**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HOC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**THỊ GIÁC MÁY TÍNH**

****

### Báo cáo BTVN-02

Phát hiện biên cạnh

**GVHD: Võ Hoài Việt**

Contents

[***Báo cáo BTVN-02*** 1](#_Toc4885589)

[**A Thành viên nhóm:** 3](#_Toc4885590)

[**B Mức độ hoàn thành:** 3](#_Toc4885591)

[**C Báo cáo:** 4](#_Toc4885592)

[I. So sánh kết quả thực hiện với OpenCV: 4](#_Toc4885593)

[1. Hình 1: 4](#_Toc4885594)

[2. Hình 2: 8](#_Toc4885595)

[3. Hình 3: 11](#_Toc4885596)

[4. Hình 4: 14](#_Toc4885597)

[5. Hình 5: 17](#_Toc4885598)

[II. Hướng dẫn sử dụng chương trình: 21](#_Toc4885599)

[**D** **Tham khảo:** 21](#_Toc4885600)

[References 22](#_Toc4885601)

# **A Thành viên nhóm:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | MSSV | Họ tên | SĐT | Email |
| 1 | 1612174 | Phùng Tiến Hào | 0933642694 | [tienhaophung@gmail.com](mailto:tienhaophung@gmail.com) |
| 2 | 1612269 | Võ Quốc Huy | 01258378481 | [voquochuy304@gmail.com](mailto:voquochuy304@gmail.com) |

# **B Mức độ hoàn thành:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên kết quả | Tên hàm đề nghị | Ghi chú | Mức độ hoàn thành (%) |
| 1 | Phát hiện biên cạnh sử dụng Sobel | int detectBySobel(Mat src, Mat dst,…); | Cho phép hiện thị ảnh gradient theo hướng x và y trong quá trình thực hiện thuật toán | 100 |
| 2 | Phát hiện biên cạnh sử dụng Prewitt | int detectByPrewitt(Mat src, Mat dst,…); | Cho phép hiện thị ảnh gradient theo hướng x và y trong quá trình thực hiện thuật toán | 100 |
| 3 | Phát hiện biên cạnh sử dụng Laplace | int detectByLaplace(Mat src, Mat des, int …); |  | 100 |
| 4 | Phát hiện biên cạnh sử dụng Cany | int detectByCany(Mat sourceImage, Mat destinationImage); | Chọn 5 ảnh bất kỳ. So sánh với thuật toán được cung cấp bởi OpenCV. Giải thích các kết quả. | 100 |
| 5 | Chọn 5 ảnh bất kỳ. Thực hiện các thuật toán trên, nhận xét và so sánh các kết quả thực hiện được. |  |  | 100 |
| Tổng cộng | | | | 100 |

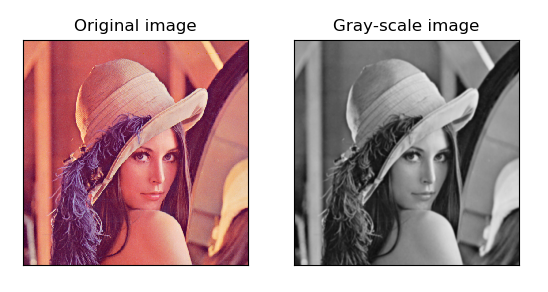
# **C Báo cáo:**

**Một vài lưu ý:**

* Ngôn ngữ sử dụng: Python
* Các thư viện sử dụng như: Numpy, openCV, matplotlip và skimage.
* Định nghĩa ký hiệu theo tài liệu nước ngoài:
  + X: Vertical axe
  + Y: Horizontal axe
  + XY: Magnitude của cả 2 hướng trên

## I. So sánh kết quả thực hiện với OpenCV:

### 1. Hình 1:



**Sobel:**



Figure Handcraft - Sobel

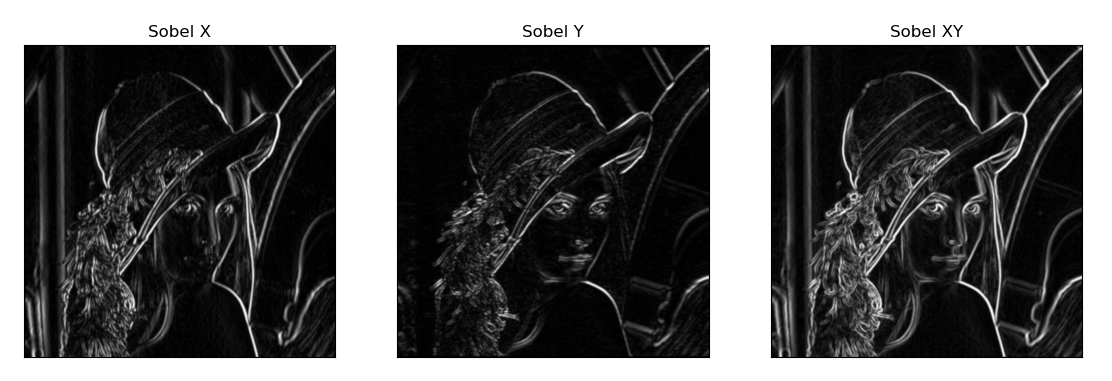


Figure OpenCV - Sobel

**Prewitt:**



Figure Handcraft - Prewitt



Figure OpenCV - Prewitt

**Laplacian:**

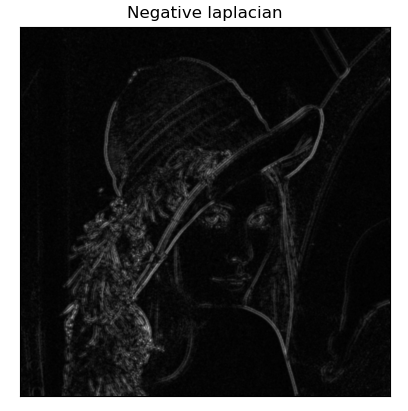


Figure Handcraft & OpenCV - Laplacian

**Canny:**

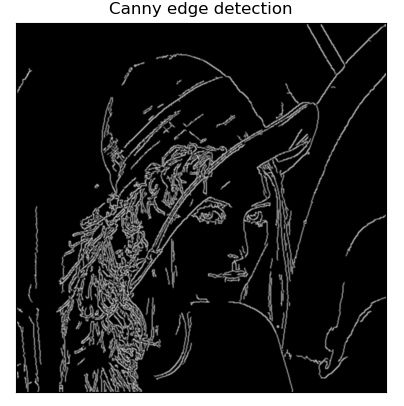
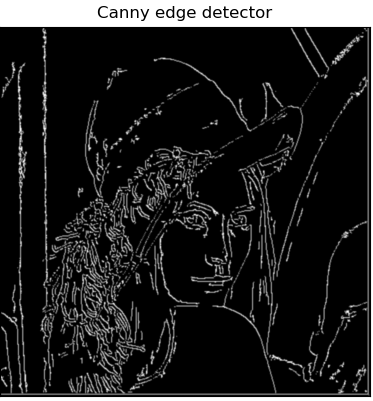
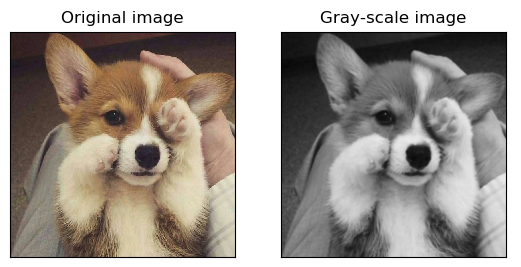


Figure Handcraft & OpenCV - Canny

|  |  |  |  |
| --- | --- | --- | --- |
| Edge detectors | Thời gian | | Nhận xét |
| Handcraft | OpenCV |
| Sobel | 5.34 (s) | 0.05 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Prewitt | 5.2 (s) | 0.05 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Laplacian | 8.47 (s) | 0.08 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Canny | 11.78 (s) | 0.06 (s) | Phát hiện được biên cạnh rất chi tiết, không thua kém nhiều của OpenCV. Nhưng thời gian xử lý lại đáng kể. |

### 2. Hình 2:



**Sobel:**



Figure Handcraft - Sobel



Figure OpenCV - Sobel

**Prewitt:**



Figure Handcraft - Prewitt

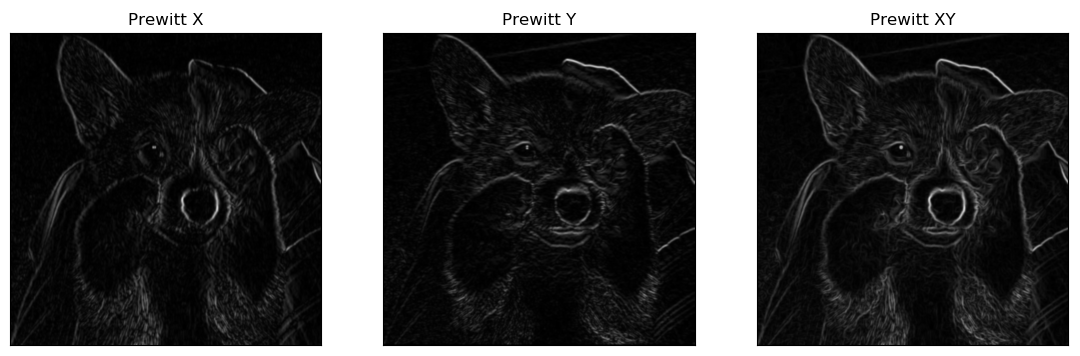


Figure OpenCV - Prewitt

**Laplacian:**

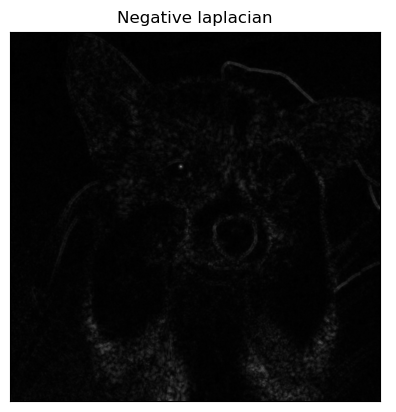
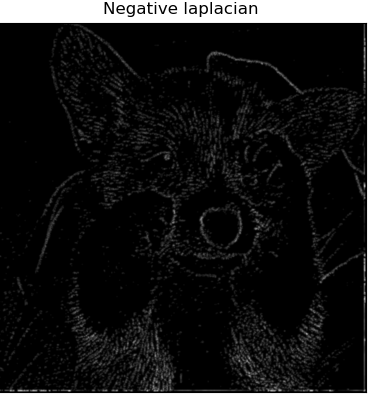


Figure Handcraft & OpenCV - Laplacian

**Canny:**

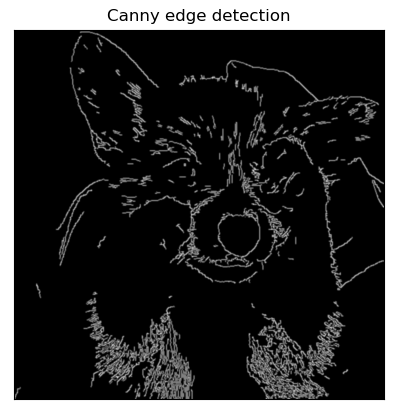
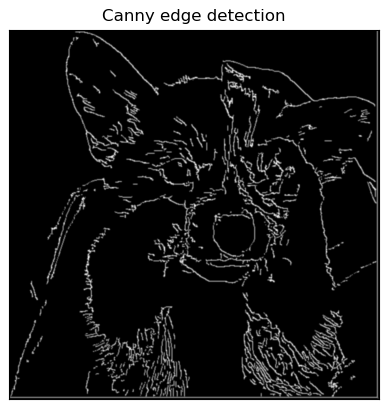


Figure Handcraft & OpenCV - Canny

|  |  |  |  |
| --- | --- | --- | --- |
| Edge detectors | Thời gian | | Nhận xét |
| Handcraft | OpenCV |
| Sobel | 7.05 (s) | 0.22 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Prewitt | 7.43 (s) | 0.18 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Laplacian | 9.57 (s) | 0.15 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Canny | 17.57 (s) | 0.16 (s) | Phát hiện được biên cạnh rất chi tiết.  Thời gian xử lý lại đáng kể. |

### 3. Hình 3:



**Sobel:**



Figure Handcraft - Sobel



Figure OpenCV - Sobel

**Prewitt:**



Figure Handcraft - Prewitt



Figure OpenCV - Prewitt

**Laplacian:**

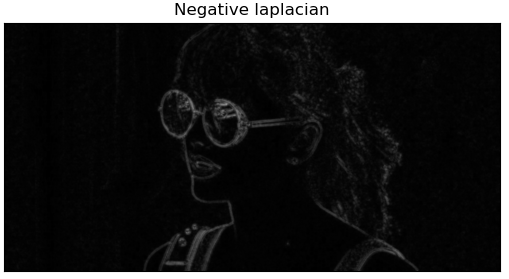
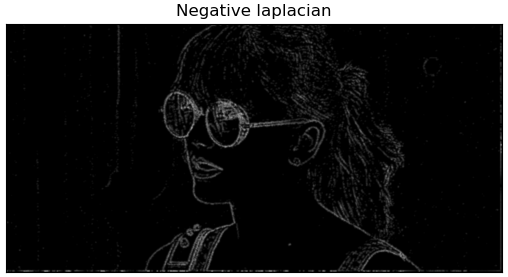


Figure Handcraft & OpenCV - Laplacian

**Canny:**

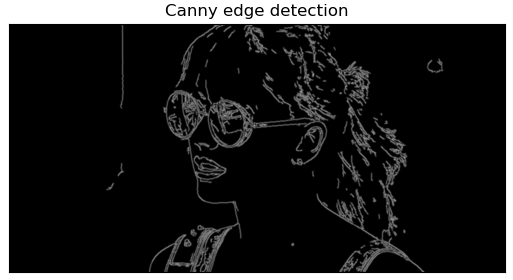
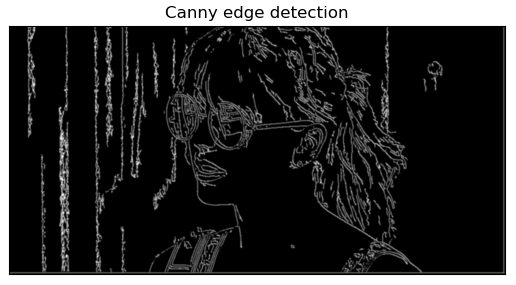
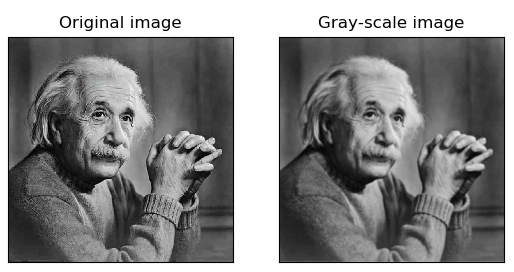


Figure Handcraft & OpenCV - Canny

|  |  |  |  |
| --- | --- | --- | --- |
| Edge detectors | Thời gian | | Nhận xét |
| Handcraft | OpenCV |
| Sobel | 12.79 (s) | 0.20 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Prewitt | 11.57 (s) | 0.19 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Laplacian | 14 (s) | 0.15 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Canny | 23.81 (s) | 0.15 (s) | Phát hiện được biên cạnh rất chi tiết, nhất là phần tóc lấy được gần trọn bộ tóc, tốt hơn của OpenCV.  Nhưng có đính nhiều chi background hơn của OpenCV.  Thời gian xử lý lại đáng kể. |

### 4. Hình 4:



**Sobel:**



Figure Handcraft - Sobel

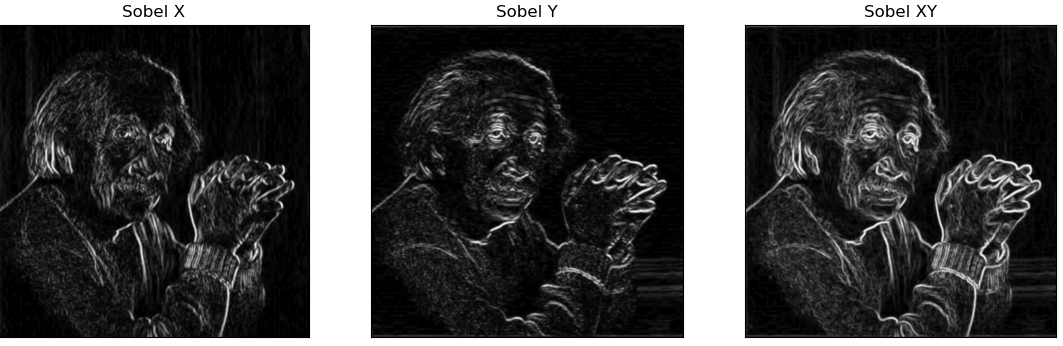


Figure OpenCV - Sobel

**Prewitt:**

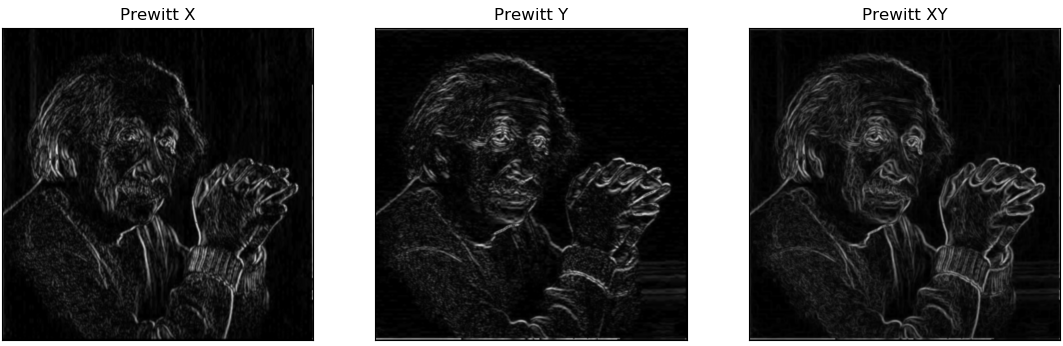


Figure Handcraft - Prewitt

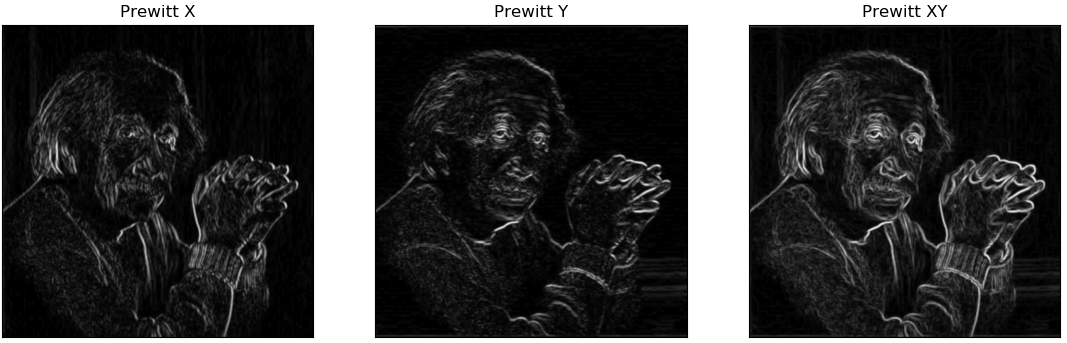


Figure OpenCV - Prewitt

**Laplacian:**



Figure Handcraft & OpenCV - Laplacian

**Canny:**

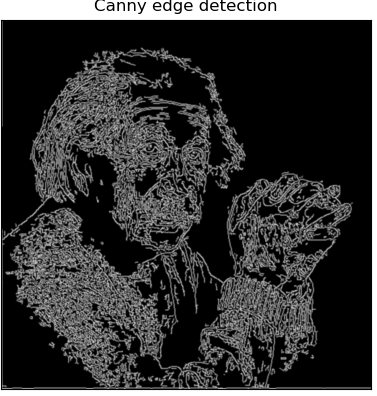
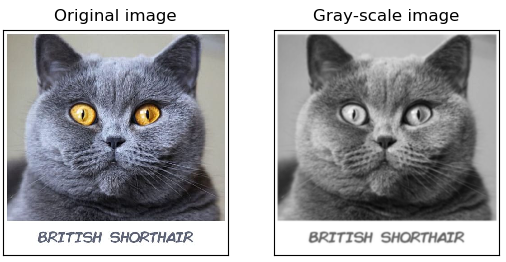


Figure Handcraft & OpenCV - Canny

|  |  |  |  |
| --- | --- | --- | --- |
| Edge detectors | Thời gian | | Nhận xét |
| Handcraft | OpenCV |
| Sobel | 6.99 (s) | 0.18 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Prewitt | 6.24 (s) | 0.18 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Laplacian | 10 (s) | 0.16 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Canny | 18.49 (s) | 0.14 (s) | Phát hiện được biên cạnh rất chi tiết, nhất là nếp nhăn.  Nhưng có đính nhiều background hơn của OpenCV.  Thời gian xử lý lại đáng kể. |

### 5. Hình 5:



**Sobel:**



Figure Handcraft - Sobel



Figure OpenCV - Sobel

**Prewitt:**



Figure Handcraft - Prewitt



Figure OpenCV - Prewitt

**Laplacian:**



Figure Handcraft & OpenCV - Laplacian

**Canny:**

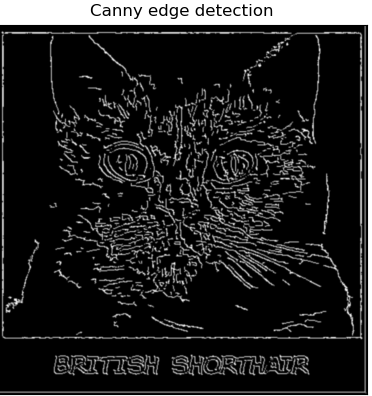


Figure Handcraft & OpenCV - Canny

|  |  |  |  |
| --- | --- | --- | --- |
| Edge detectors | Thời gian | | Nhận xét |
| Handcraft | OpenCV |
| Sobel | 5.48 (s) | 0.17 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Prewitt | 5.89 (s) | 0.18 (s) | Phát hiện được biên cạnh tốt  Thời gian thực hiện lâu hơn. |
| Laplacian | 11.78 (s) | 0.15 (s) | Phát hiện được biên cạnh tốt.  Thời gian thực hiện lâu hơn. |
| Canny | 13.77 (s) | 0.15 (s) | Phát hiện được biên cạnh chi tiết.  Nhưng phần lông mèo ngay mặt thì OpenCV nhỉnh hơn.  Thời gian xử lý lại đáng kể. |

**Tóm lại:**

* Các phương pháp phát hiện biên cạnh: Sobel, Prewitt, Laplacian và Canny đều sử dụng Gaussian blur để làm trơn ảnh. (kernel size: (5x5))
* Thuật toán Sobel, Prewitt đạt yêu cầu và phát hiện được biên cạnh khá tốt.
* Thuật toán Laplacian tách được biên cạnh khá tốt.
* Phương pháp Canny phát hiện biên cạnh ở mức khá chi tiết và đạt được kết quả rất tốt và gần với OpenCV.

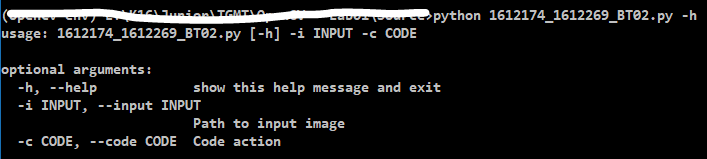
## II. Hướng dẫn sử dụng chương trình:

* Command line:

python <tenchuongtrinh> -i <duongdandentaptinanh> -c <malenh>

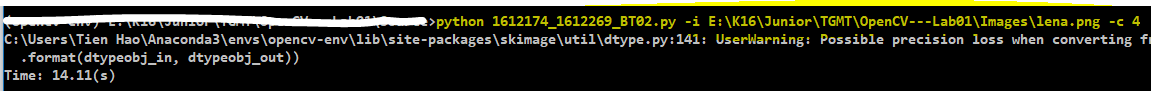
(-i, -c: Argument Parser của Python)

- Để trợ giúp: gõ “python 1612174\_1612269\_BT02.py -h”



* Các mã lệnh:
  + 1: Sobel
  + 2: Prewitt
  + 3: Laplacian
  + 4: Canny
* Ví dụ: Dùng Canny để phát hiện biên cạnh cho Lena

“python 1612174\_1612269\_BT02.py -i E:\K16\Junior\TGMT\OpenCV---Lab01\Images\lena.png -c 4”



Lưu ý: Chương trình có trả về thời gian xử lý cho 1 hoạt động.

# **D** **Tham khảo:**

# References

|  |  |
| --- | --- |
| [1] | "OpenCV-Python tutroals," [Online]. Available: https://opencv-python-tutroals.readthedocs.io/en/latest/py\_tutorials/py\_gui/py\_image\_display/py\_image\_display.html#display-image. |
| [2] | "HIPR2," [Online]. Available: http://homepages.inf.ed.ac.uk/rbf/HIPR2/hipr\_top.htm. |
| [3] | A. Rosebrock, "pyimagesearch," [Online]. Available: https://www.pyimagesearch.com/2016/07/25/convolutions-with-opencv-and-python/. |