Measurement of Ejection Fraction & Wall Thickening in myocardial perfusion Positron Emission Tomography

By Charles Malo, University of Ottawa Heart Institute, Ottawa, Ontario, Canada

With supervision by Robert A. deKemp and Ran Klein

May-August 2009

Keywords: FlowQuant, UNIX, Left Ventricle, Ejection Fraction, Volume, Wall Thickening, Myocardium, PET

# Abstract

For my first work term (SEG 2901), completed from May to August 2009, I was employed as a software developer working on adding new methods and visualization tools to a program called *FlowQuant* which studies the heart’s *Left Ventricle* functionality. My duties were to create, assisted by both supervisors, various algorithms which can facilitate the diagnosis of the heart, thus determining proper treatments. Among those algorithms was one that determines the *Ejection Fraction,* an estimation of the heart efficiency. The low precision of this algorithm was considered a problem at first. In order to improve that accuracy, another algorithm was conceptualized to calculate the myocardium *wall thickening*. This factor taken in consideration, an important improvement of the *volume calculations’* accuracy is noticeable as well as more realistic results promoting the program for *clinical use*. The program is written to be as *universal*, *automated,* as possible, with the *least assumptions* taken. Also the software is designed to *support UNIX platforms* as well. The *validity* is measured up to the acknowledged *4DM* software considered to have so far the most acceptable estimations*.*

# Left Ventricle Ejection Fraction & Wall Thickening Calculation

# Introduction

Positron Emission Tomography (PET) has become an “important non-invasive technique in cardiovascular research, offering unique insight into biochemical changes on a molecular level with excellent sensitivity” (1). The Ejection Fraction (EF), an estimation of the heart’s efficiency, is an important factor in diagnosis and determining proper treatment. Relating, it has been shown that PET perfusion images can be used to estimate the Wall Thickening (WT) throughout the Left Ventricle (LV) myocardium beating cycle, thus adding more accuracy to the Ejection Fraction calculations. In order to study the heart’s functionality, FlowQuant (FQ), which is a local under development software program at the University Of Ottawa Heart Institute (UOHI) has the significant functionality to accept Digital Imaging and Communications in Medicine (DICOM) images and various other types of scans as an input, process them in different ways, and show graphical outputs, depending on what the user wants to examine. During the summer of 2009, this program has been updated with a few methods that can, as from now, add to the list of analysis options available to FlowQuant both Ejection Fraction and Wall Thickening computing options, thus giving a better insight on the patient’s cardiac health.

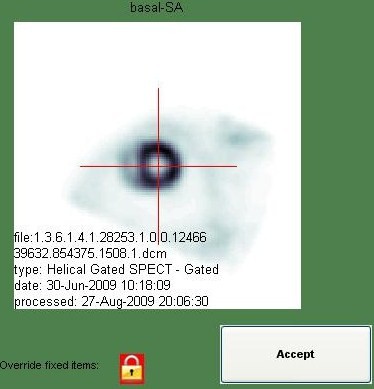
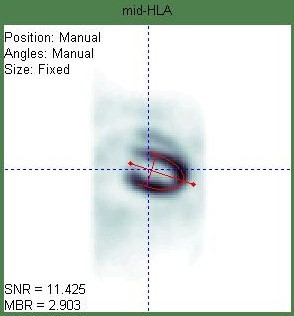
# Methodology

*Acquiring the data:*

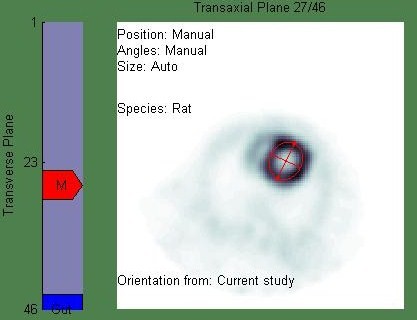
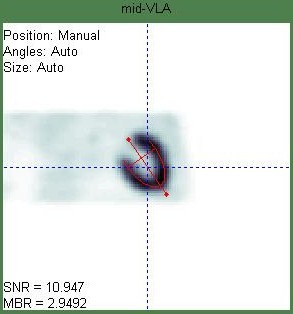
The clinical aspect of the patient examination is done prior to the study of the data by FlowQuant. The patient is injected with a radioactive substance selected depending on the study type and other factors, but usually the substance used in most studies dealt with in this report is one of the radioactive 18FDG glucose or Rubidium (RB). That substance will be carried by the blood flow into the patient’s heart, where it is absorbed by the myocardium wall (the heart muscle).

Then the patient is left to lie in a scanning machine with both built-in CT and PET cameras that can move in all planes, all around the patient, taking “snapshots” of the part of the body we wish to study, in this case the torso, for the study of the heart. The images taken of the radioactive uptake using the PET camera are then stored into DICOM files, where all values and entries of the patient’s information are saved as well, for later use by a proper software program such as FlowQuant or 4DM. Once all the clinical data as well as the uptake images in the heart muscle are stored, the scans are loaded into FlowQuant, which analyses them based on what the user wishes to study.

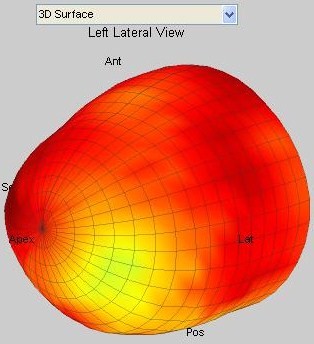
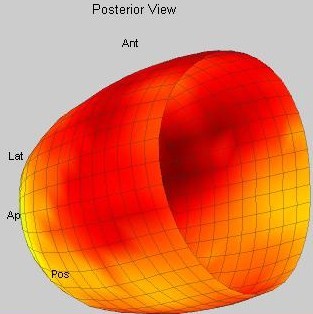
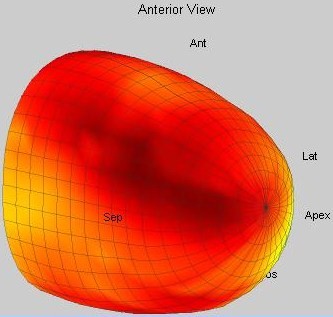
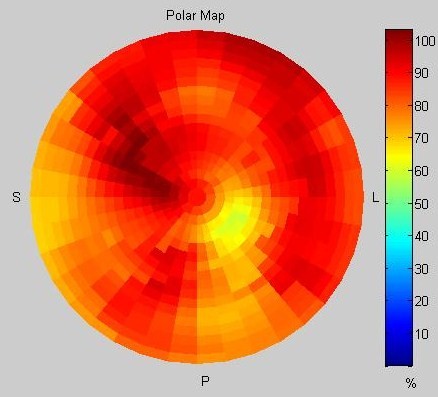
*Processing the data:*



One of the important features available in FlowQuant is a mid-myocardium contour detection based on the highest uptake points in the muscle, allowing a rough estimation of the mid-myocardium coordinates in x, y, and z directions. Based on the coordinates of the points delimiting the mid-myocardium, radii values are estimated, and written in a structure variable, containing all properties of the LV, for later use. Since the heart changes shape during the cardiac cycle, for each of the 8 frames (representing 8 different cardiac interval snapshots), different radii values and phase-dependant variables are detected for each one of these phases.



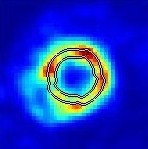
FlowQuant used to only cope with static or dynamic images of the LV, and it needed to be promoted to obtain the ability to handle *gated* images, which are animated images throughout the entire cardiac cycle. A pre-existing and rather basic function called “Gated Report” was temporary implemented into FlowQuant as a starting point, by my supervisor Mr. Ran Klein, which had the functionality of acquiring variables created or stored during the execution of the analysis. From those variables, including the radii stored, that function could create a very rough and rigid animated surface plot, as well as display slices of the LV at different view angles along the Short Axis (SA), Horizontal Long Axis (HLA) and Vertical Long Axis (VLA).



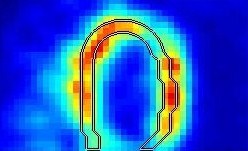
*Improving the report page*

The current Gated Report was a rather basic implementation, a “draft” of what is yet to become the new and improved Gated Report once the summer work term was done. Before getting into the Ejection Fraction or Wall Thickening calculations, the report page needed to be updated, improved to become more user-friendly and attractive for the eye. In other words, it needed to look “prettier” in order to give more intuitive understanding of the heart functionality to the clinical eye based on the “beating” of the LV model.

The Gated Report used to be a figure window showing rows containing the three slices and the primitive gated model for every scan processed simultaneously. It was recently updated to create a figure window for each scan separately in order to view more information regarding each scan. The new figure for each study is now divided into somewhat four quadrants distinguished by the type of their contents: LV Slices, LV Gated Model, LV Polar Maps (PM), and Graphic Curves.



* LV Slices:

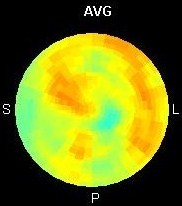
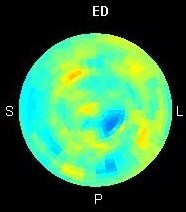
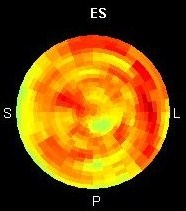
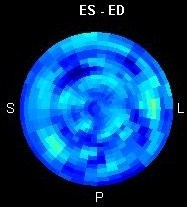
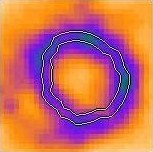
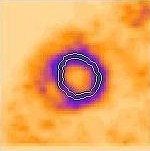
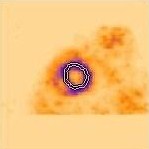


The LV slices are animated views of the LV wall at three different angles, showing the wall movements throughout the cardiac cycle, based on the radioactive uptake in the muscle itself.

All of the SA, HLA, and VLA slice images are stretched to fit their respective axes keeping aspect ratio. Inner and outer myocardium wall contours are added to the slices, with an option to turn their display off. These contours are implemented to “follow” the wall motion throughout the cardiac phases, and to accommodate the use of the zoom slider option by updating the contours to “stick” to wall borders.

Zoom

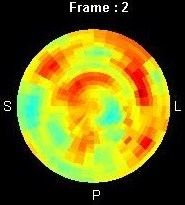
Zoom



* LV Polar Maps:



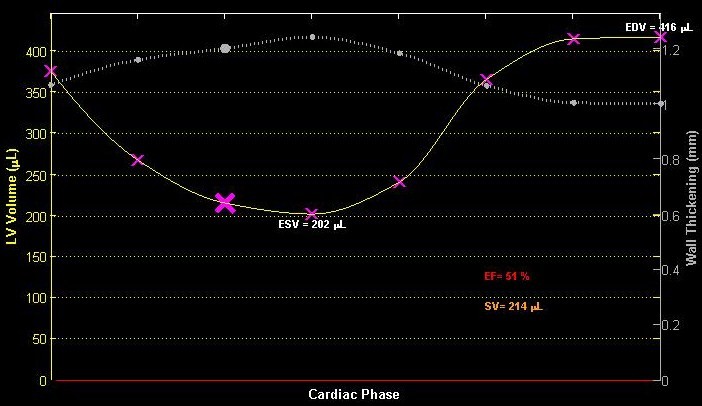
An LV polar map is considered a “flattened” version of the model, a disk view of the LV along the SA, centered at the Apex.



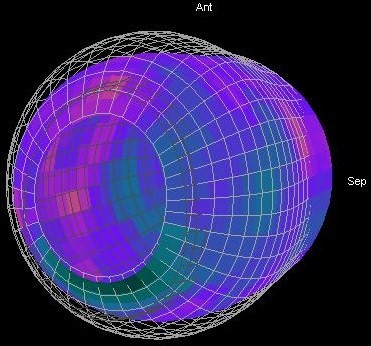
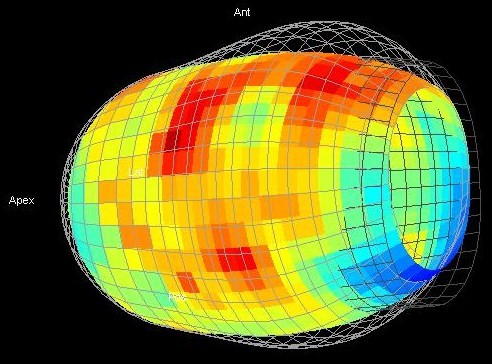
Four static polar maps are created using the uptake values stored: the ED Polar Map, ES Polar Map, Average Polar Map and the Wall Thickening (WT) Polar Map. The WT PM is represented by the subtraction of the ED PM from the ES PM, the reason is that the muscle shows more intensity in ES when it is thick and contracted, rather in ED where the muscle is thinner and dilated. Thus the difference in intensity between both polar maps is related to, not measured by, the change in wall thickness. A fifth PM is also added to the figure which is animated phase by phase.

* Graphic Curves:

After the volume of the LV at each frame is calculated using the algorithm described later, these values are plotted in the graph and an estimated curve is drawn linking these points by interpolation. Another curve is estimated similarly for the estimated Wall Thickening calculated.



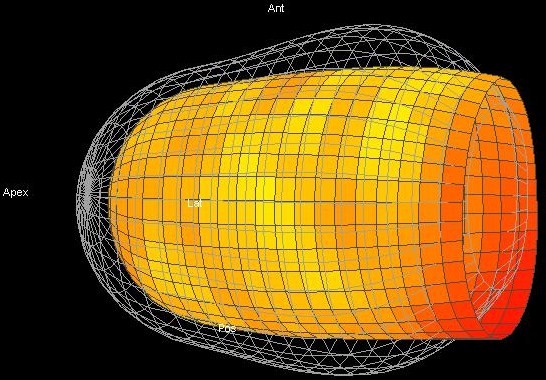
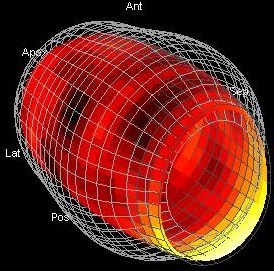
A “highlighted” mark on each curve is animated to follow the curve’s “trajectory” based on the current frame. The graph also views the calculated End Diastole Volume (EDV), End Systole Volume (ESV), Ejection Fraction (EF) percentage and Stroke Volume (SV).



* LV Gated Model:

The LV model is updated with new features among like the 3D rotation ability.

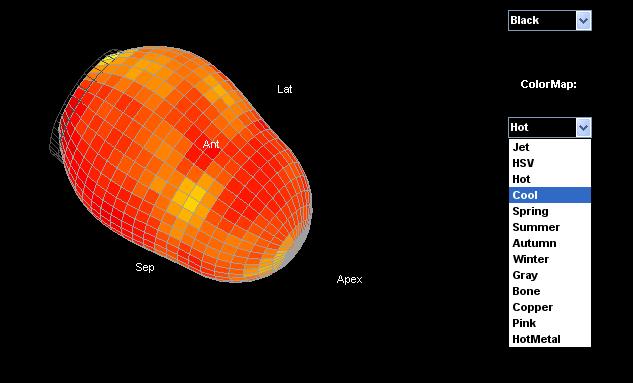
Smoothing of the surface plot was associated with the plotting function, interpolating the radii values along smaller steps to draw a smoother and more even model of the LV, giving it a nice and soft vase shape.



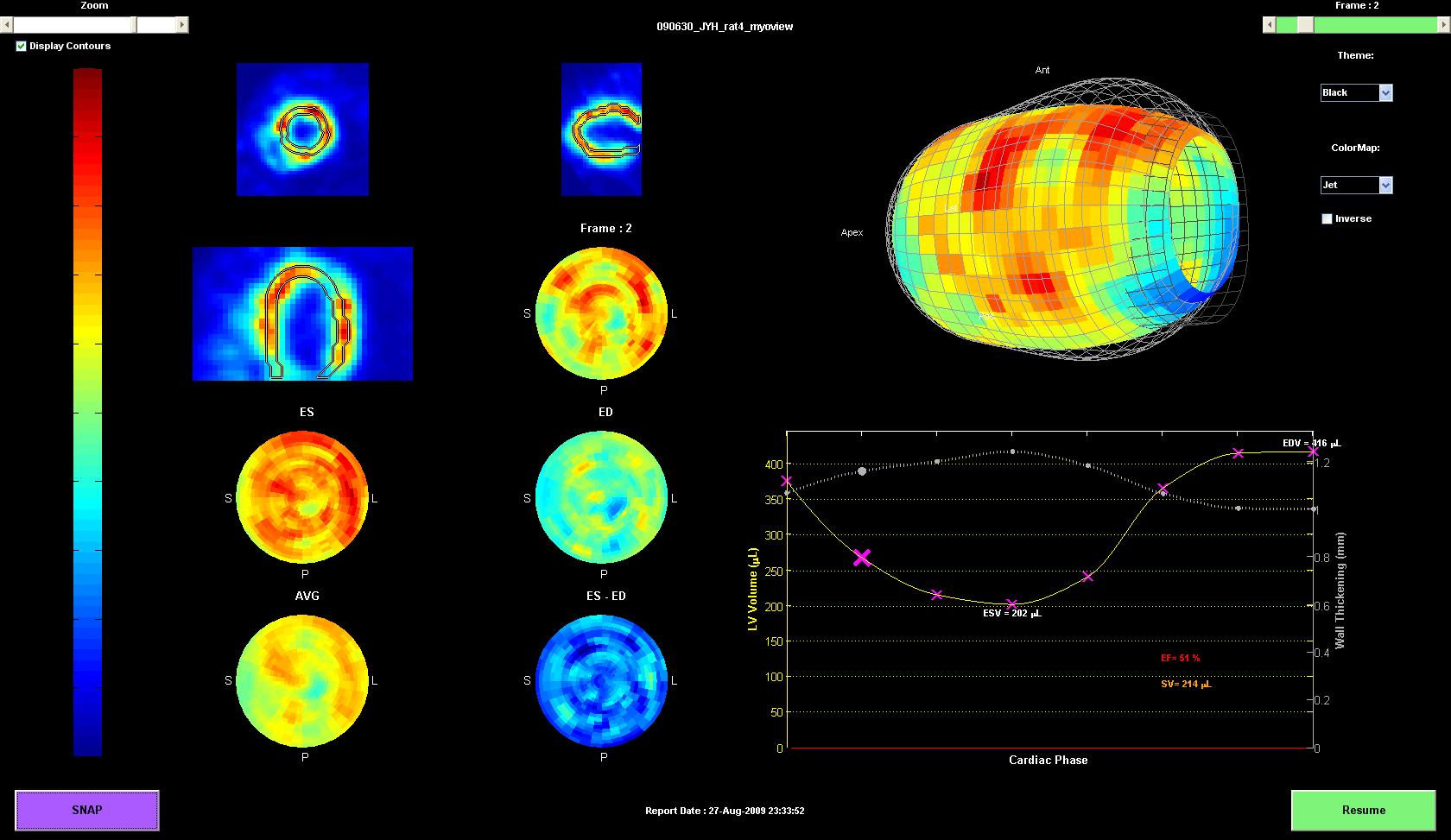
After determining the End Diastole (ED) and End Systole (ES) frames and obtaining the specific radii at those frames by a process explained later on in the report, their respective meshes were added to the surface plot, on the same axis. These meshes represent respectively the maximum and the minimum size and shape the left ventricle can take during the cardiac cycle. Having those fixed delimiting plots, the user can now have more insight on the LV wall motion and shape in respect to those grids.

Various GUI control were added to the figure to manage the output on the figure, among which are:

* A slider to control the zoom of the slice images, and a display toggle button to show the myocardium wall contours or not.
* A Frame Rate slider that control the animation speed of the entire figure, to suit the user’s needs.
* A theme List to alter the style of the figure between, so far, the default grey theme and the more appealing black theme.
* A Color Bar is associated with the currently selected Color Map to view the range of the colors used.
* A capture button was added in order to save a video capture of the entire animated figure throughout a complete cardiac cycle, offering different formats for the video output (‘avi’, ’mpeg’, or ‘gif’). When the animation is paused, the video capture button works as a snapshot button that can save an image file (in ‘jpg’ format) of the figure at the current phase.



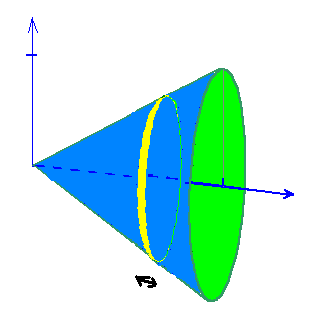
* A Pause button can temporary stop the animations in the figure, also it transforms the Frame Rate slider to a frame selector, which enables the user to chose which phase of the cycle he wishes to view.
* A Color Map List to modify the colors and their ranges used to display the graphical outputs and the ability to inverse the colors.



*Note: The previous capture of the Gated Report is of a study of a RAT Left Ventricle, not human.*

*Facts:*

* Fact 1: The volume of the LV is the biggest at end diastole, and the smallest at end systole.
* Fact 2: The myocardium wall is the thinnest at end diastole where it is “stretched” to its maximum, and the thickest at end systole, where the muscle is contracted.
* Fact 3: As consequence to fact 2, the uptake intensity in the myocardium is the highest at end systole and lowest at end diastole.
* Fact 4: The highest uptake in the muscle is recorded along the mid-myocardium wall, i.e. equidistant to both the inner and outer cardio wall.
* Fact 5: If the mass of an object and its density do not change, no matter how the shape changes, its volume is constant. This is applied for the wall muscle in my research.
* Fact 6: The myocardium wall thickness is not uniform at all points in the wall, and the thickness changes throughout the phases depending on the wall’s shape.
* Fact 7: A volume of an object is the areas of the slices of that object integrated along the axis from border to border.



*The Calculations:*

* **Basic calculation of LV volumes ignoring Wall Thickening:**

In each phase of the cardiac cycle, the heart, in particularly the LV, changes form and size when contracting and expanding, thus the coordinates of the points defining the mid myocardium vary at each frame. Therefore, radii values stored for each frame aren’t constant throughout the cycle, they grow until ED, and shrink till ES.

FlowQuant analyses the LV, and treats it as a 3D model that can be “divided” along the short axis into slices, called “Rings” in the Gated Report, even if they are not symmetric or geometric.

To calculate the LV blood volumes, an idea was originally made of getting a “best fit” function of a 3D geometric shape resembling the model, but the idea was eliminated because of the huge loss of accuracy, as well as the difficulty behind the math needed to create that function. Therefore, we thought of “slicing” the model at 1 pixel intervals along the short axis, and by that, the slices will be treated as planar polygons. The aim of that is to find a way to calculate the area inside that polygon, i.e. the LV cavity area, for each of the slices, and integrating along the SA to calculate a volume of the cavity, thus the blood capacity.

Fortunately, a pre-defined Matlab function (“poly2mask”) can convert a region-of-interest polygon into a binary region of interest (ROI) mask. A mask is similar to a binary matrix of m x n dimensions that represents the ROI polygon. “The function sets pixels in the mask that are inside the polygon to 1 and sets pixels outside the polygon to 0.

When creating a region of interest (ROI) mask, poly2mask must determine which pixels are included in the region. This determination can be difficult when pixels on the edge of a region are only partially covered by the border line” (2), yet a “self-correcting” internal algorithm verifies whether the partial pixel is within the ROI or not, thus it is relatively safe to assume that it neither overestimate or underestimate the ROI computed.

Once the binary mask is created, entries containing ones represent the area of the cavity (and a part of the Endo Myocardium Wall since the radii used delimit the mid myocardium, not the inner wall border) at each slice. Then a sum of all the entries containing Ones of all slices (integrating along SA slice by slice, using fact 7) will represent approximately the number of pixels that “fill” the shape. Multiplying the number of pixels calculated with the volume of one unique pixel will result in finding the volume of the cavity, in units of pixels, which is then converted into milliliters or micro liters depending on the subject’s species type. This is how the prototype calculations of the LV volumes are done, ignoring the effect of Wall Thickening presence and change for each phase. The stroke volume can be calculated at this point; it is defined as the positive difference between the ED and ES volumes:

SV= EDV-ESV.

Also the Ejection Fraction is defined by the ratio of the Stroke Volume to the End Diastole Volume:

EF (%) = (SV/EDV)\*100 or EF (%) = (EDV-ESV/EDV)\*100

However, the values computed show an under estimation of a large degree in comparison with the results from 4DM.

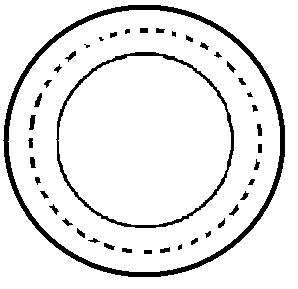
Essentially, the study of the rough volume calculations is to guess what frames represent the ED and ES frames to carry on with the next step.

Epimyocardium

* **Calculating Wall Thickening by Iteration:**

Midmyocardium

Finding the ED frame is an easy procedure, it is determined by the frame that represents the highest estimated volume, from Fact 1.



Endomyocardium

Knowing that the radii values are measured up to the mid myocardium, i.e. the middle of the heart wall, a small assumption is made and it is that the inner (Endo) wall and the outer (Epi) wall border are approximately equidistant to the mid myocardium line with a distance of half of the total wall thickening at that frame.

Cavity radii

Mid-myo radii



Epi -WT

Subtracting the Endo WT from the mid myocardium radii at each frame results in finding the radii defining the inner wall contour, thus the borders of the LV cavity. Whereas adding the Epi WT to the mid myocardium radii, at each frame, results in getting the radii defining the outer wall contour, thus the circumference of the LV.

WT

Note: Figure on the right is a representative drawing of a slice not a real image.

Circumference radii

Since no pre-existing Wall Thickening measurement algorithm is available on hand, we had to try an indirect way of computing the volumes considering Wall Thickening. An iterative approach is sought. Using all data stored in the ED frame, we aimed to calculate the volume of the myocardium wall itself using the same volume calculating algorithm with both of these sets of radii: the inner wall radii and outer wall radii. This step leads in finding the cavity volume as well as a total volume of the LV at the ED frame. The myocardium wall volume is then computed by subtracting the first smaller volume from the second larger volume.

The wall volume calculated is assumed to be the real and constant volume of the myocardium wall throughout all the phases (Fact 5) even though another small assumption is made regarding the Wall Thickness at that level.

It is assumed, for now, that the Wall Thickness is uniform all over the myocardium wall, and its value at end diastole is a specific clinically averaged value depending on the species type of the subject studied. For example, at End Diastole, a human LV wall is assumed to be 10mm thick, a rat’s WT of 1mm, a mouse’s WT of 0.5mm and so on.

Understanding Fact 2, this WT value is considered as the minimum value for the wall thickening, and thus a starting point to calculating the other WT for the other phases of the cycle.

Having at this point the assumed “real”, “precise” and constant volume of the myocardium wall, the mid myocardium radii for all phases and the minimum starting point for the wall thickening, an algorithm is created that does the following process for each frame:

For each set of radii at each frame, starting with the minimum value of the WT (i.e. the ED WT using fact 1) and iteratively increasing it with a growing step size on each iterative call, the function calculates a “trial” wall volume by means of the process described above, using the updated radii at that frame. The calculated wall volume will grow in a loop until it becomes equal to the “real” constant wall volume pre-calculated at the ED. At this point, the function will return the current wall thickening value used with that set of radii that gave the best match, and stores it in a variable for later use. This is the iterative approach of calculating the wall thickening. Another theory sought is to divide the constant wall volume by the surface area of the model at each frame, and acquiring the wall thickness, yet the surface area calculation is harder due to the limitations of the resolution of the PET perfusion scans.

Either way, the wall thickening is still considered uniform at all points of the myocardium wall.

* **Recalculation of volumes and Ejection Fraction considering Wall Thickening effect:**

Subtracting the recently calculated Wall Thickening values at each frame from its respective radii set, the volume calculation algorithm is invoked once more, and the new volume results are assumed to be the true volumes of the LV cavity, therefore the amount of blood contained in that cavity at each cardiac phase. A more accurate Ejection Fraction can now be calculated, and the results computed can now be compared with the currently acknowledged program 4DM.

*Additional detail on Calculation Methods:*

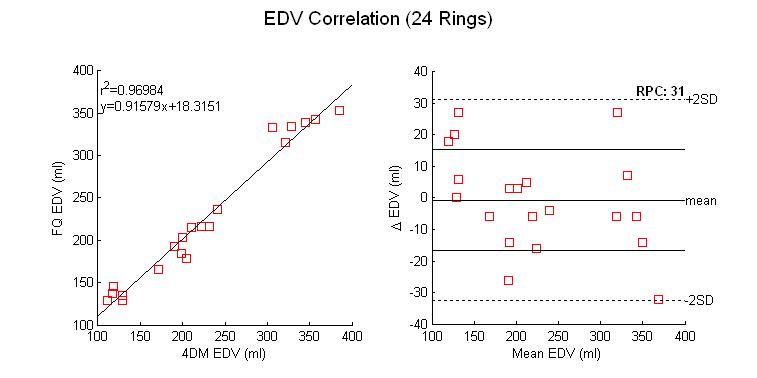
FlowQuant assumes a fixed valve plane when computing the volumes, yet it is intended later on to acquire the feature to detect the valve plane location and incline for each frame. FlowQuant used to “crop” the edge of the LV at the 18th Ring from the apex to accommodate the actuality of a moving plane. Yet to compare with 4DM, being mostly manual and taking into account the moving plane, FlowQuant was *temporarily* manipulated to run a trial computation of all results for each number of rings from 18 to 24, in order to get more insight on the volume calculation process in 4DM. The number of rings considered slightly affects the comparison results as shown later.

# *FlowQuant to 4DM Comparison and Discussion:*

Due to time shortage, only twenty human scans where randomly selected from the Hospital’s scan archives to run the comparison test between the FlowQuant and 4DM LV analysis programs. Each scan was processed in 4DM as well as in FlowQuant by me, and the results computed were stored in a comparison table and studied for their correlation using the Bland-Altman analysis method. Note that the age of most patients was, by pure hazard, high: the age range goes from 46 to 87 years old, with a mean value of about 66 years. Also, most patients had heart deficiency, therefore EF shown are relatively lower than average. By definition, the closer the r2 correlation value of the compared data and the equation slope is to 1, and the Y Intercept is to zero, the better match the results are, thus the more efficient the FlowQuant calculation algorithm is in comparison to 4DM. A closer look at the tables and graphs shows that the relationship between the 2 programs is exceptionally strong, showing high hopes for FlowQuant’s validity versus 4DM which is currently considered as the current program of choice.

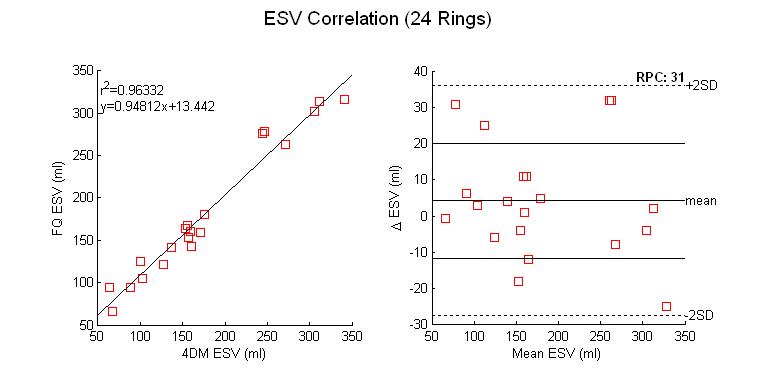


Each of the EDV, EDV and EF columns in FlowQuant are matched and compared with their corresponding column in 4DM using the Bland-Altman analysis method.

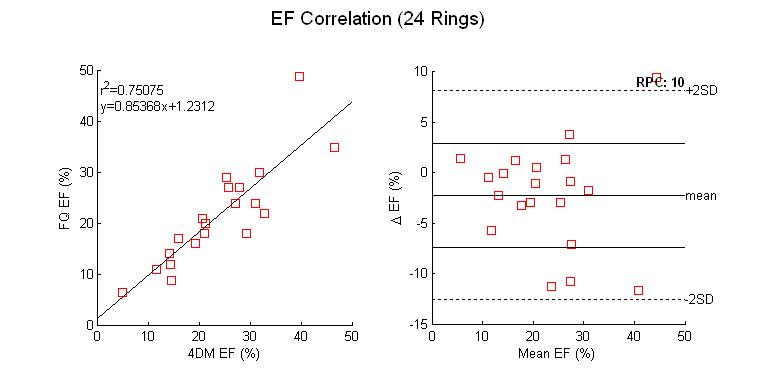


From the first graph, the EDV r2 value is near 0.97 which is remarkably fine (best r2 value is 1), the line’s slope is similarly good (best slope value must be equal to 1), and finally the y-intercept is relatively close to zero (18.3 is relatively a small amount in comparison to the y-values of the other points).

The Bland-Altman comparison graph illustrates a mean value of the difference in volumes calculated by both programs close to zero with a very negligible underestimation of the FlowQuant calculated EDV.



Similarly, the ESV comparison shows high hopes as well for the validity of FQ compared to 4DM. The second graph demonstrates mean value slightly over zero.



The EF correlation is strong as well, with a slight under estimation.

Due to lack of space, the table and graphs above show only the results computed when considering all 24 slices alone the SA as stated previously. The same comparison test is simulated for each number of rings applicable, and the results are stored in the following table.



The table shows the slight effect of the number of rings used in FQ calculations on the results computed. The best values are highlighted in yellow, and clearly it is seen for now that adopting all 24 rings of the LV in FlowQuant’s calculation gives more similar results to 4DM. This makes sense since the user who manually processes the scan in 4DM subconsciously sets the study range to include the entire LV from its apex to the most distant edge of the inclined valve plane as a starting point for the algorithm to detect the valve plane location, yet not its incline angle. However the detection of the myocardium wall is done automatically in FlowQuant with the least user intervention, but the valve plane is assumed to be fixed during the cardiac cycle for now.



The last three graphs show the variation of the correlation factors in respect to the number of rings considered during the volume computations. We observe that the r2 value for the EDV and ESV doesn’t vary greatly with different number of rings, whereas the EF r2 value can change up to 0.8 reaching its maximum with 23 slices.

A similar comparison for the equation’s slope can be seen, for the EDV and ESV, the more rings considered, the nearest the slope it is to 1, yet the EF slope is the highest at 21 rings.

Finally, the EDV and ESV y-intercept comparison for different number of rings considered show values closest to zero with 18 rings and goes higher with the number of rings. However the values are still considered relatively minor in comparison to the high volume values.

# FlowQuant advantage over 4DM?

FlowQuant is as automated as possible allowing the least user intervention, yet still holds extremely good agreement with 4DM results regardless of intervention; therefore the results are not sensitive to the user in FlowQuant.

The automation is beneficial to routine clinic work allowing reduced workload as well as inner-variability and inter-variability in FlowQuant: 4DM’s manual dependence allows great variability each time the scan is processed. The experience using the program is not as essential in FlowQuant as it is to 4DM due its automation, whereas familiarity with manually defining the wall contours in 4DM is indispensable to process the scan and obtaining acceptable results.

FlowQuant has its own Wall Thickening estimation algorithm, a feature yet still unavailable in 4DM, while still showing similarities in 4DM’s regular commonly used tools and features.

It is safe to say that the updated version of FlowQuant, if continuously improved over time, can eventually even challenge 4DM to its title of “The Program of Choice”.

UNIX Support Feature

# Introduction

As for the second part of the workload done this summer, it is needed to say that UNIX platforms are currently considered as the most reliable systems, since they provide very powerful servers that can run basically at non-stop rate, while handling a lot of studies at the same time. Therefore, UOHI sought to replace the current PC platforms with UNIX platforms for their reliability, an important factor to be considered at a vital location like a hospital, where any system failure can result in catastrophic consequences. To accommodate to the hospital’s needs of the new operating systems, FlowQuant was revised and modified to achieve the ability to work flawlessly on PC and on UNIX simultaneously.

# Methodology

The UNIX operating system (OS) holds certain characteristics that are not similar to PC OS.Among those characteristics are: The Case Sensitivity in UNIX, UNIX’s forward slash versus PC’s backward slash in directories names, UNIX specific paths and locations of specific files or tools.

*Case Sensitivity Mismatch:*

Since UNIX emphasizes on case sensitivity, running the FlowQuant program before on UNIX OS resulted in crashing errors as well as logical errors, mainly due to the inability of the program to locate certain files on the computer because of a case sensitivity mismatch in the file’s name between the name used to store the file on the drive, and the used to call for it in Matlab. PC operating systems do not take into account case sensitivity, therefore these error do not show on those platforms. Therefore, in order to create a single version supporting both platforms, some hardcoded file names where modified to match the case sensitivity of the file name, whereas some files names where changed on the drive itself, copied and pasted to overwrite their non-case-sensitive aliases, for faster editing of the program.

One point is to consider from now on: during the implementation of later versions of FlowQuant, the programmers needs to verify case sensitivity matching to avoid future errors.

*Backward/Forward slash dilemma:*

UNIX based operating systems use forward slashes to define a directory or file path location. Having hardcoded backward slashes result in similar errors to the ones created by case sensitivity mismatches. PC OS does not take into account the difference between two the slashes, therefore this error never popped up before. FlowQuant now adopts a function named “filesep” that can detect the type of the operating system currently available and return a forward or backward slash to serve requirements. A MANUAL search/modification of the hardcoded backward slashes was made by me, changing all slashes into “filesep” calling methods.

*UNIX paths and tools:*

Unlike the PC operating system, the UNIX operating systems rely heavily on a console called “The Terminal”, from which everything needed on the drive can be accessed and modified. To accommodate this characteristic, I modified the program to being able to access certain path locations and files that either are located elsewhere, are named differently or do not exist on PC computers. For instance, in order to issue a specific license approval for each computer that runs FlowQuant, the internet computer address was needed to verify validity and issue the approval, however, the command used in the PC command window is “ipconfig” from which the desired internet address is detected, whereas on UNIX/Linux it is called by the terminal using the “ifconfig-all” and then the desired internet address is detected as well, taking into account that it’s name identification is different than on PC.

This process was repeated for all issues involving this kind of modification to fix them.

*Limitations:*

In the beginning, a 7 year old Fedora 5 operating system was already installed on the Unix platform computer, and it created graphical errors on the figures employed by FlowQuant, since not the same graphic simulators are used on the different platforms.

Luckily, the newly released Fedora 9 accommodates these issues, and did not show any graphical meltdown once updated.

The program is limited to the latest versions of Unix operating systems, and user knowledge and expertise on working in a Unix platforms environment is needed to set up the computer to its full potential and acceptance of FlowQuant.

Conclusion:

The newly updated version of FlowQuant with its new features has proven itself user-friendly while defending its validity against 4DM. The Ejection Fraction and Wall Thickening calculating algorithm can be clinically approved for regular daily use, providing a very accurate insight on the patient’s heart condition and functionality and therefore determining proper treatment in time.

In addition, its UNIX compatibility can satisfy the needs of highly involved researchers of a dependable operating system to run exhausting procedures without facing server meltdown, due to the highly recommended endurably and reliance of this platform, thus replacing the currently used PC platforms. Now FlowQuant can be run virtually on any operating system flawlessly, increasing the range of use of the program for, if sought, commercial use.

***Copyrights and Acknowledgements:***

*During the COOP summer work term from May to August 2009, and the continuous development and update of FlowQuant, I was assisted by Mr. Ran Klein as well as supervised by Mr. Robert deKemp, to insure proper programming as well as time and quality efficiency. I do not in any way take credit for any features available in FlowQuant prior to the Gated Report I helped build: FlowQuant has been developed for years by fellow colleagues and I do not take credit for bringing the program to its current popularity.*

*The implementation of the Gated Report was solely my work alone, assisted by Ran Klein on technical emergencies and do not deny any advices given to me by my superiors for quality insurance or unwanted deadlines. The notions available in this report should not be copied in any way or imitated without my consent.*

*“I acknowledge that this report as well as its contents is purely my own work, with assistance and revision done by my supervisors stated above, any modification or editing to this report must be approved by myself and myself only. Also this report meets full confidentiality set by my supervisors and the University of Ottawa Heart Institute, and should not be made public in total or in part of any form, shape or content without consent on my behalf.*

*Any breaking of the requests above will be considered as a violation to this copyright.”*

*Charles Malo, August 2009*

***References***

(1): Lars Stegger, Edwin Heijman, Klqus P.Schäfers, Klaas Nicolay, Michael A. Schäfers, and Gustav J.Strijkers, “Quantification of Left Ventricular Volumes and Ejection Fraction in Mice Using PET, compared with MRI”, The Journal of Nuclear Medicine,Vol.50, No.1, January 2009.

(2): Matlab help section on the “poly2mask” function, © 1984-2007 The MathWorks, Inc .