

# **Enhanced human computer interface through webcam image processing library**

2008 *Natural User Interface Group* Summer of Code Application  
Daniel Lélis Baggio

## **Abstract**

*This application is a webcam image processing library on top of OpenCV intended to generate events from user's head, hand and body movements. This library is also intended to track objects so that augmented reality can be made. In order to enhance human computer interaction, the application is going to use a single webcam, without the needs to use FTIR or Diffused Illumination techniques. Besides tracking positions, this library will also be able to provide higher level events and gestures like get 3d user position, and open hand gestures. Collision with virtual objects is also considered in augmented reality.*

## **Features:**

*2d/3d head tracking, hand/finger tracking, body tracking, gesture recognition, fiducial marks, motion velocity, augmented reality object tracking.*

*Although OpenCV already provides lots of low-level functions like 2d head-tracking, higher level functions are still needed for programmers to take full advantage of enhanced human computer interactions. Besides that, lots of algorithms are not implemented in OpenCV. Most of available algorithms are only accessible through vendors, while some are closed source or depend on prototype research projects. It must be noticed that some events could also be available through TUIO protocol.*

*Aligned with Natural User Interface idea of benefiting artistic and educational applications, this library's use could range from helping people severely paralyzed or afflicted with diseases such as ALS (Lou Gehrig's disease) to revolutionary user interface paradigms. 3d head-tracking can create an environment so that Internet browser contents could be zoomed in and out as well as visualize 3d models from user perspective of view. Hand and finger tracking can be used in eye-toy like games as well as scrolling book pages or photos from distance, as when showing photos to friends or in PowerPoint presentations. Similar functionalities like Touchlib's could also be used. Augmented reality can bring a brand new revolutionary paradigm of user interface through an unexplored mix of virtual and real objects for never seen interaction mechanisms. Library demo applications like these should be included. An interesting trait of this library is that users won't need to wear any equipment!*

**Name:**Daniel Lélis Baggio  
**e-mail:** danielbaggio@gmail.com  
**Timezone:** GMT-3  
**Age:** 24

**Education:**

2008 - present: Instituto Tecnológico de Aeronáutica, Brazil  
PhD, Image Processing

2006 - 2007: Instituto Tecnológico de Aeronáutica, Brazil  
Masters, General Purpose Programming for Graphic Processing Units (GPGPU)

2002 - 2006: Instituto Tecnológico de Aeronáutica, Brazil  
Computer Engineering Major

**Blog url:** danielbaggio.blogspot.com

**Location:** São José dos Campos, São Paulo, Brazil

**Active NUI member?** Just introduced

**Previous experience:**

My graduation thesis (<http://umn.dl.sourceforge.net/sourceforge/ivussnakes/TG.pdf>) was about medical image processing. I have developed an algorithm for intra vascular coronary lesion detection. The algorithm finds edges in a video sequence and traces the shortest path from one segmented point to another using Dijkstra's algorithm and snakes technique. Besides working with images I have developed my master thesis in GPGPU based algorithms. My master thesis is available here (<http://code.google.com/p/gpuwire/> and <http://gpuwire.googlecode.com/files/Master%20Thesis%20-%20Updated%20February%2015th.pdf>).

As I've seen in your project ideas, I could also collaborate with the "GPU accelerated blob detection and tracking algorithms", since I have developed GPGPU applications. I am also developing Ps3 Computational Fluid Dynamics applications (<http://code.google.com/p/ps3hacking/>). My previous experience with open source programming include my ImageJ plugin (<http://ivussnakes.sourceforge.net/>) with around 2,000 downloads and my previous participation in SoC 2007 (<http://code.google.com/soc/2007/lurie/appinfo.html?csaid=DF6F247475AC2839>) and I have been an Ubuntu Portuguese translator. I have also packaged some Debian software, so I could also help out with Linux support for TouchLib.

My previous experience with HCI is related to an old blog post (<http://danielbaggio.blogspot.com/2008/01/webcam-opengl-opencv-head-tracking.html>) and its enhanced version (<http://danielbaggio.blogspot.com/2008/03/enhanced-version-of-head-tracking-and.html>).

These videos show my OpenCV and OpenGL based applications that create a virtual environment showing 3d models based on the user's point of view. The software uses OpenCV head tracking function to determine where is the bi-dimensional position of the user and displays the 3d box model according to this position. Only webcam image processing is used, so no other equipment is required to be worn.

I have also been working with realtime chroma keying (<http://www.youtube.com/watch?v=aE0F4F5WlU&NR=1> and <http://www.youtube.com/watch?v=eBS0SUN4vKA>).

Besides working with image processing before, I have always competed in algorithm contests. One of the most important was back in 2004 ACM International Collegiate Programming Contest in which my team has drawn with MIT and CalTech (<http://icpc.baylor.edu/past/icpc2005/Finals/Standings.html> and [http://icpc.baylor.edu/dmt/media/indexed/0406-2350-IMG\\_7493%5BLO%5D.jpg](http://icpc.baylor.edu/dmt/media/indexed/0406-2350-IMG_7493%5BLO%5D.jpg)).

My industry background has been an internship at IBM in the Extreme Blue project, Google Summer of Code 2007 and developing in a military software factory.

As I have been an advocate for Open Source philosophy from the first time I got in contact with it

and that my proposal could not only produce innovative software, like NUI does, but also help people with disabilities, I believe that my background experience with the help from NUI could make of me the best person to accomplish this project.

### **Development methodology:**

Produced software will be a C++ library on top of OpenCV with demo applications and documentation. A TUIO protocol layer might be added. Main features and a brief description of the scientific background needed to develop them are listed as follows:

-*Augmented reality*: track 3d models so that they can interact with virtual objects. A paper about it is available here: <http://www.bmva.ac.uk/bmvc/2000/papers/p66.pdf>

Some videos showing amazing effects obtained: <http://www.youtube.com/watch?v=enXTKvhE7yk>  
<http://www.youtube.com/watch?v=g8Eycccww6k>  
<http://www.youtube.com/watch?v=kM6QDd0XqQ4>

-*2d head tracking*: track bi-dimensional user frontal and profile face

This feature is already implemented in OpenCV through Hidden Markov Models as described in the paper "Face Recognition Using An Embedded HMM" (<http://citeseer.ist.psu.edu/rd/98354121%2C238175%2C1%2C0.25%2CDownload/http://citeseer.ist.psu.edu/cache/papers/cs/10636/http%3A%2FzSzzSzusers.ece.gatech.edu%2Fsz%2F7EarazSzavbpa99.pdf%2Fnefian99face.pdf>).

I have already developed an application that deals with this, which is demonstrated here(<http://danielbaggio.blogspot.com/2008/03/enhanced-version-of-head-tracking-and.html>)

-*3d head tracking*: since only one webcam is used, no stereo vision is available. In order to overcome this, eye distance can be measured to get depth information as well as combine with 2d head tracking, so that this task can be accomplished.

Resources:

<http://www.youtube.com/watch?v=DXlCA995sjY>  
<http://www.chrisharrison.net/projects/leanandzoom/1059-harrison.pdf>  
<http://www.kuubee.com/>

-*Hand tracking*: hand and finger position can be acquired through image processing. Some ideas focus the use of two webcams ([http://www.youtube.com/watch?v=9\\_cm2itidzU&feature=related](http://www.youtube.com/watch?v=9_cm2itidzU&feature=related)), but some attempt similar to this (<http://server.cs.ucf.edu/~vision/papers/fg2000.pdf>) might be tried. More info: [http://www.cs.toronto.edu/~smalik/downloads/2503\\_project\\_report.pdf](http://www.cs.toronto.edu/~smalik/downloads/2503_project_report.pdf)

-*Body tracking*: OpenCV already has some algorithms for body tracking.

-*Gesture recognition*: most of the background available from Natural User Interface can be used to create gesture recognition algorithms.

More info: [http://ibm-cvut.felk.cvut.cz/srp2/gesture\\_recognition/doc/doc.pdf](http://ibm-cvut.felk.cvut.cz/srp2/gesture_recognition/doc/doc.pdf)

-*Fiducial marks*: interesting points of user face, such as eyebrows or lips can be tracked, so that interesting events can be provided to programmers such as mouth opening, eye blinking or facial expressions.

OpenCV already implements Lucas Kanade algorithm ([http://en.wikipedia.org/wiki/Lucas\\_Kanade\\_method](http://en.wikipedia.org/wiki/Lucas_Kanade_method)). More info:

[http://robots.stanford.edu/cs223b04/algo\\_tracking.pdf](http://robots.stanford.edu/cs223b04/algo_tracking.pdf)

A demo tracking mouth and eyebrow movements is available here: <http://www.youtube.com/watch?v=zNqCNMefyV8>

-*Movements velocity*: use optical flow to track gesture velocity.

OpenCV already provides `cvUpdateMotionHistory`, but higher level functions like `handMotion` or `headMotion` are still required.

More info: [http://robots.stanford.edu/cs223b05/notes/CS%20223-B%20T1%20stavens\\_opencv\\_optical\\_flow.pdf](http://robots.stanford.edu/cs223b05/notes/CS%20223-B%20T1%20stavens_opencv_optical_flow.pdf)

### **Brief list of deliverables:**

#### *1<sup>st</sup> Month:*

Hand tracking and hand gesture recognition classes. Zoom and rotation features. Documentation of classes through tutorials, code documentation and demos.

#### *2<sup>nd</sup> Month:*

Head and body tracking facade classes. 3D head tracking class. Documentation and small OpenGL demos.

#### *3<sup>rd</sup> Month:*

Motion flow and 3d model wireframe tracking classes. Documentation. Project packaging through Google Summer of Code and Natural User Interface sites.

### **Work schedule:**

#### *1<sup>st</sup> Month:*

\* Hand tracking/gesture:

- 1<sup>st</sup> week: Study and implement Viola-Jones ([http://research.microsoft.com/~viola/Pubs/Detect/violaJones\\_IJCV.pdf](http://research.microsoft.com/~viola/Pubs/Detect/violaJones_IJCV.pdf)) paper for hands
- 2<sup>nd</sup> week: Study and implement Flock-of-features (<http://www.movesinstitute.org/~kolsch/handvu/KolschTurk2004Fast2DHandTrackingWithFlocksOfFeatures.pdf>)
- 3<sup>rd</sup> week: Study and implement posture recognition and hand gestures ([http://www.movesinstitute.org/~kolsch/pubs/Dissertation\\_twoside.pdf](http://www.movesinstitute.org/~kolsch/pubs/Dissertation_twoside.pdf))
- 4<sup>th</sup> week: Test features and integrate developed code in an easily accessible C++ class. Show zoom and rotate functionalities.

#### *2<sup>nd</sup> Month:*

\* Head and body tracking

- 1<sup>st</sup> week: Facade classes for OpenCV already implemented head and body tracking.
- 2<sup>nd</sup> week: Study and implement head distance information.
- 3<sup>rd</sup> week: \* Combine 2d head tracking and head distance, so that 3d head tracking is done.
- 4<sup>th</sup> week: \* Integration tests and integrated classes. Deliver small OpenGL based demos and tutorials on how to use the framework.

#### *3<sup>rd</sup> Month:*

\* Motion flow and augmented reality

- 1<sup>st</sup> week: \* Create easy to access objects that react to motion flow, similar to the ones I've developed here: (<http://www.youtube.com/watch?v=QJvKT-NId9M>)
- 2<sup>nd</sup> week: \* Study and implement 3d model tracking through wireframes (<http://www.bmva.ac.uk/bmvc/2000/papers/p66.pdf>)
- 3<sup>rd</sup> week: \* Integrate developed research in easily accessible classes and write documentation.
- 4<sup>th</sup> week: \* Time to develop side projects as packaging TouchLib for Linux or to use in case prior time wasn't enough for some features.