

UNIVERSITÉ DE BOURGOGNE

MSC COMPUTER VISION

MEDICAL IMAGE ANALYSIS

---

# MRI Prostate Post Processing

---

*Author:*

Mohamed EISSA

April 3, 2015



## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Main Assumptions</b>	<b>2</b>
<b>3</b>	<b>Main Data Structures</b>	<b>2</b>
<b>4</b>	<b>Main Functions</b>	<b>3</b>
4.1	Reading DICOM and JPEG Images . . . . .	3
4.2	Writing DICOM and JPEG Images . . . . .	4
4.3	Writing Anonymized DICOM Images . . . . .	4
4.4	Marking Prostate Zones . . . . .	5
4.5	Calculating Prostate Zones Area and Volume . . . . .	6
4.6	Saving and Loading Prostate Zones Markings . . . . .	6
4.7	Showing 3D . . . . .	7
<b>5</b>	<b>Future Work</b>	<b>7</b>

## List of Figures

1	Main Application Window . . . . .	3
2	Transition Zone is marked, and Central Zone is in edit mode for slice number 38 . . . . .	5
3	3D representation of manually segmented zones of Prostate slices (Transition Zone, Peripheral Zone, Central Zone and Tumour . . . . .	7

# 1 Introduction

In this report I will present my MATLAB software implementation for MRI prostate post processing. Using the application, the user is able to Load MRI prostate images in DICOM Medical Image format and JPEG Image format, segment all prostate zones; transition Zone, Peripheral Zone, central Zone and the tumour for all MRI slices, show a 3D representation of his segmentation. and export it so it can be used later

## 2 Main Assumptions

The application was implemented using MATLAB software with these assumptions in mind

- The user is going to use the application with DICOM and JPEG formats only
- Make the application as much as possible usable with multiple saving functions, to use his saved segmentation later
- Hard coding the GUI instead of the MATLAB GUI Builder, this is for learning purposes and since the MATLAB GUI Builder is generating a lot of unused code.

## 3 Main Data Structures

At the beginning of the source code the following **global** matrices were created to be used for all application functions:

1. **imgs**: An array that contains all loaded images (DICOM or JPEG), every element is an image. It is sorted according to images loading name order.
2. **infos**: An array that contains all loaded images DICOM information, every element is an image. It is sorted according to images loading name order. In case the user is loading a JPEG image, a **.mat** file should also be provided containing the DICOM information.
3. **TZ**: A  $n \times 3$  matrix that contains all user manual segmentation for prostate **Transition Zone**, every row in the Matrix has 3 values.
  - x coordinate of the zone border point.
  - y coordinate of the zone border point.
  - z represents the slice index.

The matrix **is not** sorted by any value since it will be later accessed totally independent of the order of insertion.

4. **PZ**: The same data structure, but for the **Peripheral Zone**.
5. **CZ**: The same data structure, but for the **Central Zone**.
6. **Tumour**: The same data structure, but for the **Tumour**.

## 4 Main Functions

The application has multiple functions from reading DICOM or JPEG image files, save them in other format, manually segment main prostate zones, save the segmentation (for each slice and for all slices) to be used later, show 3D representation depending on segmented zones, ...etc. The main functions are detailed described as follows.

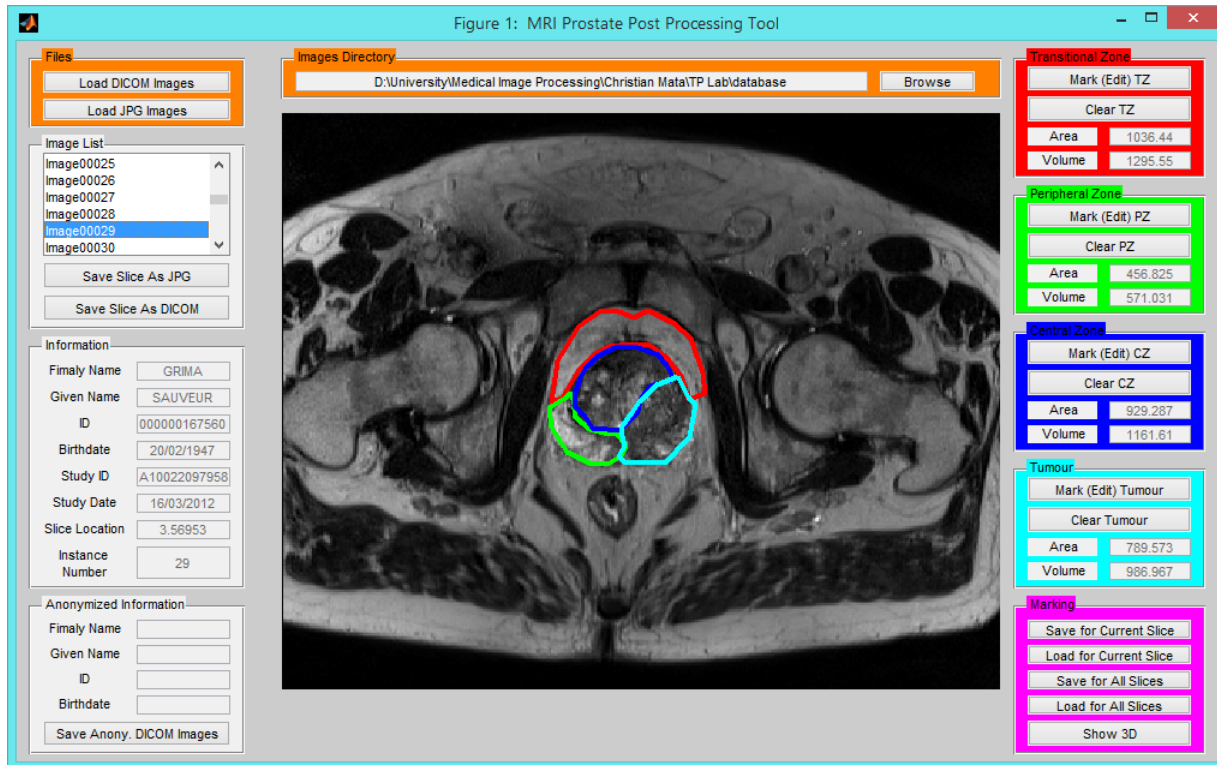


Figure 1: Main Application Window

### 4.1 Reading DICOM and JPEG Images

The application is able to read both DICOM and JPEG MRI images, it loads either DICOM or JPEG images that are located in folder path labeled as **Images Directory**.

The application loads all images located in that path in previously mentioned data structures (**imgs** and **infos**), it used to **take some time to load them all**, all images are listed in list box labeled as **Image List**.

In case JPEG MRI images are loaded, a **.mat** file containing the slice information must also be located in the same folder with the same file name.

```

1 for i = 1:length(images_names_list)
2     img = dicomread(image_path);
3     imgs(:, :, i) = img;
4     infos(i) = dicominfo(image_path);
5 end

```

Once all images are loaded, the first image in the list is displayed in the main image axes, also a set of the slice information is going to be displayed in the area labeled as **Information**. The information displayed are:

- Patient Family Name
- Patient Given Name
- Patient ID
- Patient Birthdate
- Study ID
- Study Date
- Slice Location
- Instance Number

## 4.2 Writing DICOM and JPEG Images

The user is able for each slice to write it as a new DICOM file or a JPEG file combined with **.mat** file containing the DICOM info. This could be done by 2 buttons:

1. **Save Slice As JPEG**
2. **Save Slice As DICOM**

## 4.3 Writing Anonymized DICOM Images

The application is able to save an anonymized version of all images that change the main patient personal data. The user should fill all text fields labeled as **Anonymized Information**, then click on **Save Anony. DICOM Images** to choose a folder to save the new images, the process is expected to take sometime according to the number image slices. The anonymized information are:

1. Patient Family Name
2. Patient Given Name
3. Patient ID
4. Patient Birthdate in *dd/mm/yyyy* format

## 4.4 Marking Prostate Zones

Using the application, the user is able to mark all prostate main zones; Transition Zone, Peripheral Zone, Central Zone and Tumour for each slice, this is done using MATLAB function **impoly()**, also the user is able to edit his marking again by just clicking again on the same button labeled by **Mark (Edit) TZ**, And by clicking on **Clear TZ**, he is able to deleting his marking for the current slice, and this is of course for each zone.

```

1 function markTZButton_Callback(hObject , eventdata , handles)
2     currentTZmark = clearTZ();
3     if ~isempty(currentTZmark)
4         poly = impoly(imageAxes,[currentTZmark(:,1),currentTZmark(:,2)]);
5     else
6         poly = impoly(imageAxes,TZpolyposition);
7     end
8     TZpolyposition = wait(poly);
9     delete(poly);
10    TZpolyposition = [TZpolyposition; TZpolyposition(1,:)];
11    xi = TZpolyposition(:,1); yi = TZpolyposition(:,2);
12    selected_image_index = getSelectedImageIndex();
13    zi = ones(length(xi),1) * selected_image_index;
14    TZ = [TZ; [xi, yi, zi]];
15    hold on;
16    TZplot = plot(imageAxes,xi,yi,TZcolor,'Linewidth',lineWidth);
17 end

```

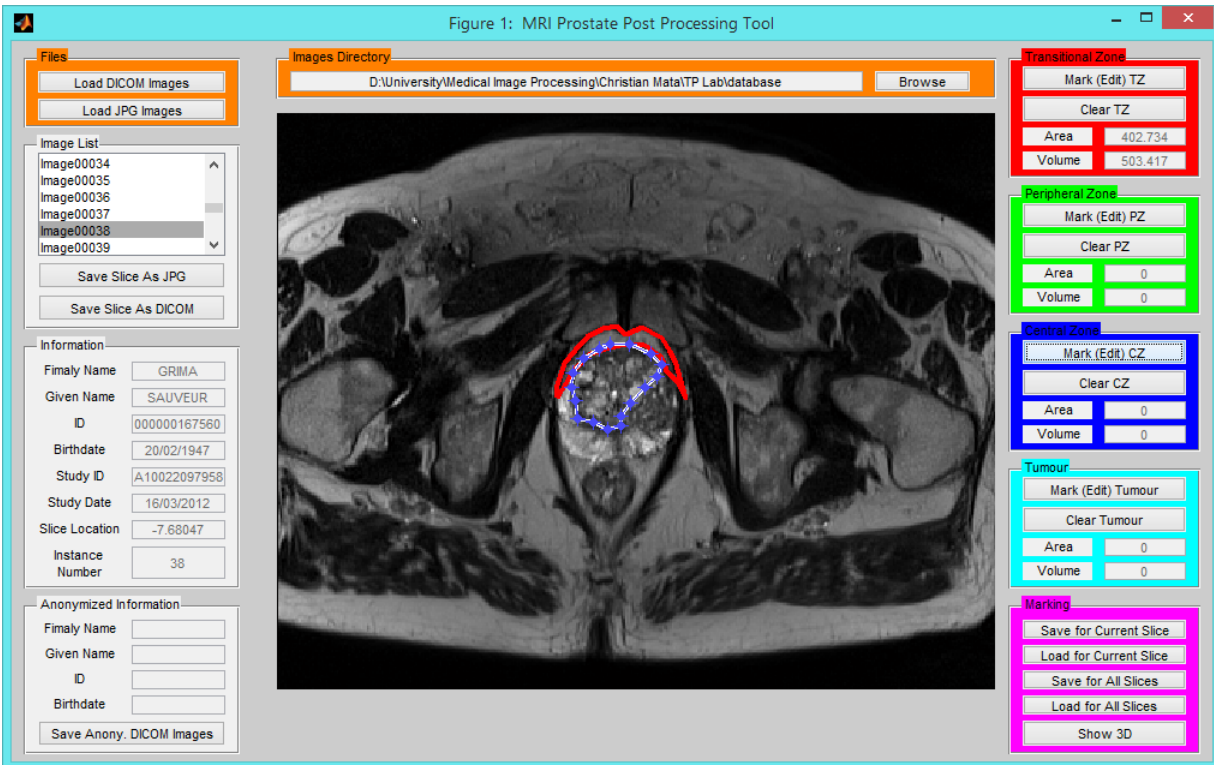


Figure 2: Transition Zone is marked, and Central Zone is in edit mode for slice number 38

## 4.5 Calculating Prostate Zones Area and Volume

The application is able to calculate Area and Volume for every marked zone by the user. Once he plots the marked, for each slice the area and volume is displayed on text fields labeled by **Area** and **Volume**. This is done by the following:

1. Area is calculated by multiplying the marked **polygon total area** by **PixelSpacing** field obtained from DICOM slice information.

$$Area = marked\_polygon\_area \times pixelspacing(1) \times pixelspacing(2)$$

2. Volume is calculated by multiplying the previous calculated **Area** by **SliceThickness** field obtained from DICOM slice information.

$$Volume = Area \times SliceThickness$$

```

1  TZslice = TZ(TZ(:,3) == selected_image_index,:);
2  xi = TZslice(:,1); yi = TZslice(:,2);
3  areaTZ = polyarea(xi,yi) * PixelSpacing(1) * PixelSpacing(2);
4  volumeTZ = areaTZ * SliceThickness;
5  set(findobj('Tag','areaTZEdit'),'String',areaTZ);
6  set(findobj('Tag','volumeTZEdit'),'String',volumeTZ);

```

## 4.6 Saving and Loading Prostate Zones Markings

Last function for the application is to save the all zone marks so it could used to edited later, this could be done for only one image slice or for all images by using the buttons labeled by:

1. **Save for Current Slice:** Save a **.mat** file containing all marked zone for the current displayed image slice.
2. **Load for Current Slice:** Load a **.mat** file containing all marked zone for the current displayed image slice.
3. **Save for All Slices:** Save a **.mat** file containing all marked zone for all slices.
4. **Save for All Slices:** Save a **.mat** file containing all marked zone for for all slices.

Saving marks for all zones for all slices in one file

```

1  slices.TZ = TZ;
2  slices.PZ = PZ;
3  slices.CZ = CZ;
4  slices.Tumour = Tumour;
5  save(filepath,'slices');

```

Saving marks for all zones for all slices from One file

```

1 data = load(filepath);
2 TZ = data.slices.TZ;
3 PZ = data.slices.PZ;
4 CZ = data.slices.CZ;
5 Tumour = data.slices.Tumour;

```

## 4.7 Showing 3D

Using the previous marked zones, the user is able to show a 3D representation by clicking on button labeled **Show 3D**

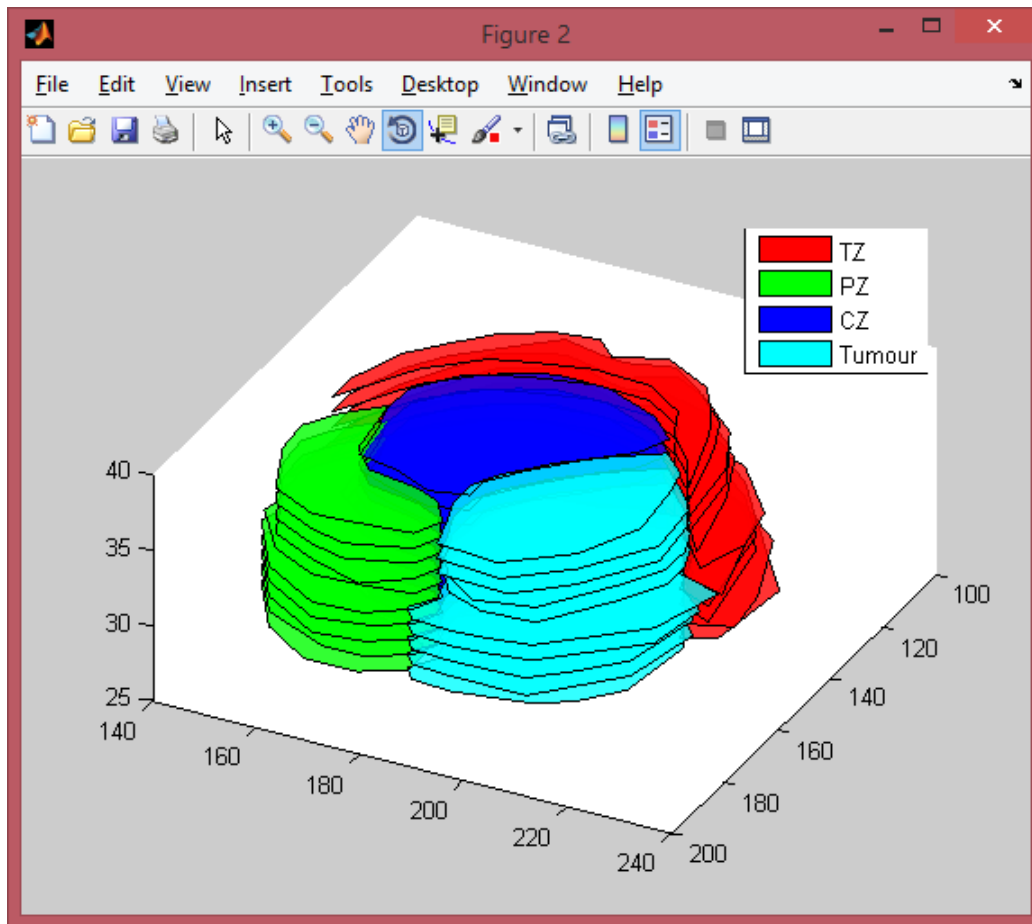


Figure 3: 3D representation of manually segmented zones of Prostate slices (Transition Zone, Peripheral Zone, Central Zone and Tumour)

## 5 Future Work

The purposed application has the basic functions to work with MRI prostate images and segment prostate main zones to render it in 3D model. However, other functions could be implemented such as Zooming, Enhancing Marking techniques, Better 3D rendering and many other features.