Université de Technologie de Compiègne
Compiègne
France

VICOMTech
San Sebastian
Spain

High content analysis of the zebra fish embryo, with multimedia information

STRICTLY CONFIDENTIAL

Student	:	Alexis Verbeke
University supervisor	:	Vincent Fremont
Company supervisor	:	Fernando Boto
UTC Semester	:	GI04
UTC Speciality	:	ICSI
Semester	:	Spring 2007

Acknowledgements

First I would like to thank Julián Florez, Jorge Posada, respectively general manager and assistant director of the research center VICOMTech, for welcoming me for my work placement.

Then I have to thank the whole team of VICOMTech, for the enjoyable atmosphere they create, for their availability for any help, and for having integrated me so nicely in their team.

In particular, I would like to thank Céline Paloc, the responsible of the biomedical department, for welcoming me in the department, and for her help that made all this possible.

And above all, I would like to give thanks to Fernando Boto, my supervisor, for his patience, his precious help, his bounty in the sharing of his knowledge, and of course for being constantly cheerful, which makes me spend my internship in the best conditions.

Abstract

This report deals with a 4th year university work placement on image processing, made during the spring of year 2007, in a spanish research center called *VICOMTech*. During 6 months, image processing methods have been developed, working inside a research project in biomedical field called *Zebrascreen*.

Methods to calculate the heart rate and segment the heart chambers from a zebra fish heart, and methods to extract features and classify zebra fishes have been developed. Every tasks were made from fluorescent videos of the heart, and photos of the fish. This report explains the unfolding of the work placement's events, details the methods and algorithms developed, and analyzes the results obtained.

The language C++ has been used, because it is well known that it fits best the problem of image processing. Microsoft Visual Studio was used as the developing environment, and MFC libraries to build Graphic User Interfaces. The library used to handle with image processing was OpenCV [8], for its simpleness of use, and its reputation of producing accurate results. For data mining and classification matters, the Torch library [11] has been used, for its well known performance.

Aperçu

Ce rapport traite d'un stage universitaire de 4ème année portant sur le traitement d'images, et effectué dans un centre de recherche espagnol, dont le nom est *VICOMTech*. Pendant 6 mois, des méthodes de traitement d'images ont été dévelopées au sein d'un projet de recherche du département biomédical, appelé *Zebrascreen*.

Des méthodes pour calculer le rythme cardiaque et pour segmenter les cavités du coeur ont été implémentées, ainsi que des méthodes d'extraction de caractéristiques et de classification du poisson zèbre. La base de toutes les méthodes sont des vidéos fluorescentes de coeur de poisson zèbre, et des photos du poisson lui-même. Ce rapport expose le déroulement de ce stage, détaille les méthodes et algorithmes développés, et analyse les résultats obtenus.

L'environnement de dévelopement utilisé au sein du projet est Microsoft Visual C++, ainsi que la librairie MFC pour créer des interfaces graphiques. Le langage C++ a été choisi car il est réputé le mieux adapté au traitement d'images. Pour ce dernier, la librairie OpenCV [8] a été employée car réputée simple d'utilisation et fournissant de bons résultats. Pour les problèmes de fouille de données et de classification la librairie Torch [11] a été choisie pour ses performances reconnues.

Contents

Chapter 1

Introduction

From the 12th of February to the 27th of July 2007, I carried out a work placement in a Spanish research center called VICOMTech, specialised in image processing, within the biomedical department. I chose this internship first because this area interested me a lot, even though I did not have any knowledge in image processing and medical imaging. Then learning Spanish and discovering a new country and culture was something really attractive to me. Finally I knew that in a research center I would get a good technical knowledge, and as the first work placement of my studies, I thought it was the best choice.

I was integrated in a project, which dealt with image processing applied to medical imaging, and more precisely with high throughput screening methods, in collaboration with a biotechnological company of new creation. As an innovative research project, it aimed to determine zebra fish phenotypes from digital images and videos.

My role in the project was to develop image processing methods, to test them, to integrate them in graphic user interfaces, and write documentations for the users. These methods aimed to solve some specific tasks included in the project. In most cases, the guidelines of the new methods to implement were given by my supervisor, but part of my work was also to work on finding new ones, and discuss them with him.

The first part of this report will present the group, the company, and the department where I worked. Then, in a second part, the project wherein I took part will be described. Afterwards the work placement's events will be explained in details. Finally the contributions of this training period will be exposed and a conclusion on the whole internship will be made.

Chapter 2

Presentation

2.1 The INI-Graphics group

2.1.1 Presentation

The INI-GraphicsNet, the International Network of Institutions for advanced education, training and Research & Development in Computer Graphics technology, systems and applications, builds worldwide one of the greatest centers for new Media and forms of communications, including the related of information and application technology.

At the locations Darmstadt (Germany), Rostock (Germany), Guimares (Portugal), Providence and Omaha (USA), San Sebastian (Spain), Seoul (South Korea), Singapore and Trento (Italy), more than 370 employees and about 550 part-time researchers gather an annual volume of research of 40 million EURO.

The center of this network of research is Darmstadt. Beside the Fraunhofer Institute for Computer Graphics (IGD) it comprises also the Computer Graphics Center (ZGDV) and the Interactive Graphics Systems Group (GRIS) as well as the INI-GraphicsNet Foundation.

The main objective of the Foundation is to promote scientific research and development within the field of computer graphics. The Foundation plays a significant role in the transfer of technology by supporting the commercialization of research results and through the establishment of spin-offs.

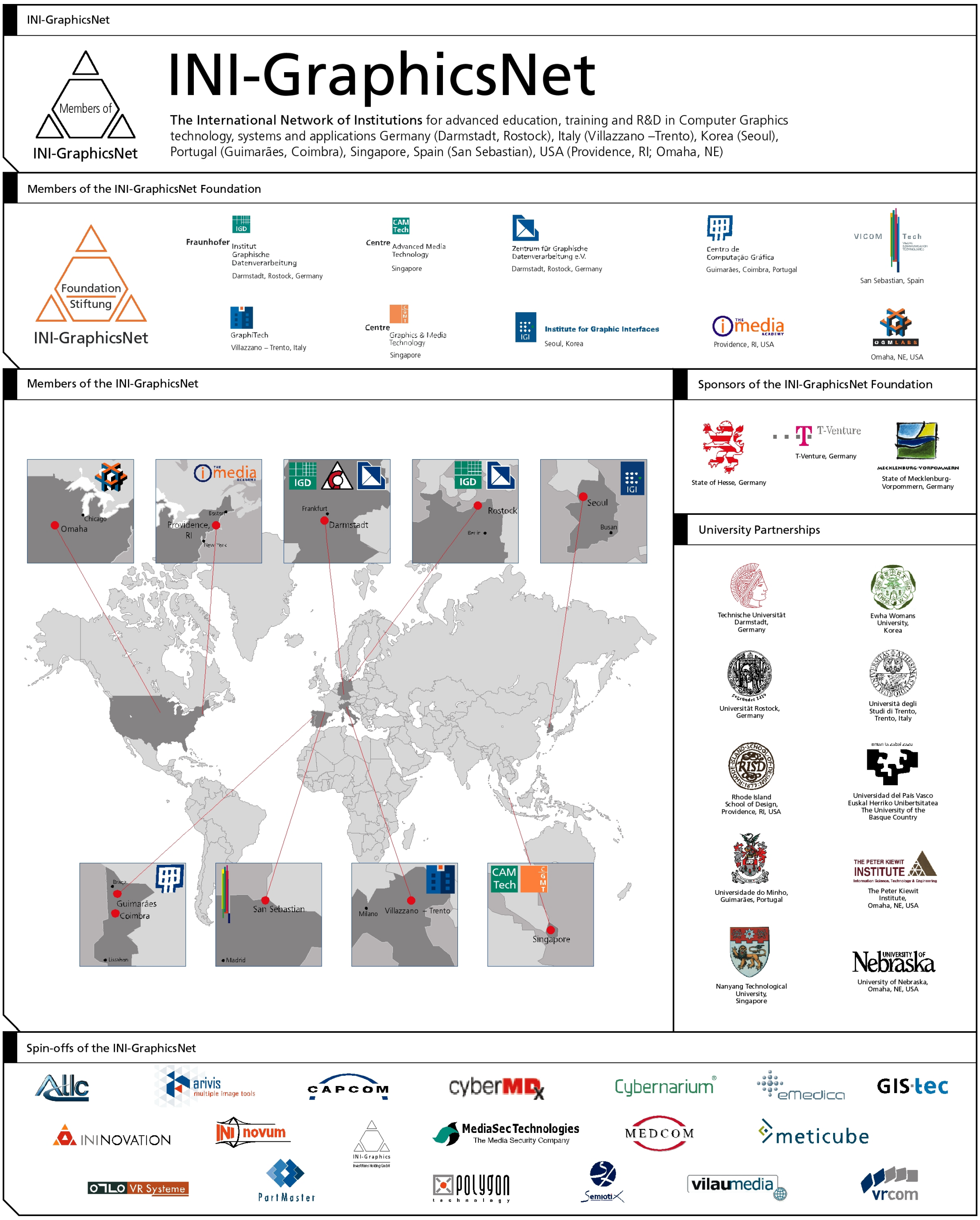


Figure 2.1. *Members, spin-offs, sponsors, and university partners of the INI-GraphicsNet (Foundation) // Membres, start-up, sponsors, et universités partenaires de (la fondation) INI-GraphicsNet.*

2.1.2 Organisation

To achieve its goals, the INI-GraphicsNet Foundation has a well-defined structure. The members, spin-offs, sponsors, and universities have each one their own place in the foundation, so that the process from the research till the market is as easy and direct as possible. The organisation intents to serve as a technology bridge between universities and industries. That means, working on applied research to allow the industry to access the basic research knowledge developed by universities.

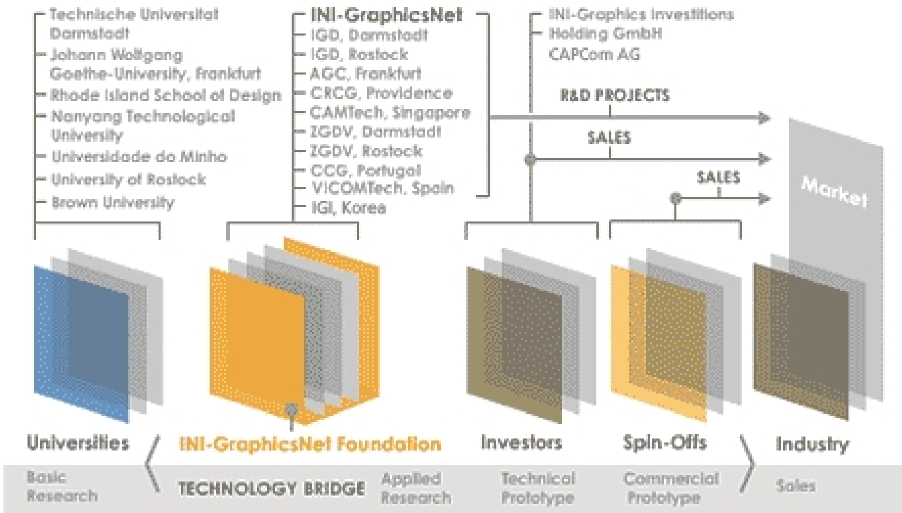


Figure 2.2. *Organisation of INI-GraphicsNET // Organisation de INI-GraphicsNET.*

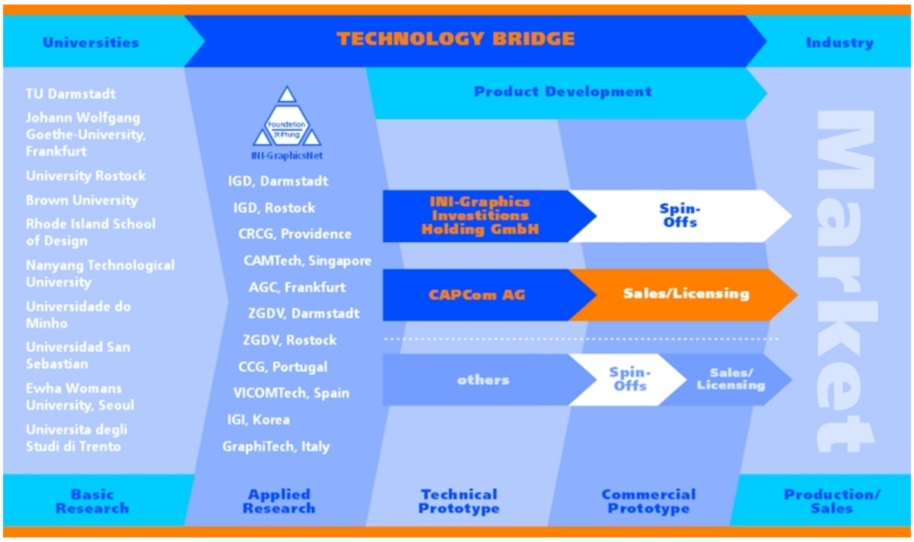


Figure 2.3. *Schema of the "Technology Bridge" // Schéma du "Pont Technologique".*

2.2 VICOMTech

2.2.1 Presentation

VICOMTech, Visual Communication and Interaction Technologies Centre, is an applied research centre for Interactive Computer Graphics and Multimedia. It is a non-profit association, founded by the INI-GraphicsNet Foundation, and the Basque Television, Radio and Broadcasting group EiTB in 2000.

Since the beginning, VICOMTech belongs to Saretek, the local science, technology and innovation network. Furthermore, the acknowledgement VICOMTech has obtained in these years has paved the way to incorporate into the IK4 alliance and also to be classified as an Innovation and Technology Centre by the Spanish Ministry of Education and Science.

Today VICOMTech runs about 60 applied investigation projects per year with local, national and European industry partners and administrations, compromises about 50 researchers, and makes an annual turnover of 2 million euros.

2.2.2 Objectives

VICOMTechs research profile is to serve as a bridge between the local and the international environments. This philosophy of applied research helps the local companies to have new opportunities to access a worldwide environment, and to benefit from the latest technological advances in the international context. At the same time, VICOMTechs participation in international projects complements and improves the main local activity of applied research.

VICOMTech's mission can be summarized in these four points :

- To come up to the needs of companies and institutions in Applied Investigation, Development and technological Innovation in Computer Graphics field.
- To develop Visual Interaction and Communication Technologies based upon the latest advances in the field of computer graphics and its deployment in specific applications and products in conjunction with the Industry.
- To contribute to the general body of knowledge on the field through the formation of new researchers and the publication of its results in internationally recognised journals and conferences.
- To pursue quality and excellence in scientific, technical, customer service, and internal organization aspects.

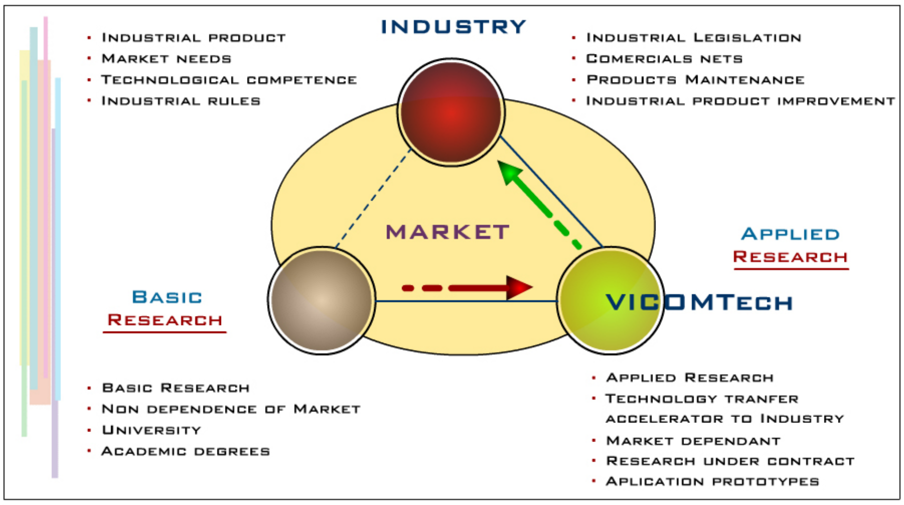


Figure 2.4. *The position adopted by VICOMTech between the basic research and the industry to achieve its objectives // La position adoptée par VICOMTech entre la recherche fondamentale et l'industrie pour atteindre ses objectifs.*

2.2.3 Areas

VICOMTech now works in 5 application areas, divided in departments :

- Digital TV and Multimedia Services : this area explores the possibilities offered by the synergies between Digital TV and Computer Graphics. Currently, VICOMTech is developing interactive applications and services for TV in order to bring the Information Society to a wider spectrum of citizens, not necessarily familiar with PCs.

- Tourism, Heritage, and Creativity : The aim of this area is to support the industry of tourism and to promote both archaeological and heritage special interest sites, as well as the promotion of the GIS systems to represent, analyse and manage geographical information data by the means of advanced computer graphics techniques (Augmented Reality and Virtual Reality), visual interaction and communication.

- Interaction for Edutainment and e-Inclusion : This area focuses on the fields of education and leisure through the possibilities of audiovisual technologies and advanced user interfaces. Multimedia technologies, Conversational 3D avatars in real time, virtual environments and e-Learning are some of the main research lines in this area.

- Industrial Applications : VICOMTech brings (via R & D projects) the possibility to use advanced Computer Graphics technologies to different sectors of the industry and the society. Ex: Virtual and Augmented Reality, image and 3D Graphics processing, digital security, multimedia interaction technologies, etc.

- Biomedical Applications : the area where I made my work placement (see section 2.3 for detailed information).

2.3 The Biomedical Application department in VICOMTech

The Biomedical Applications Department of VICOMTech carries out research and development for the healthcare and biotechnology sectors. The main research lines include the most recent advances in image analysis (image processing, segmentation, registration, fusion), visualisation (virtual reality, augmented reality) and biomedical information management (transmission, representation, standards, interface).

The unit has established close collaboration with local hospitals and private health clinics, and is working in strong synergy with a wide variety of clinicians (hepatic surgeons, pathologists, radiologist, radiotherapists).

The objective of the department is to become a source of new technological and leading developments for healthcare services, and to satisfy the ICT needs of the biomedical community. In order to market its technology, eMedica S.L. has been founded in 2004 as a spin-off of VICOMTech.

The department can be divided into the 5 following working lines :

- Image Analysis : automatic image segmentation, multi-modal image registration, measurement and characterization of biological processes.
- Advanced Interface : new visualisation and interaction techniques like virtual reality and collaborative environment.
- Data Mining and Machine Learning : knowledge representation from large databases, image classification and retrieval, decision support system, biomedical knowledge discovery...
- Simulation : physically-based simulation, soft tissue modelling, interactions and surgical tasks...
- Telemedicine : tele-assistance and remote control, data transmission and security...

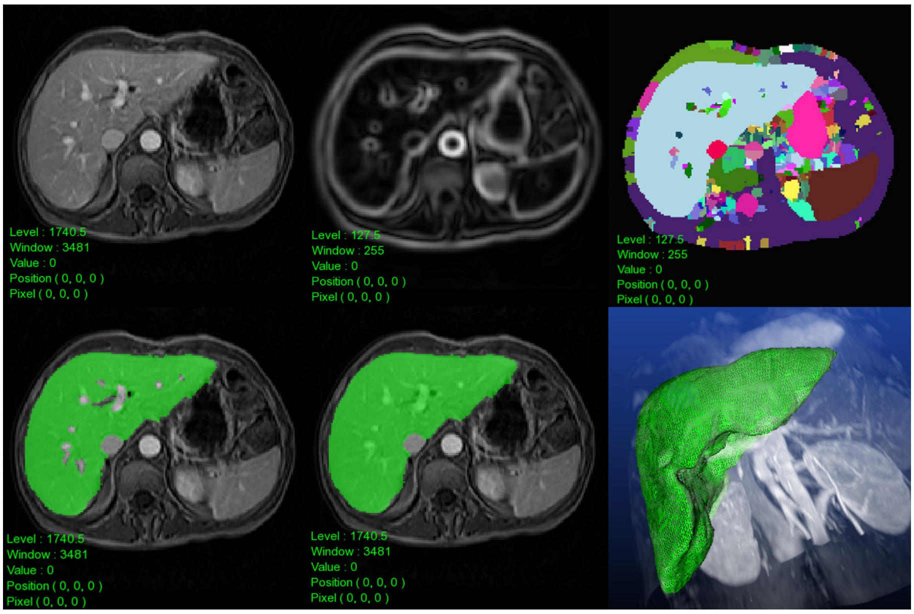


Figure 2.5. *Segmentation and 3D Visualisation of the liver from MRI image datas // Segmentation et visualisation 3D du foie à partir d'images IRM.*

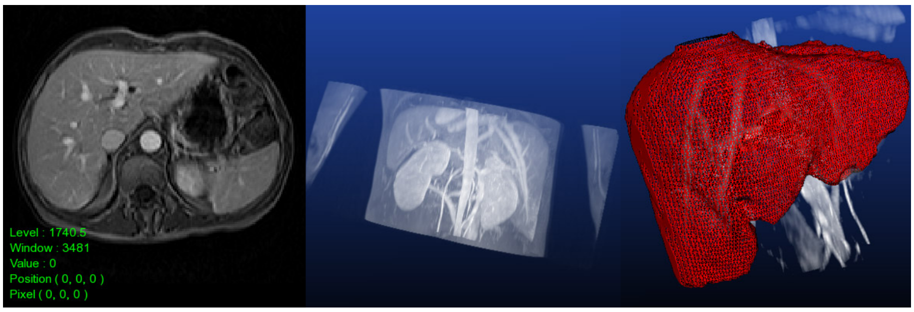


Figure 2.6. *Different kinds of visualisation of a segmentation : (from left to right) 2D, 3D, and 3D with surface // Différents types de visualisation d'une segmentation : (de gauche à droite) 2D, 3D, 3D avec surface.*

Chapter 3

The Zebrascreen project

"High-Throughput and Content Screening Platform for use with Zebra fish".

3.1 Problem presentation

The ZEBRASCREEN project aims at developing a whole animal high-throughput screening platform for testing drug effects and drug toxicity. High-throughput screens are usually using in-vitro assays (cells), whereas in the zebrascreen project a platform for zebra fish embryos in 96-well microtiter plates is used.

While tremendous advances in imaging hardware make now possible the rapid acquisition of thousands of high-resolution images of the organisms, the current method of analysing these images is still mainly the manual and visual inspection. The lack of a computational approach to identify characteristics similarities causes a bottleneck for large-scale experiments and is therefore an impediment to advancement in biology research, including genomic and drug discovery.

Automated systems for the interpretation of biological images from assays would provide three important advantages over manual practice : high-throughput performance ; quantitative and reproducible identification of phenotypes ; and therefore consistent and unbiased phenotypic information in databases.

The automatic classification of phenotypes is based on the extraction of quantitative parameters from the digital image. Such features are based on morphology, texture, or other grey-level-based measures. Selecting subsets of the many possible features is then necessary to reduce the complexity of the classifier.

3.2 Objectives

In this project, VICOMTech proposes to explore the use of advanced image analysis methods for automated identification and classification of phenotypes. Such methods should allow to quantitatively measure complex phenotypes from thousands of images in a high throughput manner.

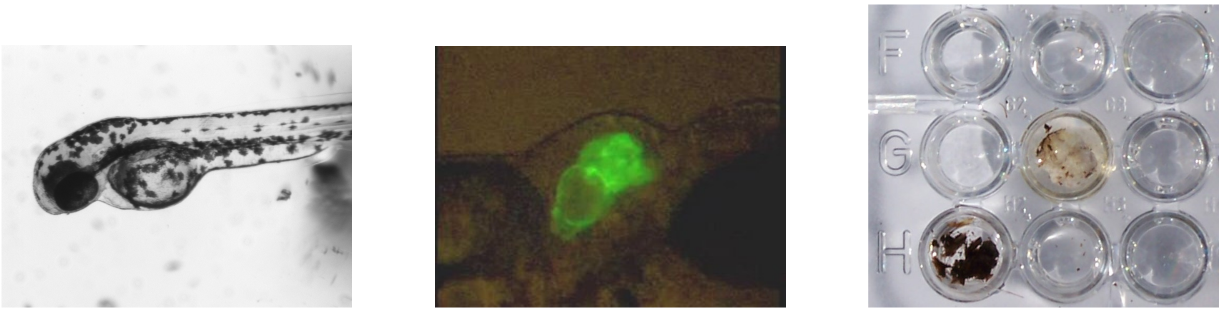
In the context of this project, VICOMTech is working in collaboration with BIOBIDE, a new biotechnological company dedicated to the automatic screening of new potential drugs using the zebra fish as animal model. VICOMTech is currently developing a prototype aiming at identifying and classifying cardiac phenotypes of the zebra fish, including morphological as well as functional phenotypes.

3.3 The zebra fish : a model organism

The zebra fish is a common and useful model organism for biomedical studies. Indeed, it has a lot of advantages :

- It is a vertebra which has a lot of homologies with other types of vertebra, included the human (till 85% of homologies).
- Its development is really fast : in 72 hours it can be considered as a larva.
- The embryos are transparent, which allows to see easily the intern organs.
- The embryos and larvae are smalls, which makes easier to do in a laboratory the fish culture, the screening, etc...

Therefore it is really interesting to use this fish as a model, to test drugs on it and analyse the results. It is also clearly well-adapted to high-throughput screening methods.



(a) *The zebra fish* **(b)** *The fluorescent zebra fish heart image* **(c)** *High throughput screening plate*

Figure 3.1. *Zebrascreen project overview // Vue d'ensemble du projet Zebrascreen*

Chapter 4

The work placement

4.1 The subject defined before my arrival

Before my arrival in VICOMTech, the subject was not clearly defined. The exact title was *"Visual Communication and interaction technologies applied to biomedical community"*. I so already knew I was going to work in the biomedical department of the company, developing image processing programs. But nothing more precise than this was established.

I spent my first two days in VICOMTech reading documentation on the company, the department and the INI-Graphics group. So I could know exactly on what works my department, how he is integrated within the others, see examples of projects, concerning biomedical applications as well as other fields, etc. After these two days, we had a small meeting with my supervisor and he explained me what I was going to do. Then we talked about it to see if it was what I expected and if I liked it, which was found to be true.

4.2 The real subject

The subject has been soon well defined, so that I have been able to begin working as early as possible. The title is *"High content analysis of the zebrafish embryo, with multimedia information"*. It has been divided in 5 different parts :

- Study of different Computer Vision methods : image segmentation and feature extraction.
- Implementation of image segmentation methods.
- Implementation of feature extraction methods.
- Integration of all the methods.
- Validation and tests.
- Documentation.

4.3 My position in the biomedical department

Two days after my arrival in the biomedical department, a meeting has been organized with all the people of the department to introduce ourselves (we were several new interns and researchers), so that everybody could present himself, explain what he was exactly doing. Therefore I have been able to know everybody and situate myself between ourselves immediately.

In the department are working 6 researchers (including the responsible of the department), and 3 interns. I was assigned to work on only one project, but I have heard a lot on other ones, due to the nice relations between everybody inside the department, and to the meetings and presentations we had. Indeed, a bi-weekly department meeting has been established since beginning of june, which was really useful to know the advancement of each project, to discuss about important points, to feel working as a team, etc.

Besides, the position I had with my supervisor was something I have appreciated a lot. It was as if we were working together, of course one giving a lot more advises and ways to take, but it was completely different of what I expected as an intern-supervisor "hierarchical" relationship. I felt myself really comfortable, able to ask everything I needed, and to propose the ideas I had.

4.4 Objectives

The objectives of the work placement are diversified, because it depends from which point of view they are seen. According to VICOMTech, the objective is that the intern integrates himself well in the work team of the project, that he produces an interesting work and so that he can be considered as a help for the project. Besides, as he is supposed to be young and dynamic, VICOMTech can also expect from him new ideas and new approaches of the problems encountered. According to the university, the objectives are that the intern acquires work experience in programming in general, as well as in team work capability, communication, and autonomy. Then, according to the intern, the objectives are much more diversified, because it regroups all the last ones, and includes as well improving foreign languages, discovering a new culture in a new country, and everything that could be positive for his future career.

4.5 Plan

In the appendix A, the work plan defined at the beginning of the work placement is presented. The real plan, representing the unfolding of the work's events, is presented on the figure 4.1. It is different from the one described in the work plan, because first the project felt a bit behind, and so the specific tasks to do were revised and changed a bit. Also the work plan is defined for 6 months exactly, whereas the work placement lasted 24 weeks, which corresponds to 5 months and a half.

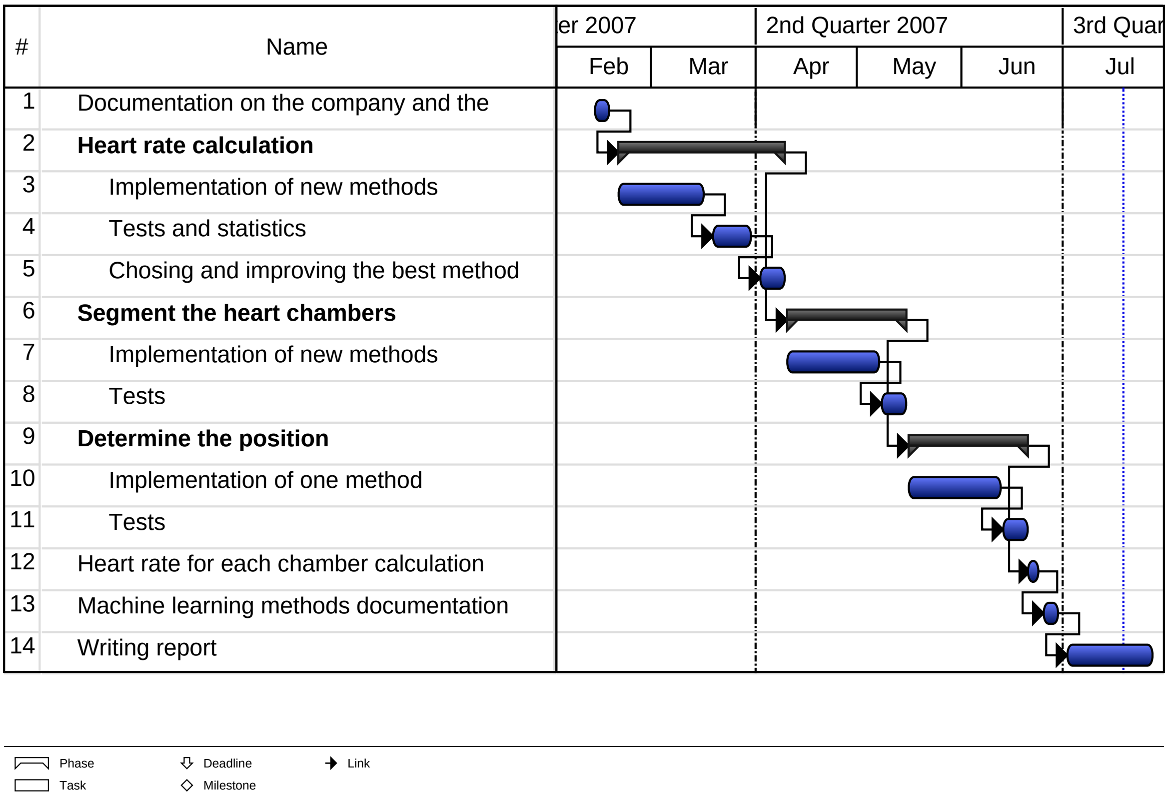


Figure 4.1. *Real plan of the work placement // Planning réel du stage.*

4.6 Bases in image processing, and medical imaging

The work placement dealt essentially with image processing. Image processing can be defined as any form of information processing for which the input is an image. The output is not necessarily an image, as it could also be any piece of data representing some features of the image. The field of image processing is really wide, and includes a lot of different techniques. The common thing is that an image is every time seen as a two-dimensional signal (matrix).

Medical imaging is the process with the objective to evaluate an area of the subject's body that is not normally visible. It may be "clinical" : seeking to examine and diagnose disease in human patients, or "research-motivated" : attempting to describe and understand processes in humans or animal models.

In the Zebrascreen project, both concepts have been used, that is to say image processing applied to medical imaging. This means applying image processing techniques on medical images in order to analyse them, to extract features formerly impossible to find, etc.

4.6.1 Binarization

Binarization is a very important concept in image processing. It is the process that allows to have a binary image. That is to say, from a colour image, or from a grey scale image, creating an image that can only have 2 intensity values (usually 0 and 255). This is really significant, because this is usually the way used to detect something in an image. It simplifies the representation of an image into something that is more meaningful and easier to analyse.

The basic method to binarize an image consists in choosing an intensity threshold value, and consider every points that have a lower intensity as 0's, and others as 1's (255). Unfortunately, there are a lot of binarization methods, and none of these works well for every problem. Especially if the object you want to detect is composed of different intensity areas, dark and bright ones for example, it becomes really difficult to have a good result. Choosing a unique threshold for every frame is so impossible. Another method is the Otsu method, which chooses automatically the threshold, minimizing the between-class variance. But this doesn't solve every problem.

4.6.2 Segmentation

Segmentation is also an important process in image processing, whose goal is to partition a digital image into multiple regions. Binarization is so a segmentation, but in only 2 regions. Thanks to this process, the processed image is more meaningful and understandable. It is often used to separate objects in an image (as it is used in this project). A lot of different methods to perform a segmentation exist, depending mainly on the purpose and the type of input image to process. For example, some well-known methods are : clustering methods, histogram-based methods, region-growing methods, graph-partitioning methods, neural networks segmentation, etc... But, unfortunately, the choice of a method is usually hard and sometimes no "general" method can be used, and a customized one has to be developed.

Examples of segmentation results can be seen in figures 2.5, 2.6, and 4.6.

4.7 Tools used

Different tools were used during the placement, depending on what task was being done at the moment. The most important ones were :

- Microsoft Visual Studio .NET : a C++ programming environment.
- OpenCV [8] : an image processing C++ library, that was used for the whole project, for all image processing matters.
- Torch [11] : a machine-learning C++ library, used in the project for data mining and classification matters.
- Carl Zeiss Axiovision : a microscope images acquisition and image processing program, used first by the company the project is aimed to serve to, and used in the project to test methods, and to do VBA applications for the company.
- Doxygen [3] : a documentation system for C++ (and other languages).

- LaTeX : a typesetting program, used for this report.
- VirtualDub [12] : a free video capture and AVI/MPEG-1 processing utility distributed under GNU General Public License.

4.8 The unfolding of the work's events

4.8.1 Calculate the heart rate

The first task was to compute the heart rate of the fish, from videos of fluorescent heart images. Before my arrival 10 methods had already been implemented. Other ones have been created, to be able to choose the best one, making statistics on a lot of videos.

Global schema of the methods

Most of the methods follow the same simple basis schema to analyse one video :

- Capture the video and its features.
- Compute the maximum number of frames we can analyse.
- For each frame we want to analyse :
  - Launch the method on this frame.
  - Add the result to the vector which represents the signal.
- Calculate the heart rate, from the signal, with the Discrete Fourier Transform.
- Write the results in a text file, formatted like a spreadsheet.

The 10 already made methods

Here is a brief description of these methods, which makes explicit the criteria used to obtain one point of the signal :

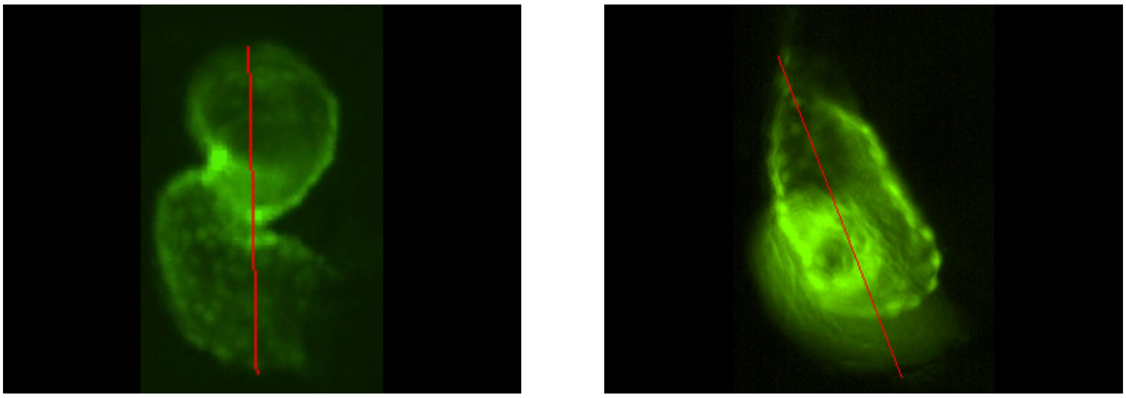
- DPI : Difference of intensity of each pixel.
- Area : Area of the heart.
- Gravity Center X : coordinate X of the gravity center of the heart.
- Gravity Center Y : coordinate Y of the gravity center of the heart.
- Central Moment : combination of both coordinates of the gravity center.
- Elliptic Fitting Height : height of the ellipse that fits the best the heart.
- Elliptic Fitting Width : width of the ellipse that fits the best the heart.
- Regression Y : coordinate Y of the line that fits the best the heart.
- Regression X : coordinate X of the line that fits the best the heart.
- Fitting Line Distance : distance from the line that fits the best the heart to the contour of it.

The method Fit Line Intensity

The initial idea of this method is based on the other methods already implemented. It was to determine the line which fits best the heart, and then to create the signal from the intensity of each points of the line, on each frame. Then we had thought to compute the mean of these intensities, for example.

After some attempts, which did not come to a conclusion, only one point on the line had to be chosen, so that the idea became this one : first determine the line which fits best the heart, using binarization, and contour detection. Then, analysing the first frame, choose one point of the line, corresponding to the one with maximum intensity. Finally, for each frame, compute the intensity of this point, which will form the signal. Algorithm of the method : (for one frame)

- Convert the image in grey-scale.
- If we are analysing the first frame :
    - Binarize the image.
    - Detect the contour.
    - Determine the equation of the best fitting line.
    - Choose "the point", as the one of the line with maximum of intensity.
- Compute and return the intensity of "the point".



(a) **(b)**

Figure 4.2. *Example of computed heart fitting lines. In red, the fitting line. In green, the heart. // Exemple de lignes représentant le coeur. En rouge, la ligne. En vert, le coeur.*

The method Fit Point Intensity

To improve the method FitLineIntensity, a new way to choose the point which "represents" the heart in the signal was searched. In this new method the point is so chosen as the gravity centre of the heart in the first frame. Algorithm of the method : (for one frame)

- Convert the image in gray-scale.
- If analysing the first frame :

- Calculate the moment vector of the frame.
- Choose "the point", as the gravity center calculated from the moments.

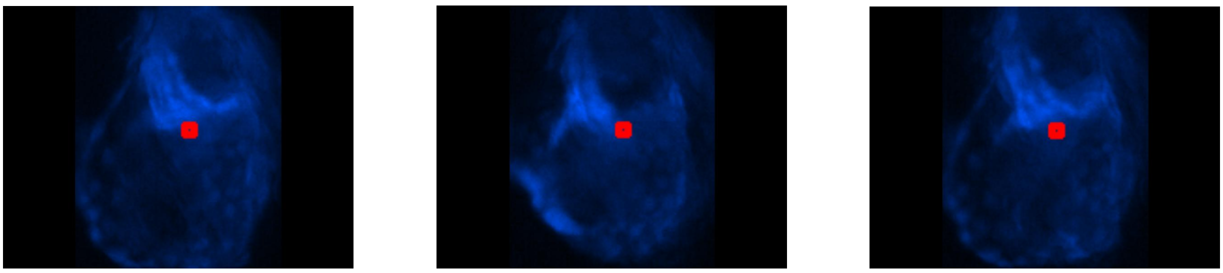
- Compute and return the intensity of "the point".

Then, a problem was the detection of the moments, which could be altered by the background of the video. Indeed, usually the heart occupies only a small part of the video, and the background, which is not totally black, can produce errors. So a cropping process on each frame has been included before processing the method. To do this, we used classes already implemented.

Here is the explanation on how works the cropping class : first it assumes that the repartition in lines and columns of the intensity of the pixels of the object follows a normal law. It computes the parameters of this normal law, and then it is able to compute the probability of each pixel to be part of the object. So, one parameter of the class is a threshold, which is this probability. All the pixels with a probability under this threshold won't be kept. Then another parameter is used, increasing the size of the region of interest of a specified percentage.

A threshold of 0.5 for the probability is used. Therefore there is not any more problems due to the background of the video. The basic schema has been changed a little. Here is the new one :

- Capture the video and its features.
- Compute the maximum number of frames we can analyse.
- If analysing first frame :
  - Calculate the rectangle to crop.
- For each frame we want to analyse :
  - Crop the frame.
  - Launch the method on this frame.
  - Add the result to the vector which represents the signal.
- Calculate the heart rate, from the signal, with the Discrete Fourier Transform.
- Write the results in a text file, formatted like a spreadsheet.



(a) **(b)** **(c)**

Figure 4.3. *Example of computed heart fitting point, seen in three different frames of one video. In red, in the center, the fitting point. In blue, the heart // Exemple de point représentant le coeur, vu dans trois images extraites de la même vidéo. En rouge, au centre, le point. En bleu, le coeur.*

The method Euclidean Difference

This method is based on the euclidean norm. The signal is created from the euclidean difference between each frame and the first one. This could have been made simply calculating this difference for each pixels. But this would have lead to a high complex algorithm.

Therefore a "zoning" process is used, already implemented in a class. Here is the principle : divide the image in rectangles larger than the size of one pixel, and compute for each rectangle the mean of the intensity of the pixels in it. Then, reconstruct the image, but formed only with the rectangles.

Then the rectangles can be used as pixels, and the complexity is really lower. Here the image is divided in 10 by 10 rectangles. The cropping process is also used, to improve at most the complexity of the method. The basis schema is so the same as for the method FitPointIntensity. Algorithm of the method : (for one frame)

- Convert the image in gray-scale.
- Apply the zoning process.
- If we are analysing the first frame :
    - Remember the image formed with the zoning process.
- Else :
    - Compute the euclidean difference between the current image and the first one and return it.

The method Unique Change Mask

This method is completely different from the others. The basis schema is not the same. The objective is also to analyse the intensity of pixels, but only for "interesting" pixels. For this we have to create a binary mask of the video, which shows us which pixels are to take in count, and those that we don't care about. So, here is the basis schema of the method :

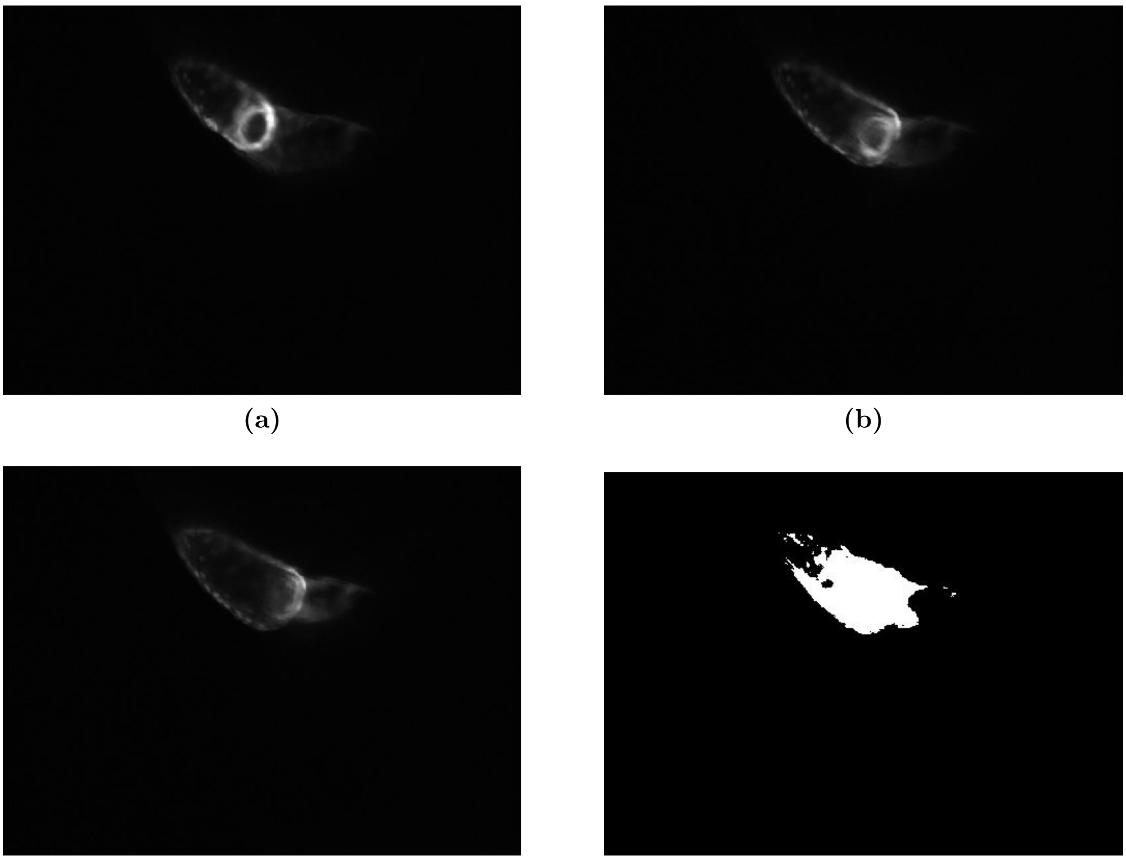
- Capture the video and its features.
- Compute the maximum number of frames we can analyse.
- Compute the mask.
- Compute the matrix with all the pieces of data of the "interesting" pixels.
- Compute the signal from this matrix.
- Calculate the heart rate, from the signal, with the Discrete Fourier Transform.
- Write the results in a text file, formatted like a spreadsheet.

The point in this method is to compute a good binary mask. To be able to have one unique mask for the whole video, we first compute one mask between each 2 consecutive frames, simply looking at the difference of intensity of each pixel, and keeping only the ones that have a difference superior than a threshold. Then, for each pixel, we look how many times it appears in all the masks. So, with a second threshold stated in percentage of the frames analysed, we keep only the "interesting" points, that is to say the ones that have an intensity which varies significantly (w.r.t. the thresholds) along the video. Algorithm to compute the unique mask :

- For each frame of the video :

    - Compute and save the temporary mask between the current frame and the previous one.
- For all masks calculated :
    - For each pixel of the mask :
        - If the pixel is part of the mask :
            - Increment the number of times this pixel appears in the masks.
- For each pixel of the unique mask :
    - If the pixel appears more than a threshold in the other masks :
        - Put this pixel in the unique mask.
    - Else :
        - Throw this pixel back from the unique mask.

Finally, we get the matrix of all the pieces of data. We form the signal as the mean of all these pieces of data.



(a) (b) (c) (d)

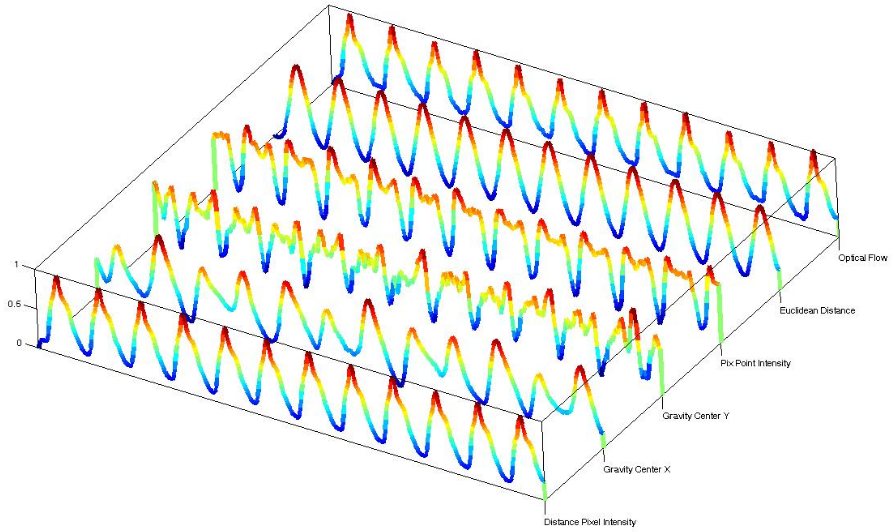
Figure 4.4. *Unique mask method example video frame at t0 (a), t1 (b), t2 (c), and binary unique mask created by the method from this video (d) // Image à t0 (a), t1 (b), t2 (c) de la vidéo exemple de la méthode Unique mask, et masque unique binaire créé par la méthode à partir de la vidéo exemple.*

Choosing the best method

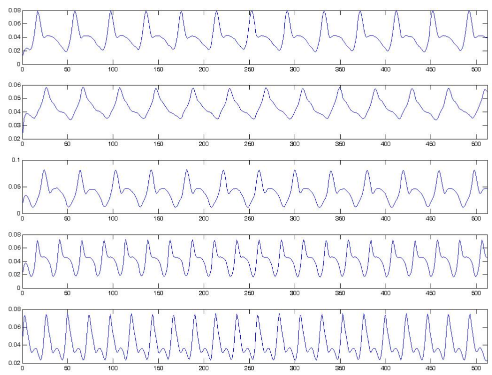
As soon as all these methods were implemented, the best one had to be chosen. In order to do it, statistics had to be made on the most videos as possible. It was clear that it was impossible to do it manually, it would have taken too much time. An automatic statistic module has so been developed.

Indeed, the videos given by the biologists are each time accompanied by an excel file, describing each feature of the video. So this new module extracts the theoretical heart rate from this excel file, and then it can compare it to the calculated value by the method. To have senseful statistics, each comparison has to be analysed as "hit or miss", so that percentages can be done for each method and the analysis becomes easy.

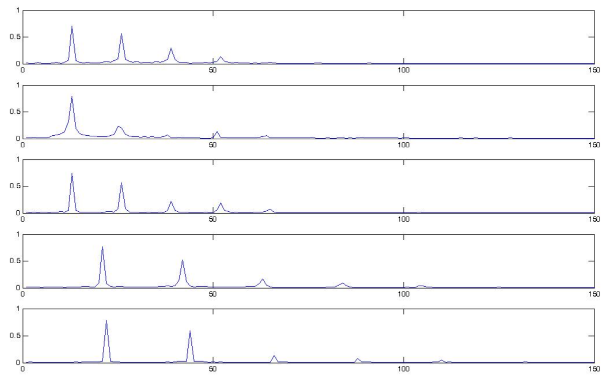
Finally the euclidean difference method has been chosen. It has been improved, regarding the calculation of the heart rate using the Fast Fourier Transform.



(a)



(b)



(c)

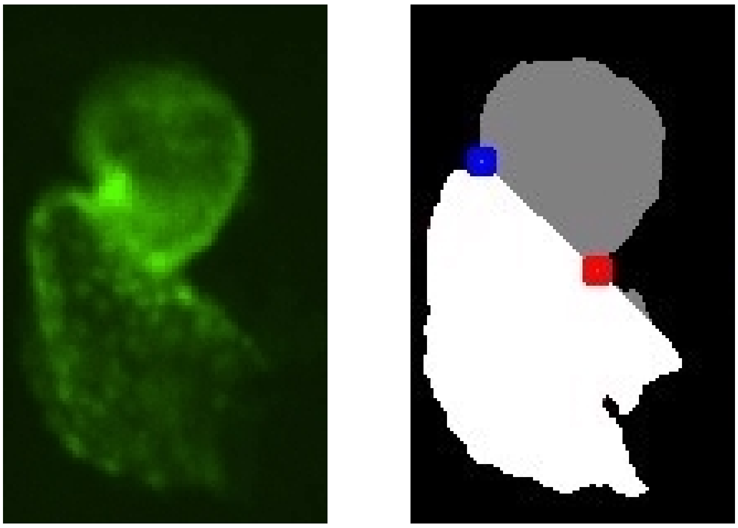
Figure 4.5. *(a) : Comparison of the signals obtained on the same video for 5 different methods ; (b) : examples of signals obtained with the chosen method ; (c) : the FFT corresponding signals // (a) : Comparaison des signaux obtenus sur une même vidéo pour 5 méthodes différentes ; (b) : exemples de signaux obtenus avec la méthode choisie ; (c) : les signaux FFT correspondants.*

4.8.2 Segment the heart chambers

The second task of my placement was to segment the heart chambers (ventricle and auricle). One method was already implemented, but other ones had to be implemented to see if better results could be obtained.

The method already implemented, based on convexity defaults

The method which was already implemented, is based on convexity defaults of the heart. That is to say, first a binarization is applied, and the convex hull of the binary image is calculated. Then, according to this convex hull, the convexity defaults are found and are supposed to define 2 points that form a line which separates the 2 cavities. Actually, this is true and works for a lot of videos, depending on the fish position, and therefore on the way you look through the heart (front, rear, transversal left, transversal right). According to this position, it can be ”easy” or extremely difficult to detect the 2 different chambers of the heart.



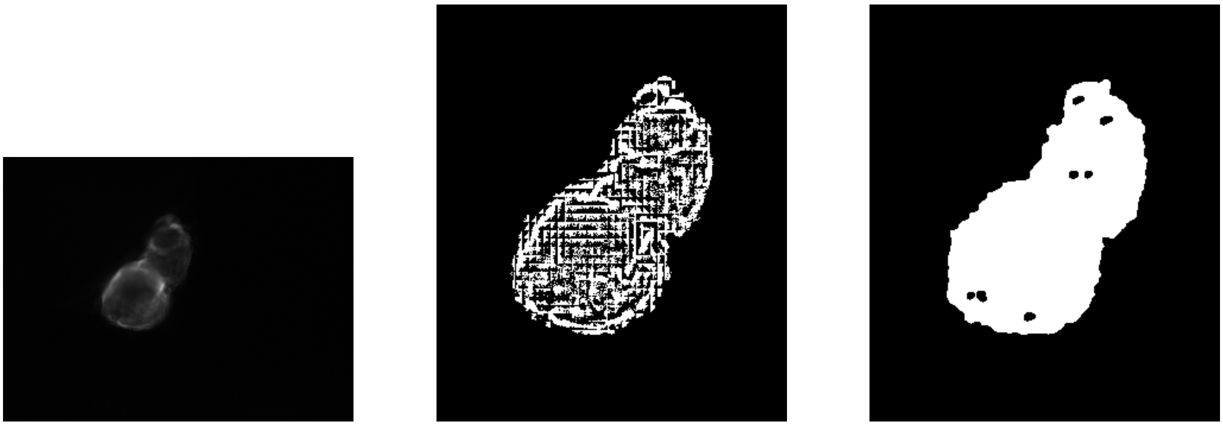
(a) **(b)**

Figure 4.6. *Example of the convexity defaults method for heart segmentation. (a) : the original frame ; (b) : the result with each chamber in a different colour, and the 2 points localizing the convexity defaults // Exemple de la méthode de segmentation du coeur basée sur les défaults de convexité. (a) : l'image originale, (b) : le résultat avec chaque cavité dans une couleur différente, ainsi que les 2 points localisant les défaults de convexité.*

Attempt for a new method of binarization

One of the problem of the segmentation method detailed in section 4.8.2 is that, sometimes, the binarization process of the heart fails, because some videos are really different from other ones and the method used is not well fitted to these changes. That's why a new method, that could work in every cases, has been searched. The idea found is based on the Otsu binarization method (cf section 4.6.1). Indeed, this method works good, but because of the high differences of intensities in the frames, the normal method does not solve the problem.

The new method developed used the Otsu normal method, but locally. That is to say, the image was divided in small rectangles, and the otsu method was applied in each one of them. Before a low threshold was applied to uniform the background to zero, and avoid to have a binarization result in all rectangles. The result at this stage was really promising, but lacked some post processes, to obtain an object without holes in it, and with continuous contours. Such processes were so implemented, using mostly morphological mathematical transformations ; the results were really close to a good binarization, but never something evidential was reached.



(a) **(b)** **(c)**

Figure 4.7. *Examples of the new binarization method, seen on one frame of a video. (a) : the video frame ; (b) : the result before the post processes ; (c) : the best result obtained // Exemples de la nouvelle méthode de binarisation, vus sur une image d'une vidéo. (a) : l'image de la vidéo ; (b) : le résultat avant les post traitements ; (c) : le meilleur résultat obtenu.*

The method based on the unique change mask and the phase of each pixel

The idea depends mostly on the phase of each pixel. This method is based on the fact that the 2 different parts of the heart have the same frequency, but are off-phase.

Beginning from the "unique change mask", then it is possible to take all the pixels from it, compute their phase, and classify them in 2 classes with these datas. First, to compute their phase, the method considers that the abscissa of the maximum of the intensity of a pixel on a heart rate period could be considered as his phase, because the most important is the comparison of the phase, not the phase alone. Then different methods to separate the formed set have been tried. The raw datas are, for each pixel, the phases for a given number of periods. 7 methods have been tried :

- For each pixel, calculate the mean of the phases. Separate the pixels with the mean of the mean of their phases.
- For each pixel, calculate the variance of the phases. Separate the pixels with the mean of the variance of their phases.
- For each pixel, calculate the standard deviation of the phases. Separate the pixels with the mean of the standard deviation of their phases.
- For each pixel, calculate the mean of the phases. Separate the pixels using the K-mean clustering method in 1 dimension with these datas.

- For each pixel, calculate the variance of the phases. Separate the pixels using the K-mean clustering method in 1 dimension with these datas.
- For each pixel, calculate the standard deviation of the phases. Separate the pixels using the K-mean clustering method in 1 dimension with these datas.
- For each pixel, calculate the mean, the variance, and the standard deviation of the phases. Separate the pixels using the K-mean clustering method in 3 dimensions with these datas.

But finally, no method gives a proper result. This idea was so abandoned to avoid losing time. As it is obvious on figure 4.8, the 2 segmented parts do not correspond to the 2 different chambers of the heart. With many tests on many videos, we got more or less the same results.

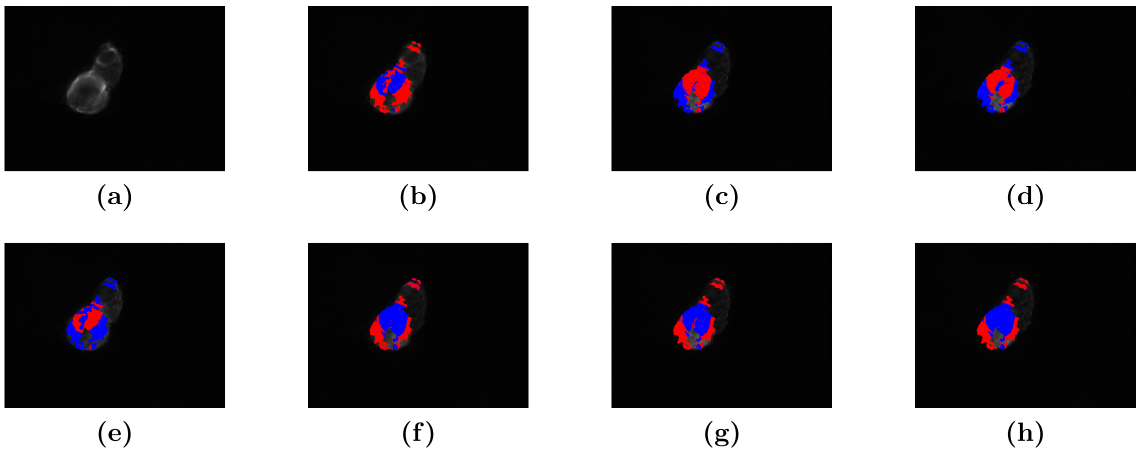


Figure 4.8. *The Unique Change Mask method results. (a) : the analysed frame ; from (b) to (h) : Segmentation of this frame with each method (from the first to the seventh respectively). // Les résultats de la méthode Unique Change Mask. (a) : l'image analysée ; de (b) à (h) : segmentation de cette image avec chaque méthode (de la première à la septième respectivement).*

Multiple Otsu threshold

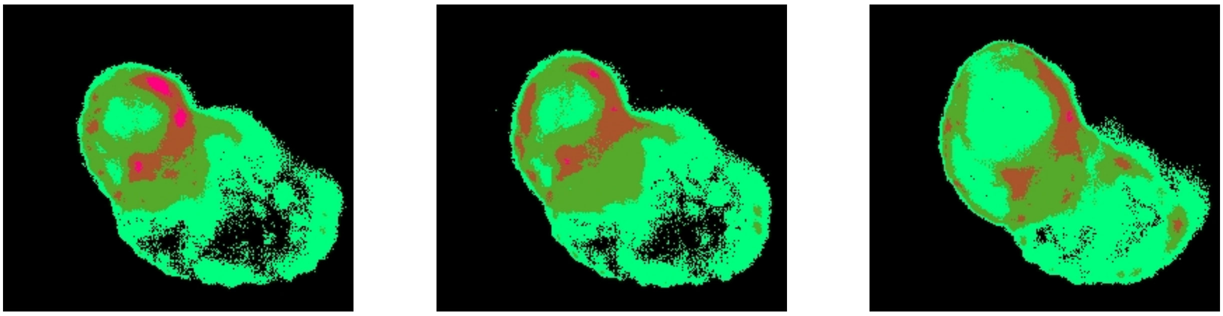
The second idea to segment the heart is the multiple Otsu thresholds method. This method proposes a criterion for maximizing the between-class variance of pixel intensity, in order to define thresholds. The Otsu binarization method is already implemented in OpenCV library, but not the multiple thresholds method. So it had to be implemented.

The problem is that the formulation of the between-class variance is really inefficient and leads to a very time-consuming algorithm. Therefore a faster algorithm, proposed by Ping-Sung Liao, Tse-Sheng Chen and Pau-Choo Chung [6], has been implemented.

First, in this algorithm, a new and equivalent criterion for the between-class variance is used. Then, in accordance with this new criterion, a recursive algorithm is defined to compute faster the thresholds. Finally, look-up tables are used to speed up the process as much as possible.

The implementation of this algorithm has taken quite a lot of time, but it did not matter so much, because even if it was not sure that it would give good results on the heart frames, it could be reused in other projects in the future. A documentation has been made with doxygen [3], so that now everyone can use the algorithm easily (cf Appendix B).

Finally the results on the heart frames were not good, and this method was not chosen.



(a) **(b)** **(c)**

Figure 4.9. *Segmentation of the heart with the multiple Otsu thresholds method, using 4 thresholds. Each colour represents one set of pixels. Each frame is extracted from the same video at different times of the heart rate period. (a) : frame 1 of the video ; (b) : frame 11 of the video ; (c) : frame 19 of the video. // Segmentation du coeur à l'aide de la méthode multiple Otsu thresholds, en utilisant 4 seuils. Chaque couleur représente un ensemble de pixels. Chaque image est tirée de la même vidéo à différents instants de la période de battement du coeur. (a) : image 1 de la vidéo ; (b) : image 11 de la vidéo ; (c) : image 19 de la vidéo.*

Other attempts

At the day this report is being written, no method that works for every case has been found to segment the heart. Other researchers and interns are currently working on it, trying new filters, creating new methods. For the moment, the convexity defaults method is being used, waiting for a better method, because it works on most cases.

4.8.3 Heart rate for each chamber

With a heart rate and a segmentation method, what was now possible and asked was to determine the heart rate of each chamber, which can be a good parameter to detect some drug effects. For instance, a chamber can die whereas the other can still be working, or sometimes a big rate difference can be observed. For this, a combination of the two processes was made : first the segmentation was applied, creating 2 binary masks representing each chamber for each frame, and then the heart rate method was applied twice, according to each mask. The result was so 2 rates, corresponding to each heart chamber.

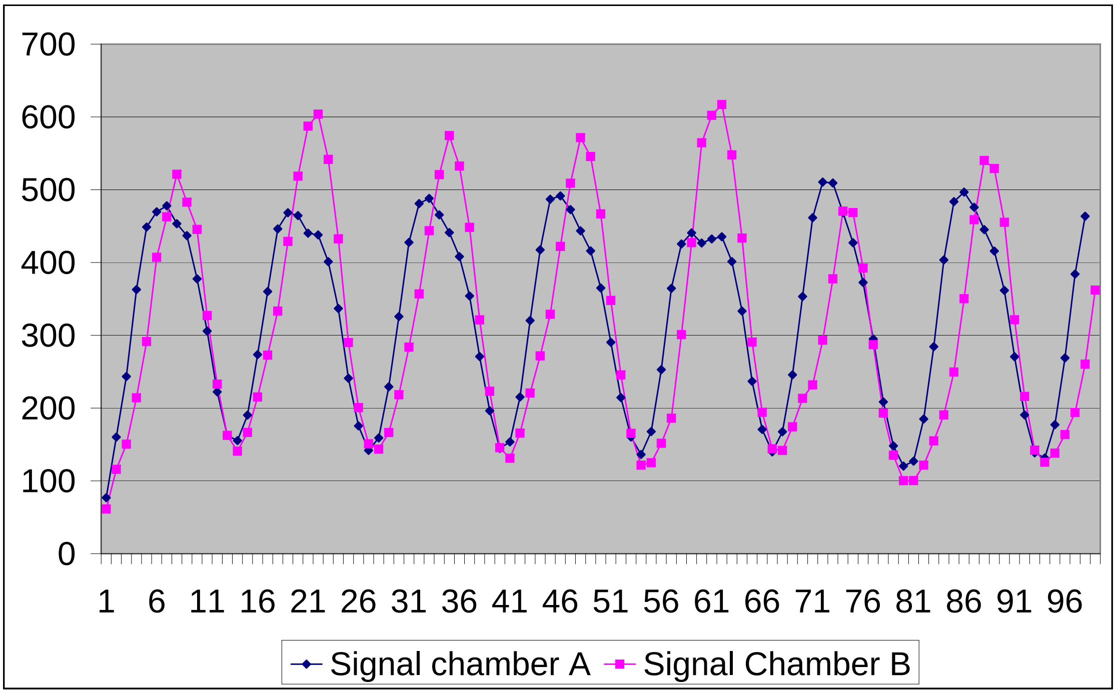
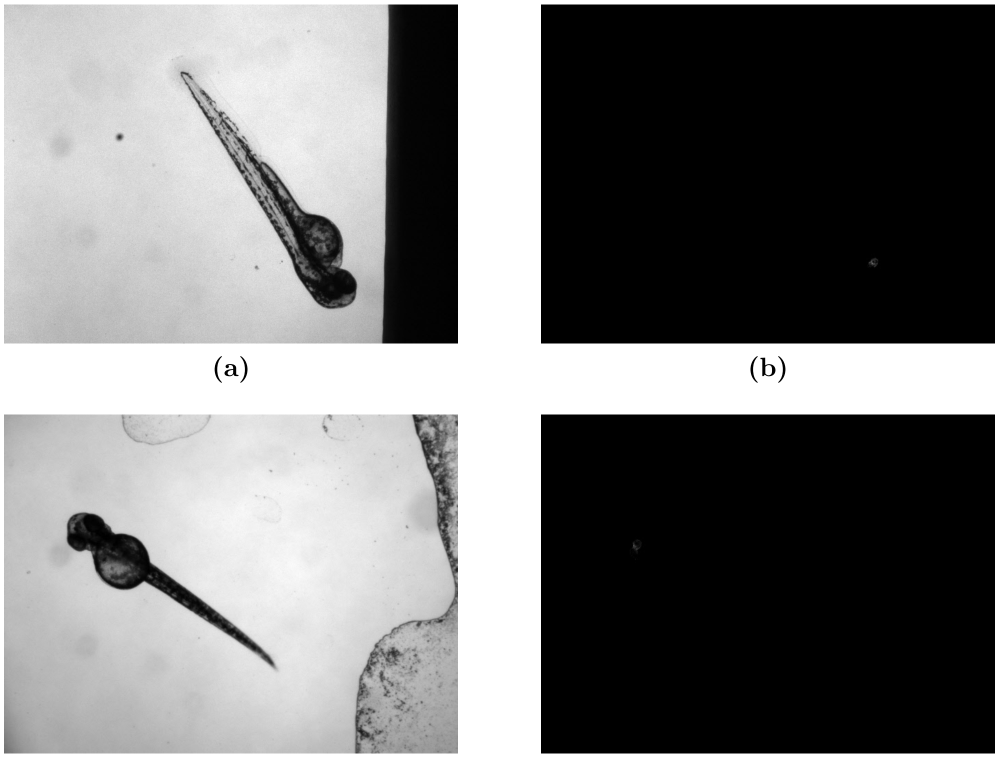


Figure 4.10. *Example of heart rate signals obtained for each chamber of the heart on one video // Exemple de signaux de rythme cardiaque obtenus pour chaque cavité du coeur*

4.8.4 Determine the position of the fish

Another task was to determine the position of the fish. That is to say : "Ventral", "Dorsal", "Lateral left", or "Lateral right", and combination of these values. To do this, we dispose of 2 images, which are the photos from the fish in the plate. One is the entire photo, and the other one represents only the heart.



(a) (b) (c) (d)

Figure 4.11. *Examples from base images for the determination of the position of the fish. (a) : image from a fish in lateral right position ; (b) : the heart image from this fish ; (c) : image from a fish in ventral position ; (d) : the heart image from this fish // Exemples d'images formant la base de la détermination de la position du poisson. (a) : image d'un poisson en position latérale droite ; (b) : l'image du coeur correspondant ; (c) : image d'un poisson en position ventrale ; (d) : l'image du coeur correspondant.*

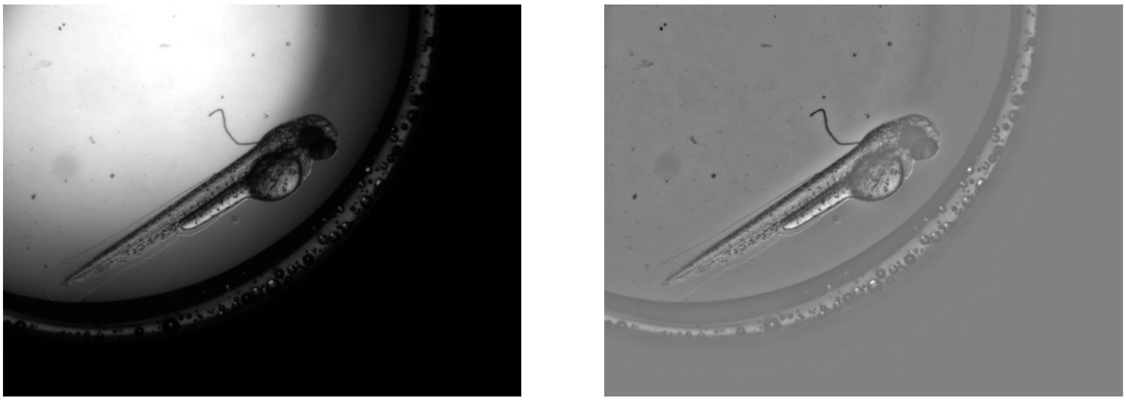
Overview of the method

The method is based on 2 filters, and a post-analysis of the results. The principle is quite easy to understand : first, detect the fish on the image, and then analyze its shape to conclude on its position. To detect the fish, 2 filters are used : one which has been made to improve the quality of the image, the Shading Correction Filter, and one which creates a binary image and a convex hull of the fish, the Incremental Components Filter. A documentation has been made with doxygen [3], so that everyone who wants to use or improve this method can do it easily.

The Shading Correction Filter

This filter has been made to improve the results of the Incremental Components Filter. Indeed, the images have strong brightness differences, which makes much more difficult to get a good output from any filter processed on them.

First a strong gaussian blur is applied on the original image. Then the obtained image is inverted, and an addition from it to the initial image is made. Finally a normalization is applied on the last image to obtain the processed image of the filter.



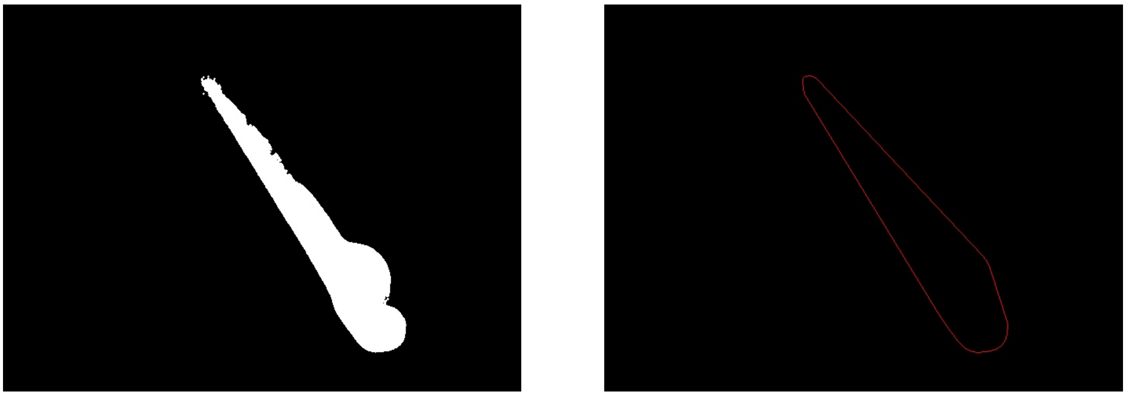
(a) (b)

Figure 4.12. *Example of the Shading Correction Filter. (a) : the original image ; (b) : the processed image. // Exemple du filtre Shading Correction. (a) : l'image originale ; (b) : l'image traitèe.*

The Incremental Components Filter

This filter uses the 2 input images (the heart one and the fish one), and detects the exact position of the fish, creating a mask image and calculating the convex hull of it. The process begins by detecting the location of the heart, which is quite easy (thanks to the heart image). Then the mask of the fish is created step by step, adding one component at each step, beginning from the heart components, as they are obviously part of the fish. This could be so compared as a customized "region growing algorithm", for instance.

In order to carry it out, first the detection of the heart is made with a simple otsu threshold binarization method ; then connected components of the heart and the ones from the fish image are determined. From them, an iterative algorithm adds to the heart components, at each iteration, the component of the fish image that is the nearest to the border of the heart. For this, the convex hull of the heart is determined at each iteration and considered as the contour of it. When it ends, thanks to end conditions, the heart components represent the fish, and the convex hull as well. The result is also being validated by some conditions analyzing the size of the components, as sometimes it can fail. Thanks to this validation, image classification is sometimes stopped, but it prevents the process from giving an absurd result.



(a) **(b)**

Figure 4.13. *Result of the Incremental Components Filter on the image 4.11 (a). (a) : the binary mask of the fish ; (b) : the convex hull of the fish // Résultat du filtre Incremental Components sur l'image 4.11 (a). (a) : le masque binaire du poisson ; (b) : la convex hull du poisson.*

The Fish Image Classifier

As already said before, the classifier is an analysis of the shape of the fish. It can differentiate 3 positions : "Only ventral", "Lateral left", and "Lateral right". The input of the classifier is the convex hull created by the Incremental Components Filter. The first step is to determine the fish direction, represented by a line going through the heart. Then 2 distances are determined, according to the position of the fish, the contour of the fish, and its direction. As it can be seen on the schemas of figure 4.14, the 2 distances allow to decide the position : if they are equal,the fish is in the "only ventral" position ; otherwise knowing which one is the longest decides on which lateral side the fish is lying.

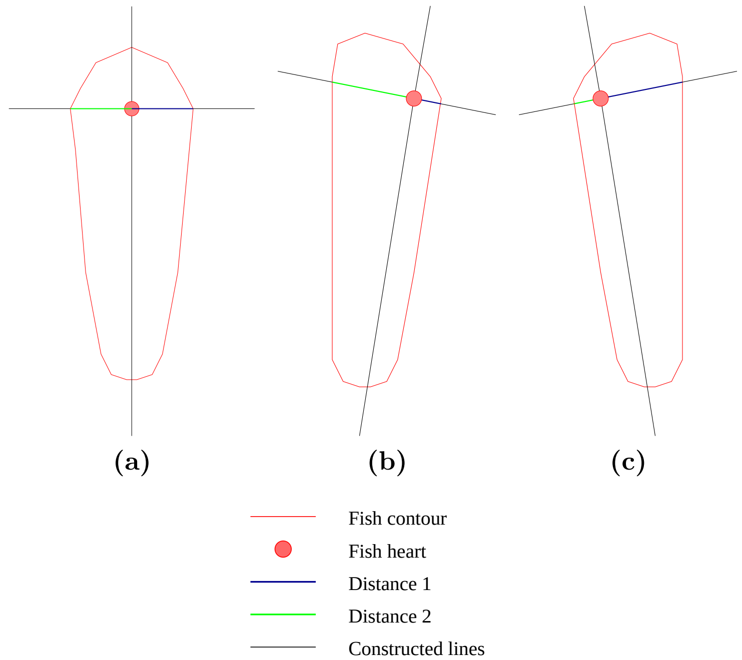


Figure 4.14. *Schemas of the analysis of the shape of the fish. (a) : "Ventral only" position ; (b) : "Lateral left" position ; (c) : "Lateral right" position. // Schémas de l'analyse de la forme du poisson. (a) : position "Ventrale seulement" ; (b) : position "Latérale gauche" ; (c) : position "Latérale droite".*

4.8.5 Create graphic user interfaces

In order to test our methods in a comfortable manner, graphic user interfaces have been made, using the MFC library. They usually had the same goals : browse the computer to select one or several videos or frames, select or enter the several features, if any, of the method to test, launch the method and visualize in a proper manner the results. With such graphic user interfaces, no time was lost to make tests on new method attempts : a graphic user interface was made at the beginning and was used till the solution to the problem was found.

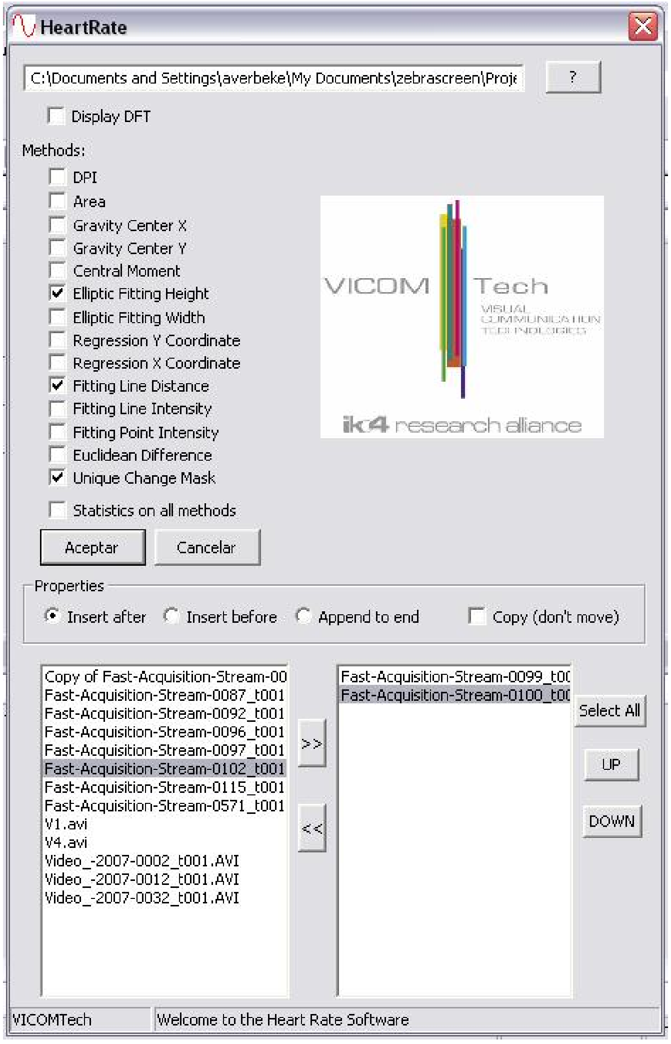


Figure 4.15. *The graphic user interface to test the methods to calculate the heart rate // L'interface graphique qui permet de tester les différentes méthodes de calcul du rythme cardiaque.*

4.8.6 Relations with BIOBIDE

The relations with BIOBIDE, the company for which the project is intended, were held by the researcher in charge of the project. He was going there regularly in order to check the advancement of the project, to discuss about problems encountered, to get new videos, etc...

After the choice of a method, it had to be sent to BIOBIDE, in order to be tested in their laboratory. The people in BIOBIDE are working with Visual Basic for Applications, integrated in the software Carl Zeiss Axiovision, so the methods had to be given in a simple VBA function, so that they could use it as easily as possible. For each method, a simple C++ function with input and output parameters was created, and exported in a DLL. Then it was possible to import it from the DLL easily in a VBA program.

4.8.7 Difficulties and problems encountered

Learning image processing was the first encountered difficulty. Indeed, I had no knowledge in this field and it was necessary to work properly in the project. Then, programming difficulties were encountered : because of the lack of experience, a lot of programming habits had to be learned in order to avoid common mistakes. For example a lot of memory deallocation problems were made, and they are really hard to find, if the memory is not managed properly. Finally, the last big difficulty was the understanding of the algorithms read in paper that had to be implemented. Actually, without being used to reading papers, it takes a long time to understand well a described algorithm.

The first problem was the use of the MFC library. Indeed, thanks to this library it is really easy and fast to create a graphic user interface, so it is well adapted for testing new methods that can eventually fail. But at the same time there are a lot of functionalities that are bad managed by the "Class Wizard" and makes the user lose a lot of time and eventually become crazy, when he wants to create a complete advanced graphic interface. Then another problem was the creation of DLL that could be used in Visual Basic for Applications (cf section 4.8.6), because no one knew how to do it and how to use VBA, it took a lot of time to create them for something simple at the end. But the main problem encountered and that was not solved completely at the end of this internship, was the quality of the videos and the images. Because of some parameters, the videos and images could be really different from one to another, sometimes bright, sometimes dark, etc... Therefore it was really difficult to create methods that can adapt themselves to every situations, and give proper results each time. This had to and still has to be improved in collaboration with BIOBIDE.

Chapter 5

Contributions of the work placement

5.1 Team work and managing skills

During the whole work placement, I observed my superiors, managing their team or directly ruling the company, and I was able to understand a lot on what they were working on, even though I did not practice this type of work. Indeed, for example, during the meetings, whether it be a department one or a general one, it is easy to listen to what is said and draw conclusions on what it is said for. Learning is possible just from listening to the subjects, seeing which ones are got onto and so seen as important, how the speaker introduces critical subjects, how he explains the critical decisions he had to take, etc. It is obvious that these meetings are really important for a team, because it introduces a cohesion between the members of it. Moreover the problems can be discussed all together, which is the best way to find optimal solutions. Everyone is aware of the orientations of his department and of his company, knows what his colleagues are working on, shares his important results, etc.

Besides, being integrated in a team inside a project made me learn a lot on how works a team. For example the repartition of the tasks : who works on what and for how long? The answer is not so easy, and it depends on the skills of everyone, their interests, the percentage of their time they can use on this particular project, the deadlines of the latter, and other specific parameters. Everyone in the team has access to the resources of the project like Gant diagrams, the budget, the table of the different phases, and a lot of other pieces of information, so that everyone knows well the environment of the project he is working on.

5.2 Technical point of view

From a technical point of view, I learned a lot on image processing, machine learning, and programing in general. I saw my progression during the 6 months, because at the end I was able to program faster and better. I had to document myself a lot on image processing algorithms, and machine learning methods, so that now I can have a global vision on these fields related to informatics. I also learned how to use several useful softwares (cf section 4.7), that I can now use almost perfectly.

5.3 Cultural point of view

From a cultural point of view, I learned the Spanish language, of which I only knew the bases before coming to Spain. Then I also discovered a new country, Spain, and a new region, the basque country, which are really different from France : the landscapes, the weather, people's habits in everyday life as well as in professional life, the hobbies, the parties, everything is made in a different way. I was also glad to be able to practice a lot my English, because we had chosen to use this language with my supervisor to work together. As VICOMTech is a company relatively international-oriented, they offer language lessons facilities, and I was also able to have German lessons and keep practicing this language.

Chapter 6

Conclusion

These 24 weeks within the company VICOMTech were really rewarding, from every point of view (cf chapter 5) according to myself. I learned a lot and I enjoyed myself working in this research center.

One weak point could be the lack of time at the end of the 6 months. Indeed, I documented myself on machine learning methods, a branch I really like, but I could not put it in practice, because of this lack of time.

The project is at this time still in progress, in its last months. People are still working to improve the methods and to find new ones for the tasks not yet solved.

But above all, I discovered what exactly means ”working in a research center”, which is something really important for my future orientation choices of my career.

Bibliography

[1] **CAPO IRIZARRY G. (Rosa I.) et al. :** *Fundamentals & Applications of Image Change Detection*, Research and Industrial collaboration conference center for subsurface sensing and imaging systems, 2006, http://www.censsis.neu.edu/RICC/2006/students/posters/pdf/Capo_POSTER%20BOSTON%20OCTUBRE%202006-RJR.pdf.

[2] **DEVELOPPEZ.COM :** *Club des développeurs*, http://www.developpez.com/.

[3] **DOXYGEN :** *Source code documentation generator tool*, http://www.stack.nl/~dimitri/doxygen/.

[4] **KIMMEL (Charles B.) et al. :** *Stages of Embryonic Development of the Zebrafish*, Developmental Dynamics, 203:253-310, 1995.

[5] **KOPKA (Helmut) :** *LaTeX Band 1 : Einführung*, Pearson Studium, München, 2002, ISBN 3-8273-7038-8.

[6] **LIAO (Ping-sung), CHEN (Tse-cheng) and CHUNG (Pau-Choo) :** *A Fast Algorithm for Multilevel Thresholding*, Journal of information science and engineering 17, 713-727, 2001.

[7] **MITTELBACH (Frank) and GOOSENS (Michel) :** *The LaTeX Companion Second Edition*, Addison-Wesley, Boston, January 2005, ISBN 0-201-36299-6.

[8] **OPENCV :** *Intel Open Source Computer Vision Library, Image Processing and Computer Vision algorithms*, http://opencvlibrary.sourceforge.net/.

[9] **PAJARES (Gonzalo) and M. DE LA CRUZ (Jesús) :** *Visión por Computador : Imágenes digitales y aplicaciones*, RA-MA, Madrid, June 2001, ISBN 84-7897-472-5.

[10] **SCHWERTE (Thorsten) et al. :** *Development of the sympatho-vagal balance in the cardiovascular system in zebrafish (Danio rerio) characterized by power spectrum and classical signal analysis*, The Journal of Experimental Biology 209, 1093-1100, January 2006, doi:10.1242/jeb.02117.

[11] **TORCH :** *The Torch Library, a machine-learning library*, http://www.torch.ch/.

[12] **VIRTUALDUB :** *a free video capture and AVI/MPEG-1 processing utility*, http://www.virtualdub.org/.

[13] **WIKIPEDIA :** *The free encyclopedia that anyone can edit*, http://en.wikipedia.org/wiki/Main_Page.

List of Figures

Appendix A

Workplan

Computer analysis of Fluorescent Microscopic Images

Alexis Verbèke (UTC) 4th-year Project

February 2007 – July 2007

Workplan

Donostia-San Sebastián, April 2007

Table of Content

1. OBJECTIVES

The project carried out by Alexis focus on research developments in the field of image segmentation and feature extraction of fluorescent microscopic images. Different image processing techniques will be studied during the internship period (02/07 – 07/07), mainly those related to image segmentation. The segmentation process allows the classification of some parts of interest in the image, in order to distinguish some pixels from the others. Feature extraction techniques are useful for preparing a strong database for the classification process, especially making a reduction of the classification space taking the best features representing the image. These techniques can be applied in many different areas, such as Medicine, or Engineering. The main objective of this project is to integrate these two concepts in a software platform to give them the ability of extraction of relevant information contained in image data.

The internship will be developed within the ZEBRASCREEN project, an industrial project in the biotechnological sector and more concretely related to microscopic imaging. The project is going to provide to the student a framework to acquire an important knowledge in image processing.

2. ZEBRAFISH IMAGE CLASSIFICATION

While tremendous advances in imaging hardware make now possible the rapid acquisition of thousands of high-resolution images of the organisms, the current method of analysing these images is still mainly the manual and visual inspection to infer potential genetic or environmental interactions by identifying functional or morphological similar characteristics (phenotypes). The lack of a computational approach to identify characteristics similarities causes a bottleneck for large-scale experiments and is therefore an impediment to advancement in biology research, including genomic and drug discovery.

Figure 1 Zebrafish Embryo

In this project, we propose to explore the use of advanced image analysis methods for automated identification and classification of phenotypes. Such methods should allow to quantitatively measure complex phenotypes from thousands of images in a high throughput manner.

The automatic classification of phenotypes is based on the extraction of quantitative parameters (features) from the digital image. Such features are based on morphology (e.g., number of objects), texture (e.g. granularity), or other gray-level-based measures. Selecting subsets of the many possible features are then necessary to reduce the complexity of the classifier. After feature extraction, a mathematical model needs to be learned from data that accurately associates image features with predefined phenotype classes.

3. WORK PLAN

This internship will be developed in 6 months, as follows.

Task 0: Project management

Duration: Project length

Responsible: Dr. Fernando Boto (technical supervisor in VICOMTech)

During this task, all the project management tasks, such as meetings or work plan validation, will be carried out.

Task 1: Study of different Computer Vision methods: image segmentation and feature extraction

Duration: 4 Weeks

Responsible: Student

In this task, basic bibliography on Computer Vision will be reviewed; the tutor will guide the student to focus on the image processing methods required by the project.

Task 2: Implementation of image segmentation methods

Duration: 8 Weeks

Responsible: Student

In this task, the student will work on the implementation of image segmentation methods allowing to extract automatically the chambers of the zebrafish embryo heart.

Task 3: Implementation of feature extraction methods

Duration: 8 Weeks

Responsible: Student

During this task, the student will work on the implementation of feature extraction methods allowing to extract automatically pre-defined information from the microscopic images.

Task 4: Integration of all the methods

Duration: 4 weeks

Responsible: Student

The algorithms and methods implemented in the previous tasks will be integrated in an unique platform.

Task 5: Validation and Test

Duration: 4 Weeks

Responsible: Student

In this period, some *Ground-Truth Data set,* obtained with other software, will be accessible. The results obtained with the software implemented in Task 4 will be evaluated by comparison with the *Ground-Truth Data.* During the validation process, different assessment parameters will be computed, such as: missclassification rate, false positive rate, or execution time, among others. Moreover, different software modifications or updates will be carried out in this period if needed.

Task 6: Documentation

Duración: 4 weeks

Responsible: Student

In this period all the developments carried out during the project will be documented in different technical reports.

3.1.1 Gantt Diagram

The following table shows the development time distribution of the project by months.

	1	2	3	4	5	6
T0: Project Management						
T1: State of the Art of Computer Vision						
T2: Implementation of image segmentation methods						
T3: Implementation of some feature extraction methods						
T4: Integration						
T5: Validation and Test						
T6: Documentation						

4. VICOMTECH DESCRIPTION

VICOMTech (Visual Communication and Interaction Technologies Centre) is an applied research centre for Interactive Computer Graphics and Multimedia located in the Technology Park of San Sebastian (Spain).

VICOMTech is a non-profit association, founded by the INI-GraphicsNet Foundation, with the Fraunhofer-IGD as founder member, and the Basque Television, Radio and Broadcasting group EiTB, It is currently formed by 16 members.

Since the beginning, VICOMTech belongs to Saretek, the local science, technology and innovation network. Furthermore, the acknowledgement VICOMTech has obtained in these years has paved the way to incorporate into the IK4 alliance and also to be classified as an Innovation and Technology Centre (ITC) by the Spanish Ministry of Education and Science.

VICOMTech's aim is to fulfil the innovation needs of the companies and institutions. For this, the Centre

- works in applied research and development of multimedia technologies for visual interaction and communication;
- fosters the mobility and formation of researchers;
- collaborates tightly with the industry, universities and institutions, and complements other technology centres.

Thus, the knowledge and technologies that VICOMTech masters, directly or indirectly via the network, give added value to its clients, because VICOMTech

- can provide them the best answer to their specific needs;
- facilitates them to take advantages from the technological opportunities that may appear;
- proposes improvements or developments for their products, based on the latest advances of scientific and technological knowledge.

Appendix B

Doxygen documentation for use of multiple Otsu thresholding algorithm

Main Page Classes

Class List Class Members

MultipleOtsuThresholding Class Reference

Multiple otsu thresholding class. More...

`#include <MultipleOtsuThresholding.h>`

List of all members.

Public Member Functions

	MultipleOtsuThresholding (void) *default constructor.*
	MultipleOtsuThresholding (**MultipleOtsuThresholding** *MOT) *constructor by copy.*
	MultipleOtsuThresholding (int NbOfThresholds, IplImage *InitialImage) *constructor with parameters.*
	~MultipleOtsuThresholding (void) *default destructor.*
void	**SetInitialImage** (const IplImage *InitialImage) *Sets the Initial Image.*
const IplImage *	**GetInitialImage** (void) const *Gets the Initial Image.*
void	**SetNbOfThresholds** (const int &NbOfThresholds) *Sets the number of thresholds to compute.*
const int &	**GetNbOfThresholds** (void) const *Returns the number of thresholds to compute.*
int &	**GetNbOfThresholds** (void) *Returns the number of thresholds to compute.*
const int *	**GetThresholds** (void) const *Returns the table which contains the thresholds computed.*
const int &	**GetNbGrayscaleValues** (void) const *Returns the number of grayscale values (usually set to 256 in constructor).*
int &	**GetNbGrayscaleValues** (void) *Returns the number of grayscale values (usually set to 256 in constructor).*
void	**SetNbGrayscaleValues** (const int &NbGrayscaleValues) *Sets the number of grayscale values.*
int	**execute** (void) *Method which executes the multilevel thresholding.*

Detailed Description

Multiple otsu thresholding class.

This class executes the otsu multilevel thresholding on one frame. Give the number of thresholds you want. Execute using the method **execute()**. The result is the values of the thresholds. Get it with the function **GetThresholds()**.

This only works for grayscale images (intensity from 0 to 255) see documentation for more information about the algorithm

Constructor & Destructor Documentation

MultipleOtsuThresholding::MultipleOtsuThresholding (void)

default constructor.

The number of Thresholds to compute is set to 1. The number of grayscale values is set to 256.

MultipleOtsuThresholding::MultipleOtsuThresholding (int *NbOfThresholds*, IplImage * *InitialImage*)

constructor with parameters.

The Highest grayscale value is set to 256.

Parameters:

NbOfThresholds The number of Thresholds to compute.
InitialImage The initial image on which you want to compute the thresholds.

Member Function Documentation

void MultipleOtsuThresholding::SetInitialImage (const IplImage * *InitialImage*) `[inline]`

Sets the Initial Image.

Parameters:

InitialImage The initial image on which you want to launch the method. !!Has to be in Grayscale!!

const IplImage* MultipleOtsuThresholding::GetInitialImage (void) const `[inline]`

Gets the Initial Image.

Returns:

The initial image.

void MultipleOtsuThresholding::SetNbOfThresholds (const int & *NbOfThresholds*) `[inline]`

Sets the number of thresholds to compute.

Parameters:

NbOfThresholds The number of Thresholds to compute.

const int& MultipleOtsuThresholding::GetNbOfThresholds (void) const `[inline]`

Returns the number of thresholds to compute.

Returns:

The number of thresholds.

int& MultipleOtsuThresholding::GetNbOfThresholds (void) `[inline]`

Returns the number of thresholds to compute.

Returns:
The number of thresholds.

const int* MultipleOtsuThresholding::GetThresholds (void) const [inline]

Returns the table which contains the thresholds computed.

Returns:
The table which contains the thresholds computed.

const int& MultipleOtsuThresholding::GetNbGrayscaleValues (void) const [inline]

Returns the number of grayscale values (usually set to 256 in constructor).

Returns:
The number of grayscale values.

int& MultipleOtsuThresholding::GetNbGrayscaleValues (void) [inline]

Returns the number of grayscale values (usually set to 256 in constructor).

Returns:
The number of grayscale values.

void MultipleOtsuThresholding::SetNbGrayscaleValues (const int & *NbGrayscaleValues*) [inline]

Sets the number of grayscale values.

Use this function to set the number of grayscale value : For example, if your image has no grayscale value superior to a threshold it will accelerate the process.

int MultipleOtsuThresholding::execute (void)

Method which executes the multilevel thresholding.

Returns:
0 if no problem has happened.

-1 if no initial image is set.

-2 if the number of thresholds is < 1.

The documentation for this class was generated from the following files:

- My Documents/zebrascreen/multiple otsu thresholding/MultipleOtsuThresholding.h
- My Documents/zebrascreen/multiple otsu thresholding/MultipleOtsuThresholding.cpp

---

Generated on Fri May 25 10:57:05 2007 for Multiple Otsu Thresholding by doxygen *1.5.2*