 2000.

8. Denney, T.S., and J.L. Prince, "Reconstruction of 3D Left Ventricular Motion from Planar Tagged Cardiac MR Images: an Estimation Theoretic Approach," *IEEE Transactions on Medical Imaging*, Vol. 14(4), pp. 625-634, 1995.

9. Kuijer, J.P.A, *et al.*, "Simultaneous MRI Tagging and Trough-Plane Velocity Quantification: A Three-Dimensional Myocardial Motion Tracking Algorithm," *Journal of Mag. Res. Imaging*, Vol. 9, pp. 409-419, 1999.

10. Kerwin, W.S., and J.L. Prince, "Cardiac Material Markers from Tagged MR Images," http://iacl.ece.jhu.edu/projects/MRmarkers/

11. Amini, A.A., *et al.*, "Measurement of 3D Motion of Myocardial Material Points from Explicit B-Surface Reconstruction of Tagged MRI Data," http://www-cv.wustl.edu

12. Young, A.A., "Model tags: direct three-dimensional tracking of heart wall motion from tagged MR images," *Medical Image Analysis*, Vol. 3(4), pp. 361-372, 1999.

13. Derleck, J., T.S. Denney, C. Ozturk, W. O'Dell, and E.R. McVeigh, " Left Ventricular motion reconstraction from planar tagged MR images: a comparison," http://www.mri.jhu

14. Magnetic Resonance Myocardial Tagging for Motion Analysis of the Human Heart, http://www.research-projects.unizh.ch/med/unit40600/area230/p816.htm

15. Axel, L., and L. Dougherty, "MR imaging motion with spatial modulation of magnetization," *Radiology*, Vol. 171, pp. 841-845, 1989.

16. Zerhouni, E.A., D.M. Parish, W.J. Rogers, A. Yang and E.P. Shapiro, "Human heart: tagging with MR imaging--a method for noninvasive assessment of myocardial motion," *Radiology*, Vol. 169(1), pp. 59-63, 1988.

17. McVeigh, E.R., "MRI of Myocardial Function: Motion Tracking Techniques; A Review Article for Magnetic Resonance Imaging," http://www.jhu.edu/tracking-mcveigh.pdf

18. McVeigh, E.R. and E. Atalar, "Cardiac tagging with breath-hold cine MRI," *Magnetic Resonance Imaging*, Vol. 28(2), pp. 318-327, 1992.

19. Ozturk, C., and E.R. McVeigh, "Four Dimensional B-spline based motion analysis of tagged MR images: Introduction and in vivo validation," *Physics in Medicine and Biology*, Vol. 45, pp.1683-1702, 2000.

20. Osman, N.F., and J.L. Prince, "Visualizing myocardial function using HARP MRI," *Physics in Medicine and Biology*, Vol. 45(4), pp. 1665-1682, 2000.

21. Osman, N.F., E.R. McVeigh, and J.L. Prince, "Imaging Heart Motion Using Harmonic Phase MRI," *IEEE Transactions on Medical Imaging*, Vol. 19(3), pp. 186-202, 2000.

22. Osman, N.F., and J.L. Prince, "Harmonic Phase Magnetic Resonance Imaging" http://iacl.ece.jhu.edu/projects/HARP/

23. Osman, N.F., W.S. Kerwin, E.R. McVeigh, and J.L. Prince, "Cardiac Motion Tracking Using CINE Harmonic Phase (HARP) Magnetic Resonance Imaging," *Mag. Res. In Med.*, Vol. 42, pp. 1048-1060, 1999.

24. Holger, B., and G. Adelmann, "A frequency-domain Gaussian filter module for quantitative and reproducible high-pass, low-pass, and bandpass filtering of images," http://www.iscpubs.com/pubs/al/articles/a9703/a9703ade.pdf

25. Osman, N.F., and J.L. Prince, "Angle Images For Measuring Heart Motion From Tagged MRI," *IEEE Signal Processing Society 1998 International Conference on Image Processing*, Chicago, Illinois, October 4-7 1998.

26. Kuijer, J.P.A., E. Jansen, J.T. Marcus, A.C. van Rossum, and R.M. Heethaar, "Improved Harmonic Phase Myocardial Strain Maps," *Mag. Res. in Medicine*, Vol. 46, pp. 993-999, 2001.

27. Goksel, D., and C. Ozturk, "Harmonic Phase Method in MR Tag Analysis," *Biyomut*, Istanbul-Turkey, 27-28 November 2000, pp. 171-175, Istanbul, 2000.

28. Guttman, M.A., E.A. Zerhouni, and E.R. McVeigh, "Analysis of cardiac function from MR images," *IEEE Comput. Graphics Appl.*, Vol. 17, pp. 30-38, 1997.

29. Medical Imaging Lab Johns Hopkins University School of Medicine FINDTAGS: A program written by Mike Guttman for tag detection, http://prospero.bme-mri.jhu.edu/~emcveigh/findtags/

30. *Findtags* MR Tagged Image Analysis Package; http://prospero.bme-mri.jhu.edu/~mguttman/findtags/

31. Guttman, M. A., J. L. Prince, E. R. McVeigh, "Tag and contour detection in tagged MR images of the left ventricle," *IEEE Transactions on Med. Imag.*, Vol. 13, pp. 74-88, 1994.

32. NCAR Advanced Study Program, "9.4.4 The cubic spline," http://www.asp.ucar.edu/colloquium/1992/notes/part1/node77.html

33. Ortendahl D. A., J. W. Carlson, "Segmentation of magnetic resonance images using fuzzy clustering," in de Gruf and M.A. Viergever. (Eds.), *Information Processing in Medical Imaging*, pp. 91-106 C.N, New York: Plenum Press, 1988.

34. Goksel, D., and C. Ozturk, "HARP Analysis in Cardiac MR Tagging," *1.National MR Conference*, 14-16 December 2000, pp. 15, Izmir, 2000.

35. Goksel, D., M. Ozkan, and C. Ozturk, "MR Tag Analysis using Harmonic Phase Method," *9. Signal Processing Conference*, 25-27 April 2001, Northern Cyprus 2001.

36. Goksel, D., M. Ozkan, and C. Ozturk, "Towards Rapid Screening Of Tagged MR Images Of The Heart," *23. IEEE-EMBC*, Poster presentation, 2001.