

Demo Programs

scaling.py

This demo program resizes the image by the specified factors in the x and y directions.

translation.py

This demo program shifts the image by a specified number of pixels in both the x and y directions. To perform translation, we specify a transformation matrix M of the form

$$M = \begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \end{bmatrix}$$

where

tx is the translation in the x direction

ty is the translation in the y direction

rotation.py

This program rotates the image by 45°. This value can be adjusted.

CV2 functions demonstrated

cv2.resize(img, dst, fx = , fy =)

This function scales the image by the factors fx (horizontal) and fy (vertical). The first argument is the image. The second argument is the size of the output image. For this application, we set that to (0,0) or None. That indicates that the size of the output image will be computed by taking into account the scaling factors.

cv2.warpAffine(img, M, output_size)

This function performs an affine transformation of an image. In the translation example, the transformation matrix, M is such that it performs only translation and no rotation. However, it can be used to perform rotation as well, as seen in the rotation example, where the transformation matrix is quite different. The output size is represented as width x height. Since columns corresponds to the width and rows corresponds to the height, it would be of the form (col, row)

cv2.getRotationMatrix2D(centre, angle, scale)

This function gets the transformation matrix for performing rotation. Here, the centre corresponds to the point of the image about which rotation is to be performed. It is (col/2, row/2), since we perform rotation about the centre of the image. The angle is expressed in degrees, and scale is used to indicate the amount of scaling. Since our examples for translation and rotation don't need to be scaled, we set scale as 1.