**Target frame Barcode TAG identification software**

Linux Setup

*Installation and Compilation*

Author: Dragos C. Popescu

Contact: [dragos.popescu@eli-np.ro](mailto:dragos.popescu@eli-np.ro)

Coordinator: Mihail O. Cernaianu

Date created: 07.03.2017

|  |
| --- |
| **Disclaimer !!!**  The source code was compiled under **Linux 14.04 64bit** using:   * **OpenCV 2.4.13** * **Zbar 0.10**   but it should work using any other Linux setup. |

**1.OpenCV Setup**

* Go through the following instructions in order to install **OpenCV**:

<http://docs.opencv.org/2.4/doc/tutorials/introduction/linux_install/linux_install.html>

|  |
| --- |
| **Warning !!!**  This will also install GCC and CMake which are required later for compiling the source code of the application |

**2.Zbar Setup**

* Open **terminal** and run:

*sudo apt-get install libzbar-dev*

* Enter your password and press **y** and **Enter** when you are prompted.

**3.Application source code compilation**

* Create a new folder.
* Copy inside **LinuxSourceCode.cpp** and **CMakeLists.txt** provided.

|  |
| --- |
| **Warning !!!**  Be careful to modify in **CMakeLists.txt** the path to your **OpenCV** installation  *ex: SET("OpenCV\_DIR" "/path/to/your/OpenCV/installation")* |

* Create inside the folder another folder entitled **build.**
* Open the **terminal** and navigate to **build** folder.
* Run:

*cmake ..*

*make*

* The application should build without any errors.

**4.Testing / using the application**

* Open **terminal** and navigate to **build** folder.
* You need to have in the same folder an input image with a barcode.
* Run:

*./* *EUCALLtag IMG\_NAME.jpg 1*

if you want to display an image with the Barcode identified, otherwise run:

*./EUCALLtag IMG\_NAME.jpg*

|  |
| --- |
| **Warning !!!**  The application can be moved to any other location and renamed |

|  |
| --- |
| **Warning !!!**  If the name of the input image contains spaces, you have to it inside commas:  *./EUCALLtag “IMG NAME.jpg” 1* |