**Project Requirement and Specification**

**On**

**Image Rotation**

Submitted to: Submitted by:

Ms. Meenakshi Mandola Hemlata Chuphal

Roll No.: 2014401

Guided by: Semester: 3rd

Ms. Meenakshi Mandola Session: 2022-2023

**DEPARTMENT OF COMPUTER SCIENCE AND ENGEENRING**

**GRAPHIC ERA DEEMED TO BE UNIVERSITY,**

**DEHRADUN**

About:

In this project I have programmed a system which will perform image rotation. This project will take path of the image to be rotated and the angle through which you want to rotate from the user and then will display the rotated image. If the path entered is invalid or if image does not exist it displays an error. This project utilizes OpenCV Library which is an open-source computer vision and machine learning library used for image processing, video capture and analysis.

Technologies Used:

1. C++ Programming language code the program.
2. OpenCV (Open-Source Computer Vision Library).
3. Visual Studio.

How to run this project?

You need to have these system requirements to run this project on:

1. Visual Studio C++ tool kit should be installed in your system. To download Visual Studio, you can go to their official site and download the latest version. And then install the C++ tool kit.
2. Download OpenCV (Open-Source Computer Vision Library) from their official site: opencv.org and then extract it to any location in your system. The tutorial to do that: <https://medium.com/@subwaymatch/opencv-410-with-vs-2019-3d0bc0c81d96>

After completing these requirements, you will be able to run this program your system. To use this program for image rotation, you need to enter the location of the image you need to rotate in your system and after that the angle through which you want to rotate the image. After entering these details, the image will be displayed after the rotation.

Various Methods used for this project:

1. Header files:
2. #include<math.h>
3. #include<iostream.h>
4. #include<opencv2/opencv.hpp>
5. RotateImage (Mat, float): This function takes two parameters a mat, and a float value. The mat is the image to be rotated in the mat from and the float value is the angle for rotation. This function uses inbuilt functions such as:
6. getRotationMatrix2D (Point2f centre, double angle, double scale): Calculates an affine matrix of 2D rotation.
7. RotatedRect (Point2f centre, Size2f size, double size)
8. warpAffine (InputArray src, OutputArray dst, InputArray M, Size dsize, int flags=1, int borderMode=0, Scalar bordervalue): Applies an affine transformation to an image.
9. boundingRect2f (): returns the minimal (exact0 floating point rectangle containing the rotated rectangle.

It returns the rotated mat after rotation to the function called.

1. ShowImage (Mat src , Mat dst): This function takes two mat values as input and then displays the mat in image form in two windows. This function displays Original image and Rotated image in two windows. It uses inbuilt functions such as:
2. namedWindow (String &winname, int flags=1): It creates a window that can be used as a placeholder for images and trackbars.
3. imshow (String &winname, InputArray Mat): Displays an image in the specified window.
4. destroyWindow (String &winname): Destroys the specified window.
5. waitKey (int delay=0): Waits for a pressed key.
6. imread (String &filename, int flags=1): Loads an image from a file.