Date 17/1/2024

Expt. No. 1 Page No. 1

Experiment-1

1) Aim:

Write a program to read, save and display images.

2) Requirements:

Python IDLE, Operating system.

3) Code:

```
from PIL import image
import matplotlib.pyplot as plt
import skimage.io as io

def read_image_pil (file_path):
    img = Image.open ("image.jpeg")
    return img

def save_image_pil (img, outpath_path):
    img_save ("output image.jpeg")

def read_image_matplotlib (file path):
    img = plt.imread (file_path)
    return img
```

Teacher's Signature : _____

Date .....

Expt. No. . .....


```
def save_image_matplotlib (img, output_path):
    plt.imsave (output_path, img)

def display_image_matplotlib (img):
    plt.imshow (img)
    plt.show()

def read_image.skit (file_path):
    image = io.imread (filepath)
    return img

def display_image.skit (img):
    io.imshow (img)
    io.show()

if __name__ = "__main__":
    file path = r "D:/joker.jpg"
    output path = r "D:/joker.jpg"

img_pil = read_image.pil (filepath)
save.image_pil (img_pil, output_path)
display_image_pil (image_pil)

image_matplotlib = read_image_matplotlib (filepath)
save_image_matplotlib (img_matplotlib, output)
display_image matplotlib (img_matplotlib)
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. ....3.....

```
img_sckit = read_image_sckit(filepath)
save_image_sckit(img_sckit, output path)

display image.sckit(image_sckit)
```

4) Observations:

- Image.open() is used by PIL to read the image. Save function is used to save it into disk and "show()" function is used to display image.
- When using matplotlib, imread() function is used to read to read the image, "imsave() to save image."

5) Result:

- PILLOW (PIL) opens the image in a new window.
- Matplotlib and sckit image plots image in plotting window.

6) Learning Outcomes:

- Using PIL to read, save and display image.
- using matplotlib to read, save, display image.

Teacher's Signature : Aarti 24/1/24

Date 24/1/2024

Expt. No. 2 Page No. 4

Experiment - 2

1) Aim:

Write a program to convert file format to other.

2) Requirements:

Python IDLE, Operating system

3) Code:

```
from PIL import Image
def convert_image(input_path, output_path,
                  output_format):
try:
with Image.open(input_path) as img:
img.save(output_path, format=output_format)
print(f"Image converted and saved to
      {output_path} in {output_format} format.")
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. 5

```
except Exception as e:
    print(f"Error: {e}")

if __name__ == "__main__":
    input_path = r"D:\8th sem\DIP LAB\image.jpeg"
    output_path = r"D:\8th sem\DIP LAB\image1.png"
    output_format = "PNG"

    convert_image(input_path, output_path,
                  output_format)
```

4) Results:

- Pillow (PIL) library is used for image processing.
- We have converted a JPEG format file into PNG format file.
- We have done a loss-less conversion.

5) Learning Outcomes:

- Using PIL to import the image.
- Learnt about python error handling.

Teacher's Signature : Sreetal 3/1/24

Date 24/1/2024

Expt. No. 3 Page No. 6

Experiment-3

1) Aim:

Write a program to convert one space to another.

2) Requirements:

Python IDLE, Operating System

3) Code:

```
def convert_spaces(input_path, output_path,
                   replacement_char = '_'):
try:
    #open the input file
    with open(input_path, 'r') as infile:

    content = infile.read()

    converted_content = content.replace(' ',
                                replacement_char)

    with open(output_path, 'w') as outfile:

    outfile.write(converted_content)
```

Teacher's Signature : _____

```
print(f"Spaces converted and saved to
      {output_path} with replacement characters
      '{replacement_char}'.")

except Exception as e:
    print(f"Error: {e}")

if __name__ == "__main__":

    input_path = "D:\8th Semester\DIP LAB\exp3.txt"
    output_path = "D:\8th Semester\DIP LAB\converted
                                    _text_Test.txt"

    replacement_char = '_'

    convert_spaces(input_path, output_path, replace-
                                    ment_char)
```

4) Results:

- We have converted space to underscore '_' character.

5) Learning Outcomes:

- Learnt about python error handling.
- Learnt how to convert one space to another.

Teacher's Signature : _____

Date 31/1/2024

Expt. No. 4 Page No. 8

Experiment-4

1) Aim:

Write a program to generate a gray scale from colour image and then generate negative of gray scale image.

2) Requirements:

Python IDLE, Operating system.

3) Code:

```
from PIL import Image

def generate_grayscale(input_path, output_path):
    try:
        with Image.open(input_path) as img:
            grayscale_img = img.convert("L")

            grayscale_image.save(output_path)

            print(f"Grayscale image generated and saved to {output_path}!")
    except Exception as e:
        print(f"Error: {e}")
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. 9

```
def generate_negative(input_path, output_path):
    try:
        with Image.open(input_path) as grayscale_img:
            negative_img = Image.eval(grayscale_img, lambda
                x: 255-x)
            negative_img.save(output_path)
            print(f"Negative Image generated and saved
                to {output_path}.")
    except Exception as e:
        print(f"Error: {e}")

if __name__ == "__main__":
    input_path = "D:\image.jpeg"
    grayscale_output_path = "D:\grayscale_image.jpeg"
    negative_output_path = "D:\negative_image.jpeg"
    generate_grayscale(input_path, grayscale_output_
                                                path)
    generate_negative(grayscale_output_path,
                        negative_output_path)
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. 10

4) Results:

- Learnt and we generated a grayscale image of a normal image.
- And, we generated a negative of grayscale image.

5) Learning Outcomes:

- Learnt how to generate grayscale image with python.
- Learnt how to generate negative of grayscale image with python.

Teacher's Signature : Aneeta

Date 21/2/2024

Expt. No. Exp 5 Page No. 11

Experiment-5

1) Aim:

To demonstrate of thresholding of an Image by three different values.

2) Requirements:

Python IDLE, Operating System

3) Code:

```
import cv2
import numpy as np

def threshold_image(input_path, output_path_template,
                    threshold_values):
    try:
        img = cv2.imread(input_path, cv2.IMREAD_GRAYSCALE)

    threshold_images = [cv2.threshold(img, threshold, 255,
                        cv2.THRESH_BINARY)[1] for
                        threshold in threshold_values].

    for i, threshold_img in enumerate(threshold_images):
        cv2.imshow(f"Threshold {i+1}", threshold_img)
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. 12

```
cv2.imwrite(output_path_template, format(i+1),
                                    threshold_img).

cv2.waitKey(0)
cv2.destroyAllWindows()
except exceptions as e:
    print(f"Error: {e}")

if __name__ == "__main__":
    input_path = "D:\image.jpeg"
    output_path = "D:\threshold_image_{}.jpeg"
    threshold_values = [100, 150, 200]

    threshold_image(input_path, output_path_template, threshold_
                                                        values)
```

4) Results:

Demonstrate a thresholding of an image by three different values.

5) Learning outcomes:

Learnt to demonstrate threshold of an image.

Teacher's Signature : 21/2/24

Date 28/2/2024

Expt. No. 6 Page No. 13

Experiment - 6

Aim → Write a program to demonstrate edit operations like cropping, resizing, scaling, and flipping.

Requirements - Python IDLE, Operating system

Code →

```
import cv2

def crop_image(input_path, output_path, x, y,
               width, height):

    try:
        img = cv2.imread(input_path)
        cropped_img = img[y:y+height, x:x+width]

        cv2.imshow("Cropped Image", cropped_img)
        cv2.imwrite(output_path, cropped_img)

        cv2.waitKey(0)
        cv2.destroyAllWindows()

    except Exception as e:
        print(f"Error: {e}")
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. 14

```
def resize_image(input_path, output_path,
                 new_width, new_height):
    try:
        img = cv2.imread(input_path)
        resized_img = cv2.resize(img, (new_width,
                                       new_height))

        cv2.imshow("Resized Image", resized_img)
        cv2.imwrite(output_path, resized_img)
        cv2.waitKey(0)
        cv2.destroyAllWindows()

    except Exception as e:
        print(f"Error: {e}")

def scale_image(input_path, output_path, scale_
                                        factor):

    try:
        img = cv2.imread(input_path)
        scaled_img = cv2.resize(img, None, fx =
                                scale_factor, fy = scale_
                                              factor)
        cv2.imshow("scaled Image", scaled_img)
        cv2.imwrite(output_path, scaled_img)
        cv2.waitKey(0)
        cv2.destroyAllWindow()
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. 15

```
except Exception as e:
    print(f"Error: {e}")

if __name__ == "__main__":

    input_path = r"C:\Jatin\input_image.jpg"
    output_path_cropped = r"C:\cropped_image.jpg"
    output_path_resized = "C:\\resized_image.jpg"
    output_path_scaled = "C:\\scaled_image.jpg"
    output_path_flipped = "C:\\flipped_image.jpg"

    crop_image(input_path, output_path_cropped,
               x=50, y=50, width=300, height=200)

    resize_image(input_path, output_path_resized,
                 new_width=400, new_height=300)

    scale_image(input_path, output_path_scaled,
                scale_factor=0.5)

    flip_image(input_path, output_path_flipped,
               flip_code=1)
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. 16

Results →

This code defines functions to crop, resize, scale and flip an input image using OpenCV in Python with example usage demonstrating each operation.

Learning Outcome -

1.) Learnt about OpenCV.

11) Learnt how to resize an image.

Teacher's Signature :

Date 27/3/2024

Expt. No. 7 Page No. 17

Experiment - 7

1) Aim:

Write a program to read, display and show the attribute and histogram of the image.

2) Requirements:

Python IDLE, Operating system.

3) Code:

```
import cv2
import matplotlib.pyplot as plt
def display_image(input_path):
 try:
     img = cv2.imread(input_path)

 cv2.imshow("Image", img)
 cv2.waitKey(0)
 cv2.destroyAllWindows()

 except Exception as e:
  print(f"Error: {e}")

def show_image_attributes(input_path):
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. ..18.....

```
try:
    img = cv2.imread(input_path)

height, width, channels = img.shape

print(f"Image Dimensions: {width} x {height}")
print(f"Number of Channels: {channels}")

except Exception as e:
print(f"Error: {e}")

def show_image_histogram(input_path):
    try:
        img = cv2.imread(input_path, cv2.IMREAD_
                                    GRAYSCALE)
histogram = cv2.calcHist([img], [0], None, [256],
                                    [0,256])
plt.plot(histogram)
plt.title("Image Histogram")
plt.xlabel("Pixel Value")
plt.ylabel("Frequency")
plt.show()

except Exception as e:
print(f"Error: {e}")

if __name__ == "__main__":
```

Teacher's Signature : _____

Date .....

Expt. No. . ..... Page No. 19

```
input_path = "D:\image.jpg"
display_image = (input_path)
show_image_attributes (input_path)
show_image_histogram (input_path)
```

4) Results:

- We read, display and show the attributes and histogram of the image.

5) Learning Outcomes:

- Learnt about openCV.
- Learnt how to read, display and show the attributes and histogram of the image.

Teacher's Signature : _____