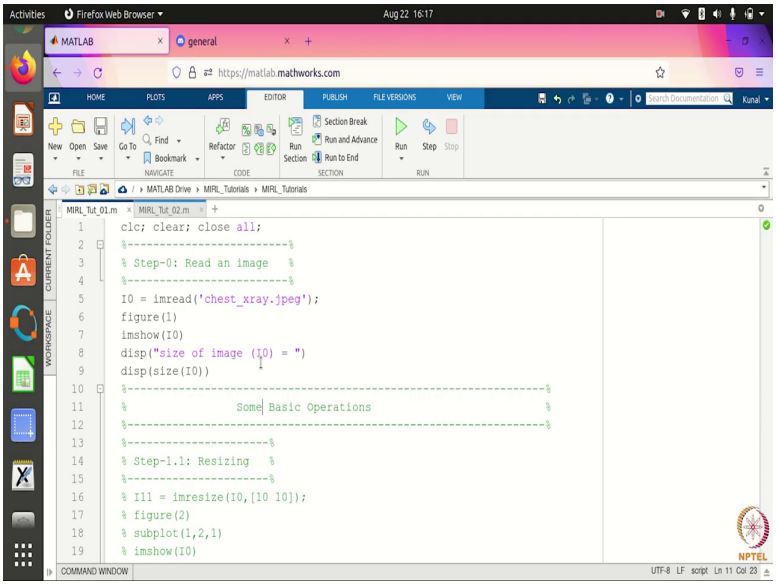
Medical Image Analysis
Professor Ganapathy Krishnamurthi
Biomedical Engineering Design
Indian Institute of Technology, Madras
Lecture 25
Basic Image Processing Technique using MATLAB

(Refer Slide Time: 00:15)

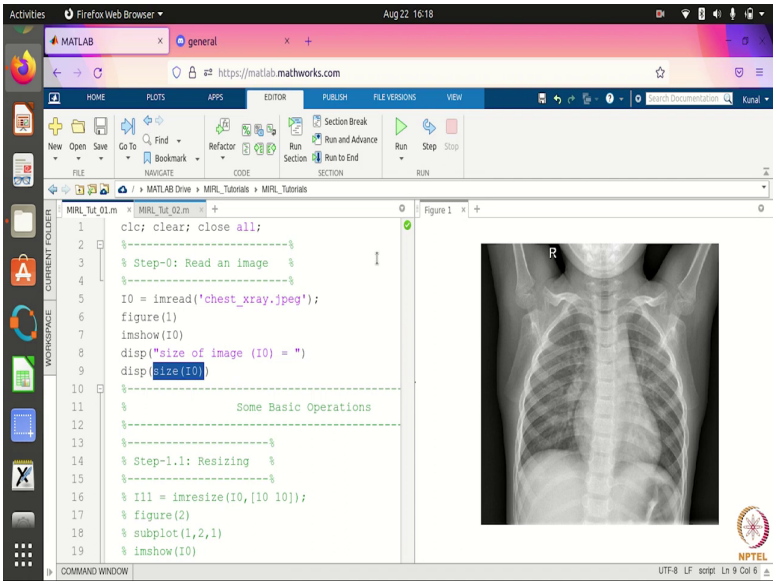


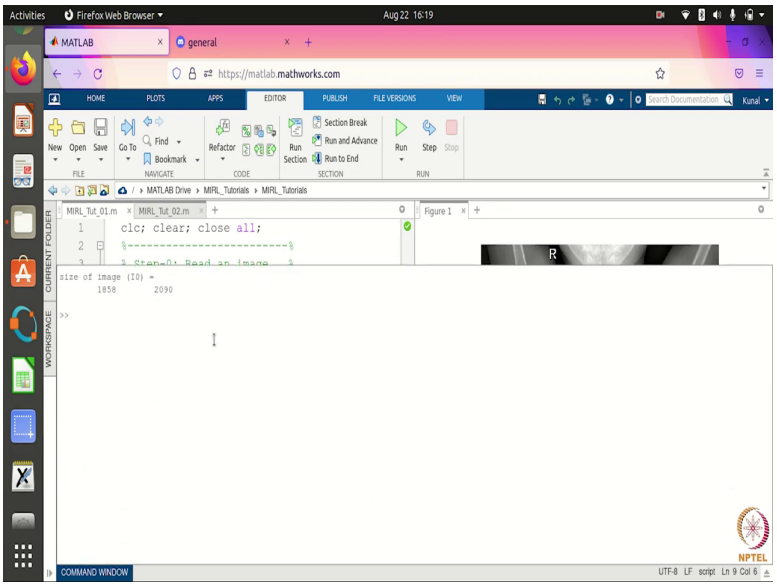
Hello everyone. Welcome to the first tutorial of Medical Image Analysis course. In this tutorial we will be showing you some basic operations that you can do in MATLAB by writing a whereby writing a few lines of code. Our main objective is not to teach you all the commands but to encourage you to explore the image processing toolbox of MATLAB yourself and I hope this tutorial will help you in that. So, let us start.

(Refer Slide Time: 00:50)

So, suppose I have an image this image chest Xray dot jpeg and I want to read this image in MATLAB so the command for this for reading an image is "imread". So, when you do this it will the image will be stored as a matrix in the variable I0 and if you want to know about the size of image you have to just write this command size of I naught (I0) and it will give you the rows and columns of I0. For example, when I run this code:

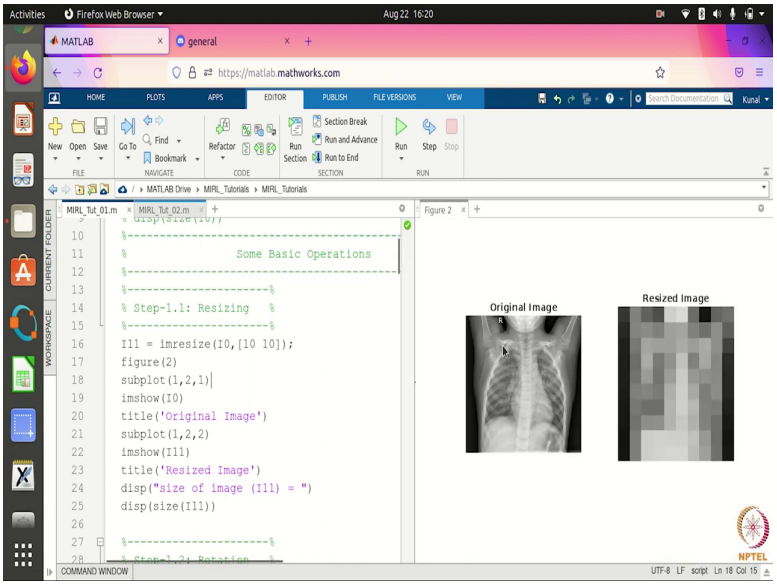
(Refer Slide Time: 01:27)

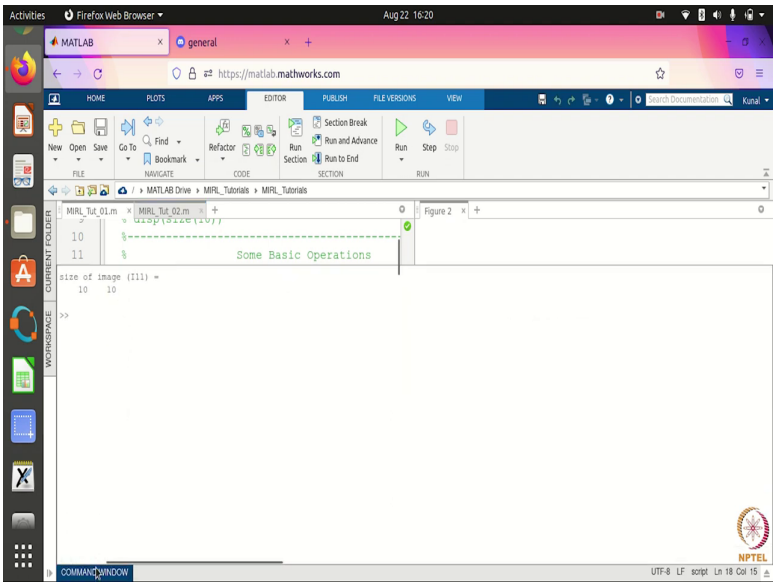




You can see it reads the image and the size of image is 1858 by 2390. Almost 2000 by 2000. So, this is the first thing or the zeroth thing. Now you can do various operations on this image for example you can resize it.

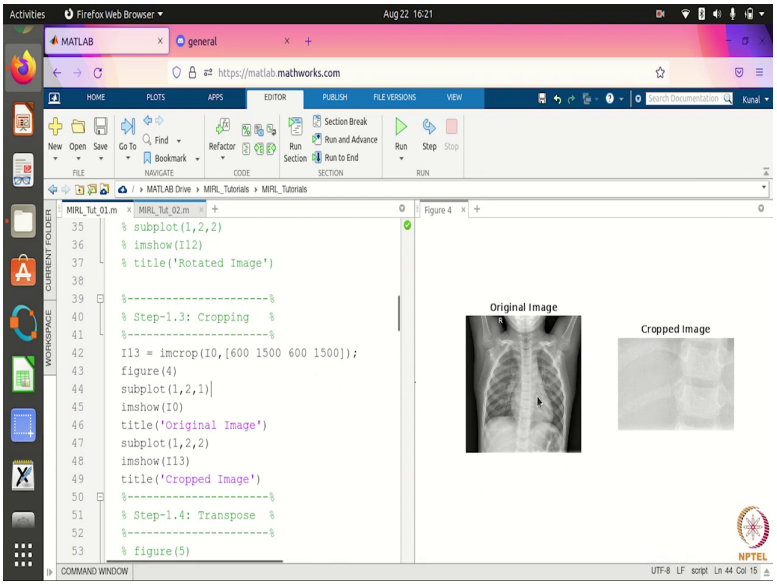
(Refer Slide Time: 02:04)





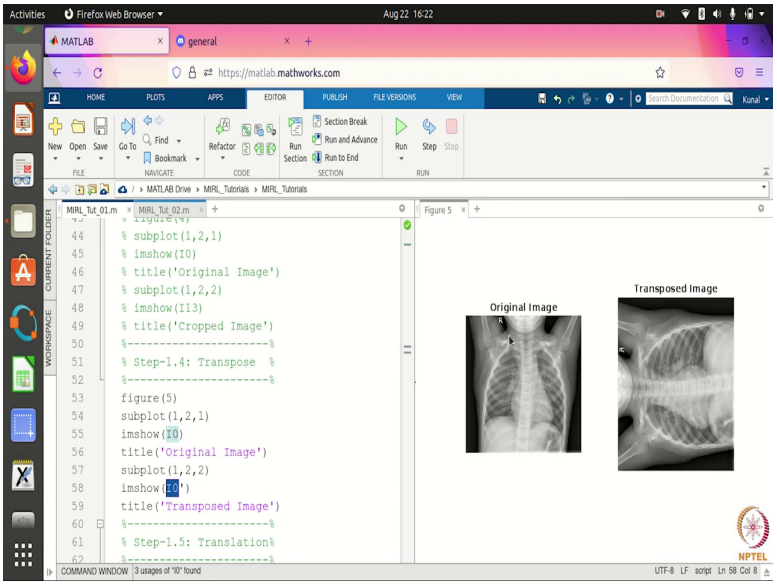
So, as we saw the original image is almost 1000 by 1000. Suppose I resize it to a very small dimension 10 by 10. So, what do you expect will happen to the image? It will be blurred out and all the details will be lost. So, for resizing the command is "imresize" and this is the original image you already have I0 and this is the size of the output image that you want it should be 10 by 10. So, when you run this command,you get the resized image and you can see the size of image is 10 by 10 and it is very blurred as we expected. Do not worry about the commands this code will be provided to you also this is just one line one command so do not worry about it. You do not have to write a code it is just one command.

(Refer Slide Time: 03:24)



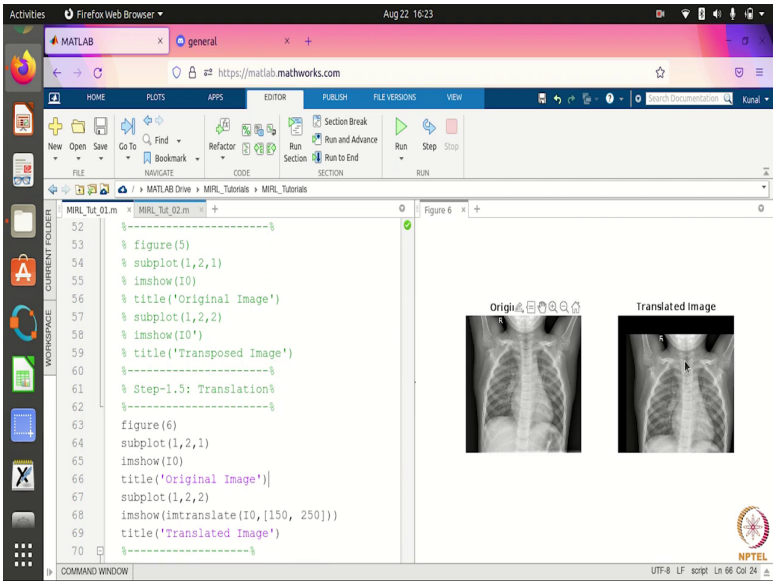
So other than resizing you can crop an image. So, for cropping the command is “imcrop” and when we crop an image, we basically select a rectangle and the dimension of the rectangle is specified here. So, when you run this command, you can extract a particular section of image.

(Refer Slide Time: 04:18)



So, you can also take transpose of the image there is no special command simply because image is a matrix so you can simply take transpose of the matrix. If I0 is the original matrix then I0 prime represents its transpose. So, when you run this see the image gets transposed. So, the rows and columns are interchanged so this is transpose.

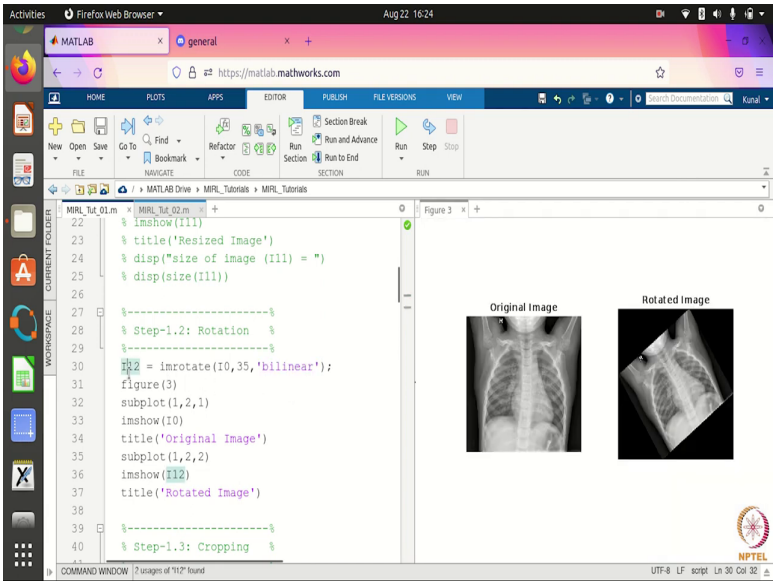
(Refer Slide Time: 05:00)



Now, we show some rigid transformations to you. Rigid transformation means the shape of the image will not be distorted. The first rigid transformation is translation. Suppose, I have an image; a chest X ray image. And I want to translate it. So this is the translation in x direction, 150 and this is translation in y direction, 250 and then the command is “imtranslate”. So, when you run this command your image gets translated by the amount that you specified.

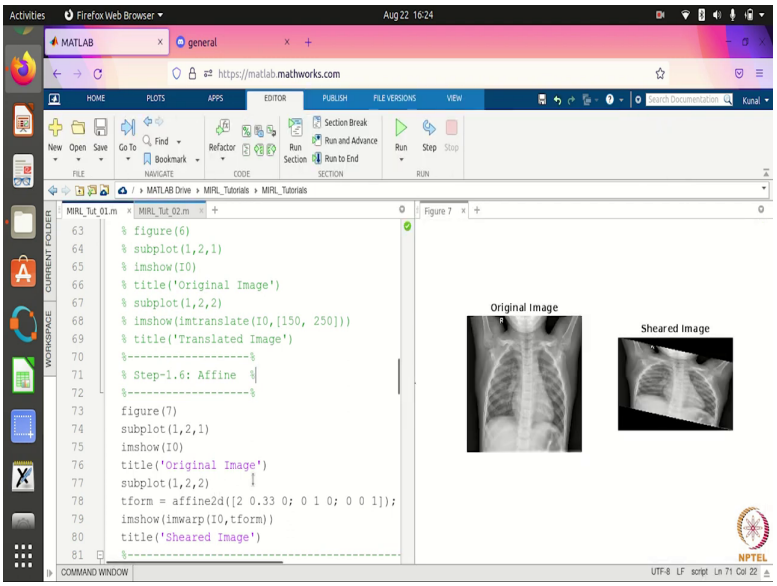
You see this is the original image and this is the translated image 150 in the x direction and 250 in the y direction. So other than that you can also rotate the image.

(Refer Slide Time: 06:05)



So just like “imtranslate” we have “imrotate”. Also you can specify the angle by which you want to rotate the image, for example here it is 35 degree, the angle of rotation .And here bilinear represents interpolation method that will be used to determine the pixel intensity in the rotated image. So, when we run this we get this. So, you can see we have a rotated image.

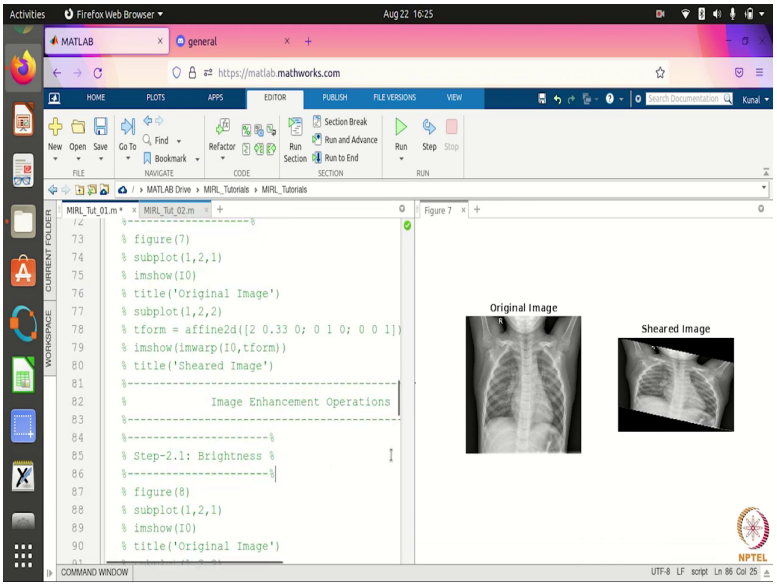
(Refer Slide Time: 06:56)

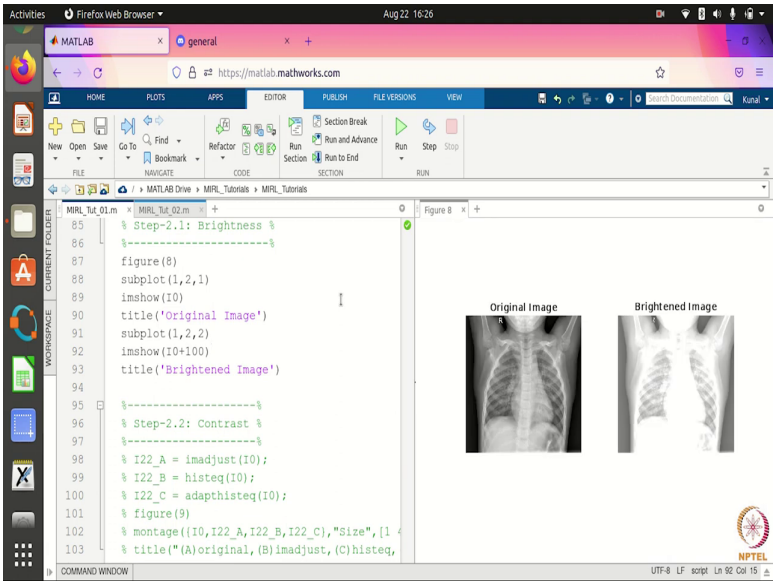


In translation as well as rotation the shape is not distorted but you can also distort the image. How? By using by introducing shear deformations in them in the image for that you can use this affine transformation and this will introduce shear in your image. Shear transformation.

So, here you define the transform. This “tform” represents the transformation. The information about the horizontal and vertical here and then you apply this using this command and here you can see the sheared image. So these are the basic operations. Now, we go to the next section.

(Refer Slide Time: 07:47)

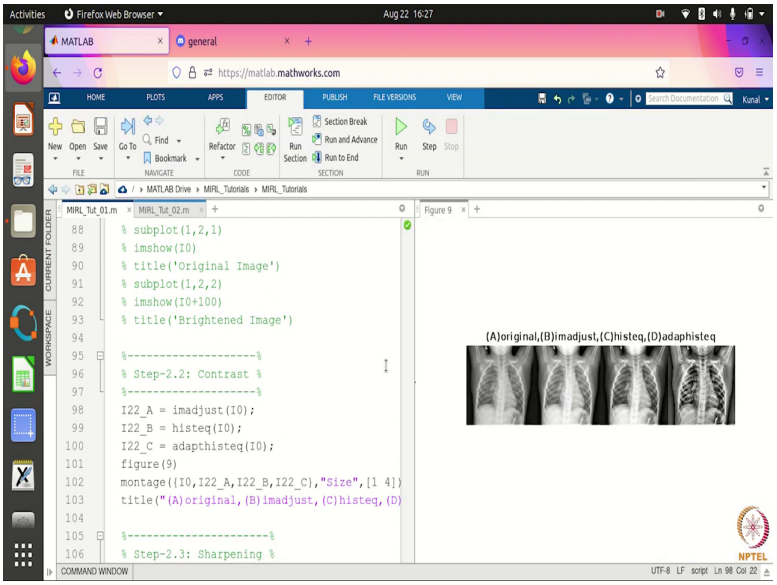




It is some image enhancement operations. So the first image enhancement operation is brightness. Suppose, you have an image and you want to increase its brightness so what can you do so the idea is to just simply add a constant simply add a constant to the original image so for example original image is I0 and to brighten it I am adding 100 to all the pixel values I0 plus 100.

Similarly, if I want to like decrease the intensity of the image I have to just subtract it so when I run this command I get the following brightened image original image.

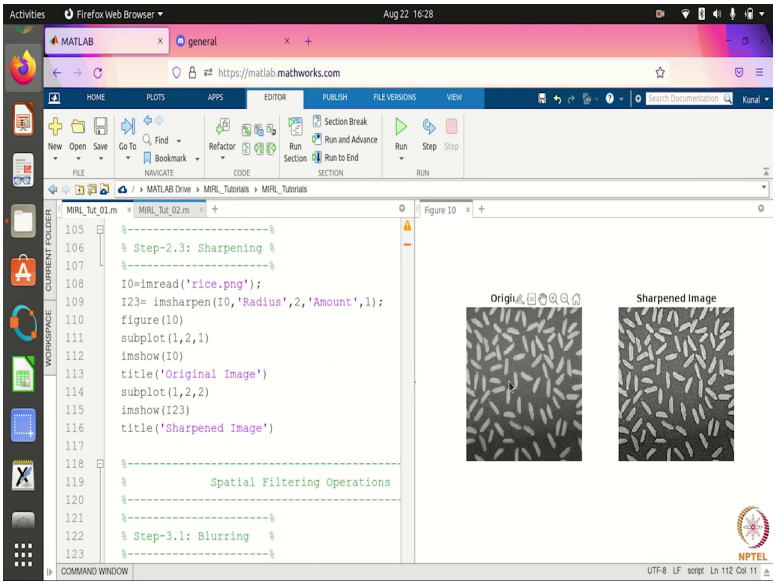
(Refer Slide Time: 08:43)



The next operation is contrast. So, you can control the contrast of an image by using various algorithms. Here I am showing just 3 of them; others are also available in MATLAB. So, these 3 algorithms are "imadjust" and histogram equalization and adaptive histogram equalization. So, it is "imadjust", "histeq" and "adapthisteq". So, you have to supply the initial image I0 and just run this function "imadjust". It will give you the image with more contrast.

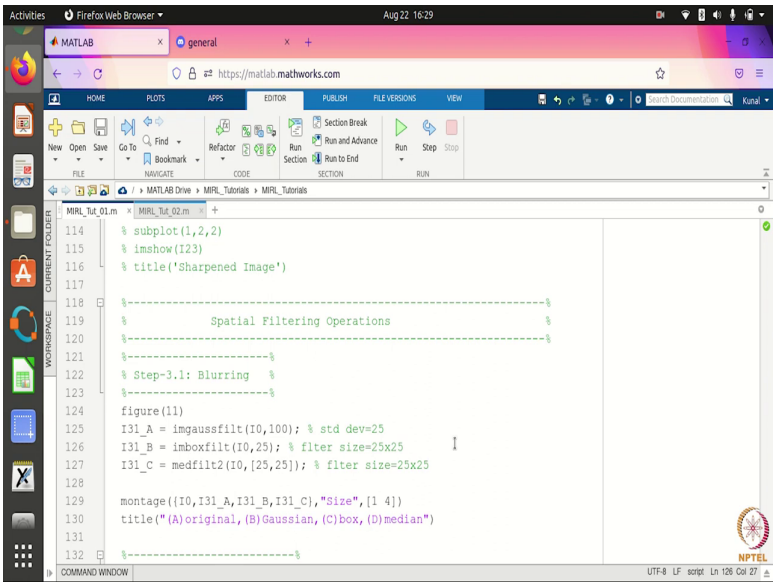
So, when you run this command when you run these commands you get the following. So, here the contrast is improved by three algorithms "iamadjust", "histogram equalisation" and "adaptive histogram equalisation". This is the original one and the other images with more contrast. So, other than contrast you can also sharpen an image with the help of command "imsharpen".

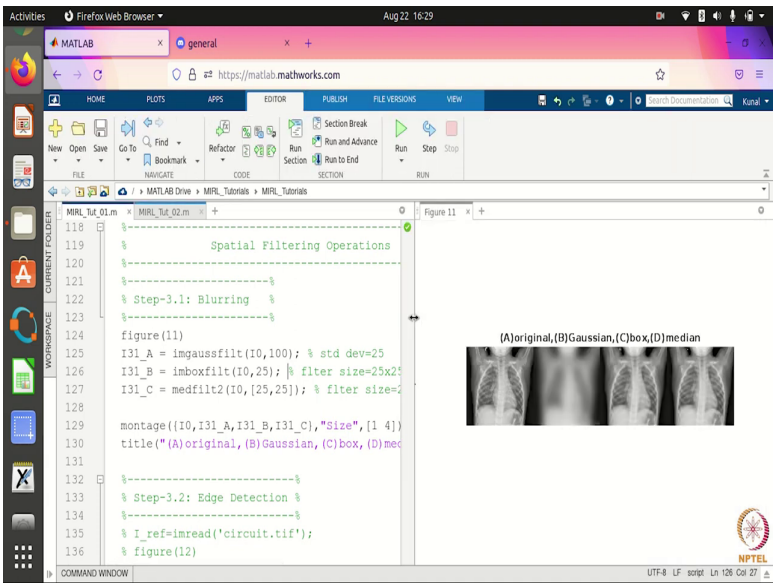
(Refer Slide Time: 10:18)



So, if we run this command "imsharpen" you will see that this is the original image and this is the sharpened image. So, clearly sharpened images better. So, these are some of the image enhancement operations that you can do in MATLAB you can control the contrast you can sharpen an image you can increase the brightness decrease the brightness etcetera here.

(Refer Slide Time: 11:07)

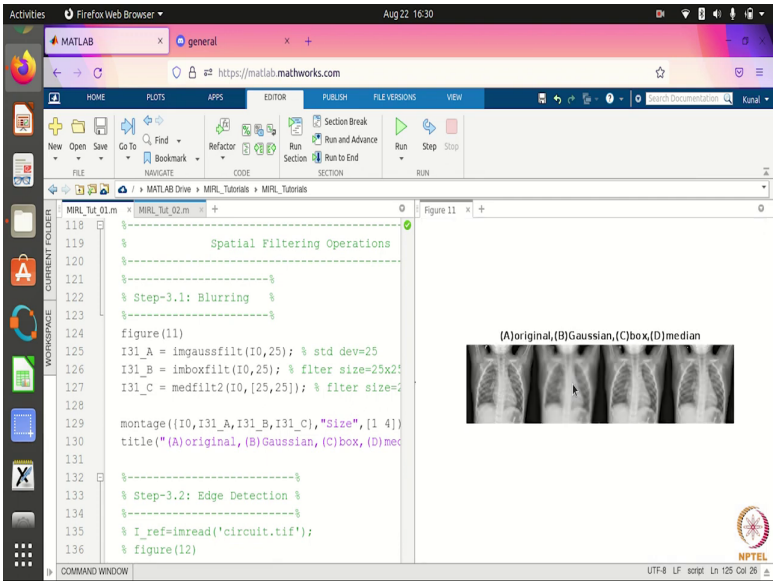




Now, we come to the third part. Some special filtering operations so the first operation is blurring. So, you can blur an image by using multiple options available in MATLAB we are showing you blending with three methods. These methods are gaussian filter, box filter and median filter. So, the parameter in gaussian filter is the standard deviation so you can control the amount of that by playing with this parameter.

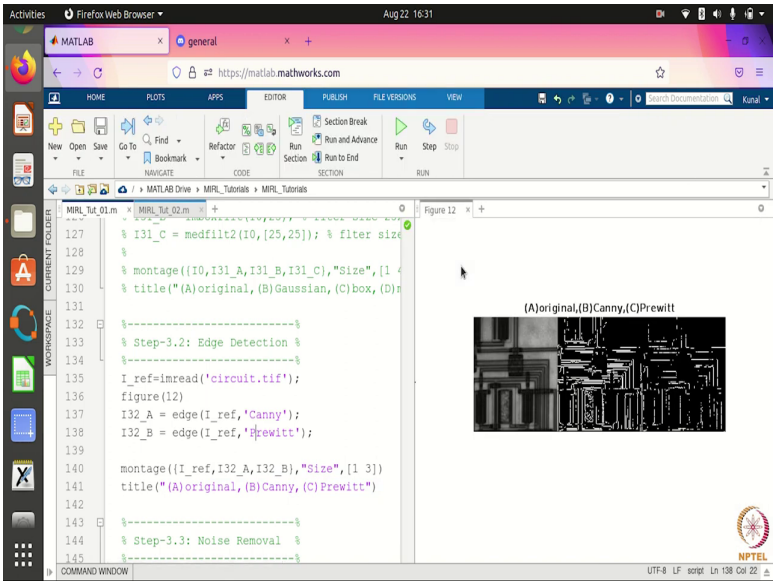
In box filter and median filter, you can specify the filter size. So, you can go to the MATLAB website and explore more about these filters. So, for these settings if I apply these filters on the I0, I will get the following. So, I get the full line you can see this is original image this is gaussian blurring and then in to box filter and medium filter.

(Refer Slide Time: 12:29)



See here if I change it from 100 to 25 the blurring will decrease and here you see the ring is decreased. So, this is about blurring.

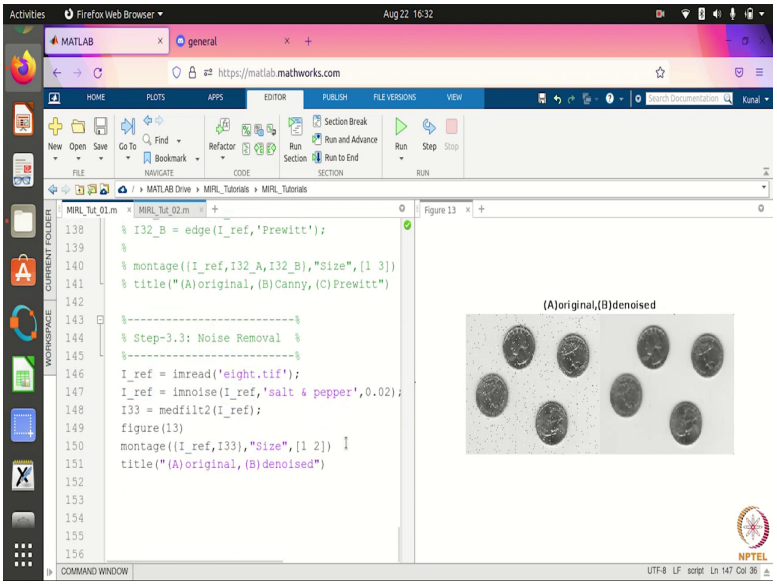
(Refer Slide Time: 12:51)



Another very important operation is edge detection again you can use multiple filters for this but I am showing you only two filtering operations. First is based on Canny algorithm second is based on Prewitt code filter not a good filter first is Canny filter second is Prewitt filter. So, there are other filters also that you can test out. So, if I try to find the edge of an image using these two filters, we will get the following result.

So, this is an image it has a lot of horizontal and vertical edges. So, with the two filters we get the following output. Clearly, here the Canny one looks better.

(Refer Slide Time: 14:08)



So, the filters that we discussed earlier for example the median filter etc, they can be used for noise removal also. So, here is an example, I take an image and add some noise to it and then I will apply a median filter to remove that noise. So for example, here is a original image: an image with some black dots. As you can see these are the noise that I have added but when I give this noisy image to median filter it leads to this image.

You can see this the noise is gone. So, many filters can be used for noise removal as well. So, these are some of the examples and you can try many more things. So, this is just to motivate you to explore the image processing toolbox of MATLAB yourself and I hope that the things which I have shown will make you like feel motivated for that. Thank you.