**THERE ARE A FEW REQUIREMENTS THAT NEEDS TO BE FULLFILLED BEFORE STARTING WITH THE PROJECT :**

**Operating System :** *Windows (In this case I have used Windows but it can also be done on Linux as well)*

**Programming Language Used :** *C++*

**Software Required :** *Microsoft Visual Studio 2019, OpenCV.*

**Steps To Configure OpenCV:**

1. *Extract the OpenCV in the root directory. Install Visual Studio as well, it will take download 2.1 GB of data to download.*
2. *Now we need to add Environment Variables in Advanced System Settings so that we can use OpenCV library with our C++ program.*
3. *The last step is to link OpenCV with Visual Studio. For this we need to add “Additional Include Directories” in Visual Studio and link it to Visual Studio.*
4. *Everything is done. Now you can test your code. Enjoy.*

**Code for Inputting an Image and Rotating it with the angle entered by the user:->**

#include<iostream>

#include<opencv2/opencv.hpp>

using namespace std;

using namespace cv;

int main()

{

float angleofrotation;

cout << "Enter the angle of rotation of image: ";

cin >> angleofrotation;

Mat outputimage;

Mat inputimage = imread("pic12.jpg",IMREAD\_UNCHANGED);

if (inputimage.empty())

{

cout << "Error!!";

cin.get();

return -1;

}

String inputwindowname = "Input Image";

namedWindow(inputwindowname, WINDOW\_NORMAL);

imshow(inputwindowname, inputimage);

outputimage = inputimage;

String outputwindowname = "Output Image";

Point2f pt(inputimage.cols / 2., inputimage.rows / 2.);

Mat r=getRotationMatrix2D(pt, angleofrotation, 1.0);

warpAffine(inputimage, outputimage, r, Size(inputimage.cols, inputimage.rows));

namedWindow(outputwindowname, WINDOW\_NORMAL);

imshow(outputwindowname, outputimage);

waitKey(0);

return 0;

}

**Code Explanation :**

**#include<opencv2/opencv.hpp> :** *This header file is used to used OpenCV library and its functions.*

**using namespace cv :** *namespace is added for using the functions defined in OpenCV library.*

*In OpenCV, image is processed in the form of a matrix and all the operation done on the image is of matrix types.(Eg. Matrix multiplication, Matrix addition, etc.) “Mat” is the data type used for this.*

**imread() :** *imread() function reads the image.*

**inputimage.empty() :** *This function checks if the image is loaded properly or not.*

**Point2f :** *It is just a class made to store Point semantic object, that is an object that holds x and y coordinates in float.*

**namedWindow(“My Flower”, WINDOW\_NORMAL) :** *This function is used to make a window inside which we can display/load our image. The name of the window here would be “My Flower”.*

**imshow(inputwindowname, inputimage) :** *This function display the image on a window.*

**getRotationMatrix2D(pt, angleofrotation, 1.0) :** *getRotationMatrix2D rotates the image according to the arguments present inside it. ‘pt’ here indicates the point about which the image would be rotated, ‘angleofrotation’ is the angle by which it will get rotated(‘+’ value of angle means anti-clockwise rotation whereas ‘-’ value of angle means clockwise rotation) and ‘1’ here means the Isotropic scale factor.*

**warpAffine(inputimage, outputimage, r, Size(inputimage.cols, inputimage.rows)) :** *The warpAffine() function mainly uses the transformation matrix(in our case matrix ‘r’ is the transformation matrix) to transform the image such as rotation, affine, translation,etc. We only need to provide a 2X3 transformation matrix ‘r’ to transform the image.*