IMAGE EDITOR

Github copy

A COMPUTER APPLICATION

BY:

ROHAN GAUTAM

Certified alien

CONTENTS

INTRODUCTION

"Digital image processing is the use of computer algorithms to perform image processing on digital images. As a subcategory or field of digital signal processing, digital image processing has many advantages over analog image processing. It allows a much wider range of algorithms to be applied to the input data and can avoid problems such as the build-up of noise and signal distortion during processing. Since images are defined over two dimensions (perhaps more) digital image processing may be modeled in the form of multidimensional systems."

Many of the techniques of digital image processing, or digital picture processing as it often was called, were developed in the 1960s at the Jet Propulsion Laboratory, Massachusetts Institute of Technology, Bell Laboratories, University of Maryland, and a few other research facilities, with application to satellite imagery, wire-photo standards conversion, medical imaging, videophone, character recognition, and photograph enhancement. The cost of processing was fairly high, however, with the computing equipment of that era. That changed in the 1970s, when digital image processing proliferated as cheaper computers and dedicated hardware became available. Images then could be processed in real time, for some dedicated problems such as television standards conversion. As general-purpose computers became faster, they started to take over the role of dedicated hardware for all but the most specialized and computer-intensive operations. With the fast computers and signal processors available in the 2000s, digital image processing has become the most common form of image processing and generally, is used because it is not only the most versatile method, but also the cheapest.

Digital image processing technology for medical applications was inducted into the Space Foundation Space Technology Hall of Fame in 1994.

In 2002 Raanan Fattel, introduced Gradient domain image processing, a new way to process images in which the differences between pixels are manipulated rather than the pixel values themselves.

Tasks:

Digital image processing allows the use of much more complex algorithms, and hence, can offer both more sophisticated performance at simple tasks, and the implementation of methods which would be impossible by analog means.

In particular, digital image processing is the only practical technology for:

- Classification
- Feature extraction
- Multi-scale signal analysis
- Pattern recognition
- Projection

Some techniques which are used in digital image processing include:

- Image editing
- Image restoration
- Independent component analysis
- Filtering
- Neural networks

- Pixelation
- Principal components analysis

Filtering:

Digital filters are used to blur and sharpen digital images. Filtering can be performed in the spatial domain by convolution with specifically designed kernels (filter array), or in the frequency (Fourier) domain by masking specific frequency regions. In this project, we have used the Blur feature from PIL.ImageFilter module to apply the blur effect on a group of images.

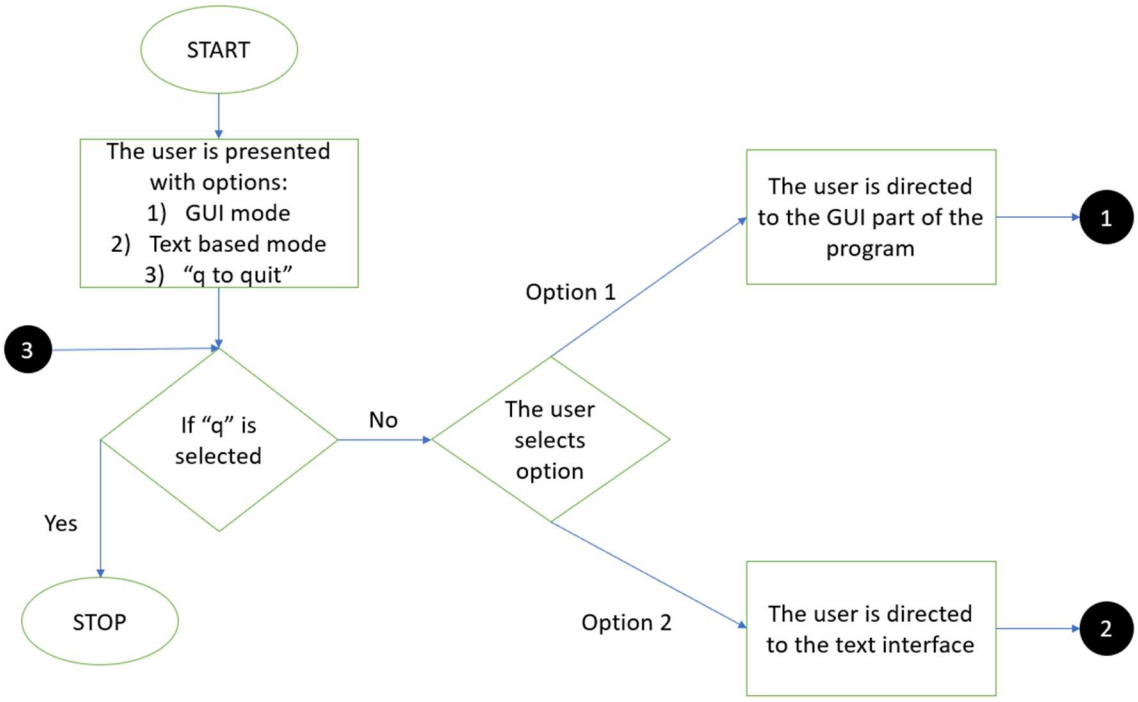
Application of digital image processing in our project:

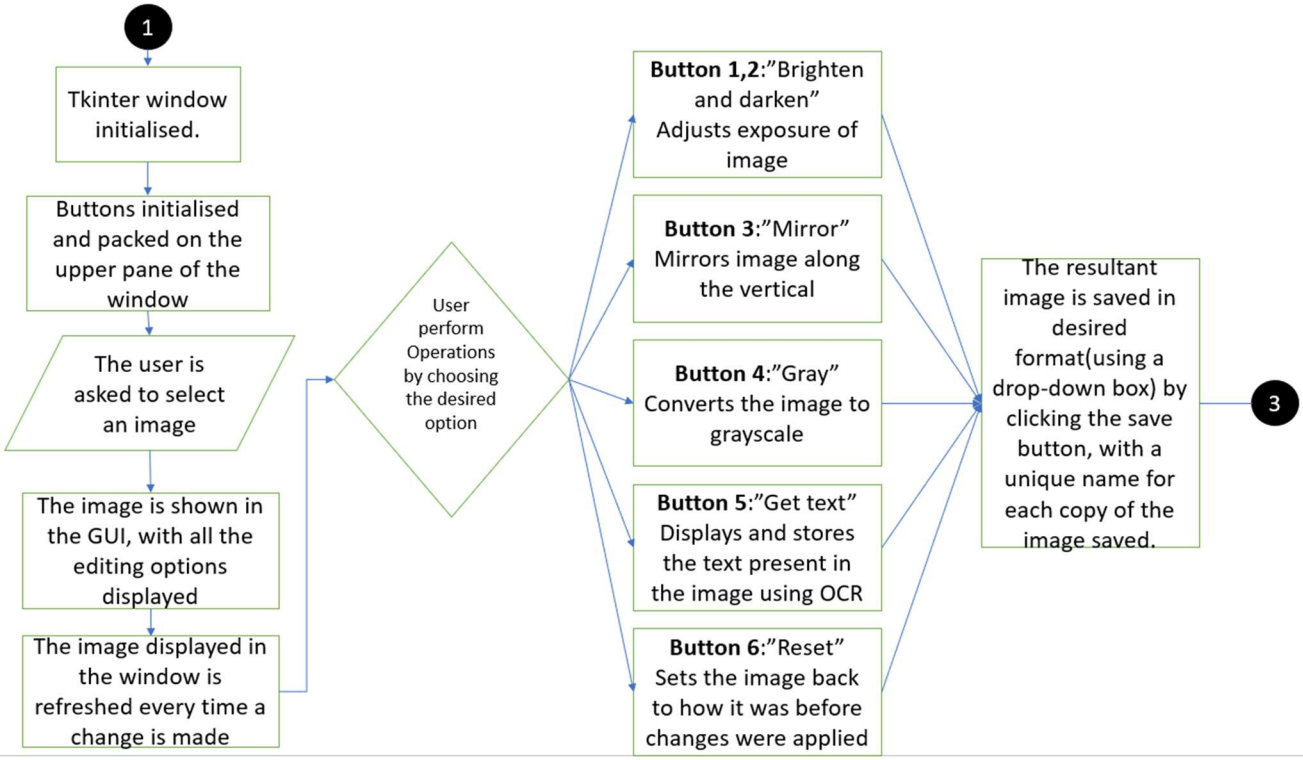
This project has made ample use of digital image processing. The digital image processing carried out in this project mostly revolves around using the python module PIL or the “Python Imaging Library” for image manipulation.

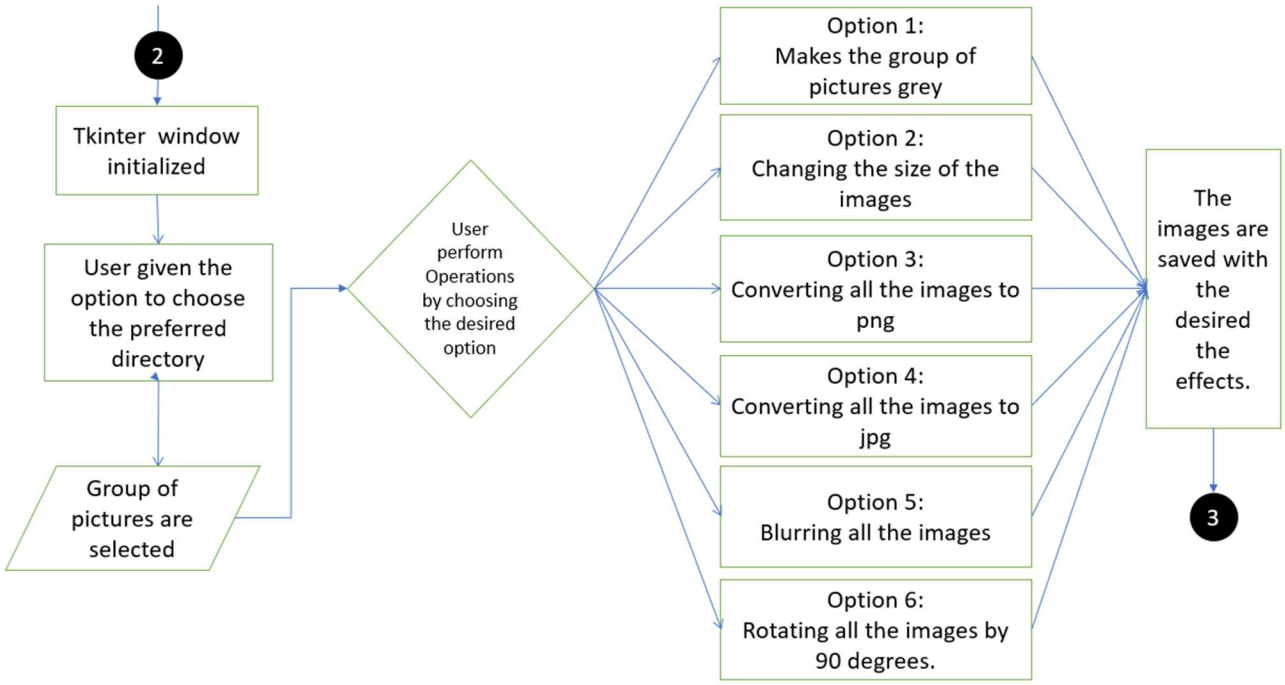
The first part of our project in the graphical interface for single image editing wherein the user can edit a single image and view the changes in real time. A few editing options include brightening, darkening, making the image gray, mirroring the image, and getting the constituent text from an image, if any.

The second text-based part includes operations like type conversion, making the image gray, rotating the image by 90 degrees, blurring the image, and resizing the image, all of which are applied to all the images of a folder.

PROGRAM FLOW






2
Tkinter window initialized
User given the option to choose the preferred directory
Group of pictures are selected
User perform Operations by choosing the desired option
Option 1: Makes the group of pictures grey
Option 2: Changing the size of the images
Option 3: Converting all the images to png
Option 4: Converting all the images to jpg
Option 5: Blurring all the images
Option 6: Rotating all the images by 90 degrees.
The images are saved with the desired the effects.
3

HOW TO SETUP

Python 2 is the version of python used in this project.

1) **Built in**: Tkinter, re, itertools, os
2) **Python imaging library**: Also known as PIL, this is a image manipulation library in python.
    a. To install, navigate to C:\Python27\Scripts and open command prompt in this location if pip is not a part of your PATH. Type the following:
    `pip install Pillow`
    This installs pillow, which is a more modern version/wrapper over the age-old PIL.
3) **Pytesseract**: This is a widely used optical character recognition (OCR) module for python.
    a. To install, navigate to C:\Python27\Scripts and open command prompt in this location if pip is not a part of your PATH. Type the following:
    `pip install pytesseract`

    There is a “Tesseract-OCR” database that you will have to place in "C:\Program Files (x86)" in your computer, which is included in this repository. After this step, you have to either set the path of “Tesseract-OCR” as an environment variable, or add the following line in your program:

    pytesseract.pytesseract.tesseract_cmd = 'C:/Program Files (x86)/Tesseract-OCR/tesseract'

And you’re ready to get this project up and running!

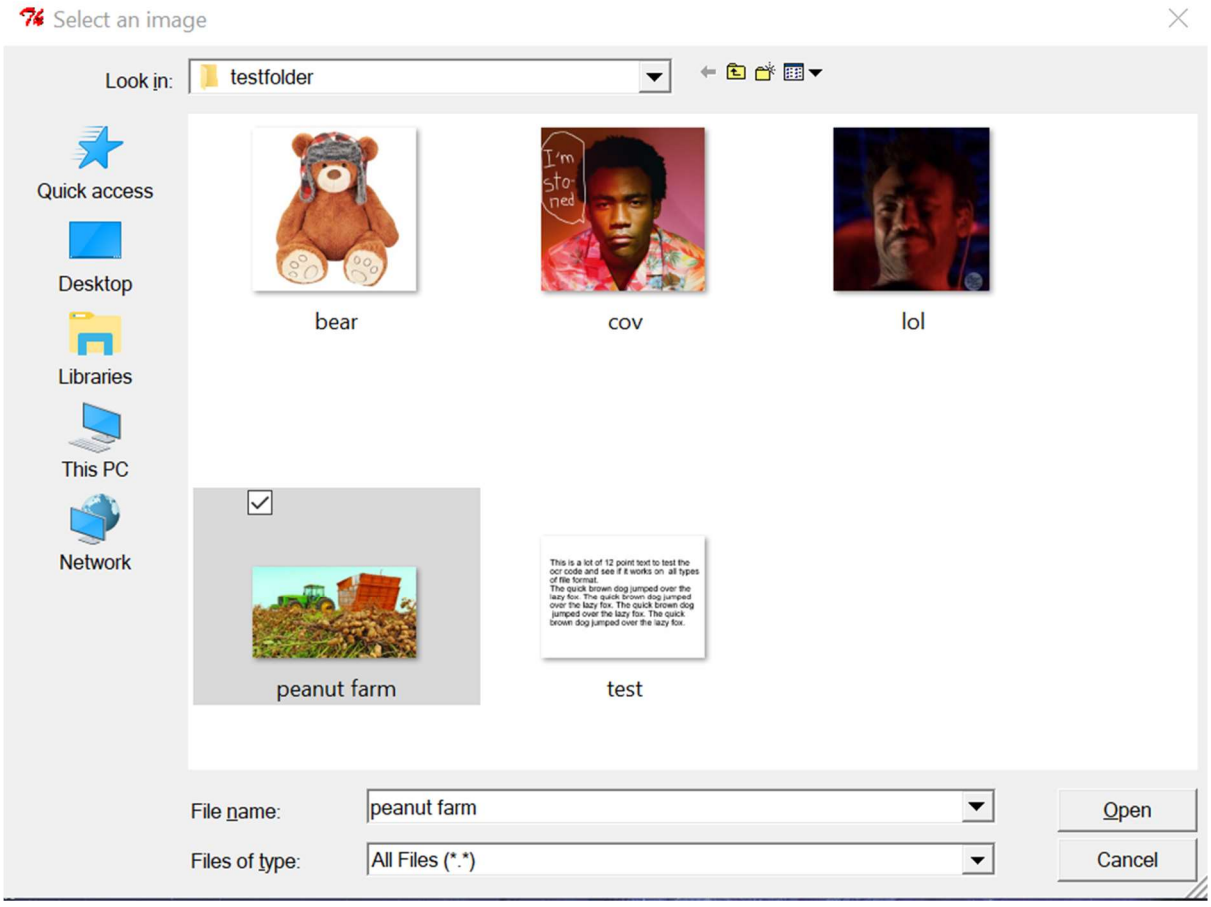
SAMPLE OUTPUT

Asking the user for desired option:

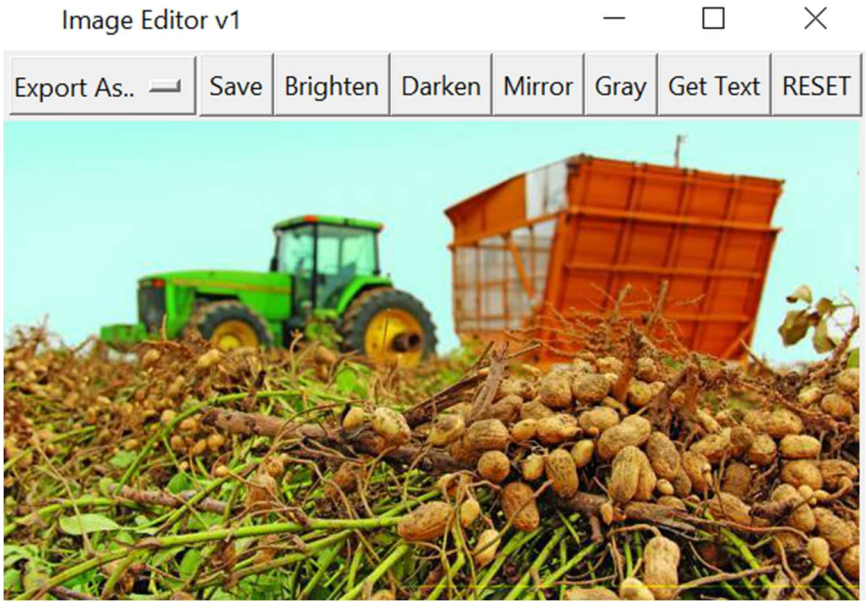
1)Assuming user chose option 1:

```
In [1]: runfile('C:/Users/Rohan/.spyder/computer_project.py', wdir='C:/Users/
Rohan/.spyder')
WELCOME TO THE IMAGE EDITING APPLICATION!!
        What do you want to do?
            1) Edit a single image (GUI)
            2) Edit a batch of images (text based)

Enter the desired option: 1
```

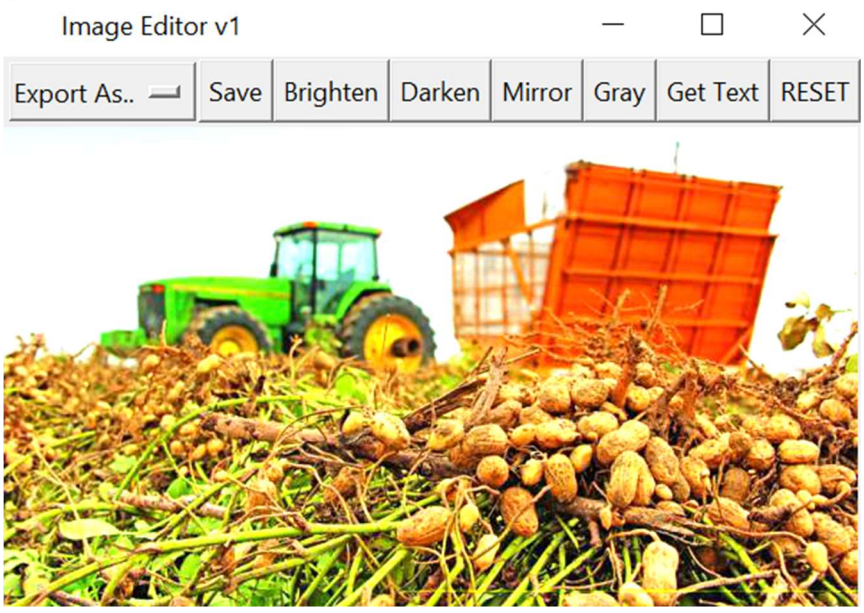
The user selects an image:



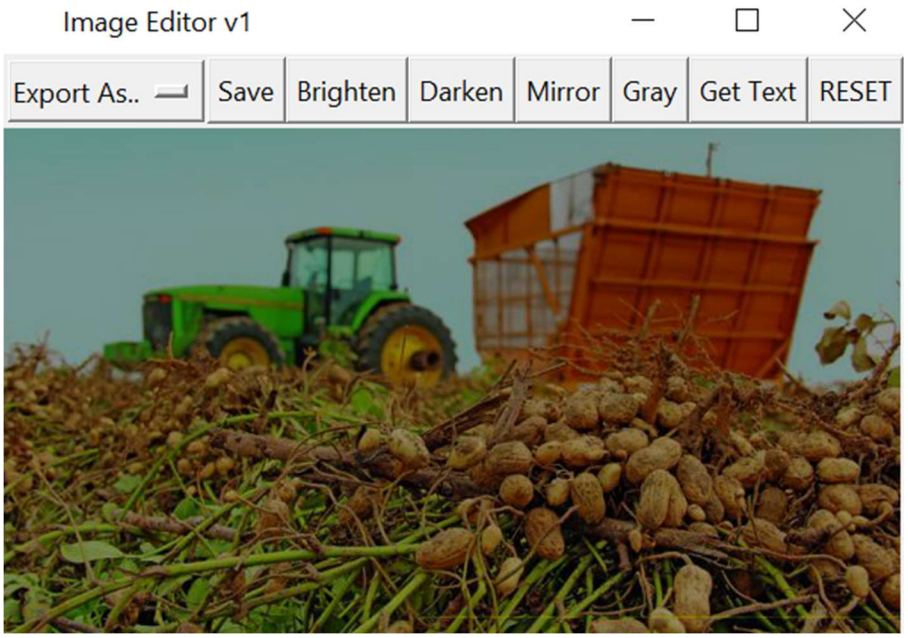
The GUI is shown below:



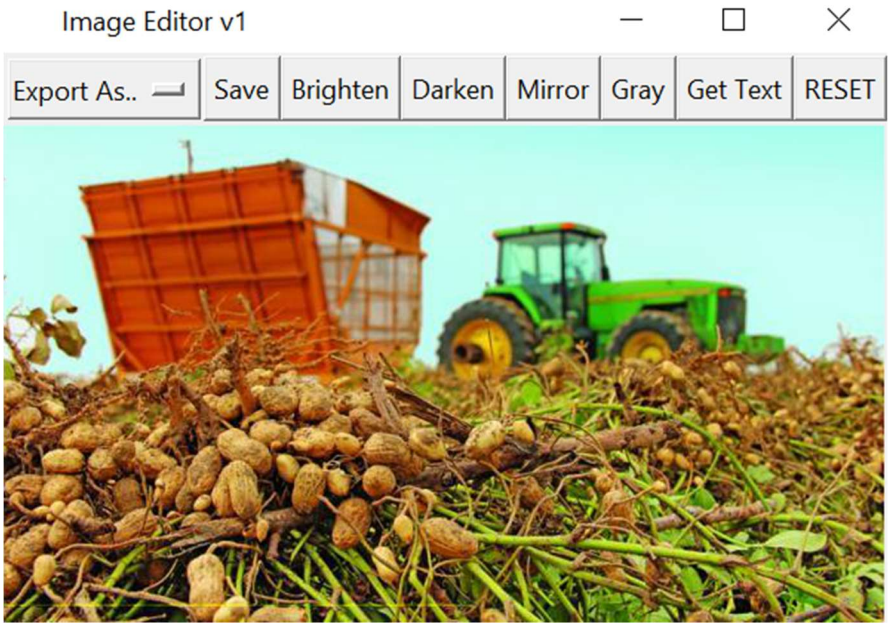
When the brighten button is clicked:



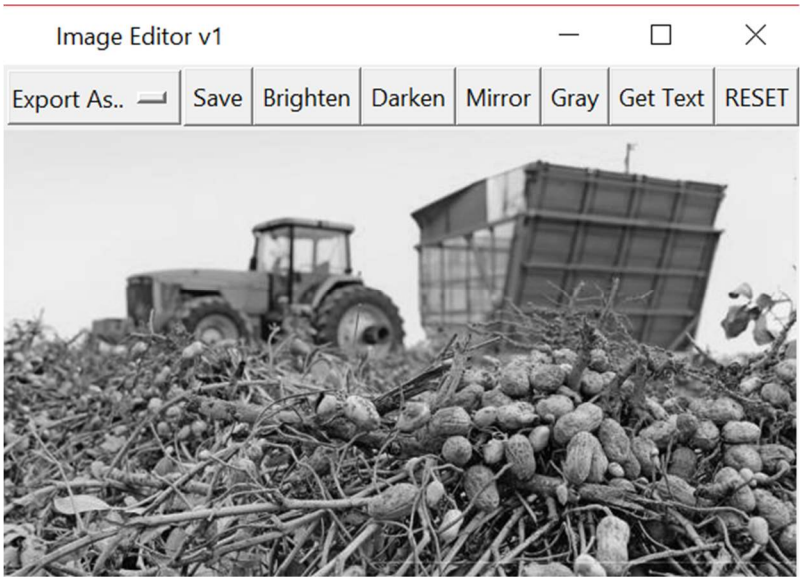
When the darken button is clicked:



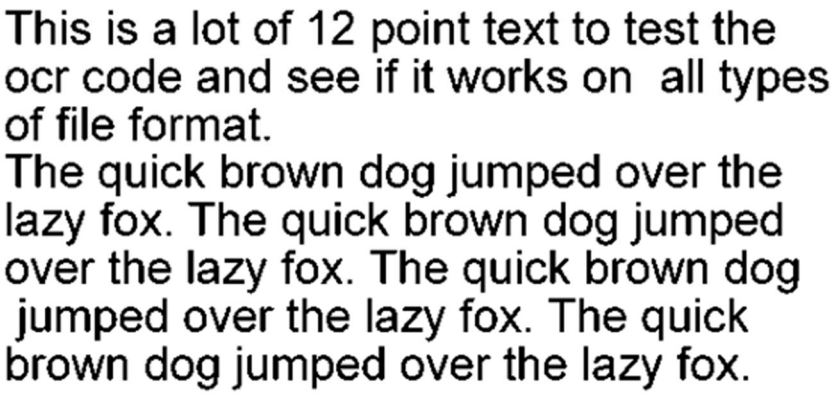
When the mirror button is clicked:



When the Gray button is clicked:



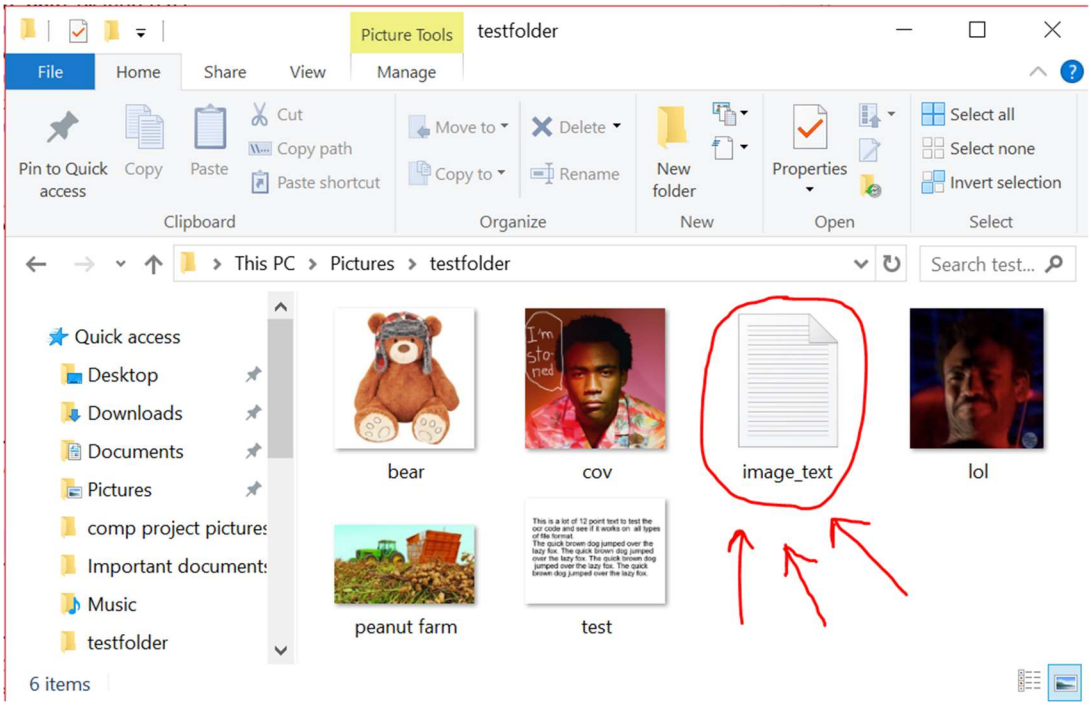
The new image used to demonstrate the “Get Text” feature/button:



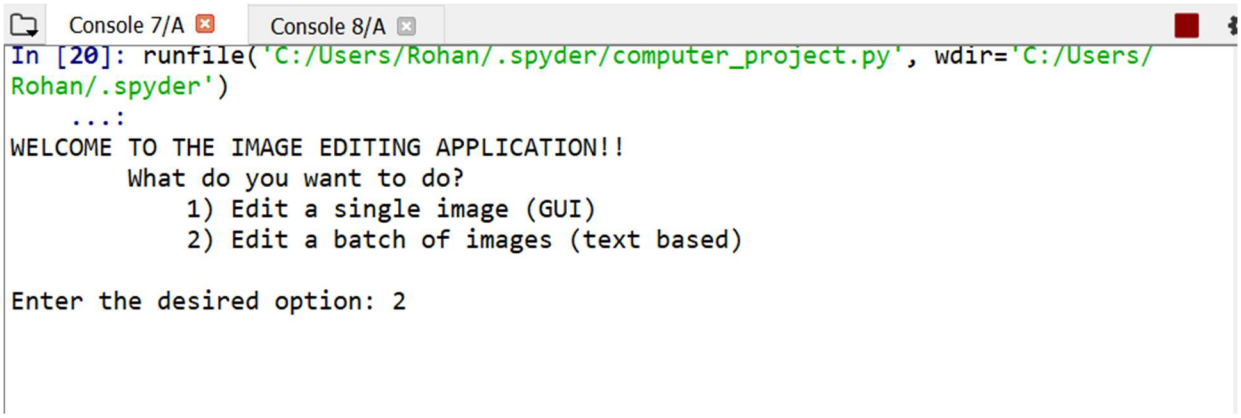
The output when “Get Text” button is pressed. The text in the image is shown as output:

```
Console 7/A   Console 8/A
In [2]: runfile('C:/Users/Rohan/.spyder/computer_project.py', wdir='C:/Users/
Rohan/.spyder')
WELCOME TO THE IMAGE EDITING APPLICATION!!
        What do you want to do?
            1) Edit a single image (GUI)
            2) Edit a batch of images (text based)

Enter the desired option: 1
This is a lot of 12 point text to test the
ocr code and see if it works on all types
of file format.

The quick brown dog jumped over the
lazy fox. The quick brown dog jumped
over the lazy fox. The quick brown dog
jumped over the lazy fox. The quick
brown dog jumped over the lazy fox.
Image text saved in the current directory as "image_text.txt"
```

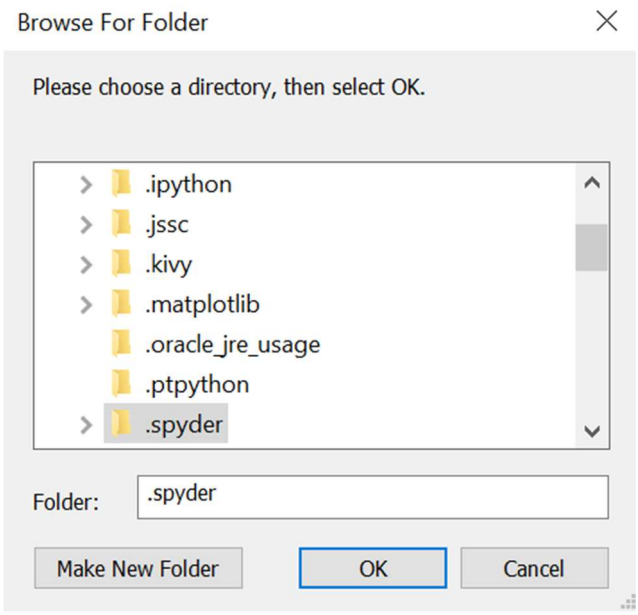
It is also saved in a new file called images.txt in the same directory.



2)Assuming user chose option 2:



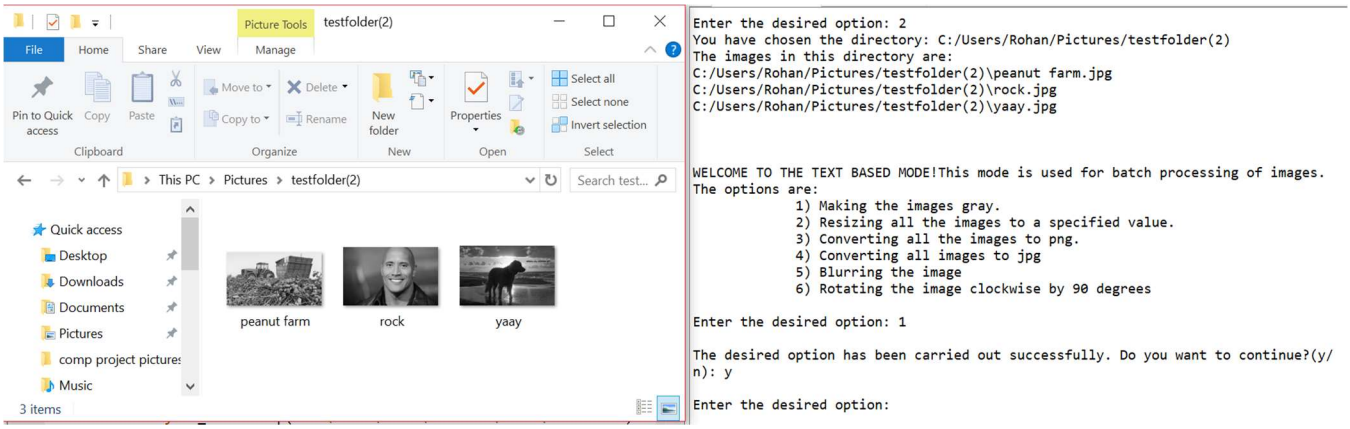
The user selects a folder:



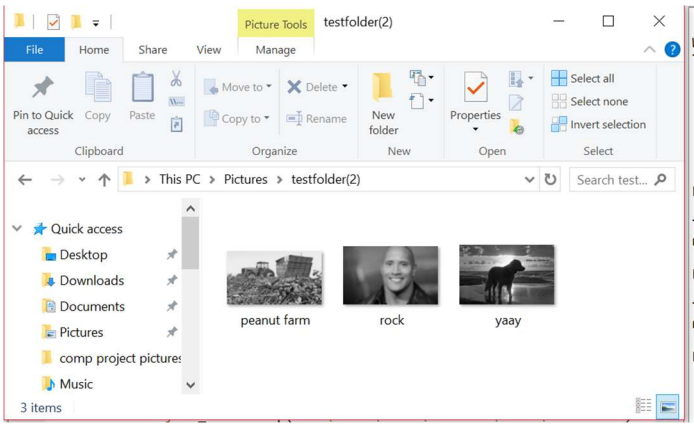
The image paths are shown. User is prompted with a menu.

```
Console 7/A   Console 8/A
Enter the desired option: 2
You have chosen the directory: C:/Users/Rohan/Pictures/testfolder(2)
The images in this directory are:
C:/Users/Rohan/Pictures/testfolder(2)\peanut farm.jpg
C:/Users/Rohan/Pictures/testfolder(2)\rock.jpg
C:/Users/Rohan/Pictures/testfolder(2)\yaay.jpg



WELCOME TO THE TEXT BASED MODE!This mode is used for batch processing of images.
The options are:
            1) Making the images gray.
            2) Resizing all the images to a specified value.
            3) Converting all the images to png.
            4) Converting all images to jpg
            5) Blurring the image
            6) Rotating the image clockwise by 90 degrees

Enter the desired option:
```

Tools: making all images gray:

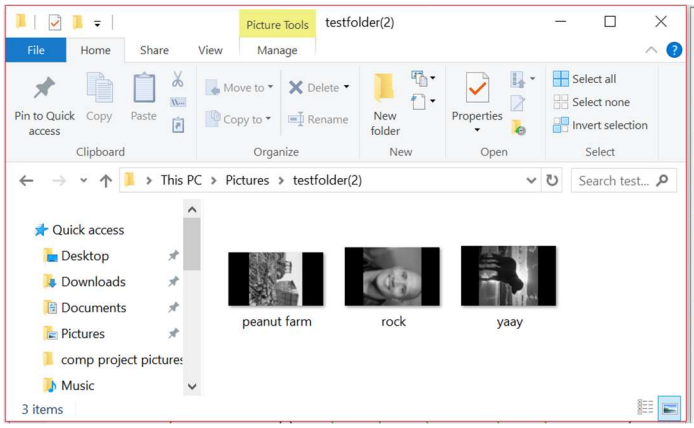


Blurring all the images:



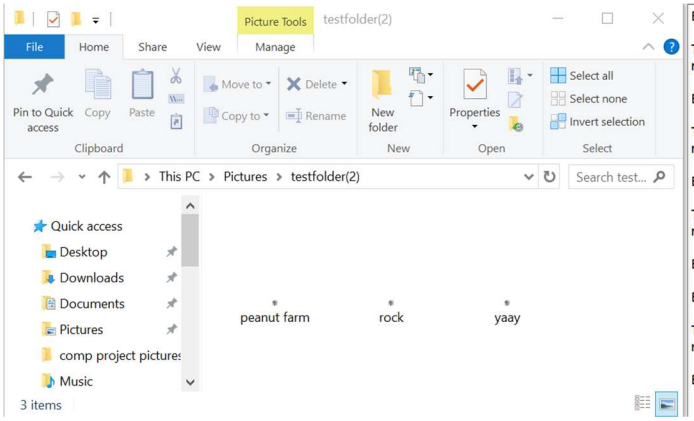

```
WELCOME TO THE TEXT BASED MODE!This mode is used for batch processing of images.
The options are:
            1) Making the images gray.
            2) Resizing all the images to a specified value.
            3) Converting all the images to png.
            4) Converting all images to jpg
            5) Blurring the image
            6) Rotating the image clockwise by 90 degrees

Enter the desired option: 1

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option: 5

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option:
```

Rotating the image by 90 degrees:




```
            3) Converting all the images to png.
            4) Converting all images to jpg
            5) Blurring the image
            6) Rotating the image clockwise by 90 degrees

Enter the desired option: 1

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option: 5

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option: 6

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option:
```

Resizing the image:




```
Enter the desired option: 1

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option: 5

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option: 6

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option: 2

Enter a,b where (a x b) is the desired size of image:1,1

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option:
```

After the operations have been carried out, the user can exit the program:

```
Enter the desired option: 2

Enter a,b where (a x b) is the desired size of image:1,1

The desired option has been carried out successfully. Do you want to continue?(y/
n): y

Enter the desired option: 1

The desired option has been carried out successfully. Do you want to continue?(y/
n): n

In [21]:
```

CONSTRAINTS AND SCOPE

The following is a list of shortcomings of our project which we are currently working on and hope to implement in near future.

- There are not too many editing options available.
- The text interface mode can be made more user friendly.
- The “Getting text from image” feature uses optical character recognition, which is not perfectly implemented in python, and does not work for handwritten text.
- The project cannot be scaled, i.e, used for a large number of pictures without reducing the speed drastically.

This project has a lot of potential. The GUI mode for editing images is perfect for an average user for quick touchups and type conversions. The text based mode is a huge help for average users as well as website developers who need many sizes of an image for thumbnails, icons, etc., and also need to apply formatting to a number of images at a time.