**Activity\_1**

1. **First transformation**

In this activity, a simple horizontal shearing transformation was used, with the following transformation matrix, T = [1 0 0; -0.3 1 0; 0 0 1], which then with affine2d() and imwarp(), the transformation was done.

Figure 1, shows the original image and the transformed image in their corresponding reference frames.

|  |  |
| --- | --- |
|  |  |

Figure 1 - The original image and the transformed image in their corresponding reference frames.

Using the images’ reference frames, the effect of the transformation is evident as the limits of the axis have changed; we can see the x-axis in the transformed image is now increased to more than 300.

However, we can show the images with the same reference frame using the ‘OutputView’ parameter in the call to the imwarp function. When using the 'OutputView' parameter with imwarp, it forces the warped image to conform to the spatial referencing of the specified imref2d object in the function, which here is the frame of the original image.

|  |
| --- |
|  |

Figure 2 - The original image and the transformed image with same reference frames.

Looking at the transformed image, it can be seen that the image is not as sharp as the original image.

During the interpolation that takes place in imwarp function, this may happen. We can break down how imwarp works as follows:

When we specify a transformation and apply it to an image using imwarp, the function computes where each pixel in the input image should map to in the transformed image space; It calculates a new location for each pixel based on the affine transformation matrix.

After the transformation step, the new pixel locations may not align with the original grid of pixel locations - they can be non-integer coordinates. Therefore, MATLAB must interpolate the pixel values for these new locations. The function resamples the pixel values at these new locations from the transformed space, using an interpolation method (nearest-neighbor, bilinear, bicubic, etc.). This process may lead to a reduction in sharpness compared to the original image.

1. **Second transformation**

For the second transformation, T = [cosd(10) sind(10) 0; -sind(10) cosd(10) 0; 10 10 1], a rigid transformation was chosen.

|  |  |
| --- | --- |
|  |  |

Figure 3 - The original image and the transformed image in their corresponding reference frames.

|  |
| --- |
|  |

Figure 4- The original image and the transformed image (rigid) with the same reference frames.