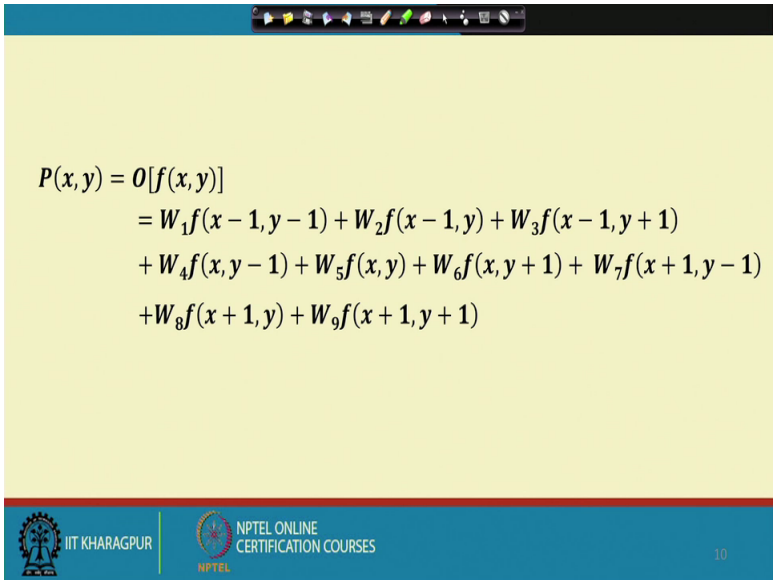
Robotics
Prof. Dilip Kumar Pratihar
Department of Mechanical Engineering
Indian Institute of Technology, Kharagpur

Lecture – 35
Robot Vision (Contd.)

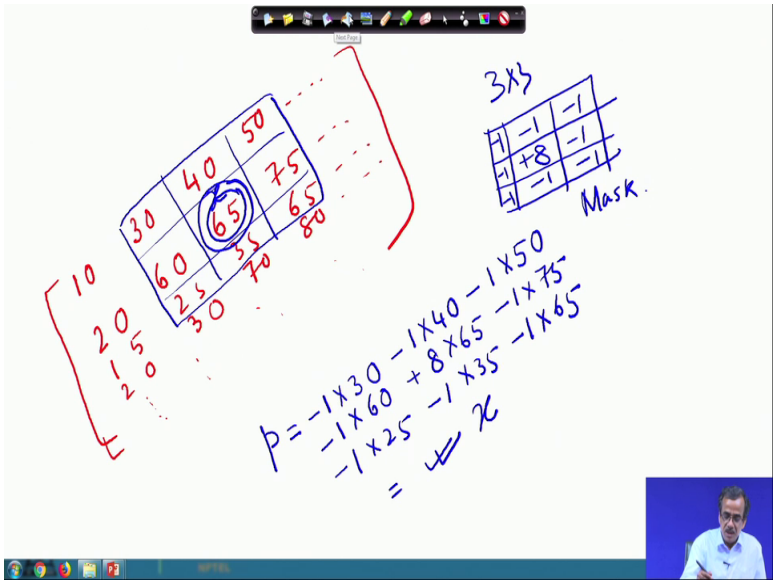
Now, I am going to solve one numerical example using the method of the masking.

(Refer Slide Time: 00:22)



Now, let me take one example one numerical example.

(Refer Slide Time: 00:34)



Supposing that we have got one the image and corresponding to this particular image. So, we have got some the light intensity values pixel wise; let me take at random some example say 10, 30, 40, 50 and there are some numerical values which I am not considering 20, 60, 65, 75; there are some numerical values then comes 15, 25, 35, 65 and there are some numerical values. Then comes 20, then comes 30, then comes 70, then comes 80; there are some numerical values and here also we have got a few other numerical values ok.

Now, here my aim is to find out what should be the preprocessed value so, corresponding to this particular any of this pixel? Now let me try to concentrate that I am just going to find out the preprocessed value corresponding to say this particular say 65 ok.

So, how to do it and supposing that I am just going to take the help of one mask like 3 cross 3 mask the way we discussed the 3 cross 3 mask is something like this. So, here we have got plus 8 minus 1 minus 1 minus 1. So, minus 1 minus 1 minus 1 minus 1 minus 1.

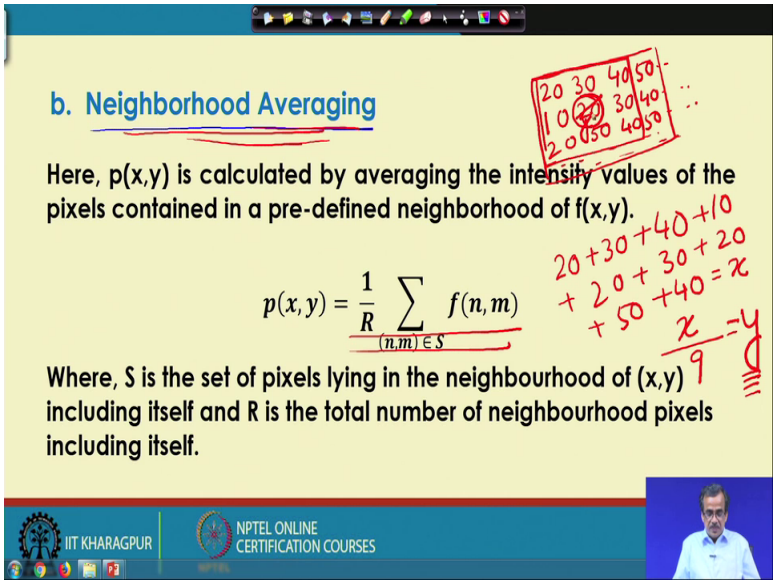
So, this is nothing, but a 3 cross 3 template or the mask. So, this is nothing, but a mask and our aim is to find out the preprocessed value corresponding to this. So, what I do is the center of the mask is made coincident with this particular 65. So, I can draw this particular template or the mask here ok.

Now, I just try to find out what should be the preprocessed value. So, p the preprocessed value; so here we can see that this is minus 1; so, if we just draw it here. So this particular template if we just draw it here. So, here we have got minus 1; so, minus 1 multiplied by 30, then comes minus 1 multiplied by 40, then comes minus 1 multiplied by 50, then comes your minus 1 multiplied by 60 plus 8 multiplied by 65 minus 1 multiplied by 75, then comes your minus 1 multiplied by 25, minus 1 multiplied by 35, minus 1 multiplied by 65.

Now, if I calculate; so, that will be the preprocessed value corresponding to this particular the pixel. So, in place of 65 supposing that I am getting x that integer value. So, I am just going to put this particular x in place of your 65. And this particular the further processing will be done with the help of; so, this particular the preprocessed value..

This is the way actually we do the masking; now for each of this particular pixel, so we will have to follow this particular the method of masking and purpose I have already told the purpose is nothing, but is your the purpose is nothing, but to remove that particular noise from the your the image.

(Refer Slide Time: 04:26)



Now, there is another method; so, this method is also very popular for preprocessing that is called the neighborhood averaging. Now here actually what you do is in neighborhood

averaging; so, what you do is. So, we try to find out your ah; so, this neighborhood averaging actually what you do?.

So, we try to concentrate on a particular neighbor we define a neighbor and we try to find out the average. Now let me take a very simple example supposing that say I have got supposing that I have got one image sort of thing. Now say this is something like this; so this is actually the light intensity values at the different pixel say; say let me consider 20 here, 30 here, 40 here, 50 here something like this and there are some numerical values 10, 20, 30, 40; there are some numerical values 20, 30, 40, 50 there are some numerical values and here also there are some numerical values.

Now, let us see how to find out, how to carry out this particular the neighborhood averaging. Now before we carry out this neighborhood averaging what you do is; we define the neighborhood first. For example, if I concentrate on this, say I will have to find out the neighborhood averaging value corresponding to this particular the 20.

So, I will have to define the neighborhood first; supposing that the neighborhood is nothing, but is 3 cross 3 neighborhood. So, if this is the 3 cross 3 neighborhood; so this is nothing, but the 3 cross 3 neighborhood ok. So, surrounding this actually we have got the 3 cross 3 neighborhood and once I have got defined this particular neighborhood; I know the light intensity values at the different pixels.

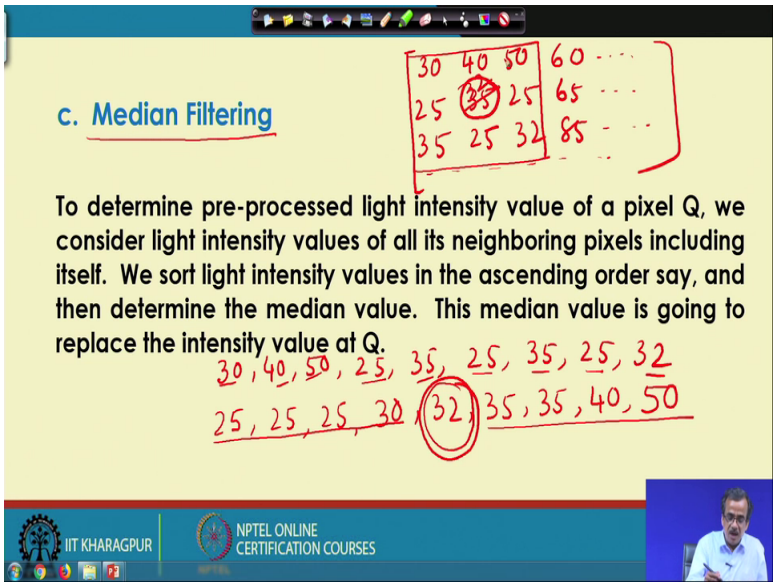
So, what you do is you sum them up all the light intensity values. So, that is 20 plus 30 plus 40 then comes your plus 10 plus 20 plus 30 plus 20 plus 50 plus 40. So, we try to find that and supposing that this is equals to x and how many entries are there? 1 2 3 4 5 6 7 8 9; so, x divided by 9. So, whatever value we get and its nearest integer will be nothing, but a numbers.

So, that is equal to say y and y is considered as the nearest integer. So, this particular y is going to replace that particular the 20; so, this is the method of neighborhood averaging. Now, mathematically actually it can be express something like this the sum of all the light intensity values contained in that particular neighborhood divided by the number of neighbors including that particular whose preprocessed value I am going to calculate ok. So, that is why I have divided by 9 by, but not 8.

So, this 1 by R multiplied by summation f n, m ok. So, this is nothing, but like how to determine the neighborhood for how to determine the average of this particular neighborhood. And this particular average is going to replace that light intensity value at that particular the pixel; now, this is also a very popular method for preprocessing.

Now this method is very simple ok. So, very easily we can implement; so, this particular the neighborhood averaging.

(Refer Slide Time: 08:37)



Now, I am just going to discuss another very popular the method of preprocessing and that is called the median filtering. Now if you see this particular the method of median filtering this is very simple. Now by median we mean like supposing there are 9 values; so, what I am going to do is 9 means there are odd number of values.

So, what I will do is I will leave the first 4 and the last 4 and I will try to find out what is there at the middle; if there are odd number if numbers. And if there are even number of number supposing that 8; so, what I will do is. So, first 3 and the last 3 we are going to remove and at the middle we have got 2 remaining because 3 plus 3; 6 plus 2 is 8. So, we try to find out the mean or the average of the middle 2; so, this is the way actually we calculate the median.

Let me take a very simple example the similar type of example; supposing that I have got an image whose light intensity values are nothing, but say 30, 40, 50, 60, 25, 35, 25, 65

and there are some other values here. Then comes 35, 25, then comes your 32, 85; there are some other values and here also there are some other values ok.

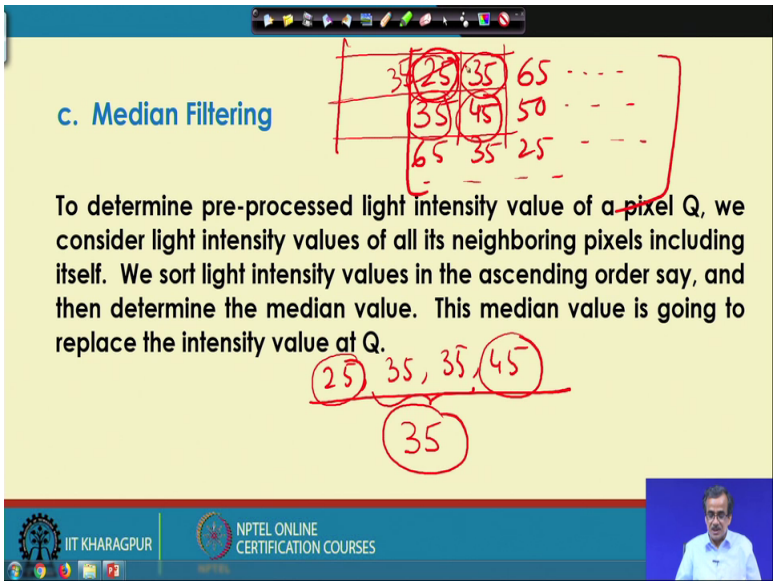
Now, once again we will have to define one the neighbor and supposing that I am just going to find out what should be the preprocessed value corresponding to this particular the 35. Now if I want to find out what should be the value corresponding to this particular 35? So, what you do is we define the neighborhood first; so, this is actually the 3 cross 3 neighborhood. And in the neighborhood we have got all the light intensity values like 30, 40, 50, 25, 35, 25, 35, 25, 32 ok.

So, the light intensity values let me write it here. So, we have got 30, then comes 40, then comes your 50, then comes 25, 35, then comes 25 then comes your 35, 25 and 32. So, there are 9 numbers 9 values 1, 2, 3, 4, 5, 6, 7, 8, 9; so, what you do is we sort them in the ascending order ok. Now if I want to sort them in the ascending order. So, what I will have to do? I will have to find out the lowest value here the lowest value is 25 and there are 3 such entries of 25.

So, let me write here 25, 25, 25 then comes your; so, 25; above 25 we have got 30 here. So, let me put 30 here then comes your 32 here, 32 here, then comes 35; 35 and here also we have got 35, then we have got 40 and after that we have got 40. So, all such 9 values these are sorted in the ascending order ok. So, lowest to the highest and there are 9 values.

So, there are odd number of values; so, what I do is. So the first 4 you neglect; so, the first 4 you neglect, the last 4 you neglect and whatever is there at the middle; so, that is 32 is the median value corresponding to that your the 9 values. So, this is the median; so we are going to replace. So, this particular 35 by this particular your 32; that is what you do in your median filtering ok.

(Refer Slide Time: 13:18)



Now, let me take another; another example similar type of example, but slightly different ok. Now supposing that I have got; so, this type of number say 25, 35, 65 and there are something say I have got 35, 45 and 50 and something then there are 65 35, 25 and something and something. So, this is nothing, but the image and supposing that I will have to find out what should be the preprocessed value according to the median filtering corresponding to this particular 25?

So, what I will have to do is. So, 3 cross 3 neighborhood if you define. So, this is the way we can define for example, say. So, this is the way we can define; so, here we have get 3 and here also we can write 3 here. So, those things are missing. So, only thing we have got only this 4 numbers 25, 35, 35 and 45 are you getting my point? And other things are 0s ok.

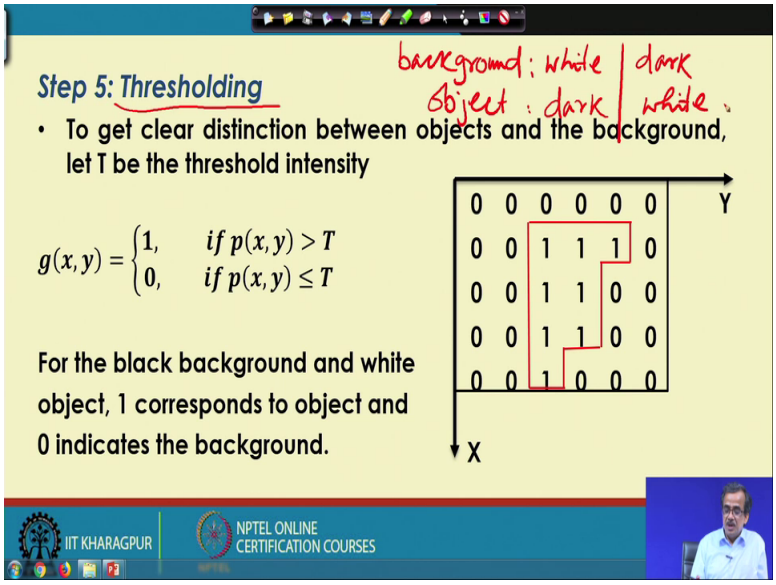
So, what will have to do it here? So, we will have to concentrate only on your. So, this particular the 4 values; that means, the values which you have here are 25, then comes 35, then comes 35, then comes 45 and they are in actually the ascending order. Now, here there are even number of numbers ok; so, what I will have to do is. So, this I will have to leave this I will have to leave and as there are even number.

So, I will have to find out the mean or the average of 35 and 35 and that is also equal to 35. So, this 35 is going to replace that particular the 25 as the preprocessed value and following the same method; so I can find out the preprocessed value corresponding to

each of this particular the pixels. So, this is the way actually we can carry out your the preprocessing. And as I told the purpose is to remove the noise and to restore if there is any such the lost information.

So, these are the methods of the preprocessing which is generally used very frequently and once I have done; so, this particular preprocessing. So, we will be getting the preprocessed data and on the computer screen actually we have got that particular [mat/matrix] matrix, the matrix of the light intensity values ok; now we will have to find out the difference between your the object and the background.

(Refer Slide Time: 16:13)

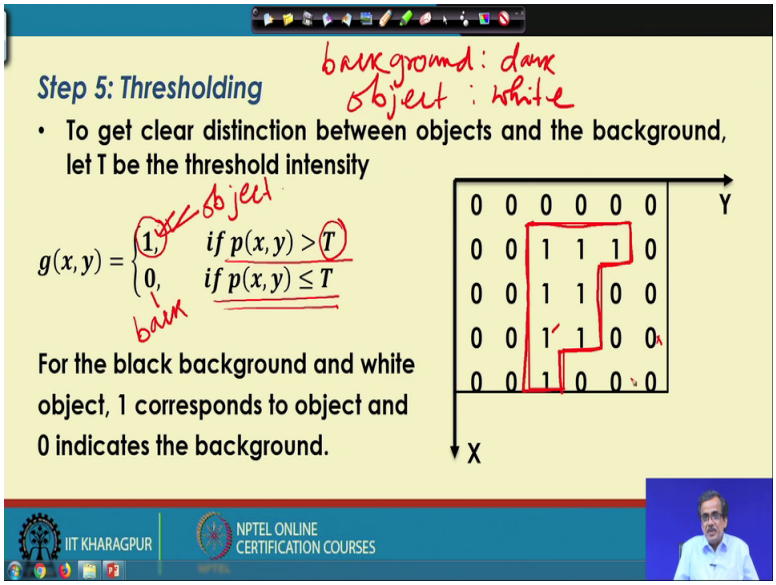


Now, as I told that here I am just going to concentrate only on the black and white picture sort of thing for simplicity ok. Now here we will have to take the help of one operator that is called the thresholding operator. So, using this particular thresholding operator; we can find out the difference between the object and this particular the background. Now there could be 2 possibilities for example, say the of the background; so, in black and white. So, this particular background; so, the background and this object, the possibilities are the background could be white and your the object could be your the dark ok.

And there is another option the background could be dark and this particular object could be the white ok. So, there are 2 possibilities and both the possibilities can be solved very easily using the principle of your the thresholding. So, let us see let us see the principle

of this particular thresholding. So, purpose is to find out the difference between your the object and this particular the background.

(Refer Slide Time: 17:47)



Now, let me consider a particular special case like the background, let me consider the dark background say the dark background. And your the object is nothing, but the white let us see what happens. White object means what? The light intensity value will be more and dark background means there the light intensity values will be less.

Now, here for this thresholding; so, we will have to defined one threshold value of light intensity that is denoted by T. Now here I am writing after thresholding; supposing that I am getting say g xy. So, initially we had f xy is the light intensity value, then we converted into p xy that is the preprocessed value and now I am just going for g xy after the application of this particular the thresholding. And I am just going to consider white object and dark background; that means, on the object the light intensity value will be more.

Now, here I am just putting the condition that if this particular p xy is found to be greater than the threshold value which is predefined by the user; then it will generate 1 and as I told that object is white. So, its light intensity value is more; that means, ones is going to indicate the presence of that particular the object. And on the other hand if p xy is found to be less than equals to T that is the threshold value. So, it is going to generate 0; so 0

means what? So, 0 means it is nothing, but the background and 1 means this is nothing, but the object.

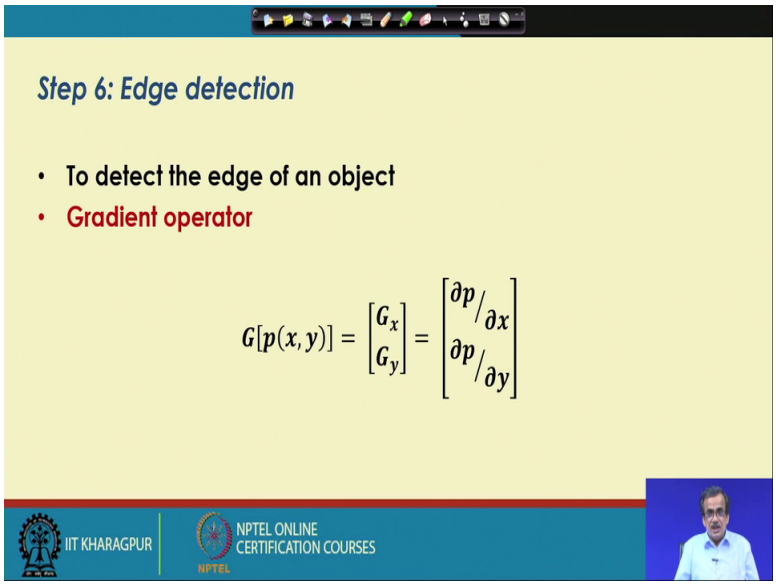
Now, previously on the computer screen; so, this is the computer screen. So, we had the matrix of light intensity values its preprocessed values. So, on the computer screen I could have seen that particular your matrix of the light intensity values. Now if you use this thresholding; suddenly we will find that on this particular computer screen, there will be a collections of 1s and 0s ok. So, there will be a connection of 1s and 0s and as I told that 1 indicate the this is the object and 0 indicate the this is nothing, but the background.

Now, if I get; so here all such a 1s I get here; then what will happen is your. So, I can just find out one boundary; so, this is nothing, but the boundary and this boundary will be the approximate boundary of this particular object on 2 D view. So, this is nothing, but and approximate the boundary for the object and 0 indicates that this is nothing, but the background.

So, on the computer screen; so, corresponding to that particular object ok; so you will be getting actually some sort of approximate picture of the objects something like this and it is in 2 d ok. So, this type of picture will be getting by using the operator like your the thresholding, this is the purpose of your the thresholding.

Now, the reverse is also possible for example, say if I consider your; the black object and your the white background that is also possible. And accordingly I will have to actuallychange this part just to take care of that and that is also possible ok; so, I can just consider the reverse also.

(Refer Slide Time: 22:04)



Now, once I have got this particular object what is the next task? The next task is how to identify or how to actually detect that particular edge that is the edge between the object and this particular the background. So, my aim is to detect this particular the edge of this particular your the object.

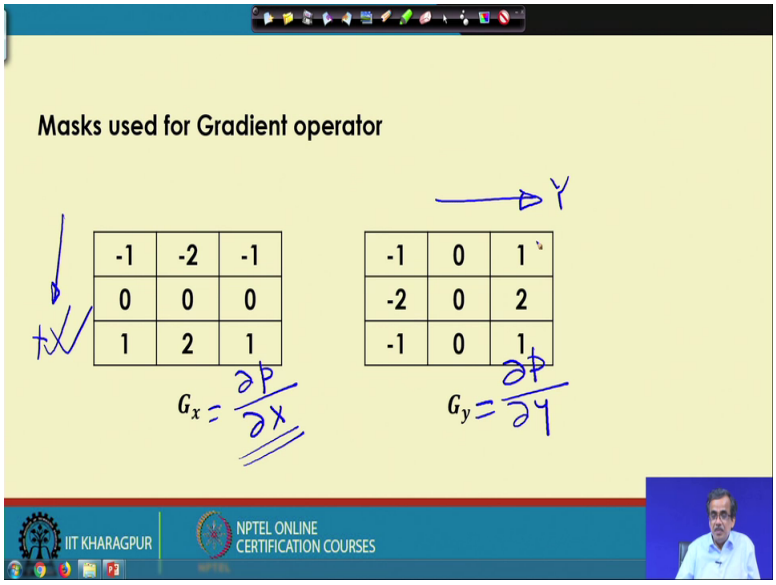
Now, let us see how to detect that particular edge? That is nothing, but the edge detection that is step 6. So, in step 6 we try to take the help of some sort of edge detection technique. Now edge detection techniques are nothing, but the gradient operator; by gradient we mean the rate of change. So, on this particular boundary, so the rate of change will be very prominent and that is why to detect the edge the edge between the object and the background we take the help of the gradient operator.

For example say. So, this particular gradient operator is very popular just to find out the difference between the object and this particular your the background; that means, to identify the edge. So, this gradient operator is working on say this particular the your the light intensity values and that is nothing, but G x, G y and G x is nothing, but the partial derivative of p with respect to x and partial derivative of y with respect to y is nothing, but actually the G y.

So, we will have to find out the partial derivative just to detect that particular the edge. Now derivative means computationally expansive and all such things ultimately we will have to write in the computer program and in the computer program if you want to carry

out. So, this type of derivative it will be computationally very expansive and that is why to carry out this particular, the derivative we take the help of some sort of templates and those templates are nothing, but so, this type of template.

(Refer Slide Time: 24:28)



For example, say if we want to carry out the G x that is nothing, but the partial derivative of p; the preprocessed light intensity with respect to x; so, this type of 3 cross 3 mass or the template we use. And here you can see that so, this is the positive x direction say..

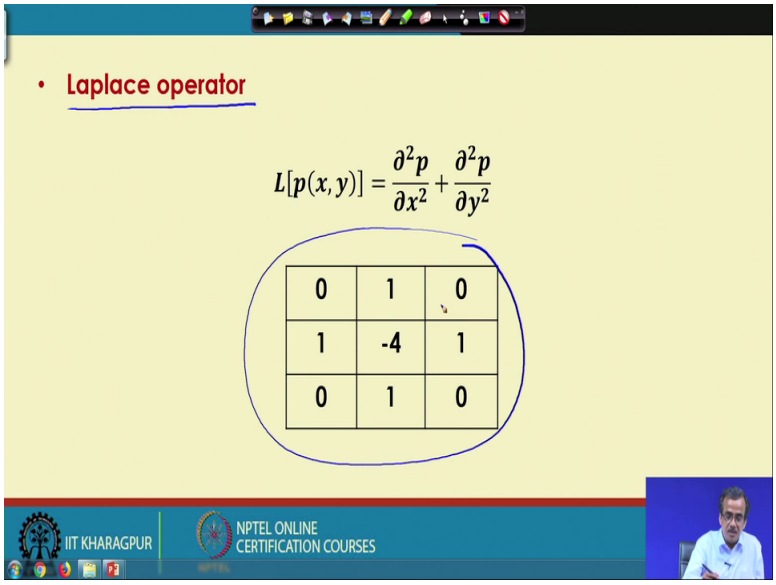
Now along this positive x direction we can see that there is change in sign for example, we have got minus 1 then 0 then plus 1. So, from minus 0 plus there is change in sign whereas, along this particular y direction; there is no change in sign. So, this is minus minus minus 0 0 0 plus plus plus ok. So, this is the way actually we try to design this particular your the mask to carry out this particular G x.

Now, here once again if we just add the mass coefficient. So, this will become equal to 0 for example, minus 1 minus 2 minus 1. So, we have got minus 4 plus 0 plus 4; so, this will become equal to 0. Now, similarly we can also find out; so this particular G y and that is nothing, but the partial derivative of p with respect to your this y and once again this is the positive y direction and along this particular y direction there is change in sign.

So, minus 1 0 1, minus 2 0 2, minus 1 0 1 and along this particular x direction; there is no change in sign. So, keeping that in mind actually we try to design; so this type of

mask or the template just to carry out. So, this type of your the derivative; so, this is actually the derivative which is very frequently used just to implement the gradient operator in the computer program.

(Refer Slide Time: 26:42)



Now, there is another operator which is also very frequently used that is called the Laplace operator. And here actually we generally go for the second order derivative. So, this L on p x y is nothing, but del 2 p, del x 2 plus del 2 P del y 2. And once again as I mention that if I want to implement on the computer program that will become computationally very expensive and that is why to implement this Laplace operator in the computer program. So, what you can do is like we will have to use some mask or the template and this is a very typical template used for Laplacian operator.

So, at the middle we have got minus 4 and on the horizontal side we have got plus 1 plus 1, vertical side plus 1 plus 1 and in there are 4 such neighboring 0s here. And the sum of the; so this particular coefficients values will be equal to 0. So, this is a very widely used operator that is called the Laplacian operator just to find out or just to indicate or identify or detect the edge of an object from the background.

Now, this is the way actually we can carry out some sort of edge detection and using this gradient operator particularly the Laplacian operator; we can find out the difference between the object and this particular background. And we can in fact, identify the edge between the background and this particular the object.

Thank you.