

Client-side Medical Image Colorization in a Collaborative Environment

Ioan VIRAG^{a,1}, Lăcrămioara STOICU-TIVADAR^a and Mihaela CRIȘAN-VIDA^a
^a*Politehnica University of Timișoara/Automation and Computer Science Faculty, Timișoara, Romania*

Abstract. The paper presents an application related to collaborative medicine using a browser based medical visualization system with focus on the medical image colorization process and the underlying open source web development technologies involved. Browser based systems allow physicians to share medical data with their remotely located counterparts or medical students, assisting them during patient diagnosis, treatment monitoring, surgery planning or for educational purposes. This approach brings forth the advantage of ubiquity. The system can be accessed from a any device, in order to process the images, assuring the independence towards having a specific proprietary operating system. The current work starts with processing of DICOM (Digital Imaging and Communications in Medicine) files and ends with the rendering of the resulting bitmap images on a HTML5 (fifth revision of the HyperText Markup Language) canvas element. The application improves the image visualization emphasizing different tissue densities.

Keywords. Diagnostic imaging, computer assisted image processing, computer assisted surgery, distance learning

Introduction

The application described in the current paper is part of a larger project that allows the visualization of the processed DICOM images inside a browser, supporting collaborative medicine, focusing on the colorization of the relevant anatomical parts. Several open source software packages such as OsiriX [1] or VolView [2] where developed to interpret the medical data stored in the DICOM format. These allow users to parse the DICOM files and to extract images and patient related data from every single slice. They also provide functionalities such as 3D volume rendering, segmentation, colorization and the animation of the rendered images. The major limitation of these programs is that they require prior installation of specific proprietary operating systems. In order to overcome such issues, browser based solutions were proposed [3, 4] that may run on mobile devices as well, a very important aspect since mobile health and computing is gaining more users every day. The paper presents work that extends the functionalities of such solutions adding a web based colorization module that highlights and isolates the regions of interest in the rendered medical images enhancing the visual aspect of the image and increasing the visibility of abnormal items related to tissue density.

¹ ioan.virag@aut.upt.ro

1. Methods

CT scanners measure the linear x-ray radiodensity attenuation values of body tissues in Hounsfield units (HU) [5], which is a measure of tissue density relative to the density of distilled water at standard pressure and temperature. Pixels in a DICOM file retrieved after CT scanning are stored in terms of relative radiodensity. Each pixel is assigned a value according to the mean attenuation of the tissue that it corresponds to in a range between -1000 and +1000 on the Hounsfield scale. Air has an attenuation of -1000 HU, lung tissue -600 HU, fat -100 HU, water of 0 HU, while denser tissues such as muscles have +40 HU and bone between +400 and +1000 HU. Table 1 shows some HU values for different tissues [6].

Table 1. Tissue types and their corresponding HU values

Tissue type	HU value
Air	-1000
Lung	-600
Fat	-100
Water	0
Soft tissue	+40
Bone	+400 to +1000

Equation (1) describes the linear transformation applied to transform the initial pixel values of the image data generated by the scanner, into input values for the web application; *stored_value* is the initial device specific pixel value that has a meaning only to the scanner that produced it, *rescale_slope* and *rescale_intercept* are values stored in the DICOM file according to the standard.

$$pixel_value = stored_value * rescale_slope + rescale_intercept \quad (1)$$

In case of a non linear transformation, a LUT (lookup table) is applied, also stored in the DICOM file.

After any Modality LUT or Rescale Slope and Intercept specified in the DICOM file have been applied, the Window Center and Window Width attributes specify a linear conversion from stored pixel values to values to be displayed. Window Center contains the input value that is the center of the window. Window Width contains the width of the window according to the DICOM standard. These two parameters specify if a pixel should be visible or not. In a process also known as windowing all the pixels outside the values specified by the window are displayed as black or white depending on the value of the Photometric Interpretation attribute also stored in the DICOM file. According to the standard the lowest and the highest visible values are calculated using Eqs. (2) and (3).

$$lowest_value = window_center - window_width / 2 \quad (2)$$

$$highest_value = window_center + window_width / 2 \quad (3)$$

The transformation of the resulting pixel data from the 2D CT slices it is achieved by mapping the HU to a range of grayscale values.

The application described in this paper retrieves the files already windowed by the NanoDICOM component on the server side. There are two files retrieved from the

server: one is a grayscale image containing the mapped pixels and one that has the relevant tissue data already isolated for the purpose of the 3D volume rendering by adding those slices as textures to parallel 2D planes provided by the Three.js component. The conversion of the mapped HU values is done using the algorithm described in [6] adapted to the web application.

The patients images will be stored in the cloud, ensuring ubiquity and any time access. The system will use only the images from the patients DICOM files. The sets have a unique identifier associated with patient related information and it is accessed only by doctors which will have the possibility to study them.

In order to use the images from the Windows Azure cloud [7], we put the images in a blob storage, which represents a binary data stored as a single entity in a database management system. Figure 1 presents the software architecture.

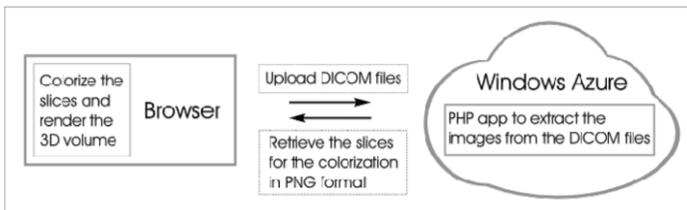


Figure 1. The software architecture

In order to keep the data private and secure we encrypt the images. The name of the blob is in fact an identifier which helps the application to identify the patient. In [8] is described an implementation of the RSA algorithm in Google cloud which can be used also for Microsoft Azure.

The application uses the ImageData interface [9, 10] which represents the underlying pixel data of an area of a <canvas> element in a web page. It is created using the ImageData() constructor which returns a newly instantiated ImageData object. For each pixel in an ImageData object there are stored four values, representing its RGBA components (Red, Green, Blue and Alpha values). Every pixel value is stored in the data property of the ImageData object which is a byte array that can be accessed from JavaScript.

Every time a new DICOM slice is uploaded to the server, the application retrieves the two PNG (Portable Network Graphics) files resulting after the image data extraction by NanoDICOM and segmentation done by the PHP script of the web application. The resulting images are drawn on the left and central canvas for comparison. The third canvas is the one that displays the resulting colored image, obtained after tweaking the min and max values generated by two range type <input> elements that have an associated *onchange* event that fires the JavaScript function responsible for the colorization of the medical images. This colorization is done based on the selected source canvas which can be changed by a <button> element that has an *onclick* event as its attribute that fires another JavaScript function responsible for the selection of the current element.

2. Results

The resulting images make it easier for the physicians to visualize different structures by colorizing the different tissues based on their density, providing a clearer way to interpret the analyzed anatomical regions. The results can be seen in Figure 2.

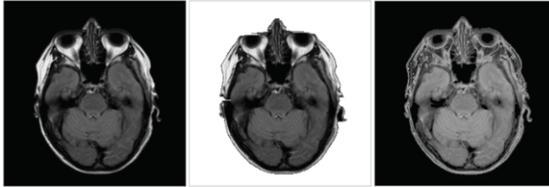


Figure 2. The left canvas is selected as source.

The importance of having such a selection possibility can be seen Figure 3 and 4 where the source image is not clear enough to produce good results.

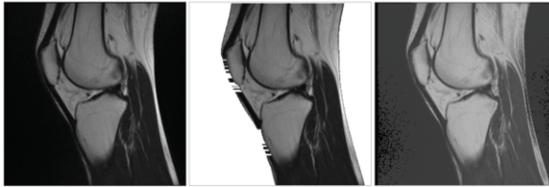


Figure 3. All the noise from the source image can be seen in the resulting image.



Figure 4. We have better results if the source is the middle, segmented image.

Further adjustment of the minimum and maximum entry values can eliminate the unnecessary noise or even add some shading to produce more realistic images. Additional tweaking of the input values makes it possible to isolate the contours of the bones in the image as it can be seen in Figure 5.



Figure 5. Isolating bone structure.

3. Discussion

As we can see from the results the web oriented implementation of the colorization algorithm allows physicians and medical students to access a more detailed and meaningful view of the generated images, thus easing the process of interpretation. This process is very useful since it allows the application to display only specific tissues based on their density. Further work is necessary to enhance the colorization algorithm and to select the region of interest, the resulting images providing the base for the volume rendering and augmented reality modules that are still under development and testing by our team.

In comparison to desktop based applications [1, 2] the main advantage is offered by the mobility of the proposed system. As a disadvantage we could mention the browser oriented architecture. Even if the mobile platforms allow magnification of different regions of the displayed images, this aspect requires some further investigation in order to make the application more suitable for the medical field.

Acknowledgement

This paper is supported by the Sectoral Operational Programme Human Resources Development POSDRU/159/1.5/S/137516 financed from the European Social Fund and by the Romanian Government.

References

- [1] OsiriX, <http://www.osirix-viewer.com/>, 16.11.2014
- [2] VolView, <http://www.kitware.com/opensource/volview.html>, 16.11.2014
- [3] E. J. Melicio Monteiro, C. Costa, J. L. Oliveira, A DICOM viewer based on web technology, *IEEE 15th International Conference on e-Health Networking, Applications & Services (Healthcom, 2013)*, 167-171
- [4] I. Virag, L. Stoicu-Tivadar, E. Amăricăi, Browser-based medical visualization system, *IEEE 9th International Symposium on Applied Computational Intelligence and Informatics (SACI, 2014)*, 355-359
- [5] J. C. Silverstein, N. M. Parsad, and V. Tsirlina, Automatic Perceptual Color Map Generation for Realistic Volume Visualization, *Journal of Biomedical Informatics* **41(6)** (2008), 927-935.
- [6] M. Martinez Escobar, An interactive color pre-processing method to improve tumor segmentation in digital medical images, master's thesis, Iowa State University, 2008.
- [7] Windows Azure, <http://azure.microsoft.com/ro-ro/>, 16.11.2014.
- [8] N. Saravanan, A. Mahendiran, N. Venkata Subramanian and N. Sairam, An Implementation of RSA Algorithm in Google Cloud using Cloud SQL, *Research Journal of Applied Sciences, Engineering and Technology* **4(19)**(2012), 3574-3579
- [9] ImageData specification defined by the WHATWG (Web Hypertext Application Technology Working Group), <https://html.spec.whatwg.org/multipage/scripting.html#imagedata>, 17.11.2014
- [10] ImageData specification adopted by the W3C (World Wide Web Consortium), http://www.w3.org/html/wg/drafts/2dcontext/html5_canvas_CR/#imagedata, 17.11.2014