Project Title

Image Processing with OpenCV, C++ and Ubuntu 16.04

Developed by: MSc. Shiva Agrawal

Place: Germany

Date: September 2018

Content

1. Introduction

1.1. Project description

The aim of this project is to collect and provide basic image processing functionalities of OpenCV library with C++. As we all know that computer vision is nowadays indispensable part of Robotics and autonomous driving. Every robot has camera installed on it. Camera is like an eye and so computer vision is like processor of the brain responsible to collect and understand the information collected by an eye.

Keeping this in mind, when I started working for OpenCV, I thought it would be a good idea, if I can collect most of the core functionalities which I am learning in form of well documented and tested code, can help me for the future projects as a reference and also can help other people.

In the project, these image processing functions are available and categorized. Each category of the functions is written as separate .cpp file. Inside each .cpp file, further each functionality is written as preprocessor #ifdef … #endif block which user can activate and deactivate as per requirement. Within each block of code, the respective function is well explained and tested using images.

For the development, I have used Eclipse CDT as IDE, OpenCV, C++ programming language and Ubuntu 16.04 (Linux OS).

1.2. Outline

- Chapter 1: Project description and outline of the project report
- Chapter 2: OpenCV and Eclipse CDT
- Chapter 3: Image Processing
- Chapter 4: Test and results
- Chapter 5: Conclusion
- Chapter 6: References

2. OpenCV and Eclipse CDT

2.1. OpenCV

[1] **"OpenCV (Open Source Computer Vision Library)** is an open source library for image processing and computer vision. It has C++, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV is designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform".

2.2. Eclipse CDT

[2] "The CDT Project provides a fully functional C and C++ Integrated Development Environment based on the Eclipse platform. Features include: support for project creation and managed build for various toolchains, standard make build, source navigation, various source knowledge tools, such as type hierarchy, call graph, include browser, macro definition browser, code editor with syntax highlighting, folding and hyperlink navigation, source code refactoring and code generation, visual debugging tools, including memory, registers, and disassembly viewers."

3. Image Processing

In the project, as mentioned in Introduction section, functions are categorized and hence each category of functions is in one .cpp file. These cpp files are available in src folder of the project repository. Also, the images used during the functions are also available in the src folder.

Further here each category is explained in short including which image processing functions are covered in it.

I have used Open Source Computer Vision for Beginners: Learn OpenCV using C++ in fastest possible way [3] and opencv.org tutorials to develop this project.

3.1. Basic Image Processing

This contains the very basic functions of the image processing as:

- Load Image
- Write image
- Create single channel image (ex. grayscale image)
- Create multichannel image (ex. RGB image)
- Finding number of channels in an image
- Split channels of an image
- Changing Color space of an image
- Binary images
- Extract Pixel values of an image
- Select Region of Interest (ROI) of an image

Filename: src/BasicImageProcessing.cpp

3.2. Draw Shapes

It contains following functions

- Draw Line on image
- Draw Circle on image
- Draw Ellipse on image
- Draw Rectangle on image
- Write Text on image

Filename: src/DrawShapes.cpp

3.3. Image Editing

It contains following functions

- Change Brightness of Image
- Change Contrast of Image
- Histogram Equalizer - Grayscale Image
- Histogram Equalizer - RGB Image
- Image Rotation
- Changing Image size

Filename: src/ImageEditing.cpp

4. Test and results

Since the project is collection of functions written as preprocessor directives blocks inside main() function, the output can be obtained just by running the corresponding file and block of code. Hence there is no separate test code written. Also, results are self-explanatory and so there is no separate test or results folder created in the repository.

5. Conclusion

The image processing functions are documented and tested for future reference.

6. References

[1] "CV," openCV, [Online]. Available: https://opencv.org/.

[2] "Eclipse CDT," Eclipse Foundation Inc., [Online]. Available: https://www.eclipse.org/cdt/.

[3] N. Faruqui, Open Source Computer Vision for Beginners: Learn OpenCV using C++ in fastest possible way, Kindle Direct Pblishing.