SAPIENZA
UNIVERSITÀ DI ROMA

Dipartimento di Metodi e Modelli Matematici
per le Scienze Applicate
Dottorato di ricerca in Modelli e Metodi Matematici
per la Tecnologia e la Società

PhD Thesis XIX cycle

Analysis of echocardiographic movies by variational methods

Candidate
Massimiliano Pedone

The Candidate

Academic year 2008 - 2009

"Se per caso ho omesso qualcosa
di più o meno giusto o necessario,
domando indulgenza, dal momento
che nessuno è senza difetti
ed esperto in ogni questione".
"If I have perchance omitted anything
more or less proper or necessary,
I beg indulgence, since there is no one who
is blameless and utterly provident in all things".
Liber Abaci, Leonardo Fibonacci (1202)

Contents

Introduction

This thesis consists of two parts. A newer dynamic approach to the well-known static variational method for the time-series medical images is presented, and a graphic software applications for synchronization between cardiac movement and electric signals (ECG) are developed.

Many approaches have been built up to process such digital images, and it is difficult to say which one is more effective than the other. Many sophisticated tools have been developed; here we focus on PDE-based and variational methods. Many issues concerning models in scientific disciplines deal with partial differential equations (PDEs). Especially, Image Processing, Computer Vision and Pattern Analysis have undergone a great development. [2], [15], [16], [33], [27].

In PDE integration, often it is not possible to get global explicit solutions, so working up methods for approximated solutions and implementing numerical codes is generally needed.

The elaboration image sequences, which represent our main topic, concerns the identification of contours of an object for segmentation and study of its movement over time.

Papers by S. Osher and J. A. Sethian [32], [34], [36] et al. have contributed to the development of numerical methods for the solution of variational problems in segmentation of images, reconstruction of surfaces and other conventional techniques of Image Processing, which prove useful in various fields of

scientific investigation.

We have studied the feasibility of applying some segmentation techniques for the determination of cardiac efficiency parameters such as the percentage of blood expulsion and the internal pressure of the left ventricle cycle. To face this first issue in order to structure, to patient's advantage, inexpensive and not invasive medical technique of analysis [38] and to build up an automatic protocol for image sequence analysis.

Standard snake models of closed curve evolution, pertaining to the Level-Set Method, have been implemented to characterize the internal ventricular area over time. We have applied standard finite difference techniques for the approximation solution of the involved eikonal equations for front evolution. Well-known techniques of Gaussian regularization, such as heat equation, have highlighted the loss of definition of ventricular edge. Thus we now present a numerical frame preprocessing technique, based on a variational model, that consists of functional minimization with a Mumford-Shah (M-S) time-dependent energy term, which is suitable to enhance ventricular edge and regularize initial data for curve evolution. This one, unlike the standard convolution with a regularizing operator, allows to compute the bright-intensity gradient of the image for velocity term of the curve evolution.

Theoretical results for closed-curve evolution on bidimensional domain our analysis is setup on are due to M. Falcone [23], [11], [12], [13], [33] et al. Approximation of the M-S functional [29] is based on work by Ambrosio and Tortorelli, instead [1], [2], [3], [4], [5].

Here we present results for different choices of parameter values, both terms of threshold values for the front evolution and favor for regularity of the contours in the minimization of the functional.

It was an Echo-Cardio-Graphic Instrument (Esaote MyLab30) to provide movie for data from real patients (Clinical Sciences Department of Umberto I General Hospital, Sapienza University).

The present thesis illustrates, in the first chapter, the formulation of the standard model problem for image segmentations by curve evolution. The Gaussian regularization of initial data and a newer approach of pretreating model by functional minimization. The applicability of the resolution method is discussed and an open mathematical problem for a new functional involving time-dependence are presented.

The second chapter presents the approximation of the above mentioned models, the stability of numerical model and the parameter choice.

Finally the last chapter describes the digital format of movies, the adaptation of single frames, the step that characterizing elaboration protocol, the quality of the results obtained with the preprocessing on a test problem. As well as the choice and implementation of reduced numerical methods in the real case and their computational cost. We compare results obtained by various preprocessing methods identified areas, curve evolution beyond the edge, ventricular area his trends over time and the evaluated Ejection Fraction. A static picture of the implemented graphic application is presented at the end of the chapter, as well as the principal use of the signal-synchronizing software. A future field of study is in the open problem subsection of the first chapter.
In the appendix a brief recall of H-J equation, Γ-convergence and SBV space function are presented.
In conclusion the aim of this work consists in generating a protocol for medical investigation to produce time-dependent profiles of the ventricular area, which could turn useful for diagnostic and research, as well as a graphical tool to synchronize and study these signals.

Chapter 1

The Active-Contours model problem

In the problem model for the individualization and recognition of contours, the front of a parametrized curve evolves from an allocated initial configuration, of the domain $\Omega \subset \mathbb{R}^2$, to stops its evolution on the external or internal edge of the object.
If instantly, the initial curve, is positioned on the external edge of the image, really to a very small distance from the edge, or is little regular closed curve in the domain and it according to the normal to every point of the curve, it will evolves to halt itself on the, external or internal, contour of the set when the local growth of the gradient of the image's bright intensity, become less than a threshold parameter. That determines the presence of the object contour in the represented set. (references in [17], [36], [34] and appendix).
Let us now recall some basic definition for a curve as isovalues of surface and its speed of propagation to which we refer the edge detection model problem.

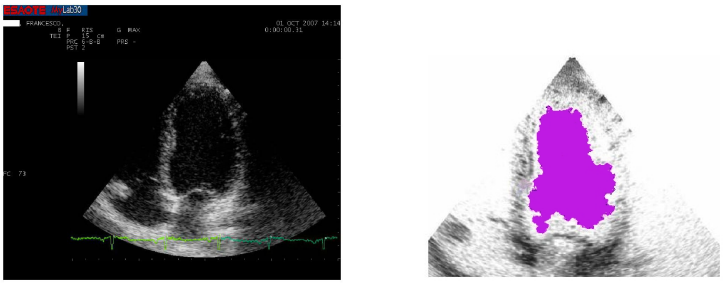


Figure 1.1.0.a: Echocardiographic frame and its recognized Area

1.1 Ventricular area recognition

Our pourposes is the automatic detection of the internal area of the ventricular cavity. This suggest us to perform the curve evolution as an initial value problem. A typical results that represent our goal is presented in the figure 1.1.0.a. In the next part we focus the attention concerning the choice of the filter function for the edge-detection and the related applicability of the speed term formulation in the continuous model problem.

1.1.1 Edge-detection

Given an image, in particular an echocardiographc image, we need to automatically detect the internal contour of object for clinical pourposes. We focus ours attention to a ventricular cavity. We consider a closed subset Ω of $\mathbb{R}^2$, having $\partial\Omega$ as the external edge of the rectangular image and being $I : \Omega \to [0, 1]$ the brightness intensity. The "edge-detector" (detector of contours) for segmentation of the image is a positive real coefficient, which is dependent of $I(x)$'s gradient at every point of the curve. In particular, the model is represented by a filter function such that

$$g : \mathbb{R}^+ \longrightarrow \mathbb{R}^+$$

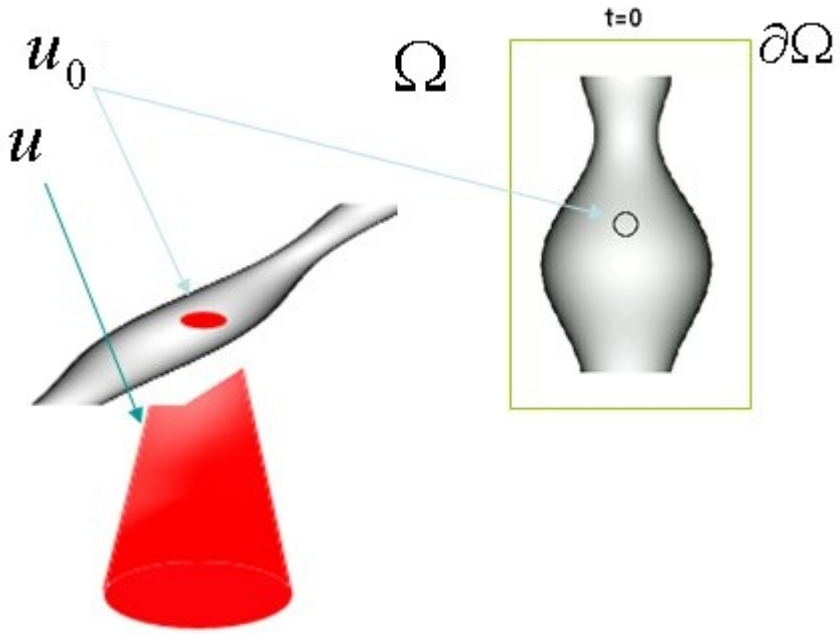


Figure 1.1.0.b: Curve at isolevel at $t = 0$

where

$$g(z) = \frac{1}{1+z}$$

decreasing with z and

$$\lim_{z\to\infty} g(z) = 0$$

such that $0 \leqslant g$ and $g(0) = 1$.

1.1.2 Segmentation case

For front evolution the $level - set$ standard model is adopted (see Sethian [36], [34], [27]et al.). A curve in $\mathbb{R}^2$ can be represented as the zero-level line of a function in higher dimension. More precisely, let us suppose that there

exists a function $u : \mathbb{R}^2 \times \mathbb{R}^+ \to \mathbb{R}$ solution of the initial value problem:

$$\begin{cases} u_t(x,t) - g_I(x)|\nabla u(x,t)| = 0 & x \in \Omega,\ t \geq 0 \\ u(x,0) = u_0(x) & x \in \Omega \end{cases} \tag{1.1.1}$$

which is the model of **eikonal equation for front evolution** for segmentation problem, typically involved in the detection for edges of objects contained in an image. We have to note that the evolving function $u(x,t)$ always remains a function as long as g_I is smooth. So we choose a different expression for the speed terms in order to limit the loss of image definition of Gaussian smoothing and by new approach with functional minimization to preserve this property. For ours pourposes we intend enough smooth a function $C^2(\Omega)$.

1.1.3 Speed choice for image processing

The speed term is dependent of the brightness intensity at every pixel and is directed along the outward normal, starting from an initial elliptic profile E centered in Ω. Insofar we define the composite function

$$g_I(|\nabla I(x)|) = g(|\nabla(G_\sigma * I(x))|) = \frac{1}{1 + |\nabla(G_\sigma * I(x))|}, \tag{1.1.2}$$

where $G_\sigma{}^*\ I(x)$ is the convolution of the image $I(x)$ with a regularizing operator G_σ whose algorithmic implementation we are going to explicit later. The operator G_σ is a filter that allows, by heat equation, to calculate the brightness-intensity gradient of the image in the presence of discontinuous data.

The discretization of the problem is performed by building a rectangular lattice, which is made fine-grained according to image's pixel definition. The curve is parametrized by means of a Lipschitz function. Therefore, the differential problem to be numerically solved will look as follows:

considering the built-in function $u_0(x)$ such that $E = \{x \in \mathbb{R}^2 : u_0(x) = 0\}$

$$\begin{cases} u_t - g_I(|\nabla I(x)|)|\nabla u| = 0 & x \in \Omega \subset \mathbb{R}^2 \times [0,T] \\ u(x,0) = u_0(x) & x \in \Omega \subset \mathbb{R}^2 \end{cases} \tag{1.1.3}$$

where T is the time horizon.
In the numerical tests the algorithm predicts a stop of the evolving curve at a threshold value th for the speed term, such that, if $g_I(x) \leqslant th \Rightarrow g_I(x) = 0$, then the threshold parameter allows curve evolution to stop in the presence of different gradient values for the regularized image.

1.1.4 Gaussian regularization of the image by heat equation

The problem representing diffusion, or transfer, of heat in a given material from an assigned initial configuration can be modeled with the aid of a second-order parabolic partial derivative equation.
If a source is present, distributed in the domain, and an initial profile is given, in the bidimensional domain the modeling follows the Cauchy's problem:

$$\begin{cases} u_t - \Delta u(x,t) = f(x,t) & (x,t) \in \Omega \times [0,T] \\ u(x,0) = u_0(x) & x \in \Omega \end{cases}$$

with boundary conditions $\partial\Omega \times [0,T]$, where $f(x,t)$ is the source function and $u_0(x)$ describes the profile at $t = 0$. We are not interested in the explicit solution that can be easily found in literature, but rather in its regularizing properties on the given image in the domain Ω. In this case we will assume $f(x,t) = 0$ and bounds fixed on the external edge of the image. This way the model problem becomes:

$$\begin{cases} u_t - \Delta u(x,t) = 0 & (x,t) \in \Omega \times [0,T] \\ u(x,0) = u_0(x) & x \in \Omega \end{cases} \tag{1.1.4}$$

with Dirichlet boundary conditions. The numerical implementation makes use of a finite-difference scheme of approximation. Solving (1.3.1) is equivalent to carring out a Gaussian linear filtering, which was widely used in signal processing. More precisely, let u_0 in $L^1([-1,1])$. Then the explicit solution of 1.1.4 (1.3.1) is given by

$$u(x,t) = \int_{\mathbb{R}^2} G_{\sqrt{2t}}(x-y)u_0(y)dy = (G_{\sqrt{2t}} * u_0)(x),$$

where $G_\sigma(x)$ denotes the two-dimensional Gaussian Kernel

$$G_\sigma(x) = \frac{e^{-\frac{|x|^2}{2\sigma^2}}}{2\pi\sigma^2}.$$

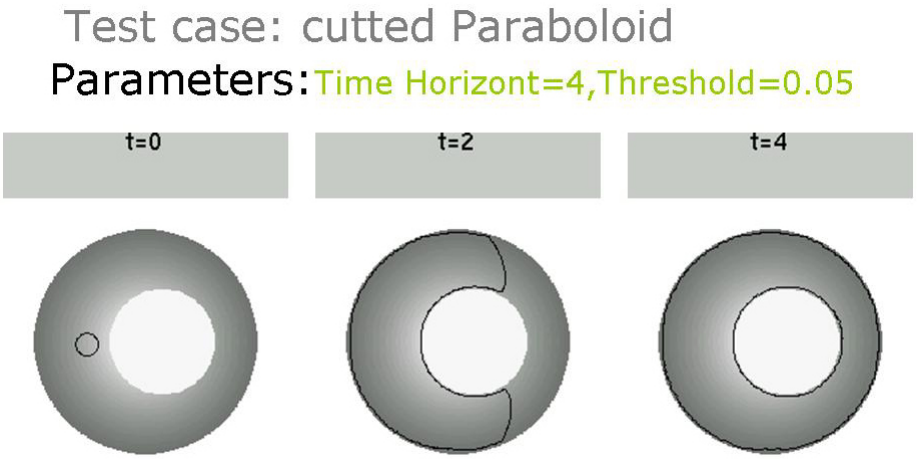
Convolution by a positive kernel is the basic operation in linear image filtering. It corresponds to low-pass filtering. This formula gives the correspondence between the time t and the scale parameter σ of the Gaussian kernel. See page 56-57 and 85-87 of [6] for better reference on mollification properties.

1.1.5 Curve evolution on image

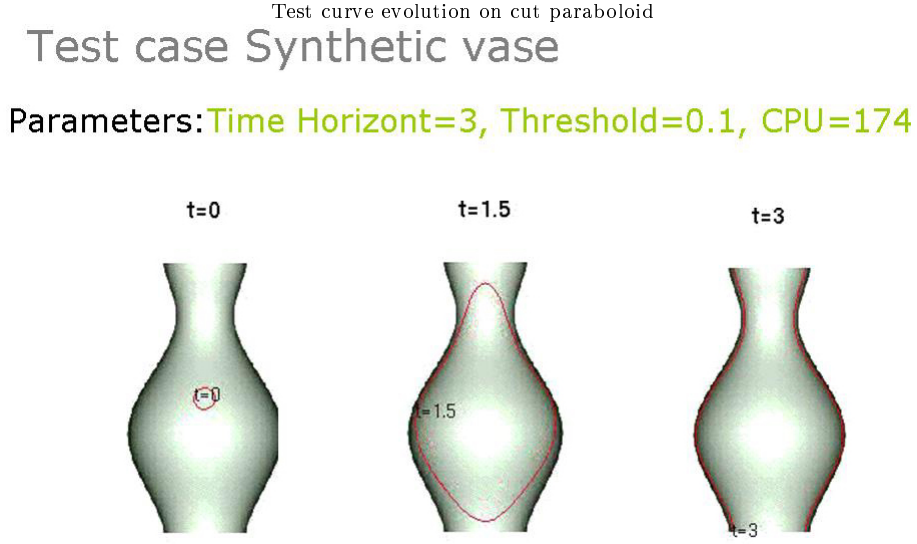
As an anticipation of what we are going to do in the case of a movie, now we want to recall in brief the evolution of a closed curve on a single image. Some results are presented here in order to assess the action of a Gaussian regularization operator in terms of its number of iterations and choice of the threshold parameter that characterizes the stopping function of the edge detector. Such results are both from test-images and echocardiographic image frames, with different CPU time needed for the frame case on different hardware systems with and without MPI parallel elaboration.

Test images

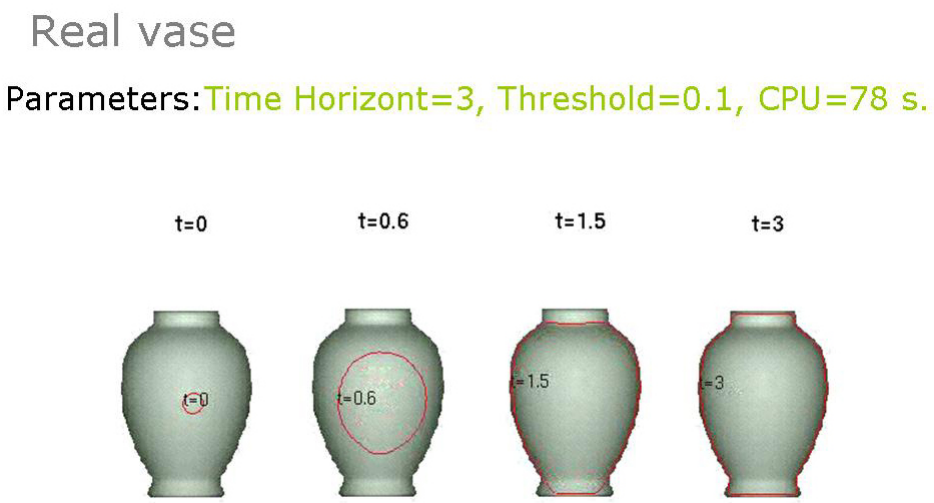
In this test case, we can see the modeled problem applied on real and synthetic images.



Test curve evolution on cut paraboloid



Test curve evolution on synthetic vase.



Test curve evolution on real vase.

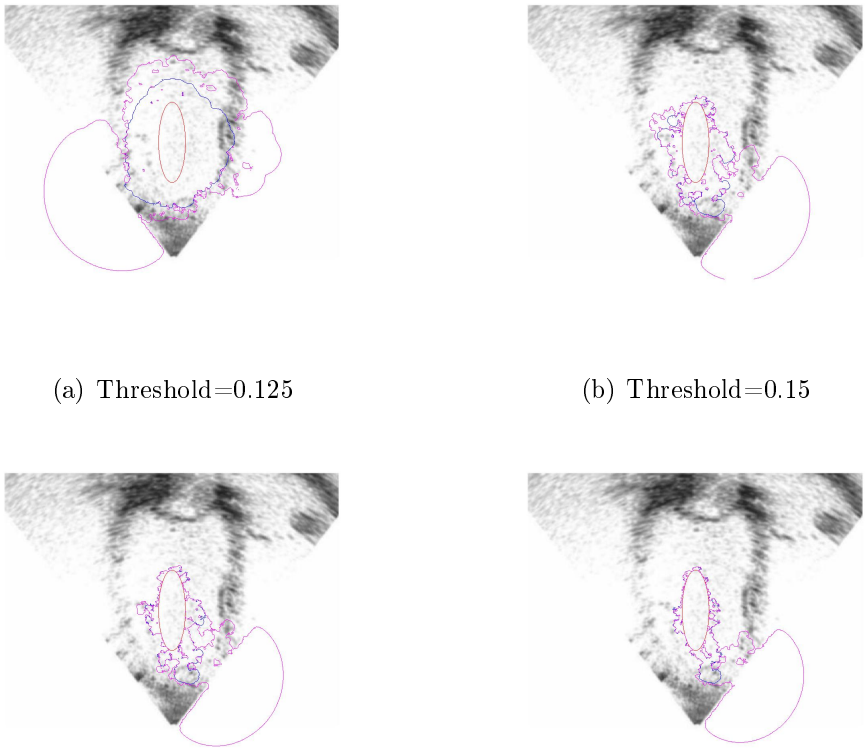
The curve stops its evolution on the edge of the object contained in the image. We use threshold values around 0.1 for curve evolution on these test-problems.

1.1.6 Real test case: Echographic image

The ventricular contour of the cardiac muscle behaves like an elastic gum, so inducing a good performance of the model for a large part of image frames, but, at its maximum expansion, the ventricle reveals low echo response from thin tissue at various locations, so causing the image-frame to be affected by the presence of a hole in the edge delimitation and the curve to go out. This kind of limit in recognition of the internal area is emphasized by Gaussian filtration of the image, which is needed to calculate the gradient for the velocity term in the model problem. A lot of curve evolution results on echocardiographic images are reported below. There we can compare different choices of threshold value and number of smoothing iterations. A detailed explanation of eikonal results from front evolution with different choices of threshold parameter for images smoothed in a variable number of heat iterations is

presented below:

Figure 1.1.6.c: Smooth-iteration=1, Time-Horizon=3



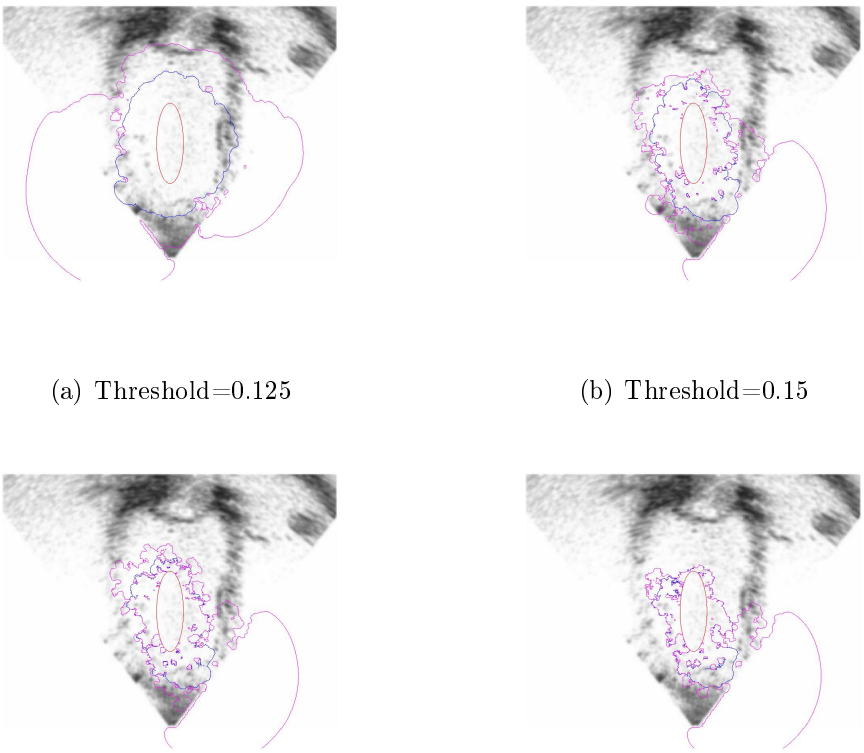
(a) Threshold=0.125

(b) Threshold=0.15

(c) Threshold=0.175

(d) Threshold=0.2

Figure 1.1.6.d: Smooth-iteration=3, Time-Horizon=3



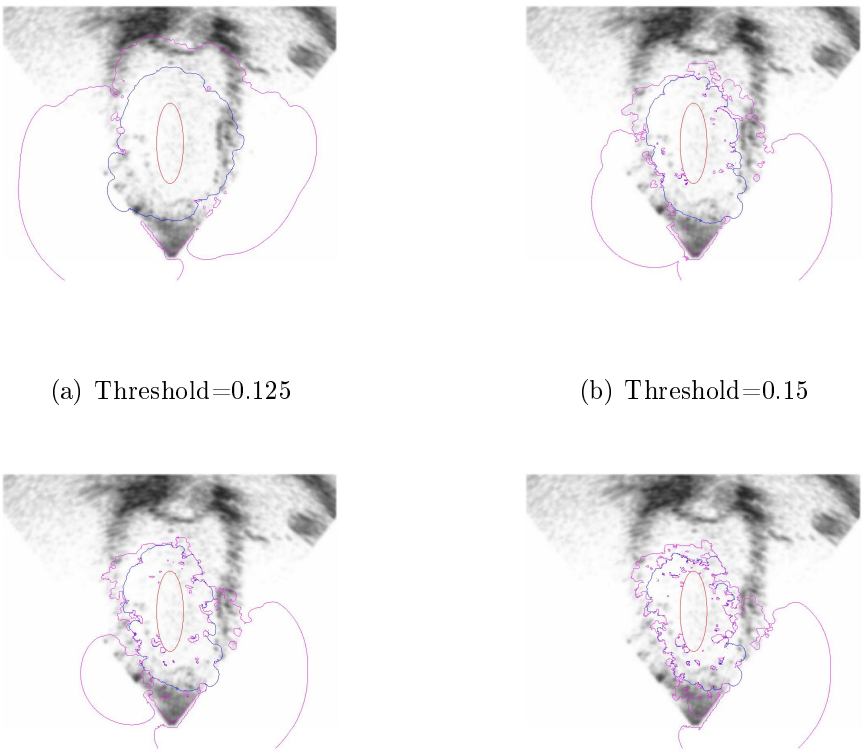
(a) Threshold=0.125

(b) Threshold=0.15

(c) Threshold=0.175

(d) Threshold=0.2

Figure 1.1.6.e: Smooth-iteration=5, Time-Horizon=3



(a) Threshold=0.125

(b) Threshold=0.15

(c) Threshold=0.175

(d) Threshold=0.2

The standard choice of threshold-parameter value at variable number of smoothing iterations, as we can see, is insufficient to give a correct stop of the curve within a small range of the pixels neighboring the ventricle edge. Indeed, this kind of image, with its amount of noise so overlapping with the studied signal, suggests us to apply a different technique in the frame analysis. Many methods and different approaches are currently available and lots of papers were written about this specific topic. An exhaustive work, due to Stanley Osher et al., is [31]. As a large part of other scientists, expert in image analysis, we choose here to investigate a newer approach including a preprocessing step, based on functional minimization, to enhance the border, following the idea that image noise is quite static over time, while studied signal (ventricle border, in our case) is rhythmically moving. Thus, let us try to apply the Mumford-Shah functional to this dynamic problem.

1.2 New approach to regularize and enhance Image

Analysis of images through variational methods finds application in a number of fields such as robotics, elaboration of satellite data, biomedical analysis and many other real-life cases. By segmentation is meant a search for constituent parts of an image, rather than an improving of its quality or characteristics. Our aim is to develop a criterium for enhancing movie frame in two possible ways by functional minimization. First, we adopt the M-S functional and its approximated form proposed by Ambrosio Tortorelli [5] to regularize data frame for curve evolution method, instead of Gaussian regularization. We call it "classic functional" since the function space the variational integral converges over contains functions that have the regularity needed for image gradient calculus. Second, we present a numerical scheme, where a

time-dependent parameter is inserted, precisely in the second integral part of functional distinguished by a "time gradient", for enhancing the internal moving parts of the object. A detailed analytical treatment and a numerical scheme for minimization of the functional, which involves some delicate conjectures and refined mathematical steps, can be found in [6]. In the following section we recall in brief the essential formulation of the model problem used to regularize and enhance image. Other particular on numerical approximation and function space are explained in the next chapter and in appendix. The reader can look up, for a complete review, the book by Morel and Solimini [30].

1.2.1 The M-S Functional

First, the formulation introduced by Mumford and Shah in 1985 is presented. It consists of an energy-based method. For a given image g, we search for a function $u : \Omega \longrightarrow \mathbb{R}$ and a set $K \in \Omega$ (the discontinuity set) minimizing

$$F(u, K) = \int_{\Omega} (u - g)^2 + \lambda \int_{\Omega \setminus K} |\nabla u|^2 \, dx \quad + \alpha \int_{K} d\sigma.$$

To obtain the existence of a minimizer and to compute it, following Ambrosio [1; 2; 3], these conditions, assuming the following regularity hypotheses to stand true:

(C1) K consists of a finite number of $C^{1,1}$-curves, meeting $\partial\Omega$ and meeting each other only at the endpoints.

(C2) u is C^1 on each connected component of $\Omega - K$.

Mumford and Shah conjectured that the functional admits a minimizer satisfying (C1), (C2). [29]

Approximation of the M-S functional by Γ-convergence

The numerical minimization of the M-S functional is a difficult problem because of the presence of unknown discontinuity set S_u. The approximation of the functional F by means of Γ-convergence is an useful tool for the numerical computation of minimizers [5; 35]. Briefly, in Ambrosio Tortorelli's theorem the functional F is approximated by a sequence of elliptic functionals. Let the functional defined by

$$\widehat{F}(u,S) = \begin{cases} F(u) & \text{if } u \in SBV(\Omega), S \equiv 1, \\ +\infty & \text{otherwise.} \end{cases}$$

Let $\{K_\epsilon\}$ be a sequence of positive numbers converging to zero such that

$$\lim_{\epsilon \to\ 0^+} K_\epsilon = 0$$

The approximating functional F_ϵ are defined [5; 35]:

$$F(u) = \mu \int_\Omega (u-g)^2 + \int_\Omega (S^2 + K_\epsilon) \mid \nabla u \mid^2 dx \quad + \alpha \int_\Omega \left(\epsilon \mid \nabla S \mid^2 + \frac{1}{4\epsilon}(1-S)^2 \right) dx.$$

Where the above term $(S^2 + K_\epsilon)$ is near to the approximation of the sequence of functional in A-T theory in its means of tubular neighbourhood and its skrinking when $\epsilon \to 0^+$ (see [5; 35]).

1.2.2 Ambrosio-Tortorelli algorithm ('90)

This section refers to M-S algorithm for the approximation of $F(u,K)$ with a sequence F_ϵ of regular functionals defined on a Sobolev space. We focus our attention on the Ambrosio-Tortorelli approximation, which is among the most used in image analysis. In this particular approach, the set K (or S_u) is replaced by an auxiliary variable S (a function) which approximates the characteristic function $(1 - \chi_K)$.

$$F_\epsilon(u,S) = \int_\Omega (u-g)^2 dx + \int_\Omega S^2 \mid \nabla u \mid^2 dx \quad + \int_\Omega \left(\epsilon \mid \nabla S \mid^2 + \frac{1}{4\epsilon}(1-S)^2 \right) dx$$

If (u_ϵ, S_ϵ) minimizes the functional F_ϵ, then the following result holds ([5]) :

$$u_\epsilon \xrightarrow{\mathcal{L}^2} u \quad \text{ed } S_\epsilon \longrightarrow 1 - \chi_K \quad \text{per } \epsilon \longrightarrow 0^+$$

We gather from the problem of minimun related to the Ambrosio-Tortorelli functional, the Euler equations, than by an appropriate approximation will give us an algorithm for minimization.
When we apply Euler equation system, if $u_0 = g$ is chosen, an enough rapid continuation method will give good results even though contours are not well defined.

1.2.3 Euler equation of the approximated functional

Given

$$\int_\Omega \psi(u, \nabla u) dx,$$

the associated Euler equation is

$$div\left(\nabla_\xi \psi(u, \nabla u)\right) - \left(\frac{\partial \psi}{\partial u}\right) = 0.$$

Using Neumann boundary conditions, we get:

$$F(u) = \mu \int_\Omega (u-g)^2 + \int_\Omega (S^2 + K_\epsilon) \mid \nabla u \mid^2 dx \quad + \alpha \int_\Omega \left(\epsilon \mid \nabla S \mid^2 + \frac{1}{4\epsilon}(1-S)^2\right) dx.$$

Euler equation system for u, considering that the quadratic gradient $2(S^2 + K_\epsilon)\nabla u^2$ is transformed into the Laplacian, looks as follows:

$$\begin{cases} div(2(S^2 + K_\epsilon)\nabla u) = \mu(u-g) & \frac{\partial u}{\partial \mathbf{n}} = 0 \text{ in } \partial\Omega \\ \alpha\epsilon\Delta S = S \mid \nabla u \mid^2 - \frac{\alpha}{4\epsilon}(1-S) & \frac{\partial S}{\partial \mathbf{n}} = 0 \text{ in } \partial\Omega \end{cases}$$

where the non linear terms S^2 and ∇u cause the system to be non-linear and elliptic, with such a structure that, when S is known, the first equation gets linear, while, if it is u to be known, it is the second equation to be linear. This suggests the adoption of a two-stage iterative scheme. Where we fix the discrete step of convergence that is enough to give back a regularized image, enhanced in its content edge, as we present in the next section.

1.3 Open problem

In course of development, with R. March and G. Riey, there is a new and more sophisticated functional that involves the time-dependent integral. The new formulation is: let
$g : \Omega \times [0,T]$, $\Omega \in \mathbb{R}^2$
$u \in \mathbb{R}^2 \times [0,T]$,

$$F(u) = \int_0^T dt \int_\Omega \left| u - g \cdot \varphi_{\eta,L}\left(\left|\frac{\partial g}{\partial t}\right|\right)\right|^2 dx + \int_0^T dt \int_\Omega \left|\widetilde{\nabla_x} u\right|^2 dx + \int_0^T dt \cdot \mathcal{H}^1\left(S_{u(t)}\right) +$$

where $\varphi_{\eta,L}$ is a cut function. Time regularization parts

$$+ \int_0^T dt \left(\frac{\partial u}{\partial t}\right)^2 dx + \int_0^T dt \int_\Omega \left|\widetilde{\nabla_x}\left(\frac{\partial u}{\partial t}\right)\right|^2 dx + \int_0^T dt \cdot \mathcal{H}^1\left(S_{\frac{\partial u}{\partial t}}\right)$$

Discretization in Ambrosio Tortorelli format.

$$F_\epsilon(u) = \int_0^T dt \int_\Omega \left| u - g \cdot \varphi_{\eta,L}\left(\left|\frac{\partial g}{\partial t}\right|\right)\right|^2 dx +$$

$$\int_0^T dt \int_\Omega S^2 \left|\nabla_x u\right|^2 dx + \int_0^T dt \int_\Omega \left\{\epsilon \left|\nabla_x S\right|^2 + \frac{1}{4\epsilon}(1-S)^2\right\} dx$$

$$+ \int_0^T dt \left(\frac{\partial u}{\partial t}\right)^2 dx +$$

$$\int_0^T dt \int_\Omega Z^2 \left|\nabla_x\left(\frac{\partial u}{\partial t}\right)\right|^2 dx + \int_0^T dt \int_\Omega \left\{\epsilon \left|\nabla_x Z\right|^2 + \frac{1}{4\epsilon}(1-Z)^2\right\} dx$$

So the second part of the functional presents the form:

$$F_\epsilon(u(x,t)) = \int_0^T dt \int_\Omega Z\left(u, \nabla_x u, \frac{\partial u}{\partial t}, \nabla_x\left(\frac{\partial u}{\partial t}\right)\right) dx.$$

We write the Euler equation for integral functional that depends to the function u and its first derivative and also to the superior order derivative. Euler equation of the functional become:

$$u - g\varphi_{\eta,L}\left(\left|\frac{\partial g}{\partial t}\right|\right) - div\left(S^2 \nabla_x u\right) - \frac{\partial^2 u}{\partial t^2} + \frac{\partial}{\partial t}\left[div\left(Z^2 \nabla_x \frac{\partial u}{\partial t}\right)\right] = 0.$$

Than we obtain the system:

$$\begin{cases} S \mid \nabla_x u \mid^2 - \epsilon \Delta_x S - \frac{1}{2\epsilon}(1-S) = 0 \\ Z \mid \nabla_x \left(\frac{\partial u}{\partial t}\right) \mid^2 - \epsilon \Delta_x Z - \frac{1}{2\epsilon}(1-Z) = 0. \end{cases}$$

The existence of minimum and the approximation are in course of development.

Chapter 2

Numerical approximation of the models

We are going to describe, in this chapter, the numerical approximation of the models introduced above. In details, our aim is to present the semilagrangian scheme for curve evolution by numerical solution of eikonal eq. The standard mollifier is constructed by a Gaussian operator as a numerical scheme for heat eq. in two dimensions. The last section reports the procedure to build the approximation of the Mumford-Shah functional by Ambrosio-Tortorelli scheme.

2.1 Semi-Lagrangian (SL) scheme for Eikonal eq.

Semi-Lagrangian (SL) schemes try to mimic the continuous behavior by constructing the solution at each grid point by back integration along the characteristic trajectory passing through the point and reconstructing the value at the foot of the trajectory by interpolation. The numeric dependence of

the domain contains its continuous dependence without any additional condition on Δt and Δx (space lattice is usually defined for finite differences as the pixel resolution of images). This allow a larger time steps than other schemes where the CFL condition has to be imposed for stability guaranty. The numerical method for eq. (1.1.3) following the semilagrangian scheme by M. Falcone and M. Sagona [21]. In our implementation we use a threshold value that has been chosen and discussed in a work cited earlier [33]. The SL-scheme is strictly connected with Hopf-Lax representation formula (A.1.5) for exact solution of (HJ1). The Hamiltonian, for eikonal equation, we consider is $H(p) = |p|$, is continuous and convex but the assumption (A.1.4) is not satisfied, in fact

$$\frac{H(p)}{|p|} \to 1 \quad \text{when} \quad |p| \to +\infty. \tag{2.1.1}$$

So the trasformated $H^*(p)$ is defined only on a subset of the space where we search the solution. When The typical SL-scheme, for D=2 such as a rectangular domain, is defined in terms of a lattice $L(\Delta x, \Delta y, \Delta t)$ such that

$$L \equiv \{(x_i, y_j, t_n) : \ x_i = i\Delta x, \ y_j = j\Delta y, t_n = n\Delta t; \ i, j \in \mathbb{Z}, n \in \mathbb{N}\}$$

where $(x_i, y_j, t_n) \in \mathbb{R}^2 \times \mathbb{R}^+$, $\Delta x, \Delta y$ and Δt are space and time steps. In case of images, the step is equivalent to space definition (since a grid based on image resolution is taken), while Δt is the time step. We can write (HJ1), first order equation as follows:

$$u_t + \sup_{a \in \mathbb{R}^2} \{\nabla u \cdot a - H^*(a)\} = 0 \tag{2.1.2}$$

To obtain the SL-scheme we consider the approximation :

$$-\nabla u(x_i, y_j, t_n) \cdot a = \frac{u(x_i - a_1\Delta t, y_j - a_2\Delta t, t_n) - u(x_i, y_j, t_n)}{\Delta t} + O(\Delta t)$$

If we denote an approximation of $u(x, y, t); i, j \in \mathbb{Z}$ by standard $u^n_{i,j}$ and make use of forward finite differences and directional derivatives from previous

equation as [23], then we obtain:

$$\frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t} = \min_{a \in B_2(0,1)} \left[\frac{u^n(x_i - a_1\Delta t, y_j - a_2\Delta t, t_n) - u^n(x_i, y_j)}{\Delta t} + H^*(a) \right]$$

which leads to the time explicit scheme

$$u_{i,j}^{n+1} = \min_{a \in B_2(0,1)} [u^n(x_i - a_1\Delta t, y_j - a_2\Delta t) + \Delta t H^*(a)]$$

Several steps are needed, in general, to compute the solution, since the value of u on the right-hand side has to be computed first by an interpolation procedure on the nodes and then $H^*(a)$ must be determined in order to finally compute the minimum for $a \in \mathbb{R}^2$. Really the search for a minimum can be reduced to a bounded set in many cases, as shown in the explicit scheme, to the minimum on the unit ball. The reconstruction step makes use of a calculation procedure at foot of characteristics on grid-nodes. A high-order polynomial interpolation cannot be used since it could introduce oscillations in non-regular solutions; we, thus, apply the scheme by M. Rorro (CASPUR) using ENO, WENO interpolation. We construct the protocol step of curve evolution by an oriented adaptation to medical application of HJPACK parallel OpenMp Fortran programming language.

2.2 Gaussian image smoothing

The multidimensional model for parabolic partial derivative eq. in a rectangular domain, where $x \in [a, b]$ and $y \in [c, d]$ for the linear problem (1.1.4) and $\Omega \equiv [a, b] \times [c, d]$, is resumed according to steps exposed in the following discussion.

We proceed by creating a space mesh of the domain, dividing abscissas into M parts and ordinates in N parts spanning $\Delta x = (b-a)/M$ and $\Delta y = (d-c)/N$, respectively. A subdivision of time axis is created with spacing Δt. At this point, the method performs the calculus of u-values in every node of the

space mesh. At the n-th step, the computation of the approximate value at the n+1-th step is performed according to the scheme:

$$u^{n+1} = u^n + \Delta t \frac{u^n_{i+1} - 2u^n_i + u^n_{i-1}}{\Delta x^2} + \Delta t \frac{u^n_{j+1} - 2u^n_j + u^n_{j-1}}{\Delta y^2} \tag{2.2.1}$$

This is a five-point approximation scheme performed for $t \in [0, \epsilon]$ with a time step Δt. The stability of the explicit scheme is assured by specific conditions concerning the space mesh taking in relation to the time-step unit between two successive approximations.

2.2.1 Stability and convergence of the explicit scheme

The stability of the explicit scheme is influenced by a convenient choice of mesh parameter values in connection with the established time step. The truncation error is measurable and is not increased during computation, so it does not cause the numerical solution to diverge. Stability is assured by Courant-Friedrichs-Lewy (CFL) condition

$$\Delta t \leq \Delta x^2. \tag{2.2.2}$$

The illustrated explicit scheme is convergent if the calculated solution has the right value at each pixel within the selected mesh.

The numerical scheme of the Gaussian-filter operator

The convolution of the image with the operator G_σ decribed in section (1.1) is realized by discretized heat equation reported above, by putting:

$$Img_{smooth} := G_{\sqrt{2}} * I(x)$$

through which is regularized $I(x)$ given for the computation. Where $\Omega \in \mathbb{R}^2$, $\forall x \in \Omega / \partial\Omega$ for $x = (x_1, x_2)$ we obtain:

Let $I(x_1, x_2) = img(h, k)$

The numerical scheme that performs the convolution is:

```
DO l=1,ItNum                    Iteration of the heat eq.
   DO k=nys,nyd            Steps in the domain image
      DO h=nxs,nxd
         u(h,k)=img(h,k)+(img(h+1,k)-2img(h,k)+img(h-1,k))+
         +(img(h,k+1)-2img(h,k)+img(h,k-1))
      ENDDO
   ENDDO
   img=u
ENDDO
```

At every iteration we allocate the smoothed image u for the successive smoothing iteration step.

Where $\sigma = \sqrt{2ItNum}$ Than results $Img_{smooth}^{ItNumber}$ where $ItNumber$ is the values of ItNum in the code. From the relation between the number of iteration given to the code we wont to remark that the values of the kernel for Gaussian filter is established by considering at every time step Δt the values of $\sigma = \sqrt{2\Delta t}$ than for the given CFL condition it implies the values of the $(\Delta x, \Delta y)$.

2.3 Ambrosio-Tortorelli approximation of the M-S functional

The numerical scheme is made by dividing in two coupled parts with $u_0 = g$ and $S_0 = 1$.

At every step we calculate u_1 for $S_0 = 1$ solving a linear elliptic equation and, this way, we find S_1 from the second equation; this process is repeated for a fixed number of iterations.

2.3.1 System discretization

We use a numerical scheme based on explicit finite differences, over the rectangle Ω with step h; this way we obtain $(x, y) = (ih, jh)$ for $0 \leqslant i, j \leqslant N$. Reducing Ω to a square of side 1 and taking $h = \frac{1}{N}$, discrete coordinates will become: $u(ih, jh) \cong u_{i,j}$ and $S(ih, jh) \cong S_{i,j}$. We use an approximated scheme that is enough to enhancing little areas, characteristic of the echographic image. Some scheme for the Ambrosio-Tortorelli segmentation problem can be found on Spitaleri et al. article [35].

In order to determine the minimum of the functional we adopt the schema given, for a finite element, in Birindelli and Finzi Vita [10] to a finite difference meshgrid through the following iterative scheme:

given a maximum number of iteration Nit and a tolerance ϵ, then we construct:

-$S_0 = 1, u_0 = g$

-for $n = 1, 2, ..., Nit$ find u_n, by solving:

$$\begin{cases} div(2(S_{n-1}^2 + K_\epsilon)\nabla u_n) = \mu(u_n - g) & \text{in } \Omega \\ \frac{\partial u_n}{\partial \mathbf{n}} = 0 & \text{in } \partial\Omega \end{cases}$$

and S_n by solving:

$$\begin{cases} \alpha\epsilon\Delta S_n = S_n \mid \nabla u_n \mid^2 - \frac{\alpha}{4\epsilon}(1 - S_n) & \text{in } \Omega \\ \frac{\partial S_n}{\partial \mathbf{n}} = 0 & \text{in } \partial\Omega \end{cases}$$

-stop for $n = Nit$.

From minimization theorem 3.1 Proposition 2.1 in [10], it respectively follows that:

S_u is a piecewise C^2 submanifolds of $\mathbb{R}^2$

for any $n > 1$ there exists u_n an S_n solution of the respective system which satisfy the bounds:

$$\|u_n\|_{L^\infty} \leq \|g\|_{L^\infty}.$$

Then we discretize the equations by finite differences.

The discreet Divergence

In a numerical scheme a function $u \in \mathbb{R}^2$ can be approximated by finite difference, its first order variation on the x direction is:

$$\frac{\partial u}{\partial x} \cong \frac{u_{i+1,j} - u_{i,j}}{h}$$

and for y direction is:

$$\frac{\partial u}{\partial y} \cong \frac{u_{i,j+1} - u_{i,j}}{h}.$$

The divergence of a function $Z(x, y)$ by second order of centered difference is given by

$$div(Z(x,y)\nabla u(x,y)) \cong$$

$$\cong Z_{i+\frac{1}{2},j}(u_{i+1,j} - u_{i,j}) - Z_{i-\frac{1}{2},j}(u_{i,j} - u_{i-1,j}) +$$

$$+Z_{i,j+\frac{1}{2}}(u_{i,j+1} - u_{i,j}) - Z_{i,j+\frac{1}{2}}(u_{i,j-1} - u_{i,j}).$$

where

$$Z_{i+\frac{1}{2},j} = \frac{1}{2}(Z_{i+1,j} + Z_{i,j}) \;\; ; \;\; Z_{i-\frac{1}{2},j} = \frac{1}{2}(Z_{i,j} + Z_{i,-1j})$$

$$Z_{i,j+\frac{1}{2}} = \frac{1}{2}(Z_{i,j+1} + Z_{i,j}) \;\; ; \;\; Z_{i,j-\frac{1}{2}} = \frac{1}{2}(Z_{i,j} + Z_{i,j-1})$$

By applying to the system of Euler equation for the n^{th} approximated item of the sequence of the Ambrosio-Tortorelli functional we obtain: the term $Z(x, y) = (S^2(x, y) + K_\epsilon)$ become by fixing every direction of the space, with K_ϵ neglected as in [35], we get: for x

$$\frac{\partial}{\partial x}\left(\left(S^2 + K_\epsilon\right)\frac{\partial u}{\partial x}\right) \cong$$

$$\frac{1}{2h^2}\left[(S^2_{i+1,j} + S^2_{i,j})(u_{i+1,j} - u_{i,j}) + (S^2_{i,j} + S^2_{i-1,j})(u_{i-1,j} - u_{i,j})\right],$$

for y

$$\frac{\partial}{\partial y}\left(\left(S^2 + K_\epsilon\right)\frac{\partial u}{\partial y}\right) \cong$$

$$\frac{1}{2h^2}\left[(S^2_{i,j+1}+S^2_{i,j})(u_{i,j+1}-u_{i,j})+(S^2_{i,j}+S^2_{i,j-1})(u_{i,j-1}-u_{i,j})\right].$$

In Image Processing the lattice has a spatial density which is equivalent to image resolution, so we use here finite differences with node resolution equal to the spatial grid-step, so the discretized equation will look as follows

$$\begin{gathered}(S^2_{i+1,j}+S^2_{i,j})(u_{i+1,j}-u_{i,j})+(S^2_{i,j}+S^2_{i-1,j})(u_{i-1,j}-u_{i,j})+\\ +(S^2_{i,j+1}+S^2_{i,j})(u_{i,j+1}-u_{i,j})+(S^2_{i,j}+S^2_{i,j-1})(u_{i,j-1}-u_{i,j})=\\ =\mu h^2(u_{i,j}-g_{i,j})\end{gathered}$$

The discreet laplacian for a function u

$$\Delta u(x,y):=u_{xx}+u_{yy}\cong$$

$$\cong\frac{1}{h^2}\left[u_{i+1,j}+u_{i-1,j}-4u_{i,j}+u_{i,j+1}+u_{i,j-1}\right]$$

Within Ω, the second equation, using a five-point stencil, will look

$$\alpha\epsilon\left(S_{i+1,j}+S_{i-1,j}+S_{i,j+1}+S_{i,j-1}-4S_{i,j}\right)=h^2S_{i,j}|\nabla u|^2_{i,j}-h^2\frac{\alpha}{4\epsilon}(1-S_{i,j})$$

Being the problem elliptic, we use a discretization such that we get centered finite differences; therefore for 2 points stencil: let $|\nabla u|^2_{i,j}:=|\widetilde{\nabla}u|^2_{i,j}$

$$|\widetilde{\nabla}u|^2_{i,j}\cong\frac{1}{4h^2}\left((u_{i+1,j}-u_{i-1,j})^2+(u_{i,j+1}-u_{i,j-1})^2\right)$$

for nodes inside Ω. Or by other formulation by 3 points stencil: let $|\nabla u|^2_{i,j}:=|\widetilde{\widetilde{\nabla}}u|^2_{i,j}$

$$|\widetilde{\widetilde{\nabla}}u|^2_{i,j}\cong\frac{1}{4h^2}\left(\frac{(u_{i+1,j}-u_{i,j})^2+(u_{i,j}-u_{i-1,j})^2}{2}+\frac{(u_{i,j+1}-u_{i,j})^2+(u_{i,j}-u_{i,j-1})^2}{2}\right)$$

then

$$S_{i,j}=\frac{\alpha\epsilon\,\widehat{S}_{i,j}+\alpha\frac{h^2}{4\epsilon}}{4\alpha\epsilon+h^2|\nabla u|^2_{i,j}+\alpha\frac{h^2}{4\epsilon}}. \tag{2.3.1}$$

where $\widehat{S}_{i,j}=S_{i+1,j}+S_{i-1,j}+S_{i,j+1}+S_{i,j-1}$.

The discretized u become:

$$u_{i,j}=\frac{(S^2_{i,j})(u_{i+1,j}+u_{i-1,j}+u_{i,j+1}+u_{i,j-1})}{(\mu h^2+4S^2_{i,j}+S^2_{i+1,j}+S^2_{i-1,j}+S^2_{i,j+1}+S^2_{i,j-1})}+ \tag{2.3.2}$$

$$+\frac{(S^2_{i+1,j})(u_{i+1,j}) + (S^2_{i-1,j})(u_{i-1,j}) + (S^2_{i,j+1})(u_{i,j+1}) + (S^2_{i,j-1})(u_{i,j-1}) + \mu h^2 g_{i,j}}{(\mu h^2 + 4S^2_{i,j} + S^2_{i+1,j} + S^2_{i-1,j} + S^2_{i,j+1} + S^2_{i,j-1})}.$$

Boundary conditions

In this context we impose that Neumann boundary conditions are satisfied at the edge. Null derivatives at the border are generated by introducing dummy nodes along the perimeter, for j=0 and i=1,..,N with j=-1 and y=-h

$$\begin{cases} u_{i,0} - u_{i,-1} = 0 \\ \qquad\qquad\qquad i = 1, .., N-1 \\ S_{i,0} - S_{i,-1} = 0 \end{cases}$$

and in the same way for the remaining sides.
The equation is written the same way for nodes on the edge. In this case, symmetrical terms disappear in correspondence to dummy nodes introduced. A linear algebraic system with $2(N+1)^2$ unknown variables is then obtained. By fixing S first and then U, we would get a system of the type:

$$\begin{cases} AU = b \\ BS = e \end{cases}$$

We use a recurrent two-step algorithm, which uncouples the system at every iterative step and calculates the first equation, replaces it in the second one and then replaces the result back into the first equation.
In summary, an iterative process give us, at one of its step, two equations to compute the non-linear elliptic problem related to the minimum conditions proper of the discrete approximated functionals sequence which converge to fix point.

Matrix row sorting

For the equations pair an algorithm is used, which allows a transformation of the associated matrix into a vector by performing a row sorting for $U \in$

$\mathbb{R}^{N+1^2}$. Aligning rows back to back, an array: $A \in \mathcal{M}((N+1)^2,(N+1)^2;\mathbb{R})$, with PENTA-DIAGONAL structure, is obtained. A five-point cross scheme appears in respect to the variable $u_{i,j}$:

$$A = \begin{bmatrix} 0 & a_{12} & 0 & \dots & 0 & a_{1,N+1} & 0\dots 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 & \dots & 0 & a_{2,N+2} & 0 \\ 0 & a_{22} & \ddots & \ddots & 0 & \dots & 0 & a_{3,N+3} \\ \vdots & 0 & \ddots & \ddots & \ddots & & 0 & \\ 0 & \vdots & & \ddots & \ddots & \ddots & & \\ a_{N+1,1} & 0 & & & \ddots & \ddots & \ddots & \\ 0 & a_{N+2,2} & & & & & & \\ \vdots & 0 & \ddots & & & & & \\ 0 & & & & & & & \end{bmatrix}$$

The system can be solved by an iterative method.

Jacobi or Gauss-Seidel relaxation scheme

The choice of the iterative method to be adopted for the computation of the functional does not involve analytical considerations. We can equally use either one scheme or the other, the choice depending upon computational resources available and the chance to parallelize the code. Indeed, Jacobi method is directly parallelized within cycles, while Gauss-Seidel method can be parallelized at a lower level for a limited number of cycles. Nevertheless, the general interest here is just the applicability of an iterative method and its convergence. Jacobi method looks:

$$\begin{cases} AU = b \\ BS = e \end{cases}$$

$$U_k^{(n+1)} = \frac{1}{a_{k,k}} \left(- \sum_{\substack{l=1 \\ l \neq k}}^{(N+1)^2} a_{k,l} U_l^{(n)} + b_k \right) \tag{2.3.3}$$

If we consider $U_k^{(n+1)}$ the matrix of the k^th item of the functional sequence of the function $u_{i,j}$ such that $i,j = 1,..,n$ for square image, where $n+1$ is the iteration order, and set $u = g, S = 1$ starting from the initial instant Iter=0, we notice that diagonals are not null for both equations ($a_{k,k} \neq 0$); in fact positive quantities, with minus sign, remain in the sum. For the variable $u_{i,j}^{(n)}$ and $1 < i,j < N$, we find that the k-th row in the first equation is:

$$\begin{gathered} u_{i,j}^{(n+1)}(\mu h^2 + 4S_{i,j}^2 + S_{i+1,j}^2 + S_{i-1,j}^2 + S_{i,j+1}^2 + S_{i,j-1}^2 + 8K_\epsilon) = \\ +(S_{i,j}^2 + K_\epsilon)(u_{i+1,j}^{(n)} + u_{i-1,j}^{(n)} + u_{i,j+1}^{(n)} + u_{i,j-1}^{(n)}) + \\ (S_{i+1,j}^2 + K_\epsilon)(u_{i+1,j}^{(n)}) + (S_{i-1,j}^2 + K_\epsilon)(u_{i-1,j}^{(n)}) + \\ +(S_{i,j+1}^2 + K_\epsilon)(u_{i,j+1}^{(n)}) + (S_{i,j-1}^2 + K_\epsilon)(u_{i,j-1}^{(n)}) + \mu h^2 g_{i,j} \end{gathered}$$

We notice that, over the edge, one or two terms disappear from the numerator and one from the coefficient of $u_{i,j}^{(n)}$.
The following quantity will occupy the denominator

$$\mu h^2 + S_{i,j}^2 + S_{i+1,j}^2 + S_{i-1,j}^2 + S_{i,j+1}^2 + S_{i,j-1}^2 + 8K_\epsilon.$$

Moreover:

$$D_{k,k} = S_{i,j}^2 + S_{i+1,j}^2 + S_{i-1,j}^2 + S_{i,j+1}^2 + S_{i,j-1}^2 + 8K_\epsilon$$

Convergence of the numerical scheme

To establish convergence of the method we use Array Diagonal Dominance, instead of estimating spectral ray. We have to demonstrate that:

$$|a_{k,k}| > \sum_{\substack{l=1 \\ l \neq k}}^{(N+1)^2} |a_{k,l}| \tag{2.3.4}$$

, that is, the diagonal is strictly dominant, and this is sufficient for convergence.
Excluding the principal diagonal, called

$$A = \mathcal{C} - \mathcal{D} - \mu h^2 I$$

then

$$\left(\mathcal{C} - \mathcal{D} - \mu h^2 I\right) U = \mu h^2 \mathcal{G}$$

Where $\mathcal{D}$ is the coefficient of $u_{i,j}^{(n+1)}$ and occupies the denominator.
$\mathcal{C}$ given from $\sum S_{k,k}^2$ that is the four points scheme, in cross, plus $8K_\epsilon$ that is really $D_{k,k}$

$$D_{k,k} = \sum_{l \neq k} |a_{k,l}|.$$

Thus, μh^2-term assures, for the first equation, diagonal dominance and convergence of Jacobi method.
For the second equation we get,
when $\widehat{S}_{i,j}^{(n)} = S_{i+1,j} + S_{i-1,j} + S_{i,j+1} + S_{i,j-1}$ is set,

$$S_{i,j}^{(n+1)} = \frac{\alpha\epsilon\ \widehat{S}_{i,j}^{(n)} + \alpha\frac{h^2}{4\epsilon}}{4\alpha\epsilon + h^2|\nabla u|_{i,j}^2 + \alpha\frac{h^2}{4\epsilon}}. \tag{2.3.5}$$

The denominator is strictly positive with no null $a_{k,k}$-terms. Diagonal dominance is established dividing operator $\widehat{S}$ into two parts, the diagonal and the off-diagonal part with null sub-diagonal:

$$\mathcal{C} - \mathcal{M};$$

this way, we get:

$$\left(\alpha\epsilon(\mathcal{C} - \mathcal{M}) - T - \frac{\alpha h^2}{4\epsilon} I\right) S = -\frac{\alpha h^2}{4\epsilon} \mathcal{Q}$$

where $S_{i,j}|\nabla u|_{i,j}^2 = T$ is defined as diagonal array
and $\mathcal{Q} = (1, .., 1) \quad dim(N+1)^2$
We want to evaluate the $\mathcal{C} - \mathcal{M}$ matrix that discretized the laplacian and consider $\mathcal{C}$ as the only off-diagonal contribution.

$$\sum_{l \neq k} |a_{k,l}| = 4\alpha\epsilon \qquad |M|_{k,k} = 4\alpha\epsilon.$$

The arrays have positive elements, which establish, with the contribution from other terms, diagonal dominance in the strong sense. The second equation also converges.
We observe that edge terms remove elements from the diagonal, preserving diagonal dominance in the strong sense.
Several stopping criteria for iterative methods exist : iterations can be stopped by counting a certain number of steps, for example 20 or 30, or setting a distance criterium between two successive solutions, such as:

$$\max_{i,j} \mid u_{i,j}^{(n+1)} - u_{i,j}^{(n)} \mid < \delta$$

Gauss-Seidel algorithm

As mentioned above, Gauss-Seidel numerical iterative scheme is a sophistication of Jacobi method where the computation of the solution is anticipated to (N+1)-th iteration, just considering pre-computed values.

$$U_k^{(n+1)} = \frac{1}{a_{k,k}} \left(- \sum_{l=1}^{(k-1)^2} a_{k,l} U_l^{(n+1)} - \sum_{l=k+1}^{(N+1)^2} a_{k,l} U_l^{(n)} + b_k \right) \tag{2.3.6}$$

so we get:

$$u_{i,j}^{(n+1)} = \frac{1}{D_{k,k}} \left[(S_{i,j}^2 + K_\epsilon)(u_{i+1,j}^{(n)} + u_{i-1,j}^{(n)} + u_{i,j+1}^{(n)} + u_{i,j-1}^{(n)}) + \mu h^2 g_{i,j} \right] +$$

$$+ \frac{1}{D_{k,k}} \left[(S_{i+1,j}^2 + K_\epsilon)(u_{i+1,j}^{(n)}) + (S_{i-1,j}^2 + K_\epsilon)(u_{i-1,j}^{(n)}) \right] +$$

$$+ \frac{1}{D_{k,k}} \left[(S_{i,j+1}^2 + K_\epsilon)(u_{i,j+1}^{(n)}) + (S_{i,j-1}^2 + K_\epsilon)(u_{i,j-1}^{(n)}) \right].$$

where the terms $U_l^{(n+1)} = (u_{l,1}^{(n+1)}, .., u_{l,M}^{(n+1)})$ when $l < k$ and $U_l^{(n+1)} = (u_{l,1}^{(n)}, .., u_{l,M}^{(n)})$ when $l \geqslant k$ and M is the height of the image i.e. the size of column. and the same for the values of discretized S

$$S_{i,j}^{(n+1)} = \frac{\alpha\epsilon \ \widehat{S}_{i,j}^{(n)} + \alpha \frac{h^2}{4\epsilon}}{4\alpha\epsilon + h^2 |\nabla u|_{i,j}^2 + \alpha \frac{h^2}{4\epsilon}}. \tag{2.3.7}$$

which represents the numerical scheme for this method, which, computational resources being equal, converges more quickly than Jacobi scheme.

2.3.2 Criteria for parameter choice

In the above numerical schemes some parameters are chosen conveniently in order to make correct approximations stand true. The choice of ϵ-parameter involves the amplitude of the tubular neighborhood; the "throat", used in determination of convergence, varies according to image type and values between $2.0 \cdot 10^{-3}$ and $2.0 \cdot 10^{-4}$ are quite reasonable, in general.
Image rescaling brings to larger gradient values, which force to modify both μ- and α-parameter. Due to Γ-convergence, the "throat" consists of two exponential branches and the function u does not change.
Just as a hint, we say the inequality of the Γ lim-sup is always greater than every other choice and it results squeezing under that, so we can state that the sequence which is computed numerically is the best in case of Γ-convergence.
Numerical stability requires the grid step size to be equal to the resolution of given image, which is a common choice in many of Image Processing methods. The grid must sample correctly the exponential branch, that is: $\exp(-\frac{t-b_\epsilon}{2\epsilon})$ and the exponential dies out for about 10ϵ(20ϵ in the presence of two branches), therefore we set $20\epsilon = mh$,with m grid steps.
For high m, large ϵ-values and large tubular neighborhoods are found. A minimum choice is $m = 1 \Rightarrow h = 20\epsilon$; in this case, the "throat" is sampled with just two nodes at terminal points and under these threshold values contours are lost.

2.3.3 Comment to the code

Let us consider the functional as approximated starting from the numerical code

$$F(u) = \int_\Omega (u-g)^2 + \lambda \int_\Omega \mid \nabla u \mid^2 dx \quad + \alpha\, \mathcal{H}^1(K) \qquad \frac{1}{\lambda} \to \mu$$

let's transform it into:

$$\widetilde{F}(u) = \frac{1}{\lambda}\int_\Omega (u-g)^2 + \int_\Omega \mid \nabla u \mid^2 dx \quad + \frac{\alpha}{\lambda}\, \mathcal{H}^1(K) \qquad \frac{\alpha}{\lambda} \to \alpha$$

Let us set $20\epsilon = mh$, from which $\rho = \frac{h}{\epsilon} = \frac{20}{m}$. K_ϵ-parameter is present in the energy term $\int (S^2 + K_\epsilon)|\nabla u|^2$ of Euler equation, so we get $div((S^2 + K_\epsilon)\nabla u$, and preserves Euler equation from degeneration and makes the M-S functional coercive in Sobolev spaces. According to March ([28]), we must choose $K_\epsilon = o(\epsilon)$; in practice, choosing $K_\epsilon = \frac{1}{10}\epsilon$ proves sufficient. By this method, image processing produces out a regularized image U and a jump set S. We will see, in the next chapter, how the choice of both λ- and α-parameter implies a different selection of contours and a difference between regularized images.

Chapter 3

Applicability of methods and simulation results

Recent scientific literature suggests several methods for recognition of ventricular area, but most of them are focused on semi-curve evolution from two fixed points, which is only suitable for single images. The method adopted here makes use, instead, of an ellipse as the initial profile, with no fixed points. This choice is justified by the chance of processing a multi-image sequence in order to derive the needed area at every frame. The above-mentioned protocol consists of applying segmentation models and methods to a rhythmically moving object.

We focus on the left ventricle of cardiac muscle since, in some clinical cases, the volume of blood introduced into the arterial net by the ventricle (estimated as its ejection fraction) has to be determined. This low-invasive technique gives us a chance to identify the area delimited by ventricular walls over time. Also expulsion time can be assessed by means of our technique, which proves useful to make new analytical hypotheses about blood internal pressure.

Ventricle internal area can be determined by curve-evolution from an as-

signed position by means of Hamilton-Jacobi PDE model. Other models, which make use of fixed points near to the atrial valve of the muscle, do not prove suitable to treat image sequences because would force to manually draw fixed points on every image in the sequence, due to variability in probe position and muscle movements. We assign, instead, an initial elliptic profile to the curve, centered in the frame.
We apply here a preprocessing method for frames in order to emphasize ventricle contours. Mumford-Shah functional minimization technique has been chosen due to its processing speed and sophistication, which make it suitable for reconstruction of images and several other real-life applications [2], [14], [15], [16], [17], [19], [20].

3.1 Applicability of the approximation methods

As shown above[27], the applicability of the evolutionary method based on curve evolution is conditioned, in the presence of discontinuous data, by restrictive hypotheses the punctual dependence of the speed term. Then the existence of the spatial gradient of the brightness-intensity function requests the convolution, in Ω-domain, with a standard mollifier. In the approximation of the functional, the sequence of functional minima are iterated by the system by Ambrosio Tortorelli method. The technique to fix every equation arise to a quasi-linear class of elliptic systems explained in the paper of Birindelli and Finzi Vita [10]. At every iteration we obtain a regularized pair of approximated functions dependent from ϵ. This suggests to leave Gaussian regularization aside, with smaller loss of details, using the function of the K_ϵ item of the iteration sequence as the brightness-intensity function for

curve evolution. u-function can directly be used in the eikonal eq. for front evolution. This technique is used in [17], but optical flow theory is left aside (see [6]); This choice provides the needed regularity in the fracture (see [26]) that represents the critical points of brightness-intensity function g_I of the given image.
A further step of technique refinement, in order to make ventricular walls get emphasized, consists, as mentioned above, of making the second term of the functional dependent of the gradient over the time [15].
The square gradient is point by point calculated over function u as the difference in brightness intensity between the preceding and the following frame. Globally, the variability of this term is mostly due to movements of ventricular walls. In the presence of a continuous movement in the same "scene", we are in condition to assume regularity for function g_I over time.

3.1.1 Regularity of the speed term in the eikonal equation

The method of eikonal eq. assures solvability in the presence of discontinuous data by convolution with a regularizing operator to control the gradient explosion of the brightness-intensity function $I(x)$ in the speed term of the eikonal equation 1.1.1 and the lipschitz continuity of the initial data $u_0(x)$.

$$v(x) = \frac{1}{1 + |\nabla I(x)|},$$

instead of the gaussian convolution

$$v(x) = v(|\nabla(G_\sigma * I(x))|) = \frac{1}{1 + |\nabla(G_\sigma * I(x))|}.$$

As mentioned above, for a Hamiltonian $H(x, \nabla u)$, when the element $u_h(x)$ represents the solution at every step of the elliptic system, the following proposition stands true:

Prop. If u_{k_ϵ} represents a solution at k^{th} iteration of the Ambrosio-Tortorelli sequence ([5]) given from alternate solution of the elliptic system for fixed number of iteration, then u_{k_ϵ} is enough smooth to calculate $|\nabla u_{k_\epsilon}(x)|$.

Indeed, in the internal points of the domain Ω the amplitude of the fracture is ϵ's proportional, then we can found, at every step of the iterative solution of AT_ϵ algorithm, a costant C such that

$$|\nabla u_{k_\epsilon}(x)| \leq \frac{C}{\epsilon}$$

then

$$v(x) = \frac{1}{1+|\nabla u_{k_\epsilon}(x)|} \geq \frac{1}{1+\frac{C}{\epsilon}} > 0,$$

we can observe that the function u_{k_ϵ} has in every direction passing to the internal point $x \in \Omega$, a profile regularized by a C^2 arcs in the fracture as we can see in [26] as explained in the figure 3.1.1. For the lipschitz continuity needed for the initial condition for the eikonal equation, we choose a regular ellipse centered in the ventricular cavity.

From a computational point of view the discretized image has bounded intensity profile and the difference beetween two neighbor points is at most:

$$\frac{max(I(x_1)) - min(I(x_2))}{|x_1 - x_2|} \leq \frac{256}{1} \quad \forall x_1, x_2 \in \Omega$$

for [0,..,255] gray levels images and spatial mesh-grids equal to the image definition.

3.1.2 Convergence and Existence

The extension of existence results for the minimum for the M-S functional, when function u is time-dependent, is currently in progress. Nevertheless, the basic idea consists of considering the fluid change that would ideally result

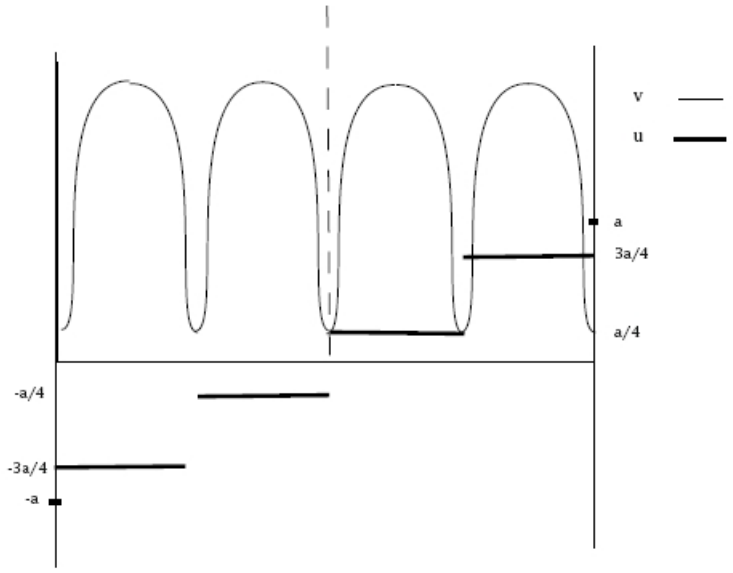


Figure 3.1.1.a: Approximation of the one dimensional fracture

from a continuous movie of some cardiac movement. We try here to reconstruct, from this point of view, the demonstrative steps mentioned above. We are brought to conjecture a tubular neighborhood, on which is based, the Γ-convergence is interpretated as it is in movement; then, for a sufficiently small time, we could imagine that the regularity of u over time is such to assure convergence of the functional sequence to the minimum. Anyway, the analytical problem is still open and, therefore, only some conjectures are allowed.

In the first tests as to the numerical approximation, for the temporal and spatial gradient calculus, we use the formulation we use the two-point following formula:

$$(|\nabla u|_{i,j}^{(f)})^2 = \frac{1}{4h^2}\left((u_{i+1,j} - u_{i-1,j})^2 + (u_{i,j+1} - u_{i,j-1})^2 + (u_{i,j}^{(f-1)} - u_{i,j}^{(f+1)})^2\right)$$

It has a quite like good performance on the ventricular walls, but a successive application with tree-point stancil, suggested by Professor Stanley Osher evidence a more reliable definition of the pixel, especially in the time dependent

elaboration:

$$(|\nabla u|_{i,j}^{(f)})^2 = \frac{1}{4h^2}\left(\frac{(u_{i+1,j}-u_{i,j})^2+(u_{i,j}-u_{i-1,j})^2}{2}\right)+$$

$$+\frac{1}{4h^2}\left(\frac{(u_{i,j+1}-u_{i,j})^2+(u_{i,j}-u_{i,j-1})^2}{2}\right)+$$

$$+\frac{1}{4h^2}\left(\frac{(u_{i,j}^{(f-1)}-u_{i,j}^{(f)})^2+(u_{i,j}^{(f)}-u_{i,j}^{(f-1)})^2}{2}\right).$$

This formulation has better results and is adopted to all tests.

3.2 Echocardiographic movie format

Modern instruments used in hospitals, as well as portable echocardiographic instruments, usually allow data exportation in various video formats. The method adopted here makes use of movies in AVI (Audio Video Interleave) format; a typical example of AVI-file characteristics follows:

Filename	'avifilename1.avi'	'avifilename2.avi'
FileSize	134158336	19728384
FileModDate	' 25-Jun-2007 14:09:10'	' 05-Oct-2007 10:06:14'
NumFrames	1602	1269
FramesPerSecond	27	26
Width	800	800
Height	600	652
ImageType	'truecolor'	'truecolor'
VideoCompression	'Cinepak'	'MP42'
Quality	0	0
NumColormapEntries	0	0

Data files can be exported and read in DICOM (Digital Imaging and Communication in Medicines) format, the world standard for all medical applications dealing with patient data and further detailed information. The standard DICOM is composed of different parts and storage levels, suitable for data exchange in the medical context.

At the present level of work, we opt for AVI format in performing image extraction, since, being supported by various operating systems with public licenses, it allows an easiest manipulation of movies. Esaote MyLab30 instrument, currently used by the team lead by Prof. Mario Curione in the Department of Clinical Sciences of "Umberto I" University Hospital, when interfaced with a PC supporting Windows o.s., exports data in different image compression formats, in terms of size and sampling frequency. A single frame looks like:

In particular, sampling frequency (FramesPerSecond:26, 27), allows medical researchers to compare data from areas with the Electro-Cardio-Graphic layout. We shortly mention here the computational protocol that handles the two phases, M-S Preprocessing and H-J Processing of images, which makes use of the well-roundness of a MatLab application in the preprocessing stage and of a FORTRAN code with better performance on single frames during processing.

Being the elaboration system implemented on a Linux system, a format conversion algorithm with no video compression is needed (compressed formats are licensed). This involves a further step in preprocessing of the entire file.

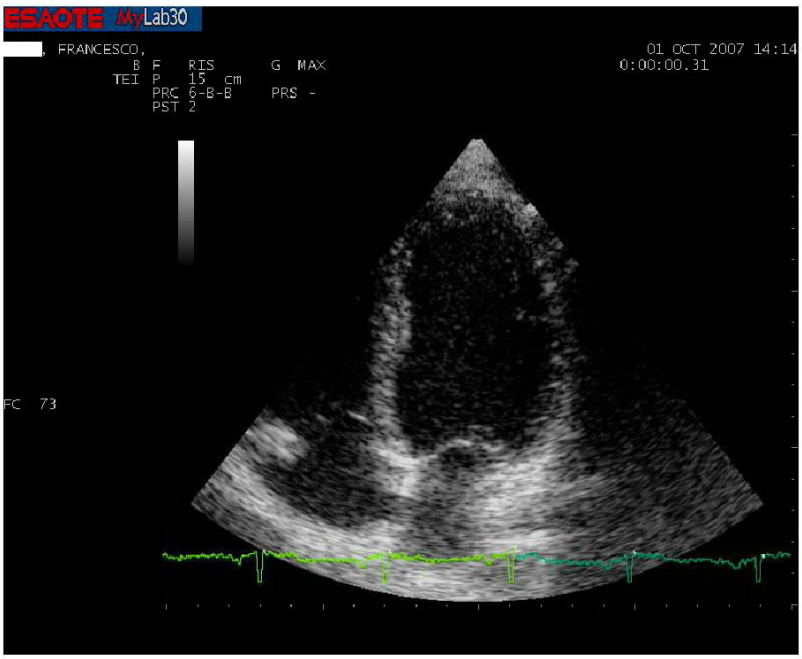


Figure 3.2.0.b: Entire echo image frame

3.3 Frame adaptation

The algorithm selects the part of the image to be cropped ("cropping" step); see Figure 3.3.0.c

After gray levels are inverted, brightness intensity difference gets positive in the presence of edges.

This way we obtain the image of the selected frame (Figure 3.3.0.d), to be later preprocessed using M-S algorithm with various parameter choices.

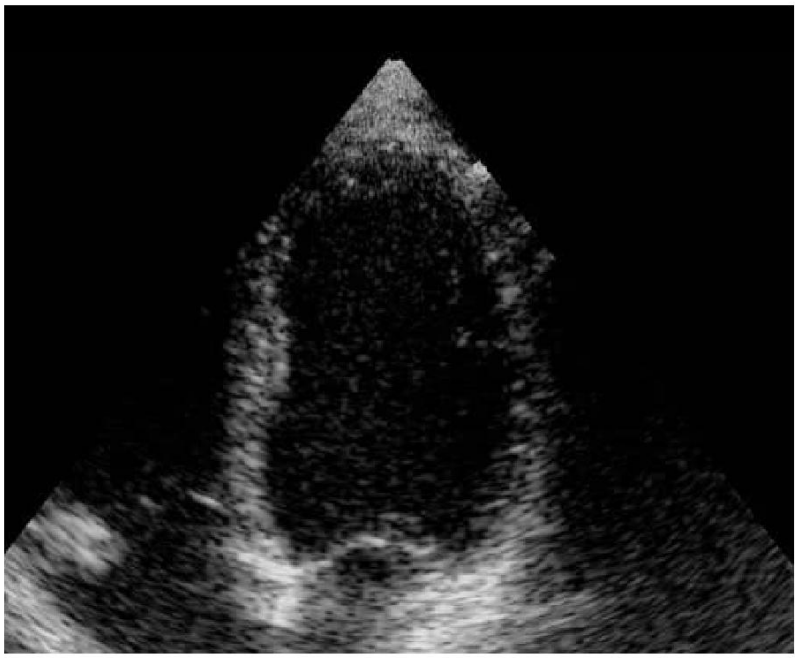


Figure 3.3.0.c: Cropped frame

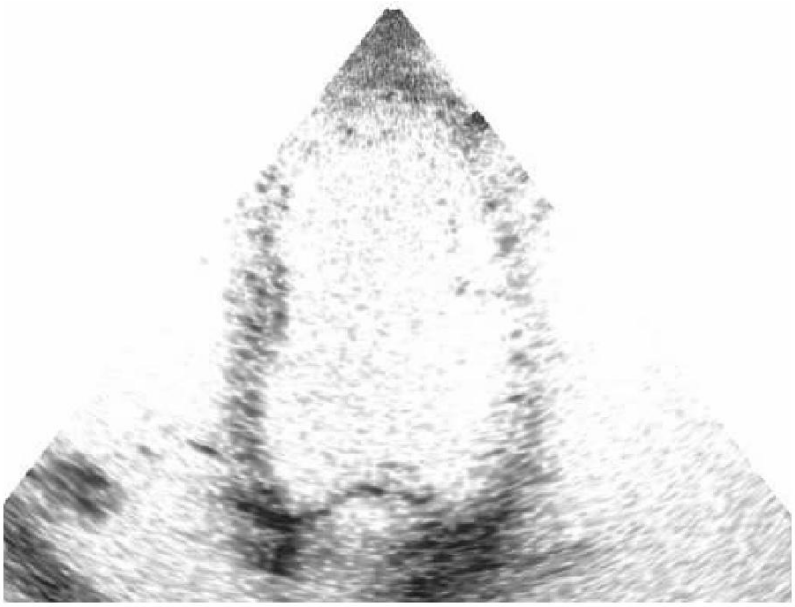


Figure 3.3.0.d: Inverse cropped frame

3.3.1 Presence of edges for H-J PDE

Preprocessed images are then uniformly distributed within the interval [0,1], in terms of gray-tone intensity (double float numerical representation). Curve evolution starts from an initial elliptic configuration u_0, centered in the frame at t=0, and goes on along the external normal direction, with speed proportional to the brightness-intensity gradient of the image at every pixel. The computation of the gradient is corrected by means of a convolution with a regularizing operator (standard Mollifier). Ref. [27], [33], [34]
The regularization is made by the diffusion heat equation, this operation is essential for the gradient evaluation, but the image lose important details

even with a limited number of iterations, therefore the curve evolution does not found a gradient value to determine its velocity shutdown [17].
To highlight the problem we show in figure 3.3.1.e the evolution in a frame on which the Gaussian smoothing was repeated for 10 iterations. Clearly, curve

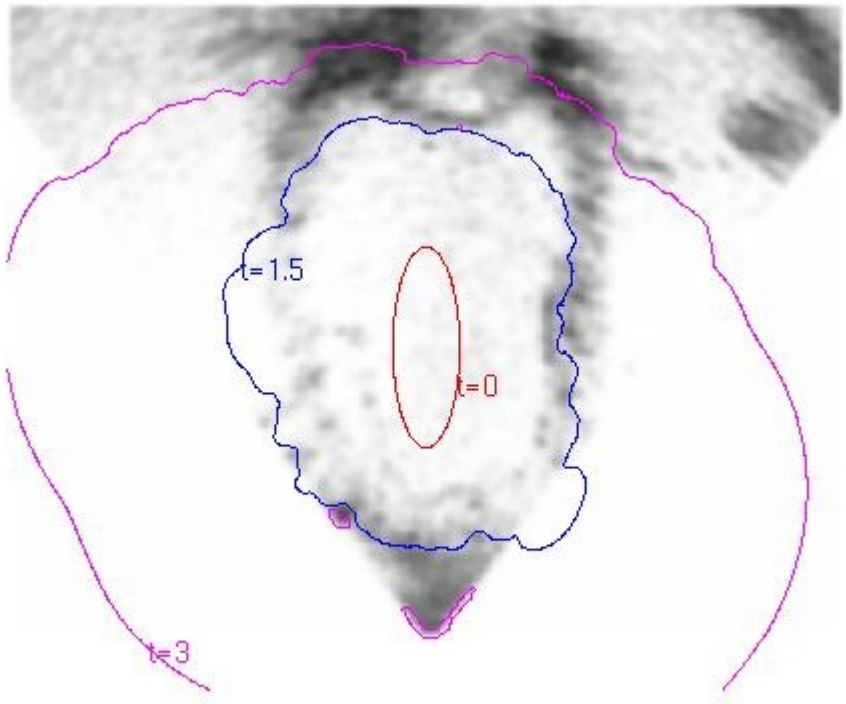


Figure 3.3.1.e: Curve evolved on 10 iteration Gaussian smoothing

evolution is not able to find a gradient value that, in respect to a stopping threshold th=0.125, can cause its stop. Even with a single Gaussian step, the gradient is not able, in some regions of the ventricular contour, to stop curve evolution (see Figure 3.3.1.f).

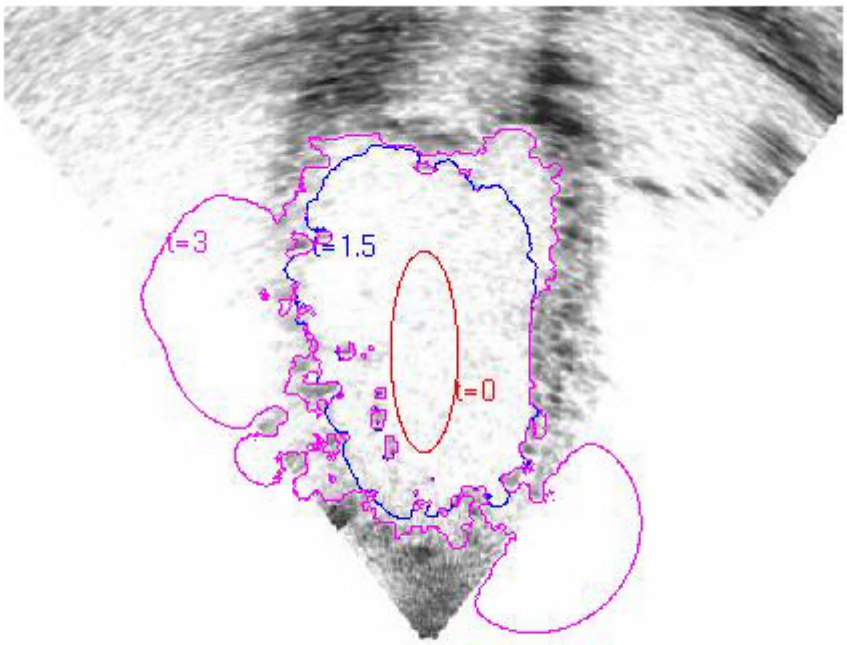


Figure 3.3.1.f: Curve evolved on 1 iteration Gaussian smoothing frame

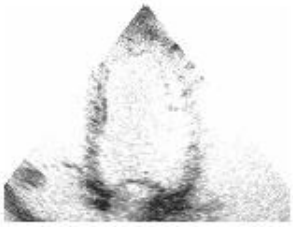
3.3.2 Preprocessing by the M-S functional

As previously stated, segmentation by solution of H-J equation fails when the probed tissue gives back a small signal represented by a small brightness-intensity gradient. Brightness-intensity gradients along ventricle edges (that is, edge info) could happen not to be visible in all regions of the given frame; in general, small signals come are generated by pixels along ventricular walls, if compared to image noise. Therefore, we decide here to improve the elaboration protocol by introducing a preprocessing phase which makes use of the M-S method. Let us illustrate our attempt to apply the M-S model, whose efficiency strongly depends on the choice of parameters λ, α (see [28]). This is particularly true for echo-graphic images, probably most due to the density of signal detector in the probe, which causes a limited spatial definition. Consequently, images from each sequence frame show very discontinuous ventricular contours.

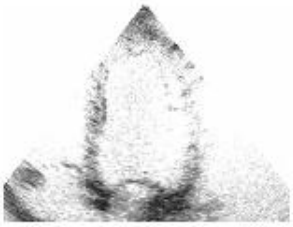
Parameters involved in the first step of M-S application

λ-parameter is directly proportional to smoothness and regularity of u-functions. On the other hand, α-parameter is directly proportional to the size of the selected region along the edge.
The presence of discontinuous and localized signal has suggested us to apply, in a first step, the M-S method to emphasize image details, choosing parameter values which favored small edges (therefore, a small λ and a small α). Let us look to some results:

Variation of λ-parameter at fixed α:



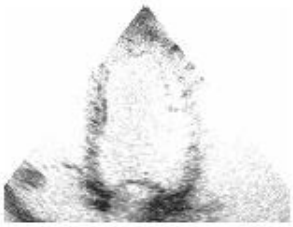
Orig frame



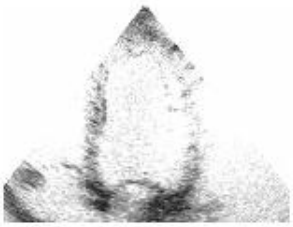
U 1st step

S 1st step

parameter:
λ =0.02
α =1e-05
$\frac{h}{\epsilon}$ =20
k_ϵ =2e-06



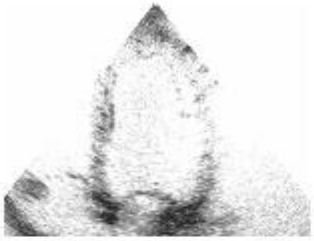
Orig frame



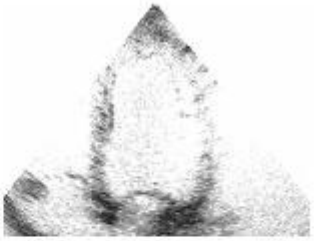
U 1st step

S 1st step

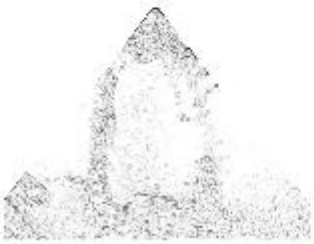
parameter:
λ =0.8
α =1e-05
$\frac{h}{\epsilon}$ =20
k_ϵ =2e-06



Orig frame

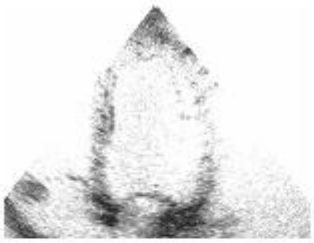


U 1st step

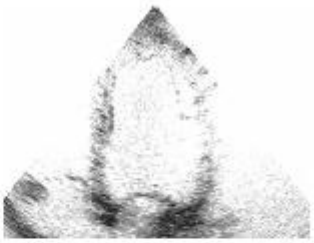


S 1st step

parameter:
λ =0.02
α =0.0001
$\frac{h}{\epsilon}$ =20
k_ϵ =2e-06



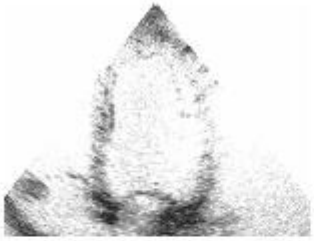
Orig frame



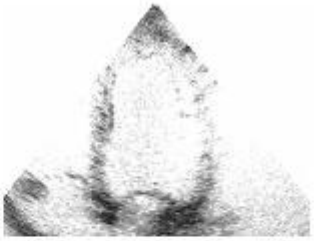
U 1st step

S 1st step

parameter:
λ =0.8
α =0.0001
$\frac{h}{\epsilon}$ =20
k_ϵ =2e-06



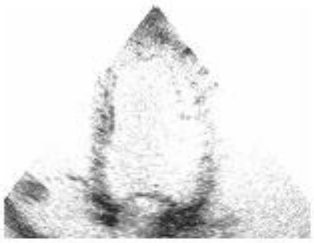
Orig frame



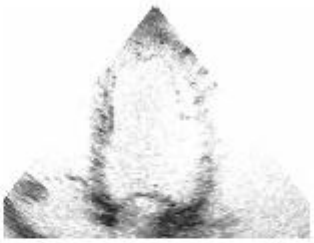
U 1st step

S 1st step

parameter:
λ =0.02
α =0.001
$\frac{h}{\epsilon}$ =20
k_ϵ =2e-06



Orig frame



U 1st step

S 1st step

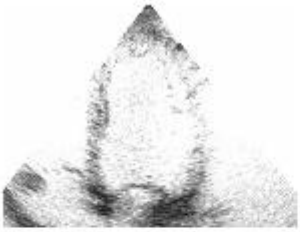
parameter:
λ =0.8
α =0.001
$\frac{h}{\epsilon}$ =20
k_ϵ =2e-06

Parameter choice proves very important in order to balance selection of image contours in respect to background noise. When small parameter values are chosen, small areas are privileged, thus we can enhance local peculiarities of Echo-Graphic images.
When parameter values are large, ventricle walls are emphasized and larger areas are selected. This suggests, as mentioned above, to perform an iterated search for edges in the preprocessing stage, in order to select small areas first and, subsequently, to reconstruct, starting from previous selection, more regular edges by means of parameters value that privilege smoother and longer contours. Therefore, we choice the following intervals: $(0.02 - 0.8)$ for λ and $(0.001 - 0.000001)$ for α.

Parameters involved in the second step of M-S application

After treating images by the above-mentioned choice of M-S parameters, we now go on to treat them again using the same method, but different values. In substance, the procedure is repeated over the jump subset S; this produces the approximation of those parts of the image which are identified as local areas, while the search for longer contours and wider areas is made with the aid of suitable parameters. This way, edges get more evident and continuous, allowing to limit the evolution of H-J model more effectively by using an optimal threshold value. In the course of the second step of M-S application, after some tests, parameters values are taken such that only a direct verification of the out coming images can be used as a measure of goodness of results (in a few words, the method requests to set parameters "with a screwdriver"). Repeated M-S algorithm.
The following paired lines of images report results from the second application of M-S method, in relation to parameter values appearing in image headings.

λ = 0.8



U 1st step

λ = 50

U 2nd step

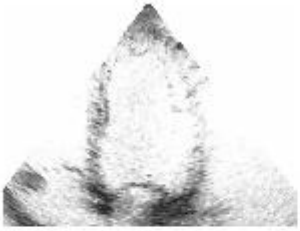
α = 0.0001

S 1st step

α = 0.05

S 2nd step

λ = 0.8

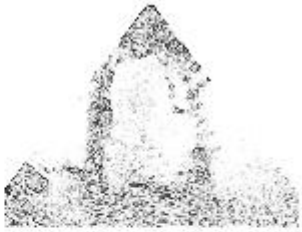


U 1st step

λ = 50

U 2nd step

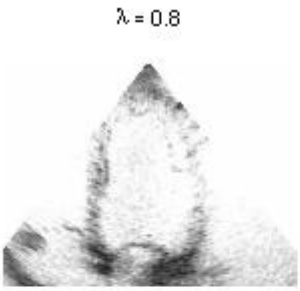
α = 0.001



S 1st step

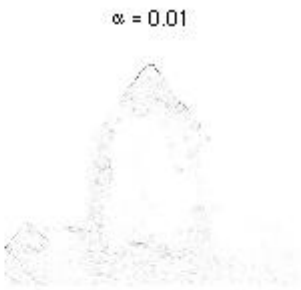
α = 0.05

S 2nd step



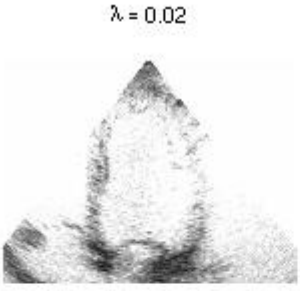
U 1st step
λ = 50

U 2nd step



S 1st step
α = 0.05

S 2nd step



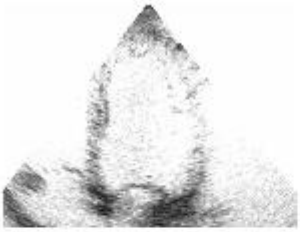
U 1st step
λ = 50

U 2nd step

S 1st step
α = 0.05

S 2nd step

λ = 0.02



U 1st step

λ = 50

U 2nd step

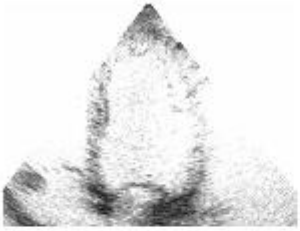
α = 0.0001

S 1st step

α = 0.05

S 2nd step

λ = 0.02



U 1st step

λ = 50

U 2nd step

α = 1e-05

S 1st step

α = 0.05

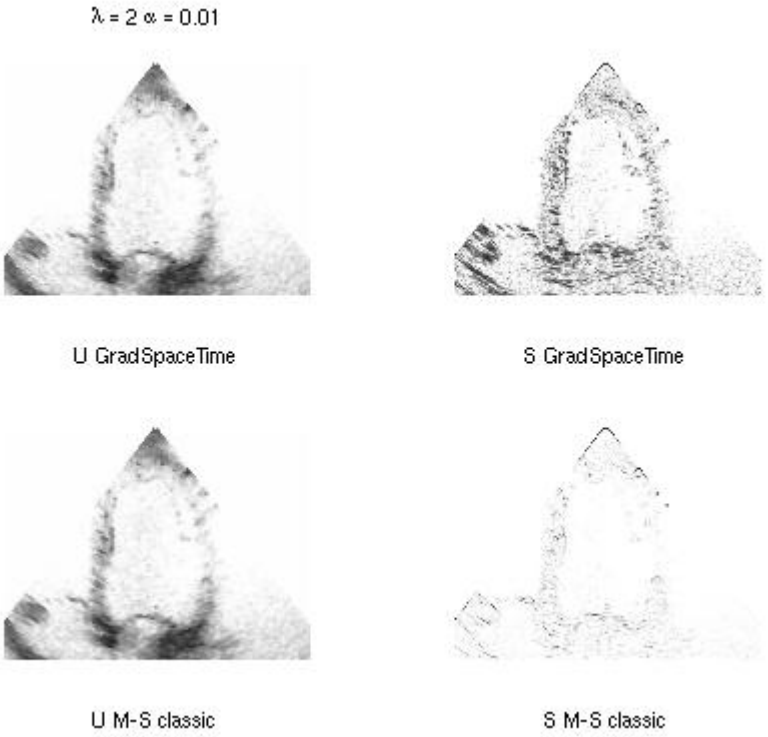
S 2nd step

Time dependence

Good results are obtained with prudent choices of parameters, but the presence of some interruptions of the ventricular profile walls results a big obstacle, to the right evolution of the curve, modeled by eikonal eq., avoiding the evolution outside the ventricular border.
Having a whole sequence of consecutive images at our disposal, we make here a sophistication of our scheme in order to enhance moving parts of ventricular walls. Those pixels, which, when comparing two consecutive images, show a variability in terms of brightness intensity, represent moving parts ventricular walls. To this aim, we put dependence on time gradient into the functional, calculated between the frame preceding the one in elaboration and the following.
As to the numerical code, various simplifications have been made in order to directly insert time dependence ∇u_t into the computation of the spatial gradient ∇u . This all has to be demonstrated in analytical terms.
The contribution ∇u_t is inserted into the computation of the gradient of function u at every spatial pixel $(x, y, \bar{t})$.
In the context of finite-difference numerical scheme with time step (frame to frame), the natural choice consists of centering the current frame in respect to those adjacent. Then, for every frame f:

$$(|\nabla u|_{i,j}^{(f)})^2 = \frac{1}{4h^2}\left((u_{i+1,j} - u_{i-1,j})^2 + (u_{i,j+1} - u_{i,j-1})^2 + (u_{i,j}^{(f-1)} - u_{i,j}^{(f+1)})^2\right);$$

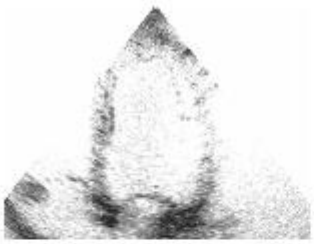
with this change gradient value is increased, as well as the minimum, as testified by larger presence of contours. In general, the jump set S finds more contours, mainly located where, proportionally, a greater variability of brightness intensity is present. The practical comparison reveals the emphatization of mobile profiles of ventricular walls. Some examples, with different choices of parameters, are shown below.



where the resulting U , regularized by the M-S method, appears at the left of the figure, while, at the right side, we can see the difference in the selection of the edges in comparison to the classical algorithm.

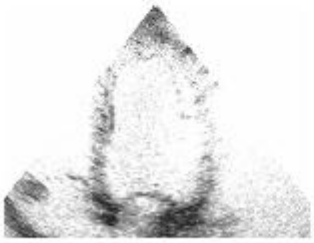
We can appreciate how edge portions not selected by the classical method are now emphasized. We notice that less mobile parts of the image - for example, pixels near to the vertex of the echographic cone - present a decrease in selection, thus hypothetically proving less representative of moving edges; on the contrary, ventricle walls get emphasized, since characterized by greater time variability in its brightness-intensity square gradient. Different choices of parameters are reported for comparison:

λ = 0.02 α = 1e-05



U GradSpaceTime

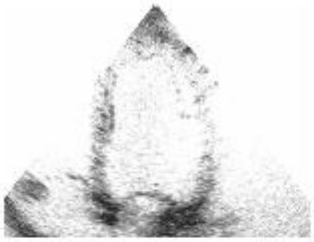
S GradSpaceTime



U M-S classic

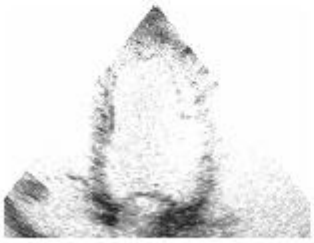
S M-S classic

λ = 0.2 α = 0.0001



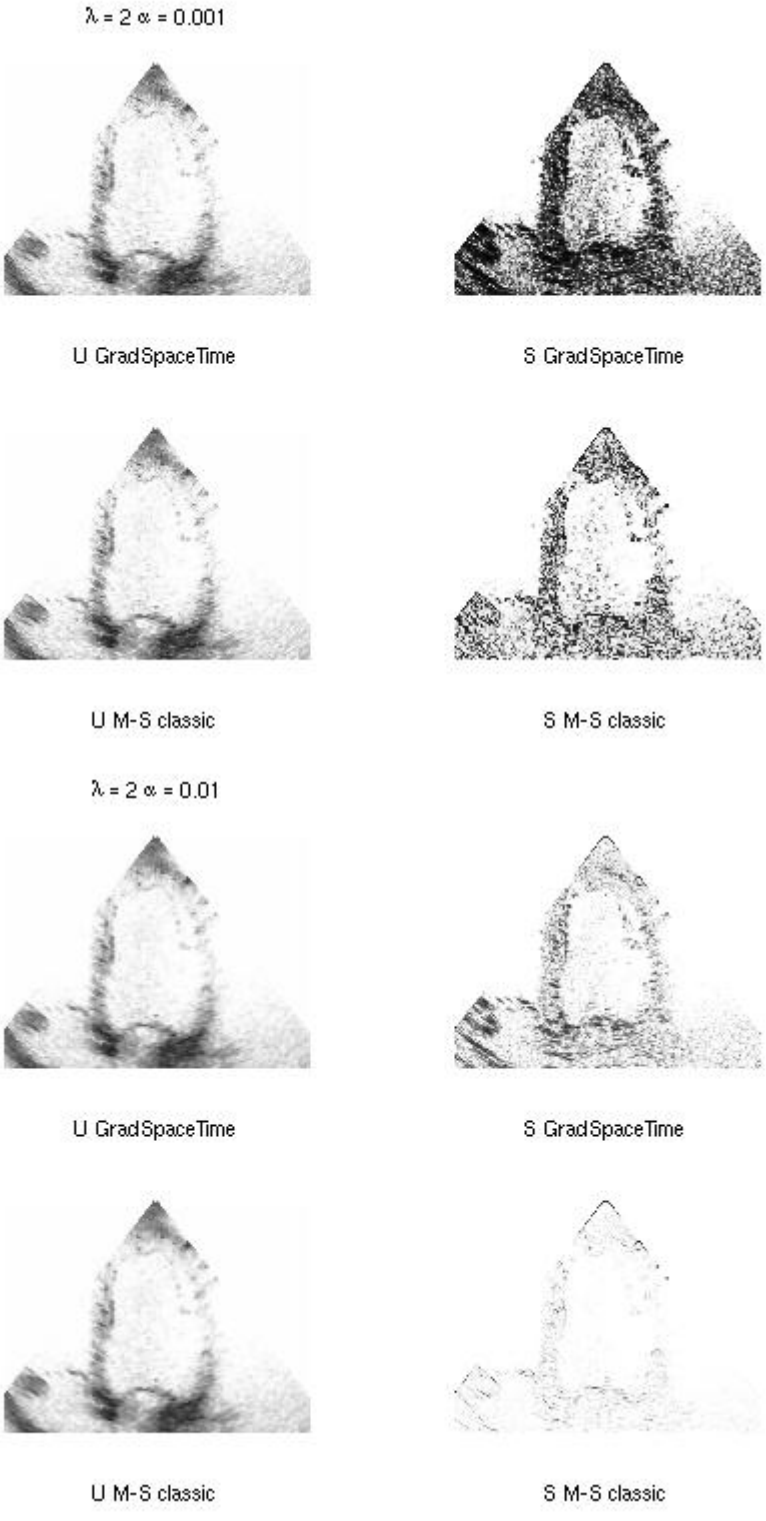
U GradSpaceTime

S GradSpaceTime



U M-S classic

S M-S classic


λ = 2 α = 0.001
U GradSpaceTime
S GradSpaceTime
U M-S classic
S M-S classic
λ = 2 α = 0.01
U GradSpaceTime
S GradSpaceTime
U M-S classic
S M-S classic

3.3.3 Algorithm code

The function performing the calculus of minima, with a term that depends on the time gradient, is a coefficient appended to the space gradient. We furnish, in input, to function the parameters and tree image-frame.

```
function [U,S] = ambrt(m,n,lamb,alpha,p,keps,iter,A0,A1,A2)

** AMBRT Ambrosio and Tortorelli's approximation of Mumford-Shah functional

** Conversion of the image frame and initialization of U and S

    G0 = double(A1);

    G1 = double(A0);

    G2 = double(A2);

    U = G0;

    U1= G1;

    U2= G2;

    S = ones(m,n);

** Solution by nonlinear Gauss-Seidel method

    for k = 1:iter

** Neumann boundary conditions

      U(1:m,1)=U(1:m,2);

      U(1:m,n)=U(1:m,n-1);

      U(1,1:n)=U(2,1:n);

      U(m,1:n)=U(m-1,1:n);

      S(1:m,1)=S(1:m,2);

      S(1:m,n)=S(1:m,n-1);

      S(1,1:n)=S(2,1:n);

      S(m,1:n)=S(m-1,1:n);

      for i = 2:m-1

        for j =2:n-1

**Computation of U
```

```
        U(i,j)=(lamb*((S(i+1,j)^2+keps)*U(i+1,j)+(S(i-1,j)^2+keps)*U(i-1,j)+...
        +(S(i,j+1)^2+keps)*U(i,j+1)+(S(i,j-1)^2+keps)*U(i,j-1))...
        (S(i,j)^2+keps)*(U(i+1,j)+U(i-1,j)+U(i,j-1)+U(i,j+1)))+G0(i,j)/...
        (1.+lamb*(4.*S(i,j)^2+S(i+1,j)^2+S(i-1,j)^2+S(i,j+1)^2+S(i,j-1)^2+8.*keps));
**Computation of grad(U)
         Ux=(U(i+1,j)-U(i-1,j))/2.;
         Uy=(U(i,j+1)-U(i,j-1))/2.;
         Ut=(U1(i,j)-U2(i,j))/2.; temporal gradient term
**New Computation of the grad(U)
         Ux=((U(i+1,j)-U(i,j))^2+(U(i,j)-U(i-1,j))^2)/2.;
         Uy=((U(i,j+1)-U(i,j))^2+(U(i,j)-U(i,j-1))^2)/2.;
         Ut=((U1(i,j)-U(i,j))^2+(U2(i,j)-U(i,j))^2)/2.; temporal gradient term
** Computation of S
         S(i,j) = (4.*(S(i+1,j)+S(i-1,j)+S(i,j+1)+S(i,j-1))+p^2)/ ...
         (p^2+16.+4.*lamb*p*(Ux^2+Uy^2+Ut^2)/alpha);
       end
     end
   end.
```

3.4 Code implementation and experimental results

The methods exposed for image-sequence movies numerically calculate the level-zero surface, which subtends the ventricular area in an optimal way according to the selected criterion. Area profiles represent an indicator of ventricular filling over time. The area of the evolved curve can be estimated in terms of number of internal pixels.

Iterating this procedure for a certain number of frames, we can estimate how area values give the variability in the periodic rhythmic ventricular movement, which is useful in medical diagnostics.
The elaboration protocol handles frames, performs the preprocessing step, lets fronts evolve and allows to appraise the change in areas over time, with a degree of definition that is almost equivalent to sampling ability of the human eye.
Sampling depends on the characteristics of the video format used for exportation by the system that controls EchoCardioGraphic acquisition. A time definition of 26-27 frames per second is common. This allows to get profiles with a resolution of 26^{-1}- 27^{-1} seconds.

3.4.1 Algorithmic efficiency and parallel calculus

To determine the efficiency of the computational system, it is necessary to control the algorithm components and their dependencies on the method, the implementation of the code and the possibility of parallelization, either direct or optimized within code compilation.

Something on computational complexity of methods exposed

We will not enter into the details of method complexity, which can be retrieved in scientific literature; nevertheless, we are able to make some conjecture to assess its order. If the pixel grid is assumed square, of size N x N=N^2, in every frame, the method for computation of the minimum of the M-S functional must iteratively solve a penta-diagonal array of size N^2x N^2 for a number of iterations about ten; therefore, we get a complexity of $o(N^4)$ per iteration.
Thus, in the case of sparse arrays, a complexity of about $o(N^2)$ is considered at each iteration. Using different implementations the number of iterations

can be reduced.

In terms of computational cost, the algorithmic complexity for the solution of eikonal equation in H-J model, resolves, at every time step, l smoothing iterations with cost N^2, calculates the gradient in N^2 computational steps and makes a function convolution in N^2 steps. Thus, every iteration involves a cost which can be approximated by $T \cdot o(N^2)$. All steps are performed on full arrays containing floating-point numbers.

Changes in calculus of the minimum of the M-S functional, translated into the numerical scheme, is a simplified algorithm prove useful for border enhancing.

It can be constructed as follows. Let us write U- and S-array into a single vector

$$(U, S) = X.$$

Let the discrete system be

$$f(X) = 0 \quad \text{with} \quad X \in \mathbb{R}^{2(N+1)^2},$$

with map $f : \mathbb{R}^{2(N+1)^2} \longrightarrow \mathbb{R}^{2(N+1)^2}$

$$x_k^{n+1} = x_k^n - \frac{f_k(x^{(n)})}{(\frac{\partial f_k}{\partial x_k})_{x(n)}}$$

the relaxation scheme with Jacobian at the denominator, becomes an iterative method for non-linear systems. If we verify that:

$$f(x) = \mathcal{A}x - b \equiv \text{JACOBI}$$

then the simplification of the method will look as follows:

$$x_k^{n+1}(x) = x_k^n(x) - J^{-1}(x^{(n)}) f(x^{(n)})$$

with J^{-1}, Jacobian matrix of punctual f ,such that off-diagonal elements are cleared out.

This means that, with a single step, the method for uncoupled equations is implemented; possible changes are introduced considering x_k's already calculated by Gauss-Seidel scheme applied to non-linear method.
Method convergence is not analytically insurable; nevertheless, in elaboration of images the solution allows a direct verification of results.
The introduction of the term dependent of the time gradient makes the choice of relaxation algorithm reasonable in view of the non-linearity of the elliptic system. Thus, the maximum estimation of code complexity is $o(N^2)$-order, which, in the case of sparse arrays, decreases to $o(N)$.

Cpu time and OpenMP parallel computing

We use different hardware platforms and software applications for our calculus.
The preprocessing step is performed over: PIII 800 Mhz, 800 MByte Ram, supporting Windows 2000 Server, Matlab R14; PIV 2000 Mhz, 1GByte Ram, supporting Linux RedHat Fedora core 2, Matlab R14b and Fortran90; CASPUR Cluster Power5, including nodes of type IBM System p5 575, 8 CPU, 1.9 GHz, 4x32 GB Ram, supporting AIX Fortran90. Preprocessing computational times are about one second quite platform-independent.
Front evolution, using the semi-Lagrangian method, requests times of the order of 300 seconds on each frame, that is, at least two orders of magnitude larger than the case of minimization of the functional.

CPU Table	Power5	PIV	PIII 800 Mhz
H-J F90 OpenMP	239.8 sec.	288.0 sec.	312.0 sec.
M-S Matlab		1.06 sec.	
M-S gradt(2 point) 3 frame		1.34 sec.	
M-S gradt(3 point) 3 frame		1.61 sec.	
Total matlab for 6 fig to H-J		5.08 sec.	

In order to save computer time, possible refinements can be performed by structuring a mesh containing finite triangular elements, which proves useful to perform domain cropping delimited on the circular sector that is typical of the echographic cone. Mesh refinement techniques are in progress; their application would improve performances, but implies interpolation of data, which often cause accumulation of numerical errors. Further improvements are, therefore, possible for the above-mentioned methods.

3.5 Results from front evolution for various types of PreProcessing methods

The block diagram summarizing the protocol is reported in the above figure. Next sections explains the different kinds of preprocessing and the consequent selected area in frame.

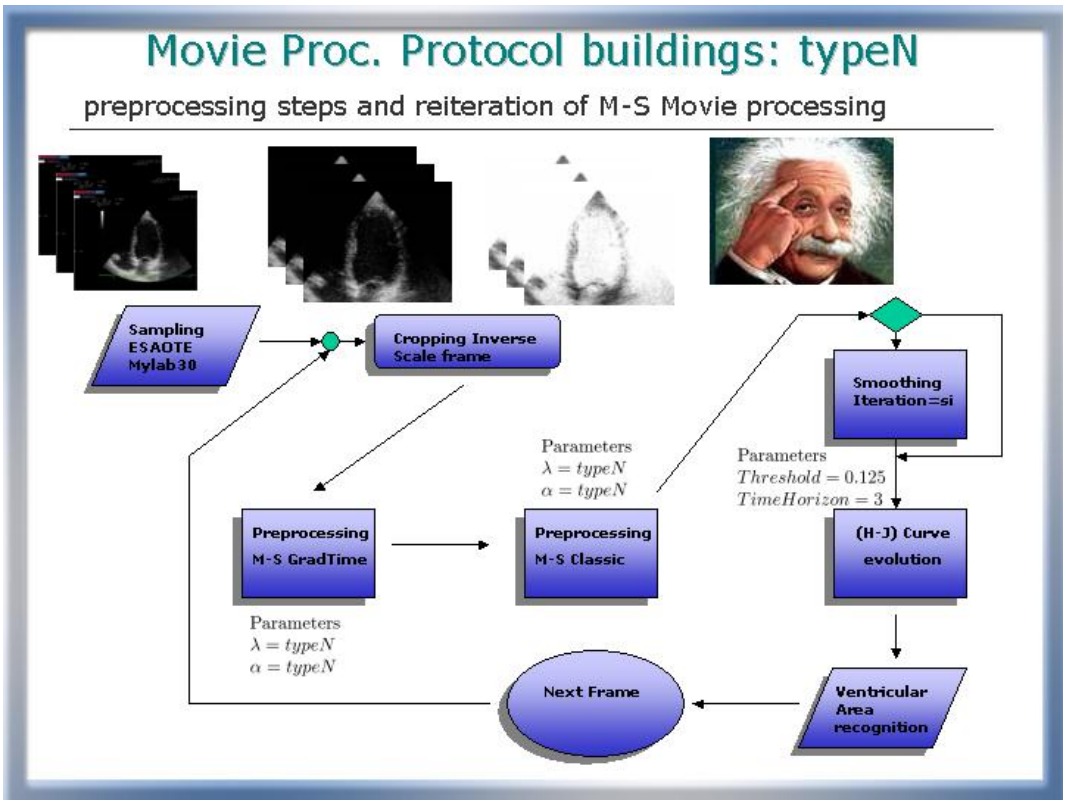
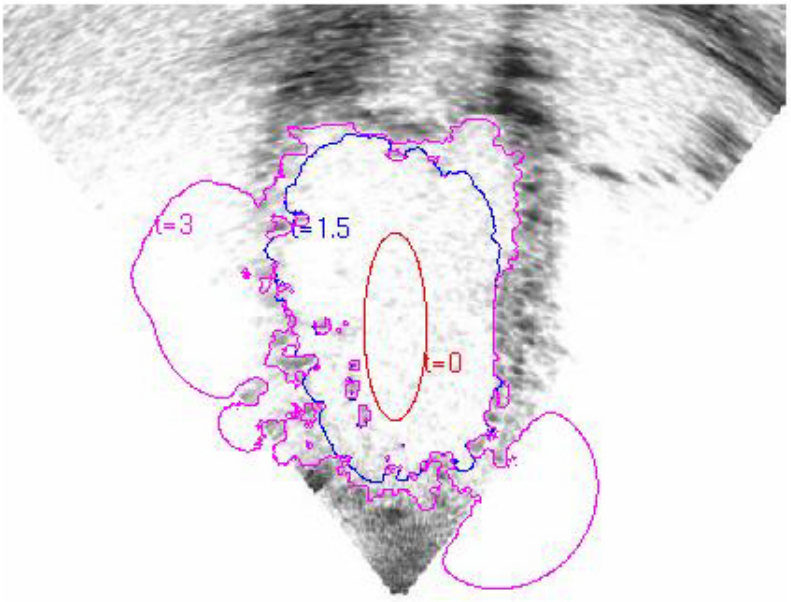


Figure 3.5.0.g: Diagram of the protocol for each Type of elaboration procedure

Type 1: Original image



Protocol of elaboration:

Cropping: Frame[800,652] $\rightsquigarrow$ Frame[501,411] [pixel]

Preprocessing: n/a

Processing:

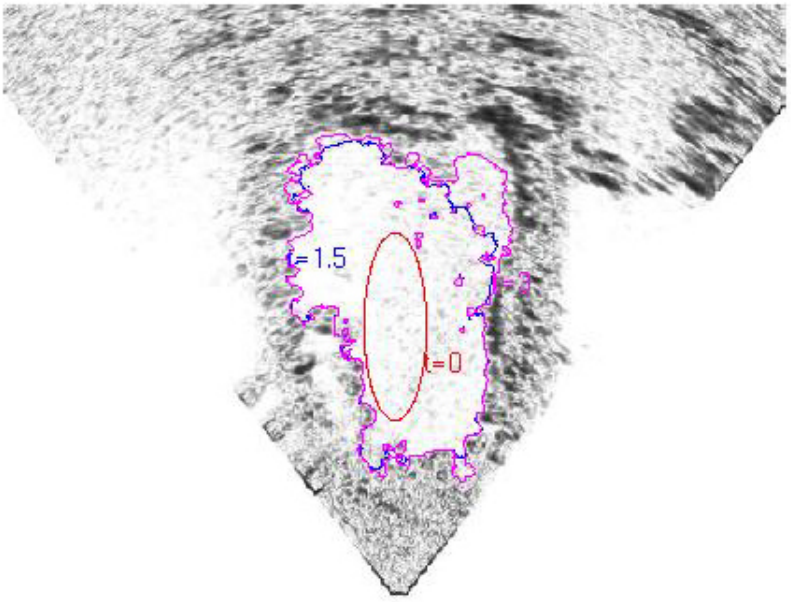
Smoothing: eq. Heat 1 iteration

H-J: Threshold (th=0.125) Time-Horizon (3, step 0.03)

Spatial mesh (0 ..5.0,0 ..4.1) Step (0.01,0.01)

The curve leaks out of the left and the right side of the ventricle, so we have to discard this choice for the protocol of area representation graph.

Type 2: Temporal gradient



Protocol of elaboration:

Cropping: Frame[800,652] $\rightsquigarrow$ Frame[501,411] [pixel]

Preprocessing: M-S Grad Temp ($\lambda = 0.02, alpha = 0.0001$)

$(UOrig(f-1, f, f+1)) \rightsquigarrow (Ureg_1(f), S_1(f))$

Processing:

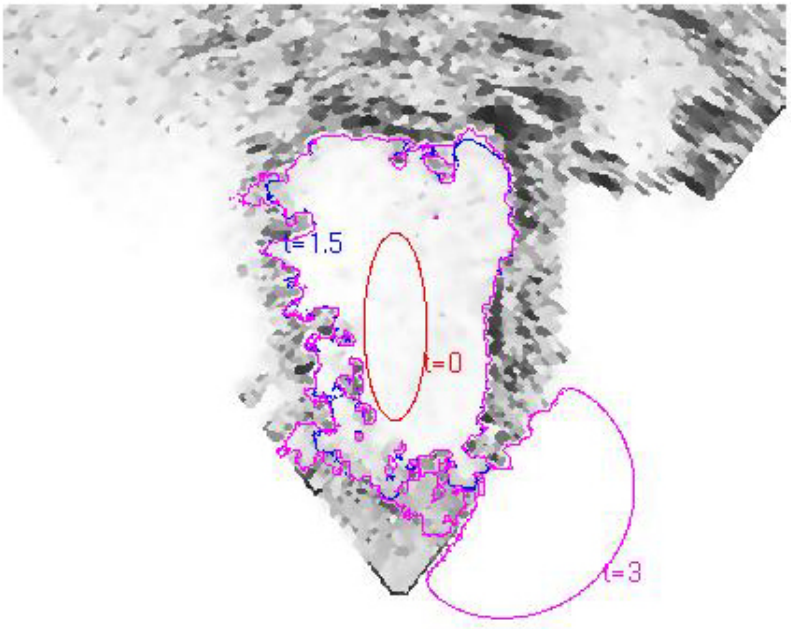
Smoothing: n/a

H-J: Threshold (th=0.125) Time-Horizon (3, step 0.03)

Spatial mesh (0 ..5.0,0 ..4.1) Step (0.01,0.01)

With this choice of parameters we obtain the best ventricular area recognition. The precision of this curve evolution is due to a correct balance between preprocessing and processing phase. In this case we apply M-S Time Grad algorithm and use the preprocessed U for curve evolution. A single step of M-S is made.

Type 3: Classical M-S on original image



Elaboration protocol:

Cropping: Frame[800,652] $\rightsquigarrow$ Frame[501,411] [pixel]

Preprocessing: Step1, M-S Time Grad ($\lambda = 0.02, \alpha = 0.0001$)

$(UOrig(f-1, f, f+1)) \rightsquigarrow (Ureg_1(f), S_1(f))$

Step2, Classic M-S ($\lambda = 50, \alpha = 0.02$)

$(Ureg_1(f), S_1(f)) \rightsquigarrow (Ureg_2(f), S_2(f))$

Processing:

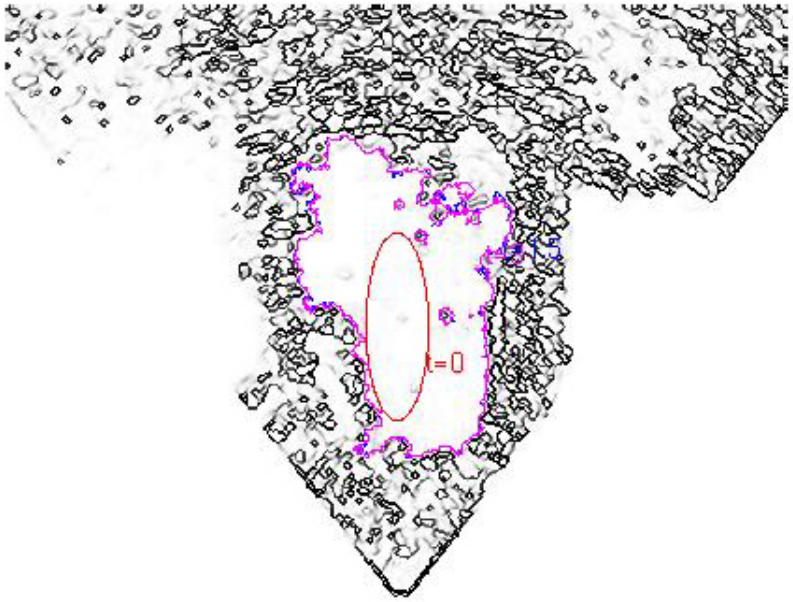
Smoothing: n/a

H-J($Ureg_2(f)$): Threshold (th=0.125) Time-Horizon (3, step 0.03)

Space mesh (0 ..5.0,0 ..4.1) Step (0.01,0.01)

In this case two steps of M-S algorithm are used with a different choice of parameters at each step. However, the curve evolves outside the ventricular border. Also this choice is not suitable for area recognition.

Type 4: M-S algorithm iterated on the jump set S



Elaboration protocol:

Cropping: Frame[800,652] $\rightsquigarrow$ Frame[501,411] [pixel]

Preprocessing: Step1, M-S Time Grad ($\lambda = 0.02, \alpha = 0.0001$)

$(UOrig(f-1, f, f+1)) \rightsquigarrow (Ureg_1(f), S_1(f))$

Step2 M-S Classic ($\lambda = 50, \alpha = 0.02$)

$(Ureg_1(f), S_1(f)) \rightsquigarrow (Ureg_2(f), S_2(f))$

Processing:

Smoothing: n/a

H-J($S_2(f)$): Threshold (th=0.125) Time-Horizon (3, step 0.03)

Space mesh (0 ..5.0,0 ..4.1) Step (0.01,0.01)

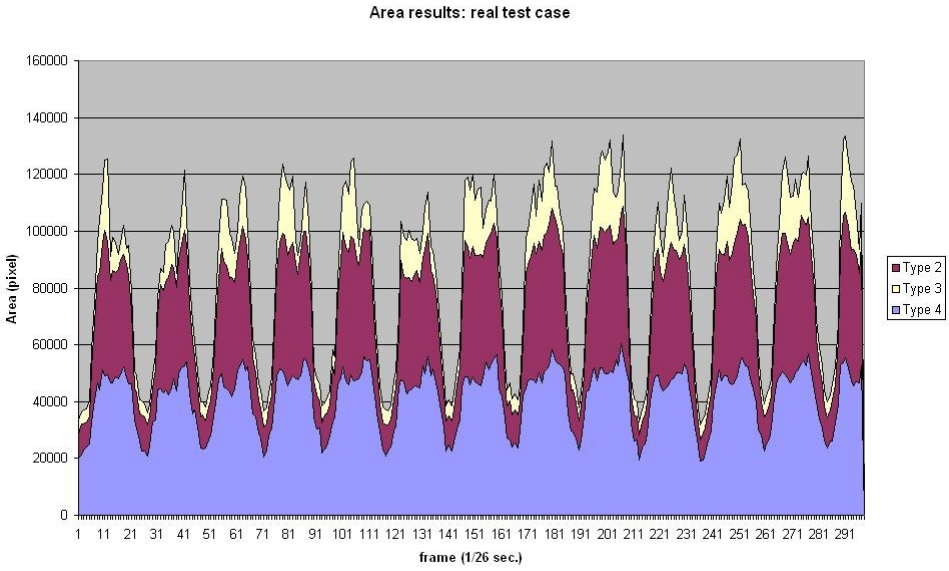
Curve evolution has a good performance on the ventricular area, but it not fills the entire cavity. This kind of choice is better than the one presented in Type2 in the case of low resolution of the image frame. The comparison

between these preprocessing methods selects which is the most viable for front evolution. The reported experiments are indicative of suitable methods for ventricular area recognition. Some of the above tests reveal how much edge emphatization is useful for moving edge recognition. The Preprocessing step allows to use small threshold values, so that front evolution fits the ventricular edge in an optimal way. Therefore, we choose to adopt Type2 and Type4 protocol for the analysis of image sequences.

3.6 Ventricular area trend in the frames

Some real-case results for ventricular areas in the case of Type2 and Type4 elaboration protocol are presented below. A graph of the CPU time is shown at the end of the paragraph.

Area results



Processed movie of 300 frames (10 sec)

3.7 Development of an application for signal synchronization

Medical research is interested in investigating the interaction between mechanic and electrical properties of cardiac muscle. To this aim, we build a Matlab graphic software (Graphic User Interface). This software is user-interactive and presents a box offering several control buttons. The main window shows an AVI video of the echographic data frame, an axis representing a multi-track of the Electrocardiographic signal over time and another axis showing the above-mentioned value of ventricular area and its related ejection fraction parameter over time expressed in seconds. The MATLAB application looks as follows.

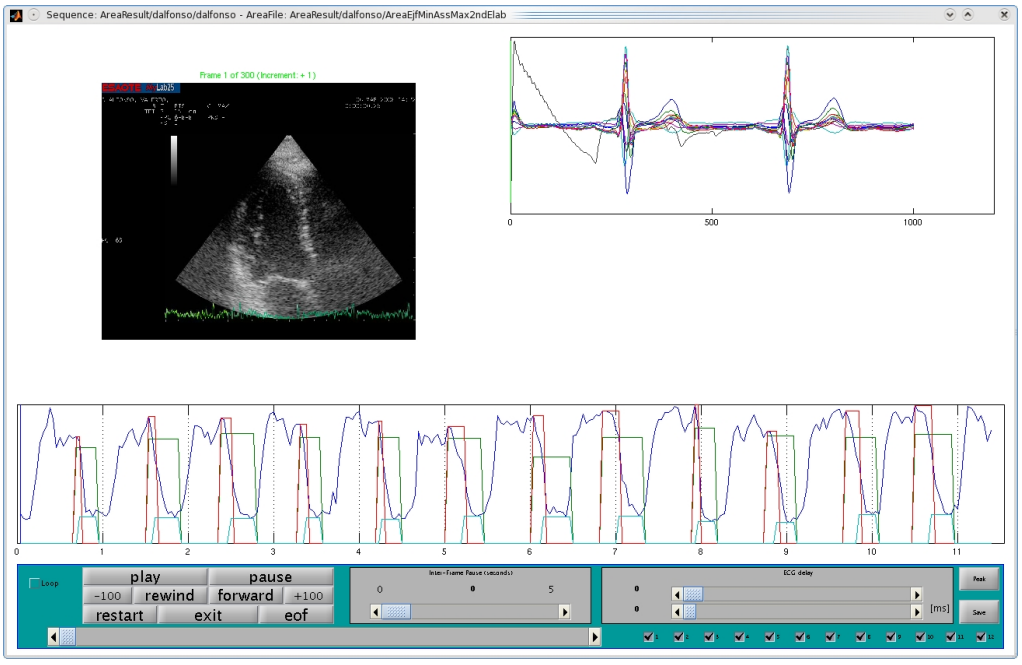


Figure 3.7.0.h: Echo ECG Synchronizer application

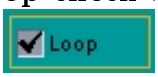
3.7.1 Commands box

The box offers buttons, a control slider and a check box for interactive manipulation of results. See figure 3.7.0.h
More detailed explain of the application components are reported below:

Control box

Movie playing control composition:

Loop check box



if checked, it restarts, after the last frame, the echo movie.

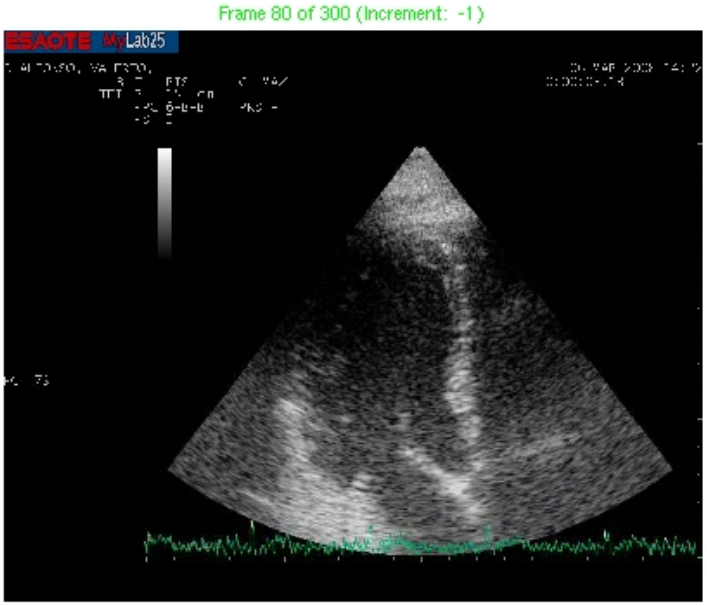
Play control

Each button controls, like a recorder, the movie playing forward or backward.

Frame slider

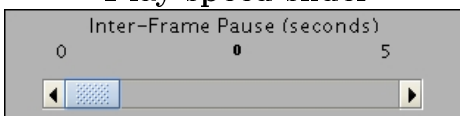
Used to search for a frame in the movie by mouse dragging.

Echo movie



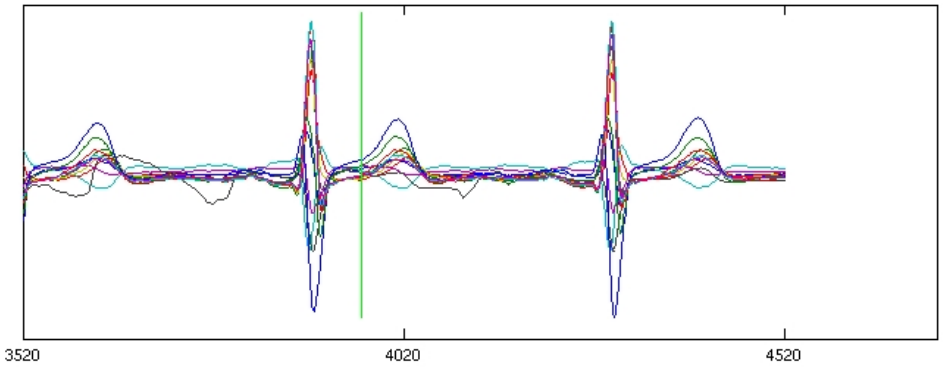
Visualize the Echo Cardio Graphic movie.

Play speed slider



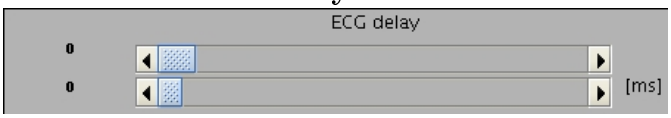
It sets the pause value between two successive frames

ECG graphic axis



It displays the Electro-Cardio-Graphic signal with relative green mark line for synchronization

Double delay ECG slider

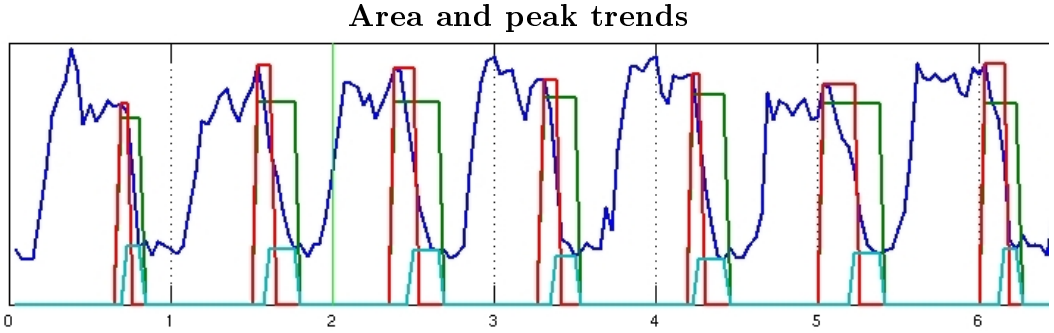


It sets both coarse and fine value for the delay between the first sample and the green mark refinement control; the upper slider sets a delay of 1 to 10 seconds, the lower one increases the position of the green marker with a step of one frame (i.e, typically 1/26 - 1/27 sec.).

Save and Peak buttons

The Save button allows to write out a text file containing synchronization data; the Peak button is aimed to a future development of a real-time ECG-

peak search function.



This main graphic axis shows results from the elaboration representing the main goal of this work: the trend of ventricular area over time. The max and min peak of expulsion for the diastolic cycle and the calculated Ejection Fraction are shown as well. We want to remark that a non-conventional information is provided: time needed for expulsion, which offers a possible field of investigation about internal ventricle pressure. In cardiovascular physiology, ejection fraction (E_f) is the fraction of blood pumped out of a ventricle with each heart beat. By definition, the volume of blood within a ventricle immediately before a contraction is known as the end-diastolic volume. Similarly, the volume of blood left in a ventricle at the end of contraction is end-systolic volume. The difference between end-diastolic and end-systolic volumes is the stroke volume, the volume of blood ejected with each beat. Ejection fraction (E_f) is the fraction of the end-diastolic volume that is ejected with each beat; that is, it is stroke volume (SV) divided by end-diastolic volume (EDV):

$$E_f = \frac{SV}{EDV} = \frac{EDV - ESV}{EDV}$$

In the last figure the values of E_f is represented by the dark green line, the red one is the maximal value of EDV and the cyan one is ESV, These values are only an appraisal, because referable to an area and not to a volume information.

Conclusions

This thesis is a first step of an extended project whose final target is the analysis of EchoCardioGraphic image sequences for non-invasive and a-posteriori medical diagnostics of heart left-ventricle diseases. The medical protocol requires the determination of local pressure and internal volume of the left ventricle during its cyclic work.
In this first step, which has many upgradable features, a specific method has been developed, based on mathematical models of image processing, for the recognition of ventricular area in different frames of EchoCardioGraphics images.
The exposed methods and their applications are an approach to answer to requests from medical diagnostics.
From the point of view of applied mathematics, numerical techniques have been used in order to obtain results which fit the problem. In echocardiographic images very ragged contours are usually available and pixel definition is low, but the described sophistication of the M-S method does not require the use of onerous numerical algorithms for convergence to the functional minimum. Indeed, those methods are typically applied in literature to skip approximation errors due to triple junctions or object occlusions in the treated images. Ref. [2],[17] et al.

The mathematical framework has been focused on some open problems about

the applicability of variational methods to time-dependent image streams. The proposed arguments have been developed in a preprint paper about segmentation and denoising of image time series by P. Barone and R. March [9]. It deals with small images of biological tissues affected by cancer growth and offers a theorical study on Bounded-Variation functions which is close to our specific interest in the existence of global minima, in order to develop a theory for the dynamic contribution of time gradients in the formulation of the M-S functional.

In the numerical solution of H-J equation, we have applied "partially standard" techniques suggested by articles [21], [36]. The approximation of front-evolution methods in the implementation of numerical codes has produced satisfactory results. The precision of results from eikonal eq. requires, according to image type, a very large elaboration time, if compared to the preprocessing phase obtained with the M-S method.

In order to reproduce a reliable contour of ventricular walls, closed-curve evolution requires a processing time which is one order of magnitude larger than the preprocessing phase. A further development of H-J technique could be obtained by means of the implementation of Fast-Marching methods [37]. In the preprocessing and processing phase, the direct examination of resulting images for the evolved curves and edge-enhancing allows back-regulation of parameters and verification the quality of our choices.
Various cardiologists are already trying to use our time-evolving image-processing technique by synchronization with the ECG signal in order to validate some medical hypotheses on ventricular diseases. In agreement with cardiologists involved in this project, future developments could be:
-The elaboration of ventricular areas by a ventricular "short and long side" approach, in order to obtain a better volumetric reconstruction.
-The validation of theories which allow the interpretation of correspondences

between the electric signal (ECG) and ventricular volume filling (ejection fraction).

-The implementation of this protocol for an Internet service for reconstruction of the ventricular area profile, which would prove useful for modern e-healthcare.

-The study of mobility profiles of ventricular walls to draw speed vector-field curves during cardiac periodical movement; this would prove useful for the individuation of muscular portions affected by diseases.

-The possibility to model, by finite-element analysis, the ventricular pump to make internal pressure and volume fit experimental results in the comparison between real data and simulation.

From a computational point of view, Fast Marching Methods are a possible improvement to reduce CPU time for curve evolution by eikonal eq.

In conclusion, in this thesis I have developed a project which fits reasonably well the aims of my PhD school in mathematical methods and models for technology and society.

Appendix A

Mathematical recall

A.1 Hamilton-Jacobi equation and Hopf-Lax formula

A.1.1 Fronts motion by normal direction speed

Let $\{\Gamma_t\}_{t\in[0,T)}$ a family of regular surfaces in $\mathbb{R}^3$, the speed in the outward direction of a given point $x \in \Gamma_t$ is defined by:

$$V(x,t) = n(x,t)C'(t)$$

where $C : [0,T) \to \mathbb{R}^2$ is a regular curve such that $C(t) = x$ and $C(s) \in \Gamma_s$ for every $s \in [0,T)$ and $n(x,t)$ is the outward normal direction versor of Γ_t at point x.
Suppose that the family $(\Gamma_t)_{t\in[0,T)}$ evolve from an initial configuration Γ_0 with assigned speed

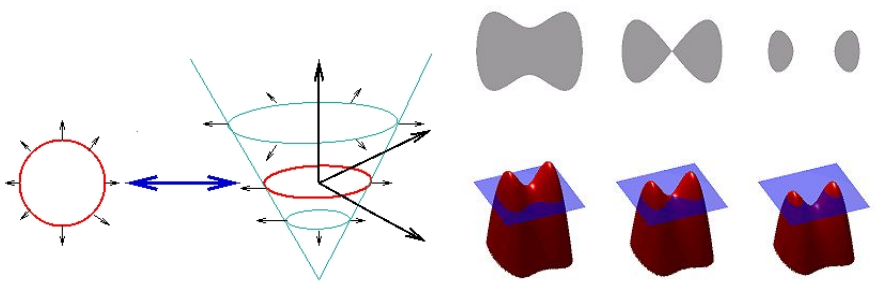
$$V(x,t) := v(x,t,n(x,t)), \qquad \forall x \in \Gamma_t, \qquad t \in [0,T) \tag{A.1.1}$$

where v is a given function on $\mathbb{R}^2 \times [0,T) \times S^1$. If exists a regular function $u(x,t)$ and if for every $t \in [0,T)$ the open set $\Omega_t \subseteq \mathbb{R}^2$ with Γ_t its frontier such that

$$\Omega_t = \left\{x \in \mathbb{R}^2 : u(x,t) > 0\right\}, \qquad \Gamma_t = \left\{x \in \mathbb{R}^2 : u(x,t) = 0\right\}$$

if, at least $\nabla u(x,t) \neq 0$ on Γ_t. Then we have

$$n(x,t) = \frac{-\nabla u(x,t)}{|\nabla u(x,t)|}.$$



So for every $t \in [0,T)$, if $C : [0,T) \to \mathbb{R}^2$ is a regular curve such that $C(t) \in \Gamma_t$ we obtain, by diffentiation with respect to t:

$$0 = u_t(x,t) + \nabla u(x,t)C'(t) = u_t(x,t) - |\nabla u(x,t)|n(x,t)C'(t).$$

This give us the relation between the front speed V and the function u; level set at the 0 level of Γ_t

$$V(x,t) = \frac{u_t(x,t)}{|\nabla u(x,t)|}, \qquad \forall x \in \Gamma_t.$$

Than we obtain the Eikonal equation

$$u_t - v(x,t, \frac{-\nabla u(x,t)}{|\nabla u(x,t)|})|\nabla u(x,t)| = 0$$

A.1.2 First-order H-J and viscosity solution

We consider the following first-order evolutionary equation. Let Ω be an open subset of $\mathbb{R}^2$, with boundary $\partial\Omega$ C^1,

$$\begin{cases} u_t(x,t) + H(\nabla u(x,t)) = 0 & x \in \Omega,\ t \geq 0 \\ u(x,0) = u_0(x) & x \in \Omega \end{cases} \tag{HJ1}$$

where $H : \Omega \to \mathbb{R}$ is a convex and continuous function[22]. Being the initial data element u_0 discontinuous and H non-linear, the problem has no solution in the classical sense. Nevertheless the problem admits more solutions among differentiable functions that satisfy (HJ1) for x almost everywhere, that is weak solutions. In particular, we are interested to *viscosity solutions* of the type $u(x,t)$ that

$$u(x,t) = \lim_{\epsilon \to 0} u^\epsilon(x,t)$$

which

$$u_t^\epsilon(x,t) + H(\nabla u^\epsilon(x,t)) - \epsilon \Delta u^\epsilon(x,t) = 0 \qquad x \in \Omega,\ t \geq 0. \tag{A.1.2}$$

The *viscosity solution* will be the only weak solution such that:
for every test function $\phi \in C^1(\mathbb{R}^N)$, when $(u - \phi)$ has
a local maximum at the point (x_0, t_0), then $\phi_t(x_0,t_0) + H(\nabla\phi(x_0,t_0)) \leq 0$
when it has a local minimum at the point (x_0, t_0),
then $\phi_t(x_0,t_0) + H(\nabla\phi(x_0,t_0)) \geq 0$.
Thus the *viscosity solution* is defined as an extreme of functions, which are not necessarily regular and uniformly continuous, but satisfy these rules for every test function ϕ.
We now shortly recall *Legendre-Fenchel Transformation* (see [25]), which is necessary to the formalization of the representation formula for *viscosity solution* to H-J equation.

$$H^*(p) = \sup_{q \in \mathbb{R}^N} \{\, p \cdot q - H(q) \,\}. \tag{A.1.3}$$

For a convex continuous function H

$$\frac{H(p)}{p} \to +\infty \quad \text{when} \quad |p| \to +\infty. \tag{A.1.4}$$

$H^*(p)$ can be found such that $(H^*(p))^* = H(p)$ for every $p \in \Omega \subset \mathbb{R}^N$. We can use, for convex and continuous H , *Hopf-Lax-Oleinik* representation formula:

$$u(x,t) = \inf_{y \in \mathbb{R}^N} \left\{ u_0(y) + tH^*\left(\frac{x-y}{t}\right) \right\} \tag{A.1.5}$$

A detailed discussion about this kind of problems with Hamiltonians of the form $H(x, \nabla u)$ can be found in [7].

A.2 Functional minimization

A.2.1 The mathematical model for image segmentation

In the analysis of variational problems with free discontinuity the discontinuous solution (discontinuity curves which are not a-priori known and geometric problems) induces a search for minima and approximations of the M-S functional through non-standard methods such as Γ-convergence [1], [2], [3]. In images analysis, which is reported here, we consider Ω as an open and limited subset of $\mathbb{R}^2$ (rectangular images). We define the image as the brightness-intensity function (the brightness-intensity of an image is indicated here by a different notation than in previous paragraphs):
$g : \Omega \longrightarrow \mathbb{R}$
(that is the same as the previous $I(x)$ scaled between $0, .., 2^s - 1$, for s gray-tone) considered in gray scale; for color images a vector field could be used in $\mathbb{R}^3$. Then let $g \in L^\infty(\Omega)$ be ess. limited.
In general, the identification of contours inside Ω domain allows the reconstruction of edges for recognition of contours of the represented object. We build the mathematical model which, when g is given, finds K, that is, the expected result.

Approximation of the image according to a metrics

We individualize the problem model for $u \in C^1(\Omega \setminus K)$ with

$$K = \bigcup_i \gamma_i, \quad \gamma_i \in C^1$$

to determine, when g is known,(u, K)-couple by optimization.
We then define a functional cost that measures the optimization of the couple (u, K) with distance L^2. Let us set

$$F(u, K) = \int_\Omega (u - g)^2 + \lambda \int_{\Omega \setminus K} |\nabla u|^2 \, dx \quad + \alpha \, | K |$$

to minimize, for $\lambda > 0$ and $\alpha > 0$, the tree addenda, which, in order, can be explained as follows :
1) L^2-distance of u from the given g, which evaluates how much our image solution is "near" to the given image.
2) To make small areas, delimited by small contours, not distinguishable, our u must be C^1 in large subsets. This corresponds to an energy criterium for the functional. The presence of λ-parameter allows us to smooth function u as much as we like. Following the optical flow model (see page 184 [6], we introduce dependence on time in the gradient term ($|\nabla u|^2$). In particular, the gradient of u is calculated both at the instant that precedes the examined frame and at the one that follows.
3) In this addendum only essential parts of contours are allowed to survive, that is, we add a measure of segmentation spread by considering the length of curves. When the value of parameter α is very small, small contours are privileged, while larger values privilege large subsets. Thus, values of α-parameters are usually chosen to allow a correct recognition of forms according to expectations and type of image. Many different choices of α-parameter values have been made in our application in order to extrude data in the best way to favor the recognition of the interested area.
The integration of the second addendum over $\Omega \setminus K$, instead of the entire Ω , is needed to avoid gradient computation for a discontinuous function with no null measure (as if it would be a square Dirac-delta). So the approximation is considered on a SBV space (see appendix).

A.2.2 Euler equation for the M-S functional

We apply to the functional the rules for determination of Euler equation.

$$F(u) = \int_\Omega \psi(u, \nabla u) dx, \quad \Omega \subset \mathbb{R}; \quad u : X \to \mathbb{R} \quad \text{with enough regularity}$$

We suppose that F has minimum in a metric space, as well as that its Gateaux's derivative exists and has zero-value: $F(u, tv) = \varphi(t), \quad \varphi'(0) = 0.$

Calculating:

$$F(u+tv) = \int_\Omega \psi(u+tv, \nabla u + t\nabla v)dx \quad \text{,t parameter dependent integral;}$$

we derive under the integral sign (assuming the necessary regularity of the integrand) and, working with$\psi(x,\xi)$

$$\varphi'(0) = \int_\Omega \left(\frac{\partial \psi}{\partial u}(v) + \left\langle \nabla_\xi \psi(u, \nabla u), \nabla v \right\rangle \right).$$

Let us expand the scalar product. Being: $div(\psi A) = \psi\ div(A) + < A, \nabla\psi >$, we obtain:

$$\int_\Omega \left(\frac{\partial \psi}{\partial u}(v) + div\left(v \nabla_\xi \psi(u, \nabla u)\right) - div\left(\nabla_\xi \psi(u, \nabla u)\right) v \right) dx =$$

with regular frontier Ω, by Gauss-Green theorem; if we collect the first and third term together, we obtain $\forall v \in X$

$$= \int_\Omega \left(\frac{\partial \psi}{\partial u} - div\left(\nabla_\xi \psi(u, \nabla u)\right) \right) v\ dx + \int_{\partial\Omega} v \left\langle \nabla_\xi \psi(u, \nabla u)\ ,\ \mathbf{n}\ v \right\rangle d\sigma = 0,$$

We now choose a test function v, which is enough regular, not null in the neighborhood of point x and null outside. Let $\mathbf{n}$ be the outward normal of $\partial\Omega$. Choosing initially a compact-support test-function in domain Ω, in distributions sense, formally we obtain the equation in the Ω, therefore the homogeneous Neumann boundary condition follows using generic test functions. Then we obtain the following system:

$$\begin{cases} \left(\frac{\partial \psi}{\partial u}\right) - div\left(\frac{\partial \psi}{\partial(\nabla u)}\right) = 0 \quad \forall x \in \Omega \\ \mathbf{n}^T \nabla_\xi \psi(u, \nabla u) = 0 \quad \forall x \in \partial\Omega \end{cases}$$

stand true. Where the second equation becomes from the second integral calculated the frontier of Ω, taken with test functions, in a neighborhood of the point.

Application to the Mumford-Shah functional

We consider the non-geometric part, susceptible of easy generalization, so we get:

$$F(u,K) = \int_\Omega (u-g)^2 + \int_\Omega \mid \nabla u \mid^2 dx \quad \text{from which } \psi(u, \nabla u) = (u-g)^2 + \mid \nabla u \mid^2 .$$

Considering that:

$$\frac{\partial \psi}{\partial(\partial \nabla u)} = \left(\frac{\partial \psi}{\partial(\frac{\partial u}{\partial x})}\ , \frac{\partial \psi}{\partial(\frac{\partial u}{\partial y})} \right) = 2\left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right) = 2\nabla u \Rightarrow \frac{\partial \psi}{\partial u} = 2(u-g)$$

we get:

$$\begin{cases} \Delta u - u + g = 0 \quad x \in \Omega \\ \mathbf{n}^T \nabla u = 0 \quad x \in \partial\Omega \end{cases} \tag{A.2.0}$$

which has null normal derivative, with Neumann conditions at the border, and is numerically solvable.

The geometric part of the functional

When we consider the functional in its geometric part, we have to refer to the above-mentioned discussion on weak function space and associated distance, which can be easily found in a large part of analytical literature (see [6])

$$F(u,K) = \int_\Omega (u-g)^2 + \int_{\Omega \setminus K} \mid \nabla u \mid^2 dx \quad + \mid K \mid .$$

In investigating criteria for the existence of a minimum of the functional, we need to make some conjectures on the space and induced metrics.
We look for a space where to locate the functional; nevertheless, whatever metrics we decide to use for (u, K)-couples we are not successful in getting compactness, although K consists of regular curves (compactness would implicate completeness and convergence of Cauchy sequences).
Thus, we need to locate functions in a special Bounded-Variation (BV) function space and to adopt a specific measure for $|K|$-lengths, which allows us to extend the geometric part of the functional to the entire Ω for the numerical approximation contained in Ambrosio-Tortorelli theorem. See references for an explanation of analytical treatment of the functional and its numeric computation by Euler equation [1], [2], [3], [4].

A.2.3 Approximation of the M-S Functional

The approximation of the M-S functional to search for minima by Euler equation has been presented so far. The numerical solution of the equation is just a guarantee for local minima, while Γ-convergence assures the existence of global minima for the functional.
Let us consider the weak form:

$$F(u) = \int_{\Omega} (u-g)^2 \, dx + \int_{\Omega} \mid \nabla u \mid^2 \, dx + \mathcal{H}^1(Su)$$

Non-convexity is given by the geometric part for $\int_{\Omega} S^2 \mid \nabla u \mid^2 \, dx$.
Nevertheless, parameters generate different weights for individual addenda; their role, for limit values, determine how much the jumps for geometric part matter. Let us consider α-parameter in the original functional:

$$F(u) = \int_{\Omega} (u-g)^2 \, dx + \int_{\Omega} \mid \nabla u \mid^2 \, dx + \alpha\mathcal{H}^1(Su).$$

When $\alpha \to +\infty$, the term $\alpha\mathcal{H}^1(Su)$ vanishes over the jump set. Consequently, the functional gets convex

$$F_{\alpha=\infty}(u) = \int_{\Omega} (u-g)^2 + \int_{\Omega} \mid \nabla u \mid^2 \, dx.$$

By a continuation method, if $\alpha_0 = 10^6$ is taken, we obtain a convex functional again.
For geometric part a very large parameter α and $S = 1$ are typically used

$$\alpha \int_{\Omega} \left(\epsilon \mid \nabla S \mid^2 + \frac{1}{4\epsilon}(1-S)^2 \right) dx$$

The procedure consists in solving Euler equation system for α-values which are progressively reduced, for example $\alpha_k = \alpha_0 c^k$ with $0 < c < 1$. Thus, heuristically, one goes down along the functional gradient until the global minimum is reached, with a functional slightly deconvexed.

Thanks

Department of Mathematical Methods and Models for the Applied Sciences
for supporting study and offering courses and seminars.

Prof. Daniela Giachetti for the PhD coordination.

Prof. Paola Loreti for suggestions about analytical references and work revision.

Prof. Emeritus Antonio Avantaggiati for his passion and extraordinary communicative ability in Functional Analysis.

Dr. Marco Rorro (MeMoMat, CASPUR) for his collaboration to development of the code for front evolution.

'Guido Castelnuovo' Department of Mathematics

Prof. Camillo Cammarota for his collaboration and grant experience in signal analysis and synchronization and for managing contacts with 'Umberto Primo' University Hospital.

Prof. Maurizio Falcone, for his constant effort in organizing seminars and for giving me application references to Image Processing models.

Prof. Stefano Finzi Vita and Antonio Siconolfi, for their patience and suggestions about analytical references.

IAC CNR Rome

Massimo Bernaschi and Roberto Natalini for their constant verification of results and their suggestions about scientific references.

Riccardo March for knowledge he gave us during and after the course of Variational Methods in Analysis of Images and for his suggestions about literature on the subject.

Department of Clinical Sciences of 'Umberto Primo' University Hospital ('Sapienza' University, Rome)

Prof. Mario Curione and his team of Cardiographers for their helpfulness and for providing clinical data.

Dip. of Physics of the Sapienza Universitá di Roma

Prof. Adalberto Bonincontro, Tina Congiu and, last but not least, Giuseppe Brig-

anti for their hospitality and collaboration in revising my work. Alexandra Parmentier for the work of language revision and correction
Dip. of Physiology and Molecular Genetics
Paola Ballario for her revision and correction work and her moral support.
All PhD colleagues who have designated me to represent them in the Department Council during this study cycle. A special thanks to the image analysis guru Stanley Osher for the indications on the numerical code implementation, and the opinion about the novelty of the approach at time series image elaboration and his "good application" to the graphic software.

References

[1] L. Ambrosio, *"A compactness theorem for a new class of functions with bounded variation."*, Boll. Un. Math. Ital. **7**,3-B,pp. 857-881 (1989).

[2] L. Ambrosio, *"Variational problems in SBV and image segmentation."*, Acta Appl. Math. **17**,pp. 1-40 (1989).

[3] L. Ambrosio, *"Existence theory new for class of variational problems."*, Arch. Rational Mech. Anal. **111**, pp. 291-322 (1990).

[4] L. Ambrosio, N. Fusco, D. Pallara, *"Functions of Bounded Variation and Free Discontinuity Problems, "*, Oxford University Press, New York , (2000).

[5] L. Ambrosio, V.M. Tortorelli, *"Approximation of functionals depending on jumps by elliptic functionals via* Γ*-Convergence."*,Comm. Pure Appl. Math. **43**,pp. 999-1036 (1990).

[6] G. Aubert, P. Kornprobst, *"Mathematical Problems in Image Processing "*, Springer Verlag ,(2002).

[7] M. Bardi, L. C. Evans, *"On Hopf's formulas for solutions of Hamilton-Jacobi equations "*, Nonlinear analysis **8**, (1984).

[8] B. Bourdin, A. Chambolle, *"Implementation of an adaptive finite-element approximation of the Mumford-Shah functional "*, Numer. Math. **85**,pp. 609-646 (2000).

[9] P. Barone, M.F. Canfora, R. March, *"Segmentation, classification and denoising of time series of images by variational method"*,Preprint IAC CNR (2008).

[10] I. Birindelli, S. Finzi Vita, *"A class of quasi-linear elliptic systems arising in image segmentation"*, Nonlinear Differential Equations and Applications NoDEA. **5**,pp 445-449 (1998)

[11] F. Camilli, *"An Hopf-Lax Formulas for a class of Measurable Hamilton-Jacobi Equation "*, Nonlinear analysis, Elsevier. **57:22**,pp 265-286(2004).

[12] F. Camilli, M. Falcone, *"An approximation scheme for the optimal control diffusion process"*, Modélisation mathématique et analyse numérique **29**, pp. 97-122 (1997).

[13] F. Camilli, A. Siconolfi, *"Hamilton-Jacobi Equation with a Measurable dependence on the state variable"*, Advances in Differential Equation. **8(6)** pp. 733-708, (2003)

[14] V. Caselles, B. Coll, *"Snakes in Movement"*, SIAM Jour. Numer. Anal. **019**,Vol 33, pp. 2445-2456 (1996).

[15] T.F. Chan, J. Shen, *"Variational Image Inpainting"*, Comm. on Pure and Appl. Math. Vol LVIII, pp. 579-619 (2005).

[16] T.F. Chan, J. Shen, L.A. Vese, *"Variational PDE Models in Image Processing"*, Notice of the AMS **01**Vol 50 N. 1, pp. 14-26 (2003).

[17] T.F. Chan, L.A. Vese, *"Active Contour Without Edges"*, IEEE Transaction on Image Processing **Vol 10**, pp. 266-276 (2001).

[18] E. De Giorgi, M. Carriero, A. Leaci, *"Existence theorem for to minimum problem with free discontinuity set."*, Arch. Rational Mech. Anal. **108**,pp. 195-218 (1989).

[19] M. Droske, W. Ring, *"A Mumford-Shah Level-set Approach for Geometric Image Registration"*, SIAM Jour. Appl. Math. Vol. 66 no. 6, pp. 2127-2148 (2005).

[20] J.D. Durou, F.Courteille, A. Crouzil, P. Gurdjos *"Shape from Shading en conditions rèalistes of aquisition photographique"* Institut de Recherche en Informatique de Toulouse, Universitè Paul Sabatier Tolouse (2000).

[21] J.D. Durou, M. Falcone, M. Sagona, *"A Survey of Numerical Methods for Shape-From-Shading"*, Kluwer Academic Publishers. Netherland, (2003).

[22] L.C. Evans, *"Partial Differential Equations"*, American Mathematical Society **Vol 19** (1998).

[23] M. Falcone, *"Numerical solution of Dynamic Programming equation"*, Appendix A in the volume M. Bardi and I. Capuzzo Dolcetta, *"Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations"*, Birkhäuser, Boston, (1997).

[24] M. Falcone and I. Capuzzo Dolcetta, *"Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations "*, Birkhäuser, Boston, (1997).

[25] M. Falcone, R. Ferretti, *"Semi-Lagrangian Schemes for Hamilton-Jacobi Equations, Discrete Representation Formulae and Godunov Methods"*, Journal of Computational Physics **175**,pp 559-575 (2002).

[26] G. A. Francfort, Q. Le, S. Serfaty, *"Critical points of Ambrosio-Tortorelli convergence to critical points of M-S in the one dimensional Dirichlet Case"*, ESAIM: Control, Optimisation and Calculus of Variation **15**, pp 576-598 (2009).

[27] R. Malladi, J.A. Sethian, B.C. Vemuri, *"Shape modeling with front propagation: A level set Approach"*, IEEE TRANSACTION ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE **Vol 17**, pp. 158-175 (1995).

[28] R. March, *"Visual reconstructions with discontinuities using variational methods"*, Image and Vision Computing **10**, pp. 30-38 (1992).

[29] D. Mumford, J. Shah , *"Optimal approximation by piecewise smooth functions and associated variational problems"*, Comm. Pure Appl. Math. **42**, pp. 577-685 (1989).

[30] J.M Morel,S. Solimini, *"Variational Methods in Image Segmentation"*, Progress in Nonlinear Differential Equations and Their Applications. Birkhäuser, Basel, (1995).

[31] S. Osher, N. Paragios, *"Geometric Level Set Methods in Imaging, Vision, and Graphics"*, Springer (2003).

[32] S. Osher, J.A. Sethian, *"Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations"*, Journal of Computational Physics **79**, pp. 12-49 (1988).

[33] M. Pedone, M. Falcone, *"Alcuni algoritmi per il trattamento delle immagini basati su EDP"*, Tesi Master Calcolo Scientifico, La Sapienza. http://pedoneweb.phys.uniroma1.it/max/master/, (2003).

[34] J.A. Sethian, *"Level Set Methods and Fast Marching Methods"*, Proc. Natl. Cambridge University Press (1999).

[35] R.M. Spitaleri, R. March, D. Arena, *"Finite difference solution of Euler equation arising in variational image segmentation"*,Numerical Algorithms pp. 353-365 (1999).

[36] J.A. Sethian, A. Vladimirsky, *"Fast methods for the Eikonal and related Hamilton-Jacobi equations on unstructured meshes"*, Proc. Natl. Acad. Ski. USA **97**, pp. 5699-5703 (2000).

[37] J. N. Tsitsiklis, *"efficient algorithms for globally optimal trajectorie"*, IEEE Transactions on Automatic Control". **v40, 9**, pp. 1528-1538 (1995).

[38] S. Urheim, R. Bjornerheim, K. Endersen, et al., *"Quantification of Left Ventricular Diastolic Pressure-volume Relations During Routine Cardiac Catheterization by Two-Dimensional Digital Echo Quantification and Left Ventricular Micromanometer"*, Journal of the American Society of Echocardiography. **15**, pp. 225-232 (2002).