Web-based platform for collaborative medical imaging research

Leticia Rittner[a], Mariana P. Bento[a], André L. Costa[a], Roberto M. Souza[a],
Rubens C. Machado[b] and Roberto A. Lotufo[a]

[a]School of Electrical and Computer Engineering, UNICAMP, Campinas (SP), Brazil;

[b]Renato Archer - CTI Campinas (SP), Brazil

ABSTRACT

Medical imaging research depends basically on the availability of large image collections, image processing and analysis algorithms, hardware and a multidisciplinary research team. It has to be reproducible, free of errors, fast, accessible through a large variety of devices spread around research centers and conducted simultaneously by a multidisciplinary team. Therefore, we propose a collaborative research environment, named Adessowiki, where tools and datasets are integrated and readily available in the Internet through a web browser. Moreover, processing history and all intermediate results are stored and displayed in automatic generated web pages for each object in the research project or clinical study. It requires no installation or configuration from the client side and offers centralized tools and specialized hardware resources, since processing takes place in the cloud.

Keywords: Medical image computing, Web-based platform, Image Processing, Image Analysis, Collaborative research, Cloud computing

1. INTRODUCTION

Medical imaging computation usually consists of batch processing large datasets using pre-existent scripts and analysis and measurements using interactive tools. Such scenario presents several drawbacks. In the batch processing step, it is very difficult to access the quality of intermediate results, hard to check if the parameters used were adequate for that dataset and the file management must be done manually. After the interactive analysis step, it is usually impossible to retrieve the sequence of operations performed interactively and very hard to check for errors, when a suspect value appears in the analysis. Additionally, some problems arise from the fact that through all steps, spreadsheets and intermediate results have to move from one computer (environment) to another, and often have to be converted from one format to another.

With the recent advances in Internet connections, web interactivity tools such as HTML5 and cloud computing infrastructure, there is an increasing number of medical and neuroimaging web-based projects. The great majority of such projects aims to manage, visualize and share medical data.[1–7] Only a few platforms are intended to perform image processing and analysis.[8–10] The advantages of a web-based analysis, processing and visualization software are its independence of operating systems, the fact that it is not limited to the processing power and storage of the workstation and it is readily accessible from wherever the Internet is available.

Following this tendency of web-based tools, and having in mind the limitations of traditional medical imaging research scenario, our goal is to develop a collaborative web environment for writing text and medical image processing programs that promotes multiple researchers to exchange scientific results along with the algorithms implementation sharing specialized computational resources.

2. METHODS

The proposed environment for medical imaging research is composed by three main elements: 1) a platform called Adessowiki, that hosts the research projects and offers all the convenience of a web-based platform; 2) a repository of basic algorithms, from image processing to pattern recognition, grouped as toolboxes; and 3) the developed pipeline for each specific research project, i.e., specific applications.

Correspondence author: L. Rittner (lrittner@dca.fee.unicamp.br)

Medical Imaging 2015: PACS and Imaging Informatics: Next Generation and Innovations,
edited by Tessa S. Cook, Jianguo Zhang, Proc. of SPIE Vol. 9418, 94180B

2.1 Adessowiki platform

The proposed platform, called Adessowiki,[11] combines literate programming,[12] collaborative development and Web 2.0[13] easiness. It relies heavily on the Internet and its clouds for creation, validation and dissemination of scientific knowledge. Adessowiki is a kind of executable wiki, where the programming code that runs in the server are embedded in the text. These code fragments are executed in a sandbox, with pre-configured hardware and software. Therefore, the client only needs a web browser to develop and test his application.

The Adessowiki platform is composed of two separated web servers: the wiki server and the media server. The main components of the Adessowiki are the wiki itself, the wiki markup language, reST, and the execution module, XSandbox, responsible for the execution of code fragments in a controlled environment. The sandbox is connected to the media server, in such a way that the media created by the code execution are made visible in the web by the server.[11]

Adessowiki is an evolution of a sequence of projects aiming to develop scientific software (Adesso, conceived between 1998 and 2002; Adessoweb, a distributed environment to operate Adesso). It was originally conceived as an educational and research environment[14] within the Image Processing field. During the following years it was extensively used for teaching Image Processing courses and was also explored as a scientific writing tool, since it carries simultaneously documentation, programming code and results of its execution. Several papers,[15–18] dissertations[19] and chapter books[20, 21] have been written in Adessowiki, that turned out to be a successful platform for writing executable papers.[18]

After years collecting experiences and feedback from the scientific community and improving the platform, Adessowiki is now being tested as a new paradigm to host collaborative research projects within the medical imaging computing field. The multidisciplinary aspects of medical computing research projects are naturally addressed in such an environment, where the computer specialist writes the code and puts together the pipeline and the client (usually the medical specialist) has access to the pipeline through a web browser, without any installation or configuration by the client. The data is stored in the server and is also accessible by the research team through the browser.

2.2 Toolboxes (reusable code)

By using the Adessowiki, a kind of wikipedia of algorithms was built, containing solutions to classical problems. For each algorithm there is a page containing: a description of the problem, a description of the algorithm, their proof for correctness and other text of theoretical content; the corresponding code (mostly in Python and NumPy,[22] some in C/C++), and examples of input and output. The implemented algorithms are grouped in toolboxes, such as: Image Processing toolbox; Image Processing through Mathematical Morphology toolbox; Texture attributes extraction toolbox; among others. In addition to the developed toolboxes, Adessowiki is also integrated to third-party libraries and tools, such as: FSL,[23] Scikit-learn,[24] FreeSurfer,[25] among others. Such integration allows one to run third-party software from within the Adessowiki environment by using the *subprocess* module from Python.

2.3 Specific applications

Adessowiki, together with its toolboxes, offer an appropriate environment for each individual Medical Imaging Research project. The access to this environment is controlled through an username and password. The dataset is accessible "within a click". Editing and processing takes place in the cloud through a web browser and all the tools are integrated in Adessowiki. After running the processing pipeline, all pre-processing data, intermediate results, parameters and techniques are accessible through automatically generated web-pages. All the history of every single processing step, even manual interventions, is kept. Once an application is developed, it can be used immediately as the programming code is stored in Adessowiki without the need of further installations or configurations.

3. EXPERIMENTS AND RESULTS

Up to now three different research projects are being tested within the Adessowiki platform: 1) DTI-based segmentation and parcellation of the corpus callosum; 2) Segmentation of brain subcortical structures and cortical thickness analysis on Systemic Lupus Erythematosus (SLE) patients; 3) Etiology-based classification of brain white matter lesions.

3.1 DTI-based segmentation and parcellation of the corpus callosum

An application prototype for segmenting and parcellating the corpus callosum (CC) on diffusion MRI was built and is being tested. The implemented pipeline performs: diffusion tensor imaging pre-processing, automatic midsagittal slice detection, automatic corpus callosum segmentation and parcellation and finally, computation of diffusion properties within each CC region.

Diffusion tensor pre-processing steps (eddy-current correction and tensor estimation) were performed by FSL,[23] using the *subprocess* Python module within the Adessowiki environment. Both automatic methods for midsagittal slice selection and CC segmentation were proposed by our group[26] and implemented in Python/NumPy. The corpus callosum parcellation was also implemented by our group using Python/NumPy, based on the method proposed by Hofer and Frahm.[27] Other parcellation method were also implemented, for comparison purposes.[28]

The pipeline results are the FA mean value for each parcel of the corpus callosum, as well as the FA mean value for the structure as a whole. These values are computed for each subject in the study (dataset) and can be downloaded as a consolidated spreadsheet or can be inspected individually, in web pages containing all intermediate results (Fig. 1).

3.2 Segmentation of brain subcortical structures and cortical thickness analysis on Systemic Lupus Erythematosus (SLE) patients

An application was tested in the Adessowiki environment to allow the segmentation of brain subcortical structures and cortical thickness analysis. The implemented pipeline performs: image pre-processing, automatic segmentation of subcortical structures and automatic cortical thickness estimation.

All steps in this application are being performed by Freesurfer,[25] that was incorporated to the Adessowiki platform in order to organize the dataset and allow the inspection of each individual step of the pipeline. Since processing of each subject takes hours and, as consequence, studies comprising large datasets can take weeks to be performed, a list of the data and the pending processing steps is generated for follow up. Also, all intermediate results for each data in the dataset are stored in the server and some of them are incorporated in a report that can be inspected anytime through a web browser (Fig. 2).

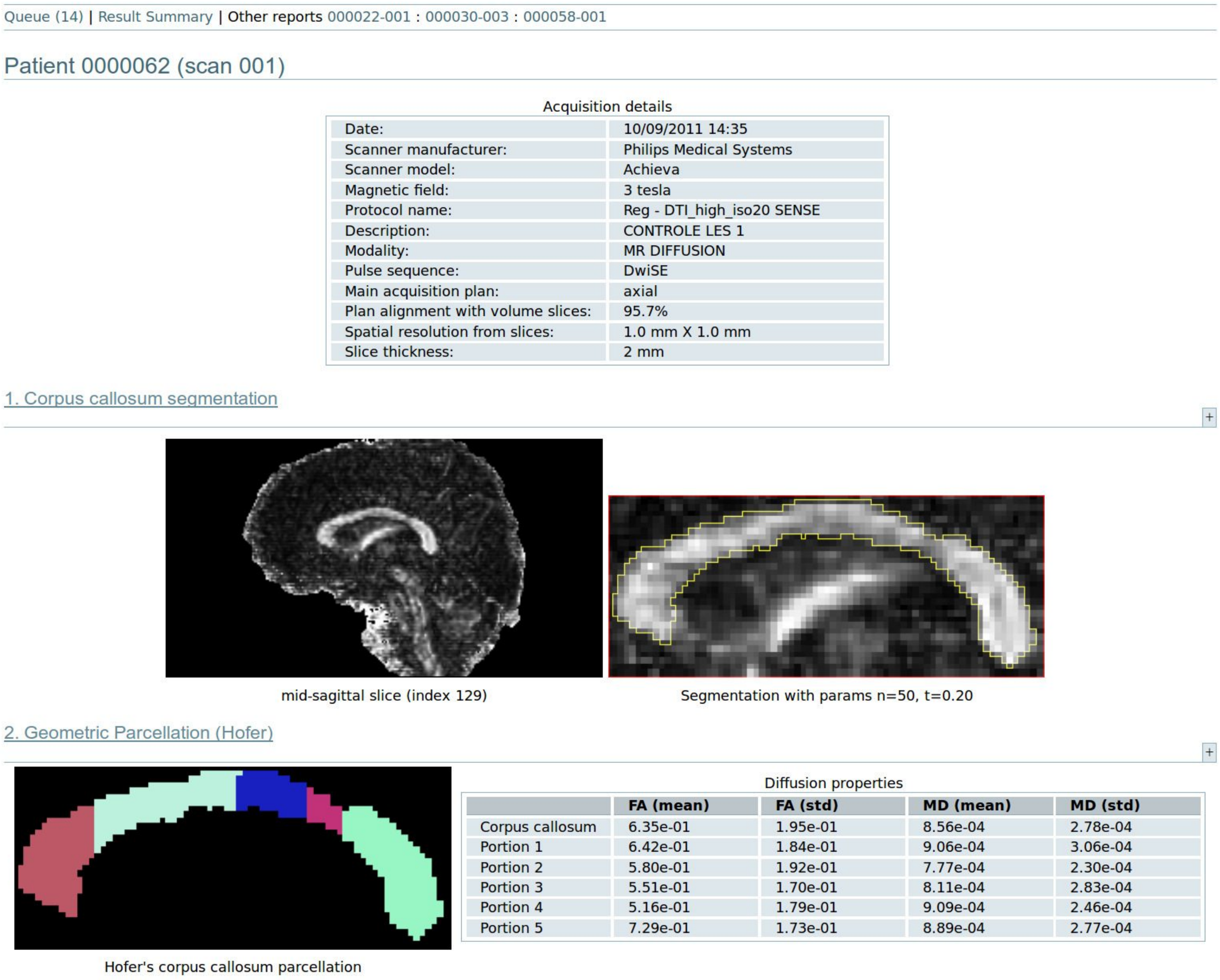
3.3 Etiology-based classification of brain white matter lesions

Another developed application within the Adessowiki platform classifies brain white matter lesions according to their etiology through texture descriptors extracted from T2-weighted MRI. The pipeline comprises: texture attributes extraction from manually selected ROIs, automatic attribute selection, and lesion classification. K-fold cross-validation (K=10) is performed to assess the classifier accuracy through randomly sampled partitions of data.

Texture attributes extraction were implemented by our group in adessowiki using Python/Numpy and are part of the *texture attributes extraction toolbox*: first order statistics based on the histogram and gradient and second-order statistics based on the gray-level co-occurrence matrix and run-length matrix.

The methods to accomplish the attribute selection step (decision tree and principal component analysis), and the classification step (linear component analysis, support vector machine, optimum path-forest and k-nearest neighbors) were performed using the scikits-learn library.[24]

The pipeline results are the confusion matrix and accuracy rates for each classifier. Adessowiki generates a report for the whole dataset, showing only the results in a web page, and hiding the code that has been executed in the server. However, it is possible to check the code of the pipeline by clicking in the "plus" sign in the top right of the result box (Fig. 3).



Queue (14) | Result Summary | Other reports 000022-001 : 000030-003 : 000058-001

Patient 0000062 (scan 001)

Acquisition details

Date:	10/09/2011 14:35
Scanner manufacturer:	Philips Medical Systems
Scanner model:	Achieva
Magnetic field:	3 tesla
Protocol name:	Reg - DTI_high_iso20 SENSE
Description:	CONTROLE LES 1
Modality:	MR DIFFUSION
Pulse sequence:	DwiSE
Main acquisition plan:	axial
Plan alignment with volume slices:	95.7%
Spatial resolution from slices:	1.0 mm X 1.0 mm
Slice thickness:	2 mm

1. Corpus callosum segmentation

mid-sagittal slice (index 129)

Segmentation with params n=50, t=0.20

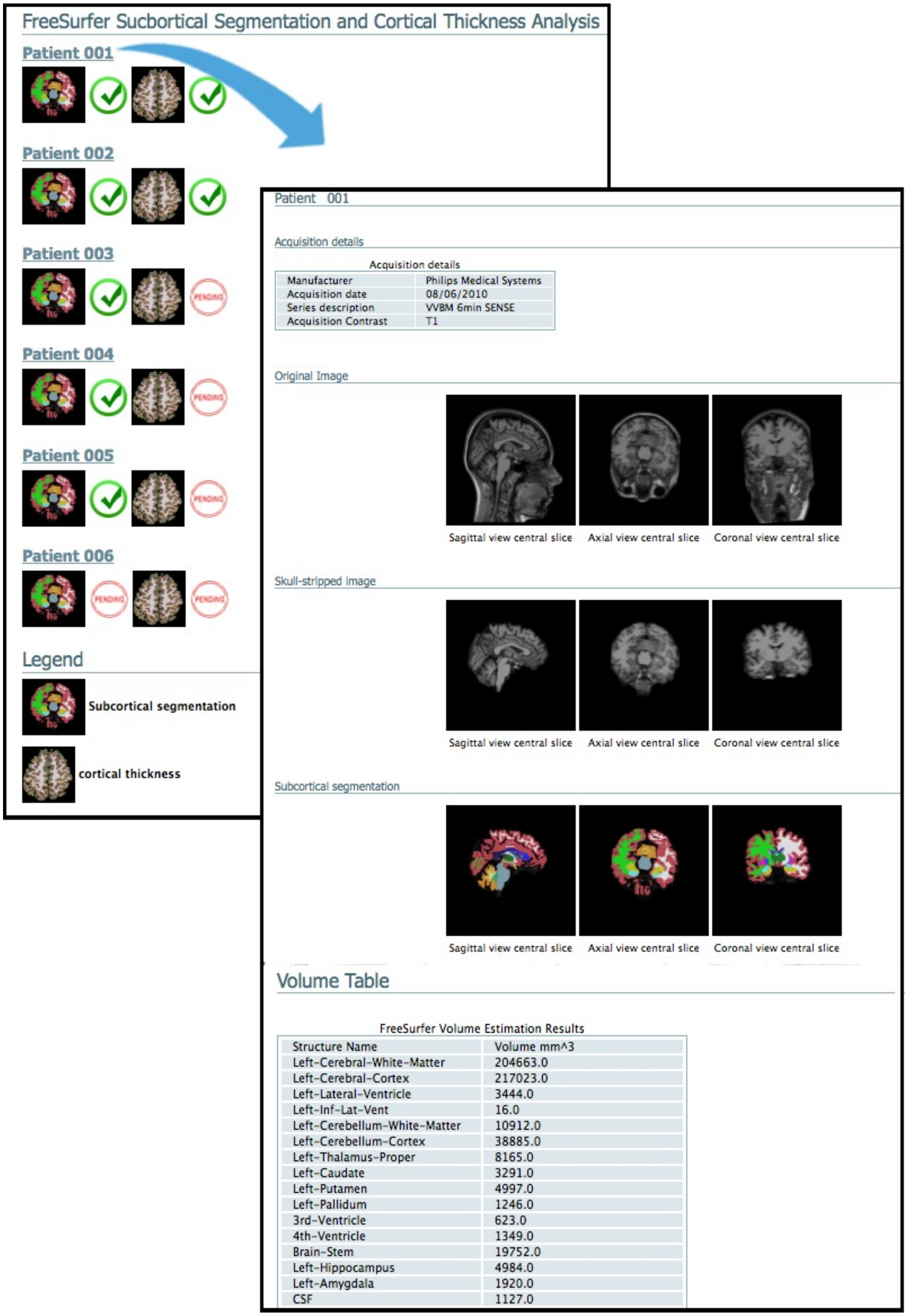
2. Geometric Parcellation (Hofer)

Hofer's corpus callosum parcellation

Diffusion properties

	FA (mean)	FA (std)	MD (mean)	MD (std)
Corpus callosum	6.35e-01	1.95e-01	8.56e-04	2.78e-04
Portion 1	6.42e-01	1.84e-01	9.06e-04	3.06e-04
Portion 2	5.80e-01	1.92e-01	7.77e-04	2.30e-04
Portion 3	5.51e-01	1.70e-01	8.11e-04	2.83e-04
Portion 4	5.16e-01	1.79e-01	9.09e-04	2.46e-04
Portion 5	7.29e-01	1.73e-01	8.89e-04	2.77e-04

Figure 1. Example of one individual report (one single subject) obtained from the pipeline for segmenting and parcellating the corpus callosum



FreeSurfer Sucbortical Segmentation and Cortical Thickness Analysis

Patient 001 ✓ ✓

Patient 002 ✓ ✓

Patient 003 ✓ PENDING

Patient 004 ✓ PENDING

Patient 005 ✓ PENDING

Patient 006 PENDING PENDING

Legend

Subcortical segmentation

cortical thickness

Patient 001

Acquisition details

Acquisition details

Manufacturer	Philips Medical Systems
Acquisition date	08/06/2010
Series description	VVBM 6min SENSE
Acquisition Contrast	T1

Original Image

Sagittal view central slice Axial view central slice Coronal view central slice

Skull-stripped image

Sagittal view central slice Axial view central slice Coronal view central slice

Subcortical segmentation

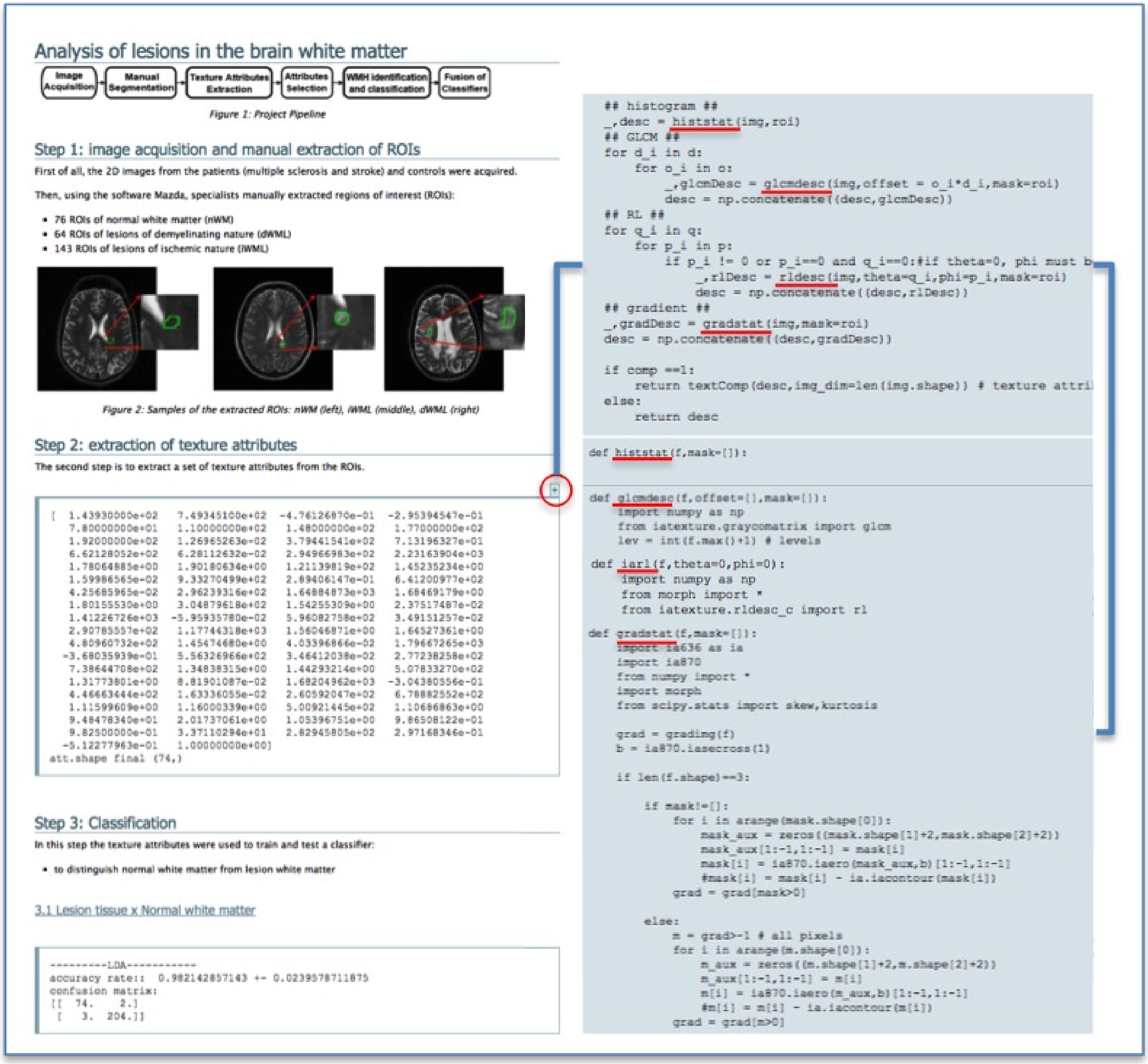
Sagittal view central slice Axial view central slice Coronal view central slice

Volume Table

FreeSurfer Volume Estimation Results

Structure Name	Volume mm^3
Left-Cerebral-White-Matter	204663.0
Left-Cerebral-Cortex	217023.0
Left-Lateral-Ventricle	3444.0
Left-Inf-Lat-Vent	16.0
Left-Cerebellum-White-Matter	10912.0
Left-Cerebellum-Cortex	38885.0
Left-Thalamus-Proper	8165.0
Left-Caudate	3291.0
Left-Putamen	4997.0
Left-Pallidum	1246.0
3rd-Ventricle	623.0
4th-Ventricle	1349.0
Brain-Stem	19752.0
Left-Hippocampus	4984.0
Left-Amygdala	1920.0
CSF	1127.0

Figure 2. Checking the progress of the pipeline. Left: dataset list and pending processing steps; right: segmentation results for patient 001, including generated images from intermediate steps (skull stripping) and table of volumes computed from segmented structures (partial list)



Analysis of lesions in the brain white matter

Figure 1: Project Pipeline

Step 1: image acquisition and manual extraction of ROIs

First of all, the 2D images from the patients (multiple sclerosis and stroke) and controls were acquired.

Then, using the software Mazda, specialists manually extracted regions of interest (ROIs):

- 76 ROIs of normal white matter (nWM)
- 64 ROIs of lesions of demyelinating nature (dWML)
- 143 ROIs of lesions of ischemic nature (iWML)

Figure 2: Samples of the extracted ROIs: nWM (left), iWML (middle), dWML (right)

Step 2: extraction of texture attributes

The second step is to extract a set of texture attributes from the ROIs.

```
[  1.43930000e+02   7.49345100e+02  -4.76126870e-01  -2.95394547e-01
   7.80000000e+01   1.10000000e+02   1.48000000e+02   1.77000000e+02
   1.92000000e+02   1.26965263e-02   3.79441541e+02   7.13196327e-01
   6.62128052e+02   6.28112632e-02   2.94966983e+02   2.23163904e+03
   1.78064885e+00   1.90180634e+00   1.21139819e+02   1.45235234e+00
   1.59986565e-02   9.33270499e+02   2.89406147e-01   6.41200977e+02
   4.25685965e-02   2.96239316e+02   1.64884873e+03   1.68469179e+00
   1.80155530e+00   3.04879618e+02   1.54255309e+00   2.37517487e-02
   1.41226726e+03  -5.95935780e-02   5.96082758e+02   3.49151257e-02
   2.90785557e+02   1.17744318e+03   1.56046871e+00   1.64527361e+00
   4.80960732e+02   1.45474680e+00   4.03396866e-02   1.79667265e+03
  -3.68035939e-01   5.56326966e+02   3.46412038e-02   2.77238258e+02
   7.38644708e+02   1.34838315e+00   1.44293214e+00   5.07833270e+02
   1.31773801e+00   8.81901087e-02   1.68204962e+03  -3.04380556e-01
   4.46663444e+02   1.63336055e-02   2.60592047e+02   6.78882552e+02
   1.11599609e+00   1.16000339e+00   5.00921445e+02   1.10686863e+00
   9.48478340e+01   2.01737061e+00   1.05396751e+00   9.86508122e-01
   9.82500000e-01   3.37110294e+01   2.82945805e+02   2.97168346e-01
  -5.12277963e-01   1.00000000e+00]
att.shape final (74,)
```

Step 3: Classification

In this step the texture attributes were used to train and test a classifier:

- to distinguish normal white matter from lesion white matter

3.1 Lesion tissue x Normal white matter

```
---------LDA-----------
accuracy rate::  0.982142857143 +- 0.0239578711875
confusion matrix:
[[  74.    2.]
 [   3.  204.]]
```

```
## histogram ##
_,desc = histstat(img,roi)
## GLCM ##
for d_i in d:
    for o_i in o:
        _,glcmDesc = glcmdesc(img,offset = o_i*d_i,mask=roi)
        desc = np.concatenate((desc,glcmDesc))
## RL ##
for q_i in q:
    for p_i in p:
        if p_i != 0 or p_i==0 and q_i==0:#if theta=0, phi must b
            _,rlDesc = rldesc(img,theta=q_i,phi=p_i,mask=roi)
            desc = np.concatenate((desc,rlDesc))
## gradient ##
_,gradDesc = gradstat(img,mask=roi)
desc = np.concatenate((desc,gradDesc))

if comp ==1:
    return textComp(desc,img_dim=len(img.shape)) # texture attri
else:
    return desc
```

```
def histstat(f,mask=[]):
```

```
def glcmdesc(f,offset=[],mask=[]):
    import numpy as np
    from iatexture.graycomatrix import glcm
    lev = int(f.max()+1) # levels

def iarl(f,theta=0,phi=0):
    import numpy as np
    from morph import *
    from iatexture.rldesc_c import rl

def gradstat(f,mask=[]):
    import ia636 as ia
    import ia870
    from numpy import *
    import morph
    from scipy.stats import skew,kurtosis

    grad = gradimg(f)
    b = ia870.iasecross(1)

    if len(f.shape)==3:

        if mask!=[]:
            for i in arange(mask.shape[0]):
                mask_aux = zeros((mask.shape[1]+2,mask.shape[2]+2))
                mask_aux[1:-1,1:-1] = mask[i]
                mask[i] = ia870.iaero(mask_aux,b)[1:-1,1:-1]
                #mask[i] = mask[i] - ia.iacontour(mask[i])
            grad = grad[mask>0]

        else:
            m = grad>-1 # all pixels
            for i in arange(m.shape[0]):
                m_aux = zeros((m.shape[1]+2,m.shape[2]+2))
                m_aux[1:-1,1:-1] = m[i]
                m[i] = ia870.iaero(m_aux,b)[1:-1,1:-1]
                #m[i] = m[i] - ia.iacontour(m[i])
            grad = grad[m>0]
```

Figure 3. Pipeline of the etiology-based classification of brain white matter lesions developed in Adessowiki. Left: pipeline description and results visualization; right: by clicking in the "plus" sign on the top right of the result box, the implemented hidden code is shown and can be inspected

4. CONCLUDING REMARKS

The Adessowiki environment combines the collaborative and flexible nature of a wikipedia and the centralized processing in an execution sandbox, which allows sharing of specialized hardware resources. Because it is accessible through Internet, it does not require any configuration and installation.

Application prototypes were built on Adessowiki: the first one for the corpus callosum characterization based on the analysis and processing of diffusion images; the second one for subcortical segmentation and cortical thickness analysis of Systemic Lupus Erythematosus (SLE) patients and the third one for brain white matter lesions classification based on texture attributes from T2 images. The developed applications are being tested in datasets composed of MR images from control and patients, as part of clinical studies.

Other applications can be considered to be developed in Adessowiki. In fact its open-source concept allows anyone (with given access) to customize the reports and inspect every and single result generated by the pipeline. New pipelines can be built by a modular approach, using the toolboxes and the existing pipelines as building blocks.

The proposed platform addresses most problems inherent to traditional medical image processing tools. Everything is kept in the same environment: raw data, hardware, source programs, compilers, executables and results. Therefore, it eliminates the need for installation and configuration by the client (user) and hardware can be in the cloud. It is also less dependent on setting standard procedures and minimizes manual procedures and interventions.

Once the data is processed, the history of every single processing step, even manual interventions is recorded. Final and intermediate results are accessible and displayed through a set of automatically generated web pages (customized reports). Software developers and clinical scientists work in the same platform, facilitating interaction between both groups. It is easier to release new versions and updates as the programs are in a single place. Errors can be monitored automatically.

ACKNOWLEDGMENTS

This work was supported by National Council for the Improvement of Higher Education (CAPES) and São Paulo State Research Funding Agency (FAPESP).

REFERENCES

[1] Poh, C., Kitney, R., and Shrestha, R., "Addressing the future of clinical information systems – web-based multilayer visualization," *Information Technology in Biomedicine, IEEE Transactions on* **11**, 127–140 (March 2007).

[2] Langer, S. G., "Challenges for data storage in medical imaging research," *Journal of Digital Imaging* **24**(2), 203–207 (2011).

[3] Chiang, W.-C., Lin, H.-H., Wu, T.-S., and Chen, C.-F., "Bulding a cloud service for medical image processing based on service-orient architecture," in [*2011 4th International Conference on Biomedical Engineering and Informatics (BMEI)*], **3**, 1459–1465 (Oct. 2011).

[4] Sánchez, A. B. and González-Sistal, A., "Design of a web-tool for diagnostic clinical trials handling medical imaging research," *Journal of Digital Imaging* **24**(2), 196–202 (2011).

[5] Kagadis, G. C., Kloukinas, C., Moore, K., Philbin, J., Papadimitroulas, P., Alexakos, C., Nagy, P. G., Visvikis, D., and Hendee, W. R., "Cloud computing in medical imaging," *Medical Physics* **40**, 070901 (July 2013).

[6] Luo, S., Han, J., and Huang, Y., "A web-based 3D medical image collaborative processing system with videoconference," in [*Fifth International Conference on Digital Image Processing*], *Proc. SPIE* **8878**, 88781V–88781V–4 (2013).

[7] Kwon, P.-J., Kim, H., and Kim, U., "A study on the web-based intelligent self-diagnosis medical system," *Advances in Engineering Software* **40**(6), 402 – 406 (2009).

[8] Matsopoulos, G. K., Kouloulias, V., Asvestas, P., Mouravliansky, N., Delibasis, K., and Demetriades, D., "MITIS: a www-based medical system for managing and processing gynecological-obstetrical-radiological data," *Computer Methods and Programs in Biomedicine* **76**(1), 53 – 71 (2004).

[9] Mahmoudi, S. E., Akhondi-Asl, A., Rahmani, R., Faghih-Roohi, S., Taimouri, V., Sabouri, A., and Soltanian-Zadeh, H., "Web-based interactive 2D/3D medical image processing and visualization software," *Computer Methods and Programs in Biomedicine* **98**(2), 172 – 182 (2010).

[10] Chen, S., Bednarz, T., Szul, P., Wang, D., Arzhaeva, Y., Burdett, N., Khassapov, A., Zic, J., Nepal, S., Gurevey, T., and Taylor, J., "Galaxy + Hadoop: Toward a collaborative and scalable image processing toolbox in cloud," in [*Service-Oriented Computing ICSOC 2013 Workshops*], **8377**, 339–351, Springer International Publishing (2014).

[11] Lotufo, R. A., Machado, R. C., Körbes, A., and Ramos, R. G., "Adessowiki - on-line collaborative scientific programming platform," in [*The 5th International Symposium on Wikis and Open Collaboration*], (2009).

[12] Knuth, D. E., "Literate programming," *The Computer Journal* **27**(2), 97–111 (1984).

[13] O'Reilly, T., "What is web 2.0. design patterns and business models for the next generation of software.." http://oreilly.com/web2/archive/what-is-web-20.html (2005).

[14] Silva, A., Lotufo, R., Machado, R., and Saúde, A., "Toolbox of image processing using the python language," in [*Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*], **3**, III–1049–52 vol.2 (Sept 2003).

[15] Vitor, G., Ferreira, J. V., and Körbes, A., "Fast image segmentation by watershed transform on graphical hardware," in [*Proceedings of 30th CILAMCE*], (2009).

[16] Körbes, A. and Lotufo, R. A., "Analysis of the watershed algorithms based on the breadth-first and depth-first exploring methods," in [*Brazilian Symposium on Computer Graphics and Image Processing*], IEEE Computer Society (2009).

[17] Körbes, A. and Lotufo, R. A., "On watershed transform: Plateau treatment and influence of the different definitions in real applications," in [*Proceedings of 17th International Conference on Systems, Signals and Image Processing*], 376–379 (2010).

[18] Machado, R. C., Rittner, L., and Lotufo, R. A., "Adessowiki - collaborative platform for writing executable papers," in [*Proceedings of the International Conference on Computational Science, ICCS 2011*], *Procedia Computer Science* **4**(0), 759 – 767 (2011).

[19] Körbes, A., *Análise de Algoritmos da Transformada Watershed*, Master's thesis, University of Campinas, Campinas, SP, Brazil (2010).

[20] Lotufo, R. A., Audigier, R., Saúde, A. V., and Machado, R. C., [*Microscope Image Processing*], ch. Morphological Image Processing, 113–158, Elsevier Academic Press, 1st ed. (2008).

[21] Lotufo, R. A., Rittner, L., Audigier, R., Saúde, A. V., and Machado, R., [*Biomedical Image Processing – Methods and Applications*], ch. Morphological Image Processing Apllied in Biomedicine, 107–128, Springer-Verlag, Berlin, 1st ed. (2011).

[22] Oliphant, T. E., "Guide to numpy." http://www.tramy.us/ (Mar. 2006).

[23] Jenkinson, M., Beckmann, C. F., Behrens, T. E., Woolrich, M. W., and Smith, S. M., "FSL," *NeuroImage* **62**(2), 782 – 790 (2012).

[24] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., "Scikit-learn: Machine learning in python," *J. Mach. Learn. Res.* **12**, 2825–2830 (Nov. 2011).

[25] Fischl, B., "Freesurfer," *Neuroimage* **62**(2), 774–81 (2012).

[26] Freitas, P., Rittner, L., Appenzeller, S., and Lotufo, R., "Watershed-based segmentation of the midsagittal section of the corpus callosum in diffusion MRI," in [*Graphics, Patterns and Images (SIBGRAPI), 2011 24th Conference on*], 274–280 (Aug 2011).

[27] Hofer, S. and Frahm, J., "Topography of the human corpus callosum revisited - Comprehensive fiber tractography using diffusion tensor magnetic resonance imaging," *NeuroImage* **32**(3), 989 – 994 (2006).

[28] Witelson, S. F., "Hand and sex differences in the isthmus and genu of the human corpus callosum," *Brain* **112**(3), 799–835 (1989).