Development and Validation of an Algorithm for Cardiomyocyte Beating Frequency Determination

Demián Wassermann and Marta Mejail

Universidad de Buenos Aires,
Facultad de Ciencias Exactas y Naturales,
Intendente Güiraldes 2160, Ciudad Universitaria, C1428EGA,
República Argentina
{dwasser, marta}@dc.uba.ar

Abstract. The Chagas disease or *Tripanosomiasis Americana* affects between 16 and 18 million people in endemic areas. This disease affects the beating rate of infected patients' cardiomyocytes. At the Molecular Biology of Chagas Disease Laboratory in Argentina the effect of isolated patient's serum antibodies is studied over rat cardiomyocyte cultures. In this work an image processing application to measure the beating rate of this culture over video sequences is presented. This work is organized as follows. Firstly, a preliminary analysis of the problem is introduced, isolating the main characteristics of the problem. Secondly, a Monte Carlo experiment is designed and used to evaluate the robustness and validity of the algorithm. Finally, an algorithm of order $O(T(N \log N + N))$ for tracking cardiomyocyte membranes is presented, where T is the number of frames and N is the maximum area of the membrane. Its performance is compared against the standard beating rate measure method.

1 Introduction

The Chagas disease or *Tripanosomiasis Americana* affects between 16 and 18 million people in endemic areas. That can be found between $42°$N and $46°$S parallels, ranging from USA to Argentina and Chile [1]. The annual death rate caused by this disease reaches the number of 45.000 people [2]. Chagas disease is considered a typical socioeconomic illness, inseparable from poverty and underdevelopment. It has been noticed that this disease affects the beating rate of infected patients' cardiomyocyte, a human or mammal cardiac cell. At the Molecular Biology of Chagas Disease Laboratory at INGEBI-CONICET Argentina, the effect of isolated and purified patient's serum antibodies is studied over neonatal rat cardiomyocyte cultures; the neonatal rat cardiomyocytes behave like human cardiomyocytes, in the case of the studied chagas antibodies [3]. The effects of these antibodies over the culture is studied on an inverted microscope connected to a digital camera. The study can be divided in two steps. In the first step the beating rate of the cardiomyocytes is measured, and according to it a volume of antibodies is inoculated to the culture. After a lapse of time a new measure of the beating rate is taken in order to measure the effect of the antibodies.

In this work a technique to measure the beating rate of the cardiomyocytes from a digital video is developed. This algorithm needs to be fast enough to produce the results

in a few minutes (short response-time) because the cardiomyocytes in the culture are dying and this affects the measures and results of the biological study. A preliminary study of the videos is performed as a first stage, a filter is applied to each video frame in order to reduce the image noise and enhance the edges. Then a set of cardiomyocyte cell membranes are manually selected and tracked over the video sequence. The results of this tracking are analyzed to measure the feasibility of extracting the beating rate of the culture from the video sequence. Once the main characteristics of the tracked objects are identified, a Monte Carlo experiment is designed to validate the procedure. Finally a short response-time algorithm based on active contours, more precisely the *fast marching* method (see [4,5]), is developed in order to track the cardiomyocyte cell membranes. This algorithm is developed in order to produce a software tool for an end-user working at the mentioned laboratory.

This work is structured as follows: In Sect. 2 a preliminary study of the cardiomyocyte culture is performed and its main characteristics isolated. Furthermore a technique to infer the beating rate is proposed. in Sect. 3 a Monte Carlo experiment is designed in order to validate the proposed technique and its results are presented. In Sect. 4 a more automated tracking technique based on active contours is presented in order to develop a short response-time software tool. Finally, in Sect. 5 the conclusions of this work are presented.

2   Preliminary Studies

In this section, image preprocessing of the cardiomyocyte culture and a technique for preliminary analysis are described, and several features of the video are characterized.

In order to find the beating frequency in the cardiomyocyte culture in a robust manner, cardiomyocyte cell membranes are tracked over a video sequence. In Fig. 1 the membranes are circled. The centroid norm and area variations over time are calculated as two one dimensional signals: $cd(t)$ and $a(t)$, where t is the frame number. The Fourier transform of these signals is then computed in order to find the beating frequency, which will be the frequency with the greater amplitude.

The first task of the preliminary analysis is the segmentation of the membranes. Here the image is convolved with a Gaussian kernel, and then thresholded. This process is illustrated in Fig. 1, where Fig. 1(a) is the original culture image and Fig. 1(c) is the image after filtering and thresholding. A region of interest (ROI) was selected for each membrane analyzed in a way such that membrane is the only object over the threshold in each ROI in every frame of the sequence. These ROIs are automatically analyzed above the video sequences by calculating the centroid and area of the pixels above the threshold in each ROI for each frame. Then the norm of the centroid and the area are normalized as follows:

$$cd_n(t) = \|cd(t)\| - \frac{\sum_{i=0}^{N-1} \|cd(i)\|}{N}, \, 0 \leq t \leq N-1$$

$$a_n(t) = a(t) - \frac{\sum_{i=0}^{N-1} a(i)}{N}, \, 0 \leq t \leq N-1$$

where $cd(t)$ is the centroid of the tracked object in frame t, $a(t)$ is the area of the tracked object in frame t and N is the number of frames. In Fig. 2 the normalized norm of the

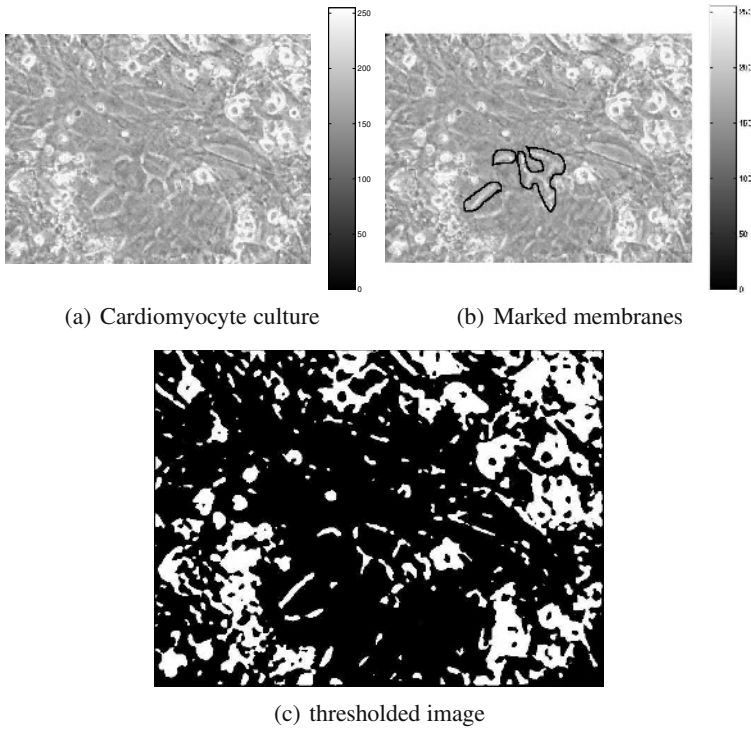(a) Cardiomyocyte culture   (b) Marked membranes

(c) thresholded image

Fig. 1. In (a) a frame of the cardiomyocyte video sequence is shown. In (b) the same frame is shown with some membranes marked. In (c) frame is shown again after it has been filtered and thresholded, the membranes marked in (b) are over the threshold and ready to be segmentated.

centroid, $cd_n(t)$, and the normalized area $a_n(t)$, functions are plotted along with their Fourier transforms. Analyzing the Fourier transform of these functions presented on Fig. 2(c) and Fig. 2(d), it can be noticed that $cd_n(t)$ is composed of several frequencies, but also there are two dominant frequencies, F_0 and F_1 where $F_1 > F_0$. The lower frequency, which also appears on the Fourier transform of $a(t)$, is a consequence of the microscope's light intensity frequency and can be eliminated by normalizing the mean grey level of all the frames in the sequence. Eliminating F_0 and discarding the low amplitude frequencies, the following characteristics can be inferred:

– There is a center of contraction and dilation in the culture. In this work this point will be referred as *beating center*.
– The normalized centroid norm, $cd_n(t)$, has a dominant frequency which indicates the beating rate: F_1.
– The area, $a_n(t)$, suffers no meaningful alteration.
– The topology of the tracked object can change in a periodic way. (i.e.: if the tracked object splits, it will merge again in the following frames.)
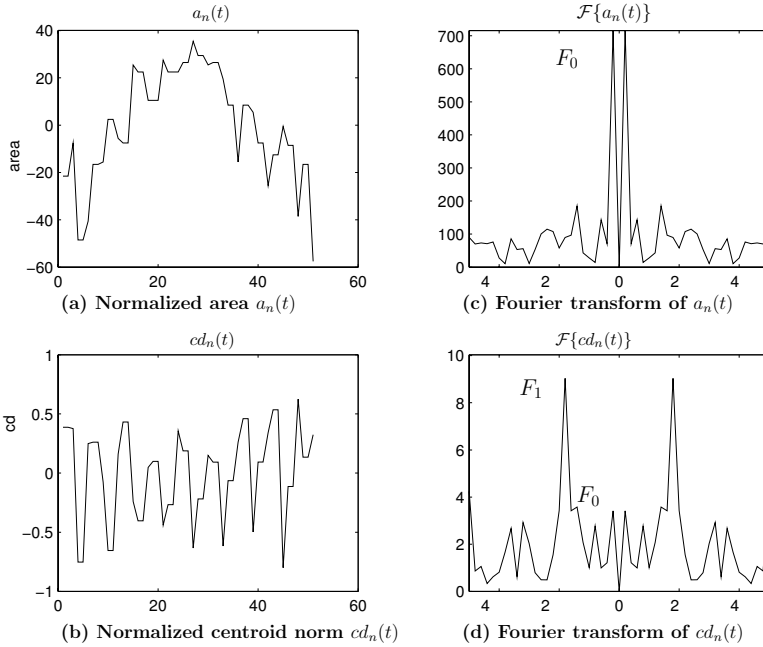
Fig. 2. Normailzed centroid norm $cd_n(t)$ and area $a_n(t)$ functions (Figures (a) and (b)) where t is the video frame number. Fourier transform of the normalized centroid norm $\mathcal{F}\{cd_n(t)\}$ and normalized area $\mathcal{F}\{a_n(t)\}$ (Figures (c) and (d)).
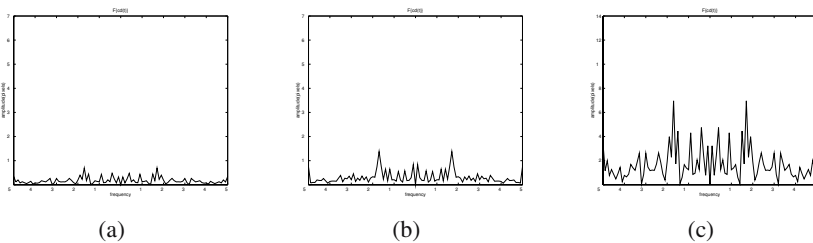

Fig. 3. Fourier transform of the normalized centroid norm of three membranes, (a) near the beating center (20 pixels or less), (b) middle distance from the beating center (between 40 and 60 pixels) and (c) far from the beating center (more than 200 pixels)

On the other hand, analyzing objects with several distances to the beating center, an increment on the amplitude of $cd_n(t)$ can be noticed when the object is near the beating center but no alteration on the beating frequency is registered. This analysis is illustrated in Fig. 3, where the Fourier transform of the normalized centroid norm of three membranes is shown. The first membrane 3(a) is less than 20 pixels far from the beating center, the second membrane 3(b) is between 40 and 60 pixels far from the beating center and the third membrane 1(c) is more than 200 pixels far from it.

3   Synthetic Data Processing and Model Validation

In this section the design and results of a Monte Carlo experiment based on the previously inferred characteristics of the problem are presented. This experiment is used to validate the procedure that finds the beating rate, taking into account the following characteristics of this particular problem:

- Each tracked object has an ondulatory movement along a line. The frequency of this movement is the application's goal.
- The amplitude of each ondulating object's movement is different proportionally with its distance from the beating center.
- The amplitude of each ondulating object's movement suffers slight variations in the same sequence (ie: noise).
- Each tracked object does not follow the previously mentioned line precisely. In fact, each object has slight displacements from the line (ie: noise).
- The area of each tracked object does not have meaningful ondulatory variations.

3.1   Monte Carlo Experiment Design

According to the presented characteristics, a Monte Carlo experiment was designed. This experiment, given the image dimensions, the number of frames to be generated, a beating center and a frequency, generates a number of non-overlapping circular particles. Each particle has a fixed area and moves in an ondulatory manner over a line oriented towards the beating center. This movement has a fixed amplitude, which can be different for each particle. Furthermore, a noise component is added to the position and frequency of each particle. The area the covered by the particles is between 20% and 80% of the image. The frame sequences are generated by Algorithm 1. The random variables used in the algorithm are the following:

- $X_c \sim \mathcal{N}(\frac{w}{2}, 0.6w)$
- $Y_c \sim \mathcal{N}(\frac{h}{2}, 0.6h)$
- $X_p \sim \mathcal{U}(0, w-1)$
- $Y_p \sim \mathcal{U}(0, h-1))$
- $Amplitude \sim \mathcal{N}(0.001, 0.05)$
- $Area \sim \mathcal{U}(minArea, maxArea)$
- $X_n \sim \mathcal{N}(0, 0.05w)$
- $Y_n \sim \mathcal{N}(0, 0.05h)$

where $\mathcal{N}$ and $\mathcal{U}$ represent normal and integer uniform distributions.

Once the Monte Carlo frame sequence is generated, a ROI is inferred for each particle in the same manner that the ROI for every tracked membrane on Sect. 2 was described. The centroid of each particle is tracked and analyzed as presented in Sect. 2.

3.2   Monte Carlo Results

Using the algorithm described on Sect. 3.1, 10000 frame sequences were generated with the following characteristics:

- 50 frames.
- 640×480 pixels per frame.
- The number of particles ranged from 8 to 20 with a uniform integer distribution.
- The beating frequency is a random variable $freq \sim \mathcal{N}(\frac{148}{60}, \frac{20}{60})$ where a common beating frequency is $148 \frac{beats}{minutes}$.

Algorithm 1 Monte Carlo frame sequence generation

1: w, h dimensions of the image I.
2: nf the number of frames to be generated.
3: qty the quantity of non-overlapping particles.
4: $freq$ the beating frequency.
5: $c \leftarrow (X_c, Y_c)$ is the beating center.
      ▷ pattern signal generation
6: $s_t \leftarrow sin\left(2\pi\frac{t}{nf}freq\right), t = [0 \ldots nf]$ is the discretized pattern signal.
      ▷ minimum and maximum areas for the particles
7: $minArea \leftarrow \min\left(\frac{(0.2)w\,h}{qty}, 2\right)$
8: $maxArea \leftarrow \max\left(\frac{(0.8)w\,h}{qty}, 4\right)$
      ▷ generation of the particles
9: $P \leftarrow \emptyset$
10: **repeat**
11:     $pos(p) \leftarrow (X_p, Y_p)$
12:     generate a particle p at (x, y).
13:     $area(p) \leftarrow Area$
14:     $amp(p) \leftarrow \min(w, h)Amplitude$ is the amplitude of p's movement.
15:     $dir(p) \leftarrow \frac{pos(p)-c}{|pos(p)-c|}$ is the unit vector which represents the direction of the p's movement.
16:     **if** $(\forall p' \in P)$ p does not touch p' while they're moving **then**
17:        add p to P.
18:     **end if**
19: **until** $|P| = qty$
      ▷ sequence generation
20: Set every frame $f^t, t \in [1 \ldots nf]$ black.
21: **for** every frame f^t **do**
22:     **for** every particle $p \in P$ **do**
23:        draw a white circle of area $area(p)$
24:        at $pos(p) + dir(p)amp(p)s_i + (X_n, Y_n)$
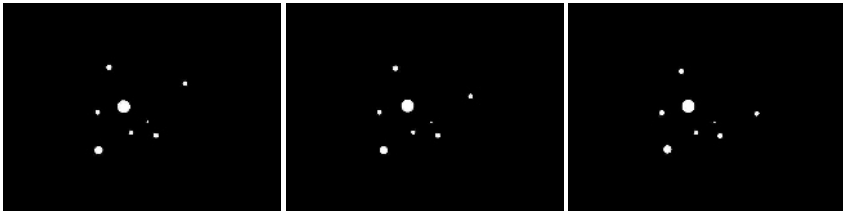25:   **end for**
26: **end for**

Fig. 4. Three frames of a Monte Carlo frame sequence generated with the algorithm 1 and the parameters specified in Sect. 3.1

Three frames from a Monte Carlo frame sequence generated with the stated parameters and algorithm 1 are shown in Fig. 4. The frequency measurement error for the n-th Monte Carlo experiment is calculated as follows. For each particle $p \in P^n$ is calculated, where P^n is the set of particles of the n-th Monte Carlo experiment and $|P^n|$ the number of particles in the set the main frequency $F_1^{n,p}$ is calculated as stated in Sect. 2:

$$e^n = \sum_{p \in P^n} \frac{(F_1^{n,p} - freq^n)^2}{|P^n|}$$

Then, the mean square difference between the true beating frequency $freq^n$ of the experiment and the frequency $F_1^{n,p}$ is calculated. A histogram presenting the frequency measuring error for each Monte Carlo experiment is presented in Table 1 where it can be seen that the error more than 94% of the experiments is zero.

Table 1. Table presenting the frequency measurement error for each Monte Carlo experiment. It can be seen that the error in more than 94% of the cases, ranges to 0.

Frequency measurment error(e)	Percentage of cases
00	94.46
04	1.23
08	0.88
12	0.80
16	0.94
20	1.16
24 and more	0.53

4   Real Data Processing

The proposed tracking algorithm based on the *fast marching* method presented in [4,5] and applied to image segmentation in [6] is presented in this section. This algorithm is developed in order to relax the restrictions placed on the algorithm presented in Sect. 2, where a ROI in which the membrane to be tracked had to be the only object over the threshold in every frame of the sequence. The short response-time requirement for the application stated in Sect. 1 makes the choice of a tracking algorithm critical. A comprehensive review of tracking techniques is presented in [7], nevertheless the techniques reviewed in that work are bounded by a high-computational cost or by topology dependence; none of these characteristics can be afforded in this work due to the non-functional requirements on the application and the possibility of a topology change in the thresholded tracked membrane. A tracking algorithm based on the *fast marching* algorithm is presented in this section.

4.1   Fast Marching Algorithms

In this section the algorithm introduced in [6,4] is briefly described. Let Γ be the initial position of a curve and let F be its speed in the normal direction. In the level set

perspective [8], Γ is viewed as the zero level set of a higher dimensional function $\psi(x, y, z)$ and its movement equation is:

$$\psi(\Gamma(t), t) = 0$$

where $\Gamma(t)$ is the Γ curve in instant t. Then, by chain rule,

$$\psi_t + \nabla\psi(\Gamma(t), t) \cdot \Gamma'(t) = 0$$

and taking F as the speed in the outward normal direction

$$F = \Gamma'(t) \cdot \frac{\nabla\psi}{|\nabla\psi|},$$

an evolution equation for the moving surface can be produced [8,4], namely

$$\psi_t + F|\nabla\psi| = 0. \tag{1}$$

Consider the special case of a surface moving with speed $F(x, y) > 0$. Now, let $T(x, y)$ be the time at which the curve Γ crosses a given point (x, y). The function $T(x, y)$ then satisfies the equation

$$|\nabla T(x, y)| F(x, y) = 1; \tag{2}$$

this simply says that the gradient of arrival time is inversely proportional to the speed of the surface. The way of approximating the position of the moving curve in the *fast marching* approach is to explicitly construct the solution function $T(x, y)$ from equation (1). The algorithm which presents this function reconstruction is presented on [6,4].

Implementation of the Algorithm. In order to solve equation (2) an approximation to the gradient given by [8,4,5],

$$\left[\begin{array}{c} max(D_{ij}^{-x}, 0)^2 + min(D_{ij}^{+x}, 0)^2 \\ + max(D_{ij}^{-y}, 0)^2 + min(D_{ij}^{+y}, 0)^2 \end{array} \right]^{\frac{1}{2}} = \frac{1}{F_{ij}}, \tag{3}$$

is used, where D^- and D^+ are the backward and forward difference operators for $T(x, y)$. The central idea behind the *fast marching* method is to systematically construct the solution T using only upwind values. Observe that the upwind difference structure of equation (3) allows to propagate information "one way", that is from smaller values of T to larger values. Plainly speaking no point in T can be affected by points of T containing larger values, taking this "causality principle" [5] into account, a method for calculating the values of T is presented in the algorithm 2.

By implementing the $Trial$ set with a particular heap sort structure [4] this algorithm takes $O(N \log N)$ computations, where N is the number of points in the grid.

4.2 Segmentation Using the Fast Marching Method

Given an image function $I(x, y)$ the objective in segmentation is to separate an object from the background. This can be done applying an image-based speed function $k_I >$

Algorithm 2 *fast marching* algorithm

1: Given a grid with the initial values of T where the points in the initial curve have a known
    value and every other point has an ∞ value.
2: Tag all points in the initial boundary value as *Known*.
3: Tag all points with a neighbor in *Known* as *Trial*.
4: Calculate the value of T for each point tagged as *Trial* with equation (3)
5: Tag all other grid points as *Far*
6: **while** there are points in $Trial$ **do**
7:      Let A be the $Trial$ point with the smallest value.
8:      Add A to $Known$ and remove it from $Trial$.
9:      Tag as $Trial$ all points that are not $Known$. If the neighbor is in Far, remove.
10:     Recompute the values of T at all $Trial$ neighbors of A according to equation (3).
11: **end while**

---

0 such that it controls the outward propagation of the initial curve in a way that it stops in the vicinity of shape boundaries. Mathematically this corresponds to solving equation (2) where $F_{ij} = k_{Iij}$:

$$|\nabla T| = \frac{1}{k_{Iij}},$$

where k_{Iij} approaches to 0 as it gets closer to shape boundaries; in every other case it approachos to 1. In the case of this particular work where $I(x, y)$ is a thresholded binary image k_{Iij} can be defined as

$$k_{Iij} = \frac{1}{I_{ij} + \epsilon}, \quad \epsilon > 0,$$

where I_{ij} is 1 inside the object to be segmentated and 0 outside of it.

4.3   Tracking by Fast Marching Methods

The *fast marching* method presented on Sect. 4.1 propagates a curve. The segmentation task, introduced in Sect. 4.2, is performed by manually selecting a set of initial curves or points inside the objects to be segmentated and then propagating these curves until the border of the object is reached. This process is illustrated on Fig. 5

The facts that the area of the objects do not suffer a meaningful variation and the displacement performed by the objects between frames is bounded were stated in Sect. 2. Thus, the initialization of the propagating curve inferred from the previously segmented frame is obtained eroding the resulting curve with a circular kernel, see Fig. 5(d), obtaining an initialization curve that can change its topology. For each segmented membrane, the normalized centroid norm $cd_n(t)$ for the video sequence is calculated in order to find the beating rate of the membrane. This is done by analyzing the Fourier transforms of these functions and taking the frequency F_1 as stated in Sect. 2.

The overall cost of processing a membrane in a frame can be calculated as the cost of segmenting the membrane $O(N \log N)$ plus the erosion $O(Nk)$, where N is the area of the segmented membrane and k is the size of the erosion kernel. The cost of calculating the centroid is $O(N)$ -linear in the area of the membrane-, then the overall cost of processing a frame is $O(N \log N + Nk + N) = O(N \log N + N)$.
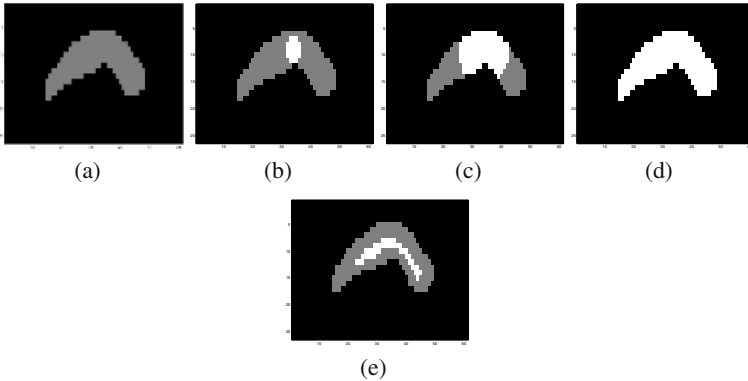
(a)   (b)   (c)   (d)

(e)

Fig. 5. Segmentation of a membrane using the *fast marching* method. The membrane to be segmentated is shown in (a). An initial curve is selected and propagated until it reaches the border of the object. The evolution of the segmentation progress is shown in (b),(c) and (d). In (b) The membrane to be segmented in grey and the initial curve in whitem in (c) the evolving curve-white- growing inside the membrane-grey-. In (d) curve has grown to fill the membrane. After the segmentation, the obtained figure is eroded in order to initialize the next frame, shown in white in (e), is used as the initial curve for the next frame.

4.4 Real Data Processing Results

The algorithm described in Sect. 4.3 was applied over several rat cardiomyocyte culture frame sequences with a sampling frequence of 25 frames per second and 10 frames per second. Several membranes were tracked in each video. The results were compared with the measures obtained with the standard method by the staff of the Molecular Biology of Chagas Disease Laboratory at INGEBI-CONICET Argentina. Their method consists of a trained member of the staff who measures the beating rate with a cronometer by looking at the culture over a microscope. The mean measuring difference was of $0.1Hz$ in the 25 frames per second videos and of $0.77Hz$ in the 10 frames per second videos. This measuring difference are one order of magnitude smaller than the measured beating rates.

5   Conclusions and Further Work

In this work the base for an image processing application to measure the beating rate of a rat cardiomyocyte culture is presented. It can be divided in three stages. In the first stage, a preliminary analysis of the problem is introduced, isolating the main characteristics of the problem. In the second stage, a Monte Carlo experiment is designed using this characteristics to evaluate the robustness and validity of the algorithm which attains a rate above 87% of measurements with zero measurement error. Finally, a short-response time $O(T(N \log N + N))$ algorithm for tracking cardiomyocyte membranes is presented, where N is the maximum surface of the tracked membrane in pixels and T is the number of frames in the video sequence. This algorithm is implemented in a testbed application and the beating rate measures obtained with it were compared against the

measures obtained by the standard procedure over several videos. This comparison resulted in a mean error of $0.1Hz$ in 25 frames per second videos and in a mean error of $0.77Hz$ in 10 frames per second videos representing a measuring error one order of magnitude smaller than the measures taken by both methods.

Two tasks are currently being addressed as further work: In order to improve the comparison between the standard method and the testbed application more video sequences are being generated and measured by the Chagas Disease Laboratory, the results of this measures will be compared against the measures of the proposed application using the Bland and Alman method [9]. Furthermore this algorithm will be implemented in an end-user application for its use at the Molecular Biology of Chagas Disease Laboratory at INGEBI-CONICET Argentina in Chagas' disease investigation.

References

1. Bonomo, R., Salata, R.: American Trypanosomiasis (Chaga's Disease: Trypanosoma cruzi). In: Nelson Textbook of Pediatrics. W. B. Saunders (2000)
2. Kirchhoff, L.: Trypanosoma species (American Trypanosomiasis, Disease): Biology of Trypanosomes. In: Principles and Practice of Infectious Diseases. Churchhill Livingstone (2000)
3. Levin, M.: Proyecto genoma de trypanosoma cruzi asociado al proyecto de genoma humano. In: V Muestra de Ciencia y Técnica, IX Jornadas de Becarios. (1996)
4. Sethian, J.A.: A fast marching level set method for monotonically advancing fronts. Proc. Nat. Acad. Sci. **93** (1996) 1591 1595.
5. Sethian, J.A.: Level Set Methods and Fast Marching Methods. second edn. Cambridge University Press (1999)
6. Malladi, R., Sethian, J.A.: An o(n log n) algorithm for shape modeling. Proc. Nat. Academy of Sciences, USA **93** (1996) 9389–9392
7. Paragios, N.: Geodesic Active Regions and Level Set Methods: Contributions and Applications in Artificial Vision. PhD thesis, INRIA Sophia Antipolis (2000)
8. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. Journal of Computational Physics **79** (1988) 12–49
9. Bland, J.M., Altman, D.G.: Measuring agreement in method comparison studies. Statistical Methods in Medical Research **8** (1999) 135–160