LRD Upgrade

Amir Salama Davide Morgagni

Biomedical Informatics, 2014

LRD / LAR RUNNING DEMO

What is LRD

LRD is a software that enables automatic model extraction from a set of medical images (like MRI). Such feature is achieved through a process in which a user gives input and other parameters to start the computation. The whole software is based on the mathematical LAR framework and the OpenCL based framework “HPC LAR”.

Functioning of LRD

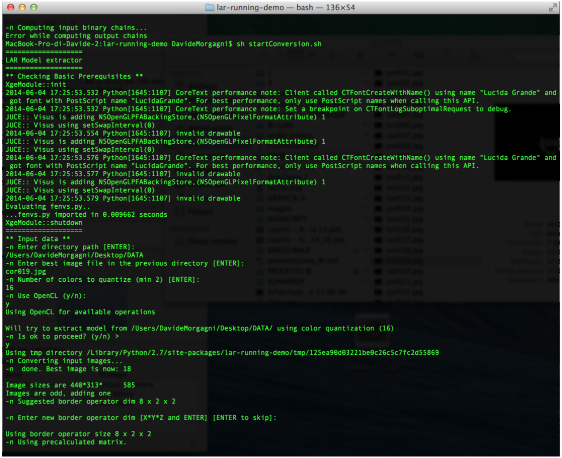
The current version of the LRD is composed of several modules that allow you to automatically extract all the possible models from a stack images (for example, from a magnetic resonance imaging). The input images must be in PNG, JPG or DCM. Executions can be controlled only from the command line. LRD will extract one of the possible models, and show it using one of the available viewers. Viewers supported are PyPlasm, MeshLav and Manta.

Functioning of LRD

The extraction of the model takes place through a filter that transforms each image from a gray scale to a bitonal image, using an edge operator. In this way the bone tissue is identified and from these images is rendered 3D model.

Defects

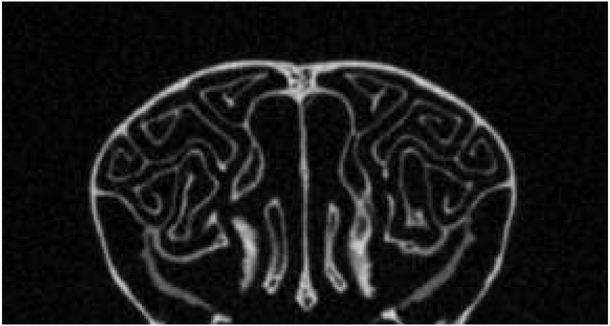
- Exclusive use from the command line



```
-n Computing input binary chains...
Error while computing output chains
MacBook-Pro-di-Davide-2:lar-running-demo DavideMorgagni$ sh startConversion.sh
===================
LAR Model extractor
===================
** Checking Basic Prerequisites **
XgeModule::init
2014-06-04 17:25:53.532 Python[1645:1107] CoreText performance note: Client called CTFontCreateWithName() using name "Lucida Grande" and
 got font with PostScript name "LucidaGrande". For best performance, only use PostScript names when calling this API.
2014-06-04 17:25:53.532 Python[1645:1107] CoreText performance note: Set a breakpoint on CTFontLogSuboptimalRequest to debug.
JUCE:: Visus is adding NSOpenGLPFABackingStore,(NSOpenGLPixelFormatAttribute) 1
JUCE:: Visus using setSwapInterval(0)
2014-06-04 17:25:53.554 Python[1645:1107] invalid drawable
JUCE:: Visus is adding NSOpenGLPFABackingStore,(NSOpenGLPixelFormatAttribute) 1
JUCE:: Visus using setSwapInterval(0)
2014-06-04 17:25:53.576 Python[1645:1107] CoreText performance note: Client called CTFontCreateWithName() using name "Lucida Grande" and
 got font with PostScript name "LucidaGrande". For best performance, only use PostScript names when calling this API.
2014-06-04 17:25:53.577 Python[1645:1107] invalid drawable
JUCE:: Visus is adding NSOpenGLPFABackingStore,(NSOpenGLPixelFormatAttribute) 1
JUCE:: Visus using setSwapInterval(0)
2014-06-04 17:25:53.579 Python[1645:1107] invalid drawable
Evaluating fenvs.py..
...fenvs.py imported in 0.009662 seconds
XgeModule::shutdown
===================
** Input data **
-n Enter directory path [ENTER]:
/Users/DavideMorgagni/Desktop/DATA
-n Enter best image file in the previous directory [ENTER]:
cor019.jpg
-n Number of colors to quantize (min 2) [ENTER]:
16
-n Use OpenCL (y/n):
y
Using OpenCL for available operations

Will try to extract model from /Users/DavideMorgagni/Desktop/DATA/ using color quantization (16)
-n Is ok to proceed? (y/n) >
y
Using tmp directory /Library/Python/2.7/site-packages/lar-running-demo/tmp/125ea90d03221be0c26c5c7fc2d55869
-n Converting input images...
-n  done. Best image is now: 18

Image sizes are 440*313*      585
Images are odd, adding one
-n Suggested border operator dim 8 x 2 x 2

-n Enter new border operator dim [X*Y*Z and ENTER] [ENTER to skip]:

Using border operator size 8 x 2 x 2
-n Using precalculated matrix.
```

Defects

- Actual version of filter does not work well for some forms of bone



Upgrading LRD

Improvements

The new features of the software will be:

- A new filter that is able to extract all types of bones
- Web interface to make the software easy to use

Requirements of filter

- Able to extract the correct model in the presence of bones in hen

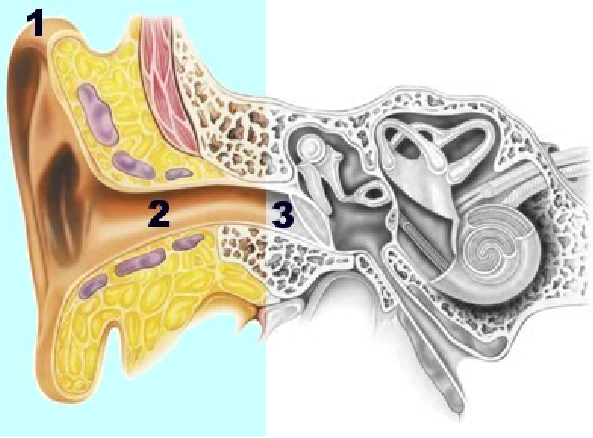


Figure : Internal structure of ear

Operation of Web Service

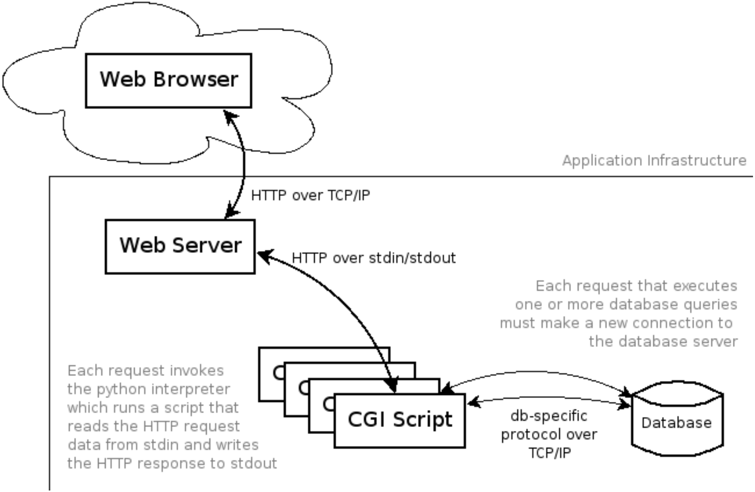

Web Browser
Application Infrastructure
HTTP over TCP/IP
Web Server
HTTP over stdin/stdout
Each request that executes one or more database queries must make a new connection to the database server
Each request invokes the python interpreter which runs a script that reads the HTTP request data from stdin and writes the HTTP response to stdout
CGI Script
db-specific protocol over TCP/IP
Database

Figure : Web service operation

Technologies

Technologies

- Back-end: Python
    - Framework (Django with WSGI)
- Front-end: CSS, HTML5

Django

Django is a free and open source web application framework, written in Python, which follows the model–view–controller architectural pattern. It provides a number of features that facilitate the rapid development of applications for managing content.

WSGI

The Web Server Gateway Interface is a specification for simple and universal interface between web servers and web applications or frameworks for the Python programming language.

WSGI vs CGI

WSGI servers handle processing requests from the web server and deciding how to communicate those requests to an application framework's process. The segregation of responsibilities is important for efficiently scaling web traffic. So WSGI is more scalable and flexible than CGI

Evolution

The doctor will use the system to select the tac you want to display the 3D model. The web service starts the application and return the model to the browser.

Roadmap of work

1. Search and selection of the best algorithms for image processing to implement a filter better than the current (Amir Salama / Davide Morgagni)
2. Filter implementation (Davide Morgagni)
3. Web service development (Amir Salama)