**Artificial Intelligence**

**LAB # 09**

**LAB OBJECTIVES**

The objective of this lab is to understand

* How to rotate images in MATLAB
* How to crop images in MATLAB

**Image cropping**

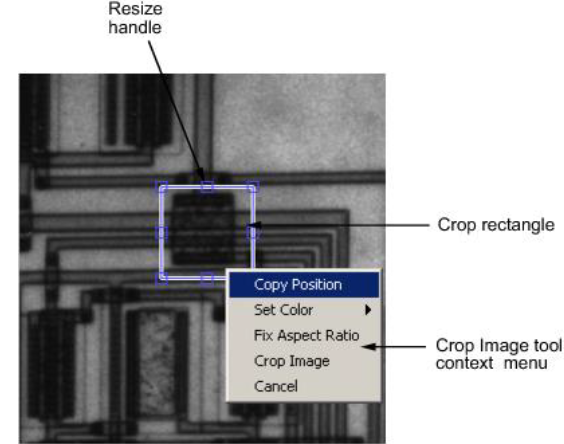
Cropping an image means creating a new image from a part of an original image. The function use to crop an image is imcrop

I = imcrop creates an interactive Crop Image tool associated with the image displayed in the current figure, called the target image. The Crop Image tool is a moveable, resizable rectangle that you can position interactively using the mouse. When the Crop Image tool is active, the pointer changes to cross hairs when you move it over the target image. Using the mouse, you specify the crop rectangle by clicking and dragging the mouse. You can move or resize the crop rectangle using the mouse. When you are finished sizing and positioning the crop rectangle, create the cropped image by double-clicking the left mouse button or by choosing Crop Image from the context menu. imcrop returns the cropped image, I. The following figure illustrates the Crop Image tool with the context menu displayed. For more information about the interactive capabilities of the tool, see the table that follows.

**Syntax**

I2 = imcrop(I)

imcrop(I) displays the image I in a figure window and creates a cropping tool associated with that image. I can be a grayscale image, a true color image, or a logical array. The cropped image returned, I2, is of the same type as I.



Imcrop(image,rect)

I2 = imcrop(I, rect) crops the image I. rect is a four-element position

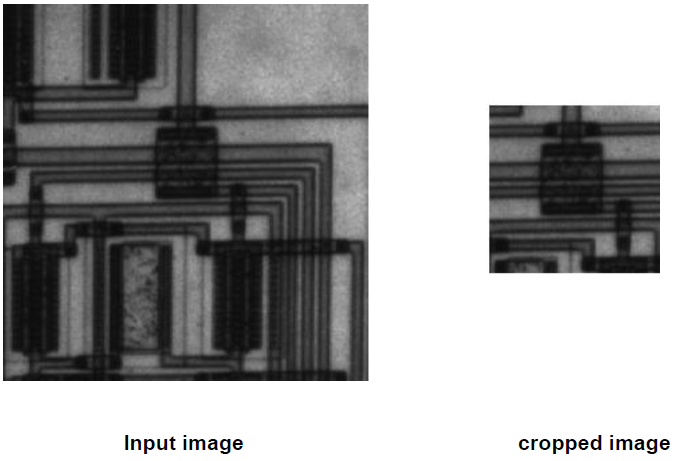
vector[xmin ymin width height] that specifies the size and position of the crop rectangle.

**Example**

I = imread(‘circuit.tif’);

I2 = imcrop(I,[75 68 130 112]);

imshow(I), figure, imshow(I2);



**Image Rotation**

**Syntax**

**B = imrotate(A,angle)**

B = imrotate(A,angle) rotates image A by angle degrees in a counterclockwise direction around its center point. To rotate the image clockwise, specify a negative value for angle. imrotate makes the output image B large enough to contain the entire rotated image. imrotate uses nearest neighbor interpolation, setting the values of pixels in B that are outside the rotated image to 0 (zero).

****

**Input image image rotated at 45 degree**

**Task 1**

Rotate the image at 45, 120 and 270 degree and show the original image and rotated images in subplot

**Task 2**

Take any three images, crop them and then display cropped images in subplot