Python and shell Programming Lab Project Report

On

**“CONVERT IMAGE TO AN PENCIL SKETCH ”**

Submitted in Partial Fulfilment of the Requirements

For the award of the Degree of

**Bachelor of Technology**

**In**

**Electronics & Computer Engineering (ECM)**

By

**20311A1985 P.SAI SIDHARTHA**

**20311A1986 B.SAI MANOJ**

**20311A1987 P.TAPASWINI**

#### Under the Supervision of

**DR. ATILLI VENKATA RAMANA**

**Associate Professor**

****

**Department of Electronics & Computer Engineering**

**Sreenidhi Institute of Science & Technology (Autonomous)**

**2021-2022**

**DEPARTMENT OF ELECTRONICS & COMPUTER ENGINEERING**

**SREENIDHI INSTITUTE OF SCIENCE & TECHNOLOGY (AUTONOMOUS)**

****

**CERTIFICATE**

This is to certify that the PYTHON AND SHELL PROGRAMMING Lab Project work entitled “**CONVERT IMAGE TO AN PENCIL SKETCH”**, submitted by **P.SAI SIDHARTHA, B.SAI MANOJ, P.TAPASWINI (20311A1985, 20311A1986, 20311A1987)** towards partial fulfilment for the award of Bachelor of Technology Degree in in **Electronics and Computer Engineering** from Sreenidhi Institute of Science & Technology, Ghatkesar, Hyderabad, is a record of bonafide work done by them during the academic year 2021-2022 under our guidance and evaluation.

**Mr. DAYAKAR KONDAMUDI Dr. A. VENKATA RAMANA**

**Lab Project Coordinator-2 Lab Project Coordinator-1**

**Dr. D. Mohan**

**HOD, ECM Department**

**DECLARATION**

This is to certify that the PSP Lab Project Report titled **“CONVERT IMAGE TO AN PENCIL SKETCH”**, is a record work done by me in the department of Electronics and Computer Engineering (ECM), Sreenidhi Institute of Science & Technology, Ghatkesar, Hyderabad.

The report is based on the project work done entirely by us and not copied from any other source.

**Name of student (Roll No.)**

P.SAI SIDHARTHA 20311A1985

B.SAI MANOJ 20311A1986

P.TAPASWINI 20311A1987

**ACKNOWLEDGMENT**

I wish to express immense gratitude to my supervisor, **Dr. A Venkata Ramana** Associate Professor and **Mr. Dayakar Kondamudi** assistant professor, **ECM Department,** for his guidance and useful suggestions, which helped us in completing our Project Work in time. His valuable suggestions and comments towards this python and shell programming project have been very much helpful in tackling various obstacles and accomplishing the major tasks.

I take immense pleasure in thanking **HOD Dr. D. MOHAN, Principal Dr. Ch. Shiva Reddy** and our **Executive Director C.V TOMY** and all faculty members of ECM department for having permitted me to carry out this Lab Project work.

**Name of student (Roll No.)**

P.SAI SIDHARTHA 20311A1985

B.SAI MANOJ 20311A1986

P.TAPASWINI 20311A1987

**ABSTRACT**

In Python, an image is just a two-dimensional array of integers. So, one can do a couple of matrix manipulations using various python modules in order to get some very interesting effects. In order to convert the normal image to a sketch, we will change its original RGB values and assign its RGB values similar to grey, in this way a sketch of the input image will be generated. In today’s era, we are surrounded by different types of photo manipulating filters in our mobile phones, apps…etc. But do you know how they do these images manipulations….? In the backend, they are using computer vision techniques. Computer vision has a wide variety of applications not only to reduce the human effort but also used for entertainment apps. Many photo editing apps like FaceApp, Instagram filters…etc are using computer vision techniques. In this article, we will try to convert a normal photo into a pencil sketch using computer vision in a python programming language. In this article, we will show how to convert an image into its corresponding pencil sketch in a few steps.

**Keywords**: Matplotlib, Open CV, RGB colors scheme, BGR color scheme.

**Team Members:**

P.Sai Sidhartha (20311A1985)

B.Sai Manoj (20311A1986)

P.Tapaswini (20311A1987)

**INDEX**

|  |  |  |
| --- | --- | --- |
| S.NO | CONTENT | PAGE NUMBER |
| 1 | Introduction | 7 |
| 2 | Objective | 8 |
| 3 | Software Requirements and Description | 9 |
| 4 | Implementation | 10-13 |
| 5 | Working | 14-16 |
| 6 | Result And Output | 17-19 |
| 7 | Conclusion | 20 |
| 8 | References | 21 |

**CHAPTER-1**

**INTRODUCTION**

Python can be used on a server to create web applications. Python can be used alongside software to create workflows. Python can connect to database systems. It can also read and modify file. Python can be used to handle big data and perform complex mathematics. Python can be used for rapid prototyping, or for production-ready software development. Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses. Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose. Python is a multipurpose language and one can do literally anything with it.

**In today’s era, we are surrounded by different types of photo manipulating [filters](https://analyticsindiamag.com/adobes-new-ai-powered-tools-make-job-photo-video-editor-redundant/) in our mobile phones, apps…etc. But do you know how they do these images manipulations…..? In the backend, they are using [computer vision](https://analyticsindiamag.com/how-do-data-scientists-create-high-quality-training-datasets-for-computer-vision/) techniques. Computer vision has a wide variety of applications not only to reduce the human effort but also used for entertainment apps. Many photo editing apps like [FaceApp](https://analyticsindiamag.com/the-ai-behind-faceapp/), Instagram filters…etc are using computer vision techniques.**

**In this article, we will try to convert a normal photo into a pencil sketch using computer vision in a [python](https://analyticsindiamag.com/comparing-python-libraries-pylearn2-vs-scikit-learn/) programming language. In this article, we will show how to convert an image into its corresponding pencil sketch in a few steps.**

**CHAPTER 2**

**OBJECTIVE**

We are using the bitwise\_not function which is used to make brighter regions lighter and vice versa so that we can find the [edges](https://analyticsindiamag.com/this-is-how-i-created-an-object-detection-model-in-less-than-5-minutes/)to create a pencil sketch. We have used the gaussian blur technique with 21 x 21 pixel and the default sigma values filter on the image to smoothen our image. By increasing the filter size, we can create thin lines for our sketch and it is used to reduce the noise in the image. By using this function, it is dividing the greyscale value of the image by the inverse of blurred image value which highlights the boldest edges. This technique is used by traditional photographers to print photos from the reel.

**CHAPTER 3**

**SOFTWARE REQUIREMENTS AND DESCRIPTION**

1. **PYTHON INTREPRETER:**

The Python interpreter is **a virtual machine**, meaning that it is software that emulates a physical computer. ... The Python interpreter is a bytecode interpreter: its input is instruction sets called bytecode. When you write Python, the lexer, parser, and compiler generate code objects for the interpreter to operate on.

Python is interactive and a compiler reads the entire program on the initial standard input. When it reads the program, it generates the object code instantly. The process is fast and effective. Whereas the interpreter does the opposite. It will read each line and doesn’t create an immediate object code like the compiler does. Compilation occurs instantly, instead. Compilation with the compiler occurs before the program is executed. The **interpreter** uses the source language and converts the program through each line of data. Then, the computer will be able to complete any command given through the program. **Python interpreter** takes an interactive command and executes it. All lines of source code are completed one line at a time.

1. **NOTEPAD OR MS WORD:**

Notepad isn't just for taking notes and opening readme files. You can also use Notepad to make basic computer programs. This can be done by creating batch files that run scripts in the Windows Command Prompt. We can also write lines of code inside Notepad.

By using Notepad and the Python interpreter, a programmer can write Python programs and execute them, or create "batch" files that can execute multiple programs, including Python scripts. A programmer can use any text editor to write a Python script. For the Windows operating system, the Notepad program exists as a plain text editor that programmers can use to write any sort of program

**CHAPTER 4**

**IMPLEMENTATION**

**INPUT:**

**import cv2**

**image = cv2.imread('Image.jpg')**

**# loads an image from the specified file**

**# convert an image from one color space to another**

**grey\_img = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)**

**invert = cv2.bitwise\_not(grey\_img)**

**# helps in masking of the image**

**# sharp edges in images are smoothed while minimizing too much blurring**

**blur = cv2.GaussianBlur(invert, (21, 21), 0)**

**invertedblur = cv2.bitwise\_not(blur)**

**sketch = cv2.divide(grey\_img, invertedblur, scale=256.0)**

**cv2.imwrite("sketch.png", sketch)**

**# converted image is saved as mentioned name**

**CHAPTER-5**

**WORKING**

## Step 1: Convert to Grey Image

Using cvtColor function of OpenCV.

grey\_img=cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

## Step 2: Invert Image

Inverting the image can be done by either of the following methods

invert\_img=cv2.bitwise\_not(grey\_img)  
#invert\_img=255-grey\_img

## Step 3: Blur image

Apply *Gaussian blur* to the image. The second argument to the function is the kernel size, if should be a pair of odd numbers.  
Larger the kernel size, more blurred the image will be and it will lose its subtle features.  
For creating sketch, we require only the prominent features (contrasting edges) from the image.  
For small images, *kernel size* of (3,3), (5,5) etc. will be sufficient, whereas for larger images, small kernel size do not create any impact.  
Appropriate kernel size can be selected by trial and error method.

blur\_img=cv2.GaussianBlur(invert\_img, (111,111),0)

## Step 4: Invert Blurred Image

Repeat step 2

invblur\_img=cv2.bitwise\_not(blur\_img)  
#invblur\_img=255-blur\_img

## Step 5: Sketch

The sketch can be obtained by performing bit-wise division between the grayscale image and the inverted-blurred image.

sketch\_img=cv2.divide(grey\_img,invblur\_img, scale=256.0)

Explaining bit-wise division in detail is beyond the scope of this article.  
You can access [more information on image arithmetic here](http://refer this page for more information on image arithmetic./" \t "https://towardsdatascience.com/_blank) .

## Step 6: Save Sketch

cv2.imwrite(‘sketch.png’, sketch\_img)

## Step 7: Display sketch

cv2.imshow(‘sketch image’,sketch\_img)  
cv2.waitKey(0)  
cv2.destroyAllWindows()

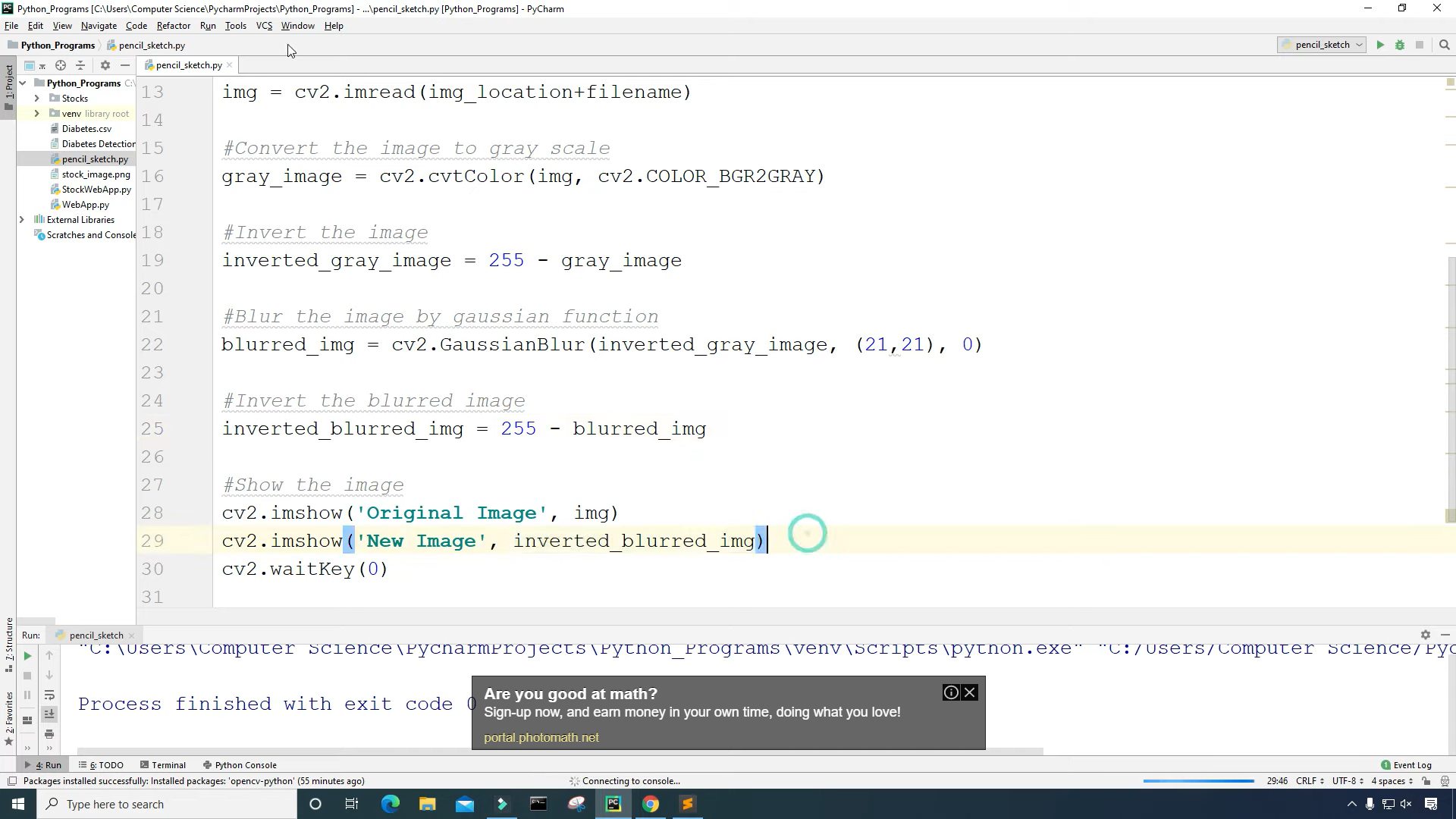
## Original Image vs Sketch

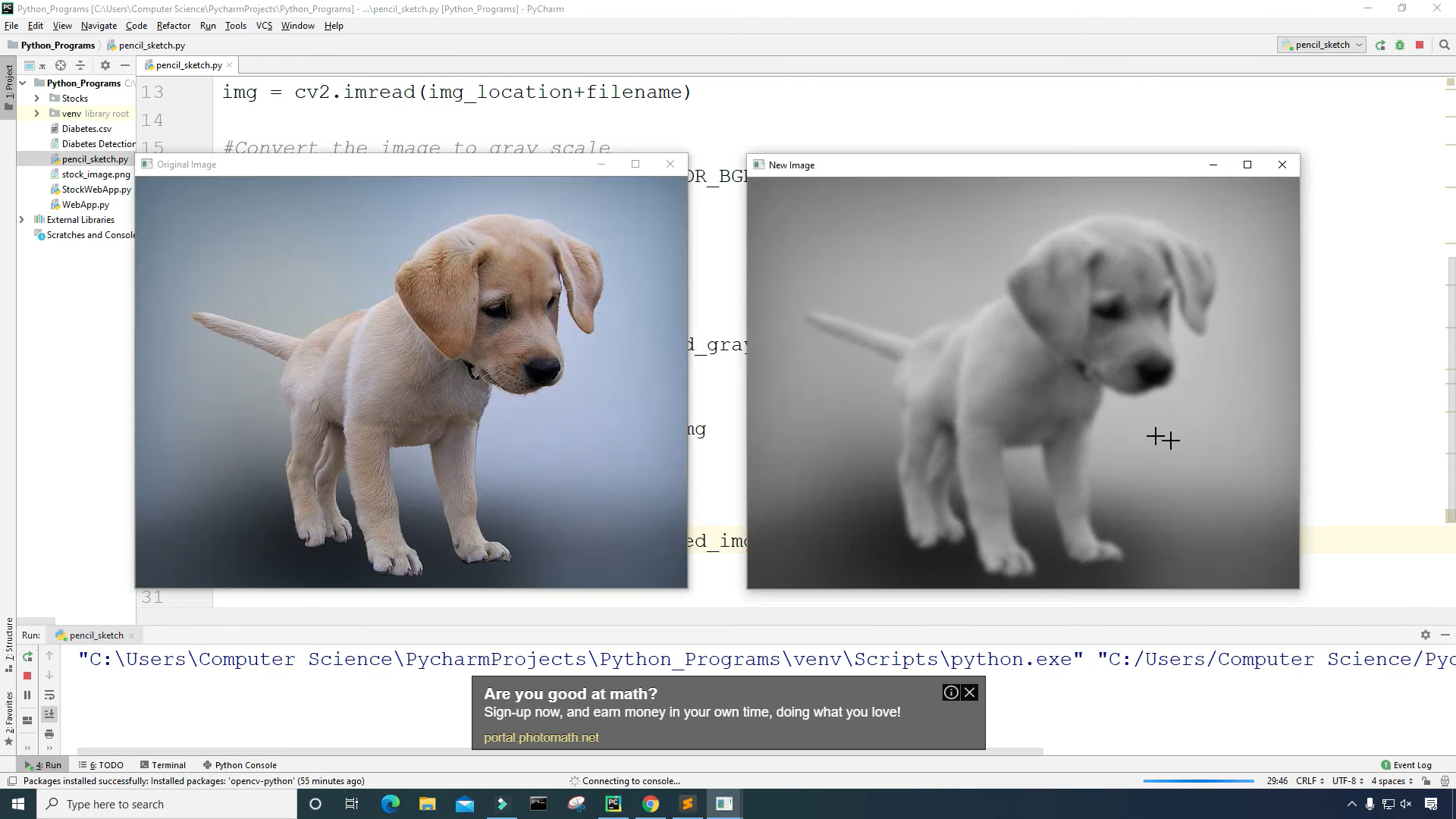
We can display the original image and sketch side by side for comparison.

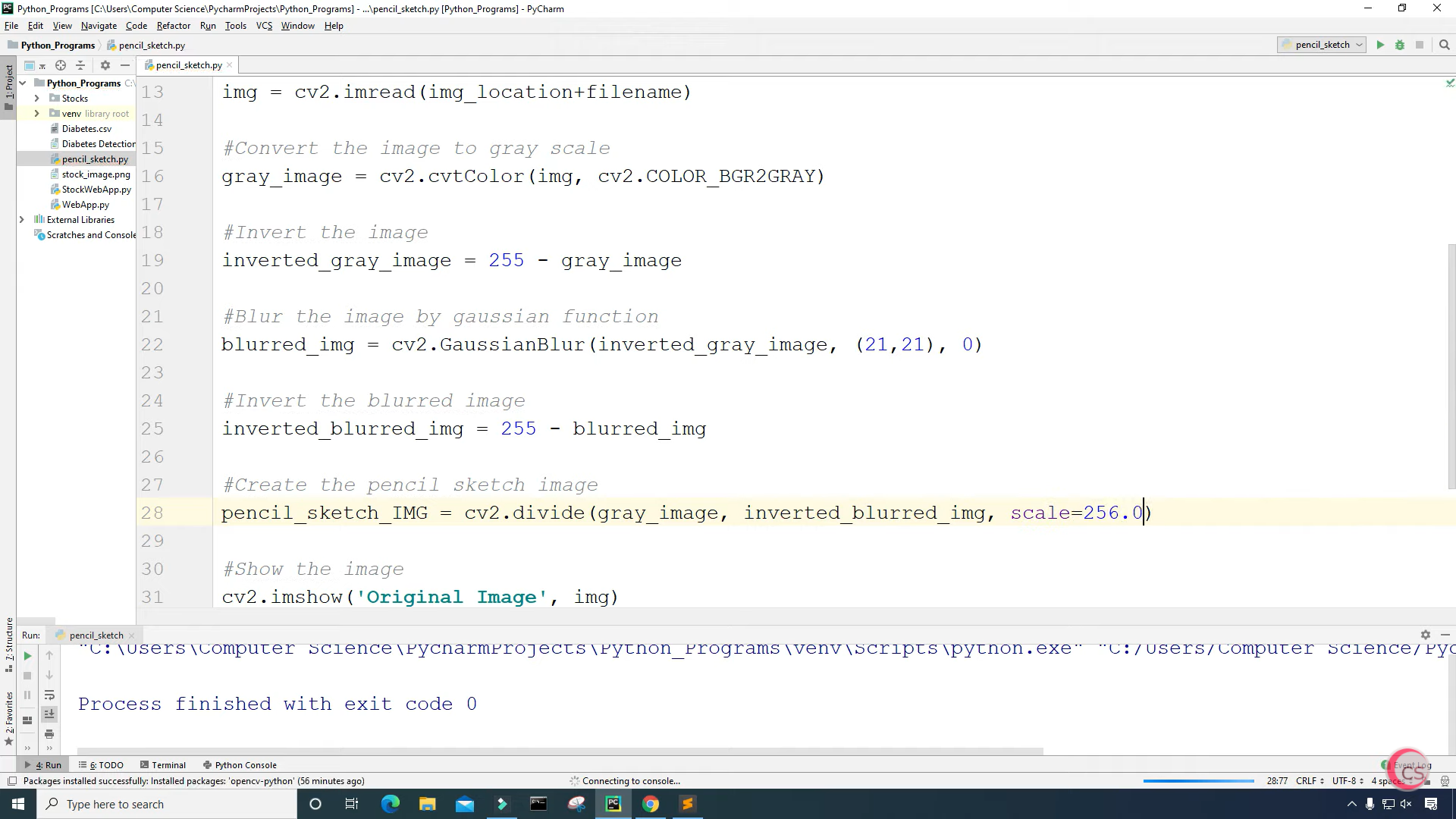
plt.figure(figsize=(14,8))plt.subplot(1,2,1)  
plt.title('Original image', size=18)  
plt.imshow(RGB\_img)  
plt.axis('off')plt.subplot(1,2,2)  
plt.title('Sketch', size=18)  
rgb\_sketch=cv2.cvtColor(sketch\_img, cv2.COLOR\_BGR2RGB)  
plt.imshow(rgb\_sketch)  
plt.axis('off')  
plt.show()

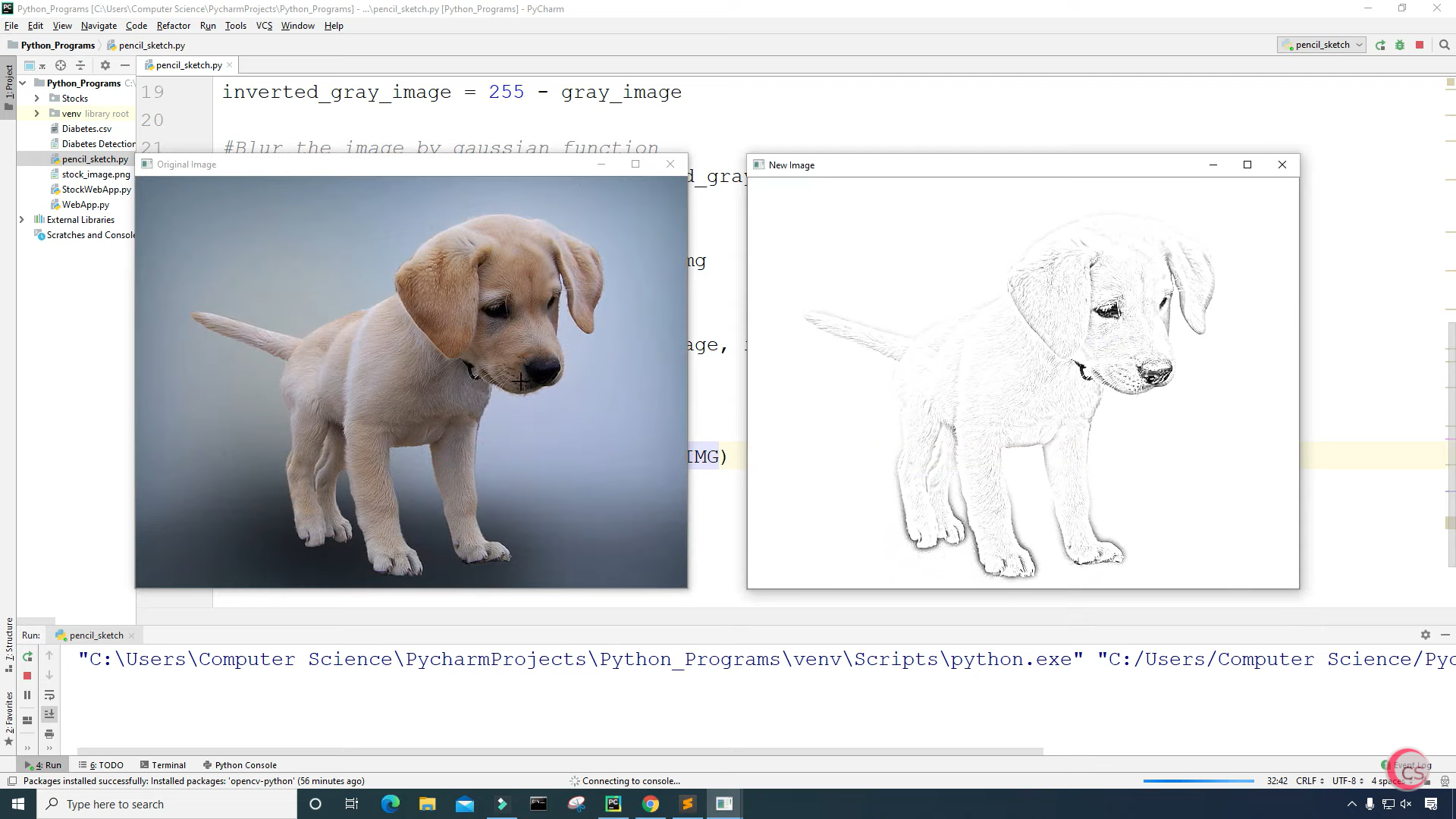
**CHAPTER-6**

**RESULTS AND OUTPUT**







****

**CHAPTER-7**

**CONCLUSION**

Understanding the syntax of Python is great and all, and Python by itself is indeed a great language, but the fundamentals of Python aren't why Python is a successful language. Python is a fully-fledged programming language.

With these steps, we have successfully created a Convet Image To An Pencil Sketch project using python. We used popular tkinter library to rendering graphics in our display window and we also learned about pyperclip and random library.

**CHAPTER-8**

**REFERENCES**

**https://learnopencv.com/category/opencv-tutorials/**

**https://www.pyimagesearch.com/category/opencv/**