Digital Image Processing

Prof. P.K. Biswas

Department of Electronics & Electrical Communication Engineering

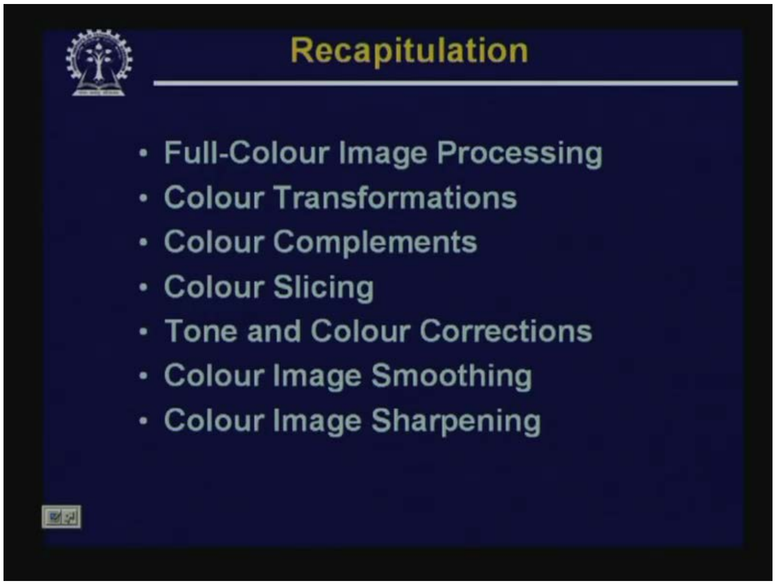
Indian Institute of Technology, Kharagpur

Lecture - 29

Image Segmentation - 1

Hello, welcome to the video lecture series on digital image processing. From today, we are going to start our discussion on image segmentation.

(Refer Slide Time: 1:06)

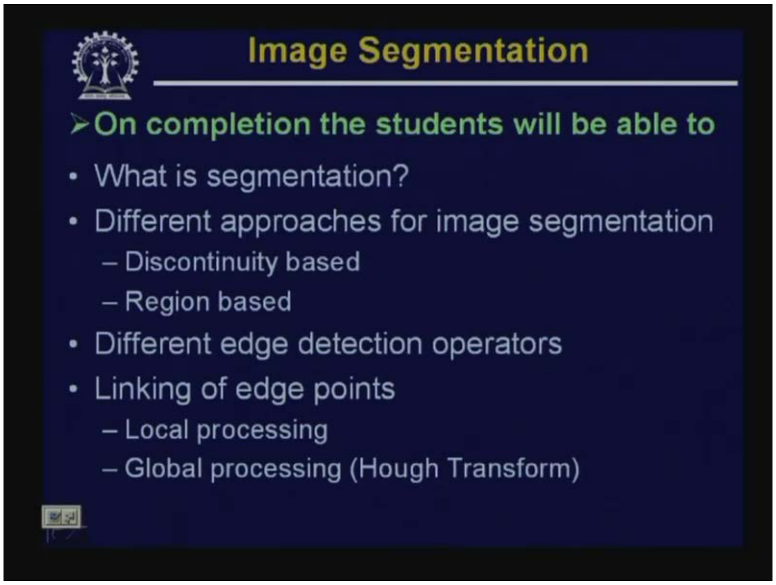


In the last class, what we have done is we have discussed about the colour image processing, we have talked about what is full colour image processing and under full colour image processing techniques, we have discussed about various topics like colour transformations, colour complements or colour negatives, we have talked about colour slicing, we have talked about what is meant by tone of a colour image and how can we go for tone and colour corrections in case there is any defect in the image tone. Then, we have also talked about processing techniques like colour image smoothing and we have also talked about how to sharpen a particular colour image.

Now, you find that till now, the type of discussions that we have done with respect to our digital image processing; there our intention was to improve the quality of the image as far as the visualization is concerned. Say for example, we have tried to sharpen the image, we have tried to enhance the image, we have tied to reduce the noise component in an image. So, in all these different techniques that we have discussed; our input was an original image and the output was a processed image. But our aim was that the processing techniques should be such that the output image is better visually than the input image.

Now today, when we start our discussion on image segmentation, we are going to talk about or we are going to start our discussion on another domain of image processing which is called image analysis. So here, our aim will be to extract some information from the images so that those informations can be used for high level image understanding operation.

(Refer Slide Time: 3:14)



So, in today's discussion, we will see what is image segmentation, we will talk about what are different approaches of image segmentation and we will see that image segmentation is mainly categorized into one of the two categories; the segmentation is either discontinuity based or the segmentation is region based.

Then, we will talk about different edge detection operations and these edge detection operations are useful for discontinuity based image segmentation technique. Then we will see that how to link those edge points which are extracted through different images, through different edge detection operators so that we can get a meaningful edge and under this linking of edge points, we will talk about two specific techniques; one is the local processing technique, other one is the global processing or Hough transformation based technique.

Now, let us see what is meant by image segmentation. By image segmentation, what you mean is a process of subdividing an image into the constituent parts or objects in the image. So, the main purpose of subdividing an image into its constituent parts or objects present in the image is that we can further analyze each of these constituents or each of the objects present in the image once they are identified or we have subdivided them. So, each of this constituents can be analyzed to extract some information so that those informations are useful for high level machine vision applications.

Now, when you say that segmentation is nothing but a process of subdivision of an image into its constituent parts; a question naturally arises that at which level this subdivision should stop? That is, what is our level of segmentation? Naturally, the subdivision or the level of subdivision or the level of segmentation is application dependent. Say for example, if we are interested in detecting the movement of vehicles on a road; so on a busy road, we want to find out what is the movement pattern of different vehicles and the image that is given that is an aerial image taken either from a satellite or from an from a helicopter.

So in this particular case, our interest is to detect the moving vehicles on the road. So, the first level of segmentation or the first level of subdivision should be to extract the road from those aerial images and once we identify the roads, then we have to go for further analysis of the road so that we can identify every individual vehicle on the road and once we have identified the vehicles, then we can go for vehicle motion analysis.

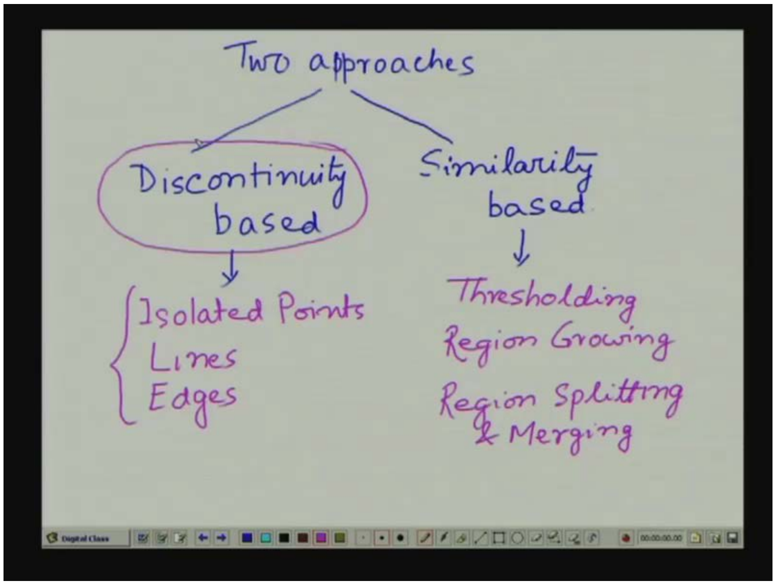
So here, you find that in this particular application, though an aerial image will contain a large area; many of the areas will have information from the residential complexes, many of the areas will have information of water bodies, say for example a sea or river or a pond, many of the areas will contain information of agricultural lands but our application says that we are not interested in water bodies, we are not interested in residential areas, neither we are interested in agricultural lands. But we are only interested in the road segment and once we identify the road segment, then we have to go further subdivision of the road so that we can identify each and every vehicle on the road.

So, as I said that our subdivision of an image at the first level should stop after we are able to extract or identify the road component, the road segments and after that we have to subdivide the road component to identify the vehicles and we need not go for segmentation of the vehicle in its constituent parts because that is not of our interest. Similarly, we should not or we need not segment or analyze the residential complexes or water bodies or agriculture lands for further subdivision into its constituent parts.

So, as we said that this segmentation or level of subdivision is application dependent; now for any automated system, what we should have is automatic processes which should be able to subdivide an image or segment an image to our desired level. So, you will appreciate that image segmentation is one of the most important task in machine vision applications. At the same time, image segmentation is also one of the most difficult tasks in this image analysis process and we will easily appreciate that the success of the image analysis operations or machine vision applications is highly dependent on the success of the autonomous segmentation of objects or segmentation of an image.

So, this image segmentation, though it is very difficult but it is a very very important task and every machine vision application software or system should have a very very robust image segmentation algorithm. So now, let us see that what are the different image segmentation algorithms or techniques that we can have.

(Refer Slide Time: 9:40)



Now, as we have just mentioned that image segmentation approaches are mainly of two different types, so we have two different approaches of image segmentation; one of the approach as we have just said is the discontinuity based approach and the second approach is what is called similarity based approach.

In discontinuity based approach, the partition or subdivision of an image is carried out based on some abrupt changes in intensity levels in an image or abrupt changes in gray levels of an image. So, on the discontinuity based approach, our major interest, we are mainly interested in identification of say isolated points or identification of lines present in the image or identification of edges. So, under discontinuity based approach, we are mainly interested in identification of isolated points or identification of lines or identification of edges.

In the similarity based approach, the approach is slightly different. Here, what we try to do is we try to group those pixels in an image which are similar in some sense. So, the simplest approach under this similarity based technique is what is called thresholding operation. So, by thresholding what we mean is as we have already said that if we have images where every pixel is coded with 8 bits, then we can have intensities varying from 0 to 255 and we can decide a threshold following some criteria, say we decide that we will have a threshold level of say 128; so we decide that all the pixels having intensities of having an intensity value greater than 128 will belong to some region whereas all the pixels having intensity values less than 128 will belong to some other region. So, this is the simplest thresholding operation that can be used for image segmentation purpose.

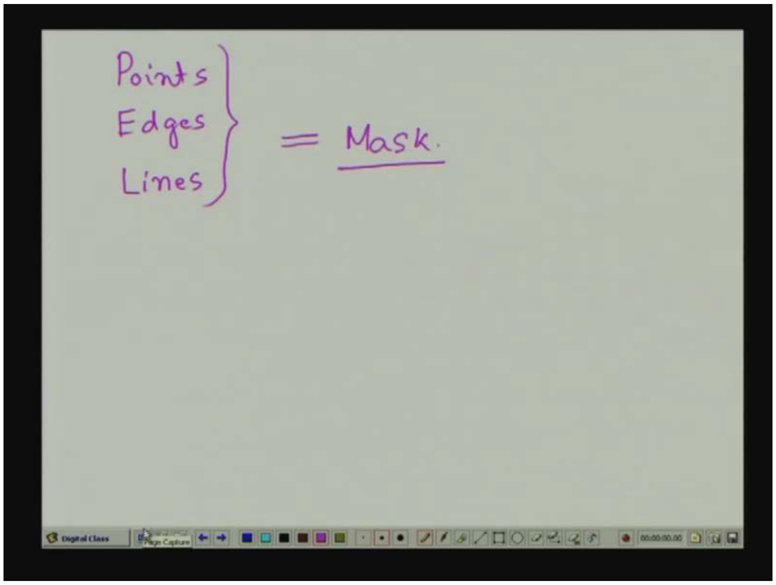
The other kind of segmentation under this similarity based approach can be a region growing based approach. Now, the way this region growing stuff works is suppose we start from any particular pixel in an image, then we group all other pixels which are connected to this particular pixel. That means the pixels which are adjacent to this particular pixel and which are similar in intensity value.

So, our approach is that you start from a particular pixel and all other pixels which are adjacent to this particular pixel and which are similar in some sense; in the simplest cases, similar in some sense means we say that the intensity value of that adjacent pixel is almost same as the intensity value of the pixel from where we have started growing the region. So, starting from this particular pixel, you try to grow the region based on connectivity or based on adjacency and similarity. So, this is what is the region growing based approach.

The other approach under this similarity based technique is called region splitting and merging. So, under this region splitting and merging, what is done is first you split the image into a number of different components following some criteria and after you have split the image into a number of smaller size sub images or smaller size components, then you try to merge some of those sub images which are adjacent and which are similar in some sense.

So, your first approach is the first operation is you split the image into smaller images and then try to merge those smaller sub images wherever possible to have a larger segment. So, these are the different segmentation approaches that we can have and in today's discussion and in subsequent discussion, we will try to see details of these different techniques. So first, we will start our discussion on this discontinuity based image segmentation approach.

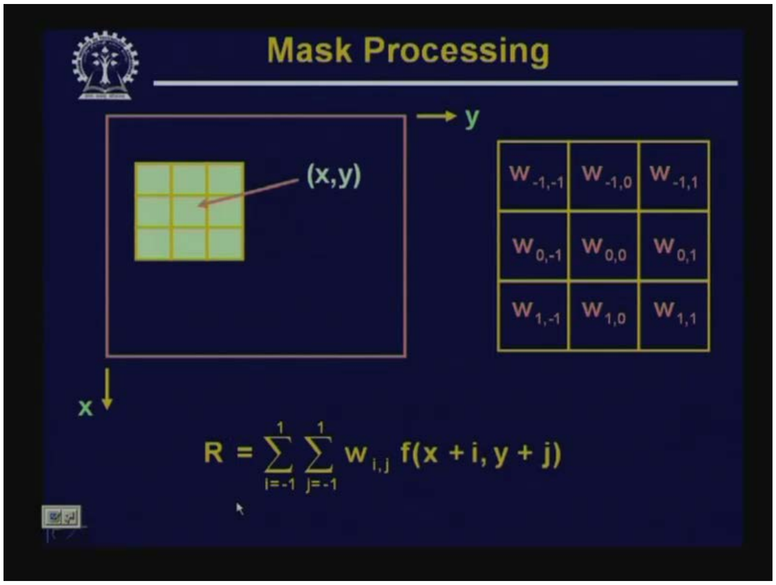
(Refer Slide Time: 15:45)



So, as we have already said that in discontinuity based image segmentation approach, our interest is mainly to identify the points isolated points or we want to identify the edges present in the image or we identify try to identify the lines present in the image and for detection of these kind of discontinuities that is either detection of points or detection of lines or detection of edges, the kind of approach that will take is use of a mask.

So, using the masks, we will try to detect isolated points or we will try to detect the lines present in the image or we will try to detect the edges in the image.

(Refer Slide Time: 16:43)

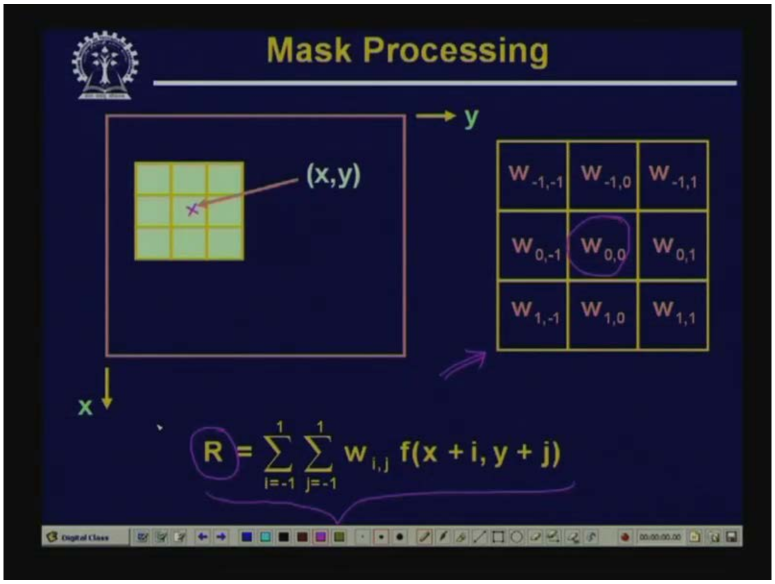


Now, this masks, use of masks, we have discussed earlier in connection with our discussion of image processing like image smoothing, image sharpening, image enhancement and so on. So there, we have said that if I consider a 3 by 3 neighborhood like this, we take a mask of size 3 by 3. So here, on right hand side, this is a mask of size 3 by 3 having different coefficient values given as $W_{\text{minus 1, minus 1}}$ $W_{\text{minus 1, 0}}$ $W_{\text{minus 1, minus 1}}$ and so on taking the center coefficient in the mask having a value $W_{0, 0}$.

Now, in this mask processing operation, what is done is you shift this mask over the entire image to calculate some weighted sum of pixel at a particular location. Say for example, if I place this mask at a location (x, y) in our original image; then using all other different mask coefficients, we try to find out and weighted some like this - R equal to W_{ij} into f Xi Y (x plus i, Y i plus j) where i varies from minus 1 to 1 and j varies from minus 1 to 1 and this component, we call as a value R. Use of this mask as I have said that we have seen in connection with image sharpening while we have taken different values of the coefficients.

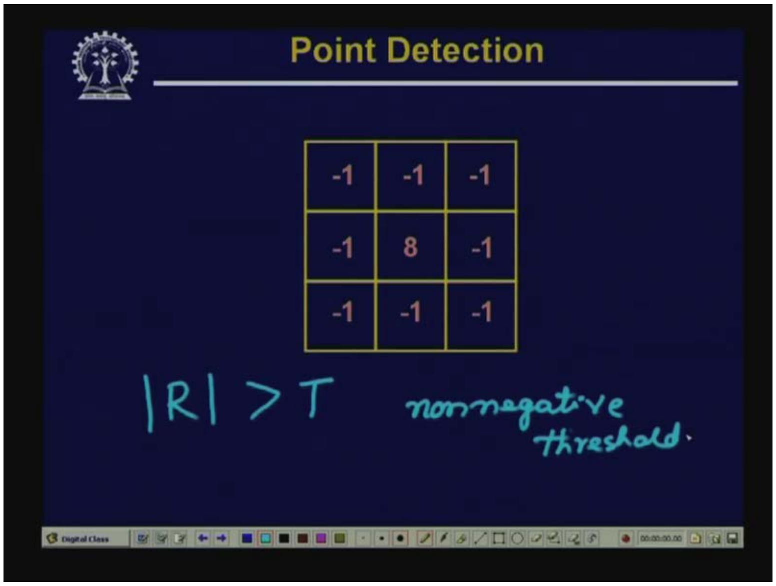
In case of image smoothing, we have taken the values of the mask coefficients to be all ones. So, that leads to an image averaging operation. So, depending upon what are the coefficient values of this mask that we choose, we can have different types of image processing operations.

(Refer Slide Time: 18:53)



Now here, you find that when I use this mask, then depending upon the nature of the image around point (x, y), I will have different values of R.

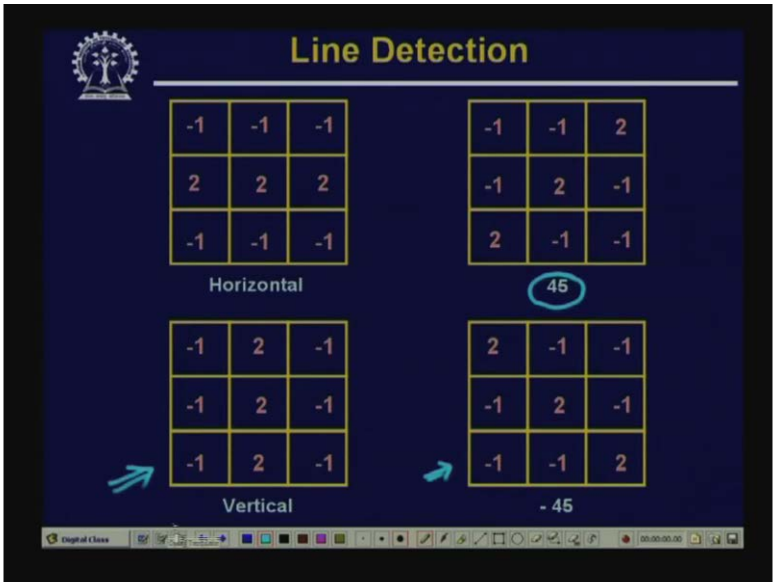
(Refer Slide Time: 19:10)



So, when it comes to an isolated point detection, we can use a mask having the coefficient values like this that the center coefficient in the mask will have a value equal to 8 and all other coefficients in the mask will have a value of minus 1. Now, we say that a point is detected at a location say (x, y) in the image where the mask is centered if the corresponding R value, we are computing the value of R; so, we say that a point is located at location (x, y) in the original image if the corresponding value of R, the absolute value of this is greater than certain threshold say T where this T is a non negative threshold value, this is a nonnegative threshold.

So, if the value of R computed at location (x, y) where this mask is centered is the absolute value of R is greater than T where T is a nonnegative threshold, then we say that a point, an isolated point is detected at the corresponding location (x, y). Similarly for detection of lines, the mask can be something like this.

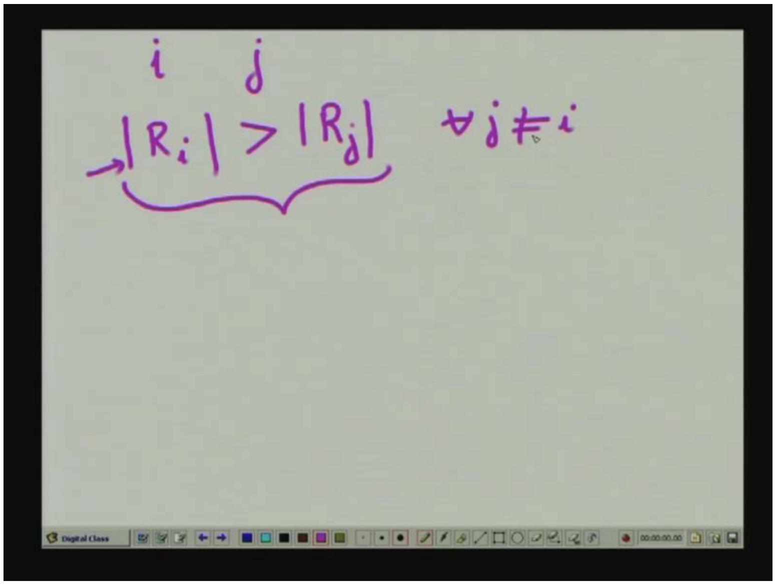
(Refer Slide Time: 20:50)



Here, for detection of horizontal lines, you find that you have used a mask at the center row or the middle row having all values equal to 1 and the top row and the bottom row is having all values equal to minus 1, all the coefficient values equal to minus 1 and by moving this mask over the entire image, it detects all those points which lies on a horizontal line.

Similarly, the other mask which is marked here as 45, if you move this mask over the entire image, this mask will help to detect all the points in the image which are lying on a line which is inclined at an angle of 45 degree. Similarly, this mask will help to detect all the points which are lying on a line which is vertical and similarly this mask will detect all the points lying on a line which is inclined at an angle of minus 45 degree. Now, for line detection, what is done is you apply all these masks, all these 4 masks on the image.

(Refer Slide Time: 21:11)



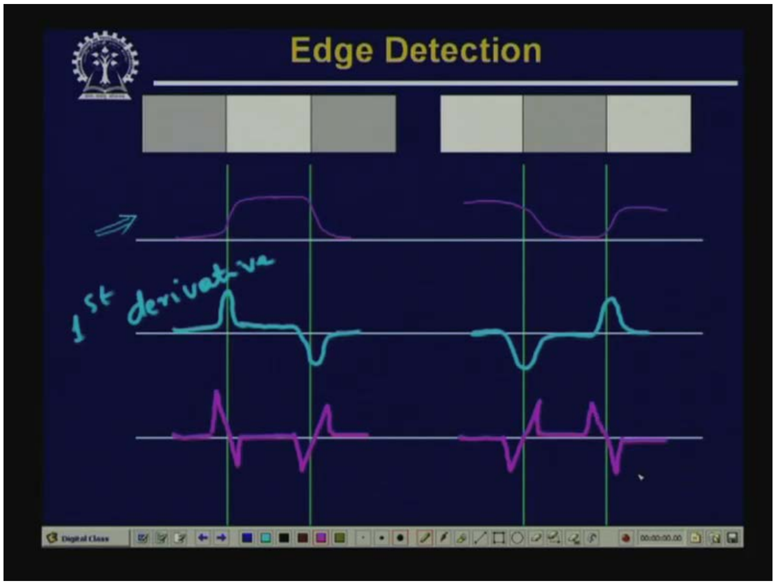
And, if I take a particular masks say, i'th mask and any other mask say j'th mask and if I find that the value computed are i with the i'th masks, the absolute value of this is greater than R_j where R_j is the value computed with the j'th mask for all j which is not equal to i. This says that the corresponding point is more likely to be associated with the line in the direction of the mask i.

So, as we said what we are doing is we are taking all the 4 masks, apply all the 4 masks on the image, compute the value of R for all these masks; now if for an i'th mask if I find that R of i, the absolute value of R_i is greater than absolute value of R_j for all j which is not equal to i, in that case we can conclude that this particular point at which location this is true, this point is more likely to be content on a line which is in the direction of mask i.

So, these are the 2 approaches; the first one we have said, given a mask which is useful for identification of isolated points and the second set of masks is useful for detection of points which are lying on a straight line. Now, let us see that how you can detect an edge in an image.

Now, edge detection is one of the most common approaches, most commonly used approach for detection of discontinuity of an image in an image. So, we say that an edge is nothing but a boundary between 2 regions having distinct intensity levels or having distinct gray level. So, it is the boundary between 2 regions in the image, these two regions have distinct intensity levels. So, as is shown in this particular slide.

(Refer Slide Time: 24:47)

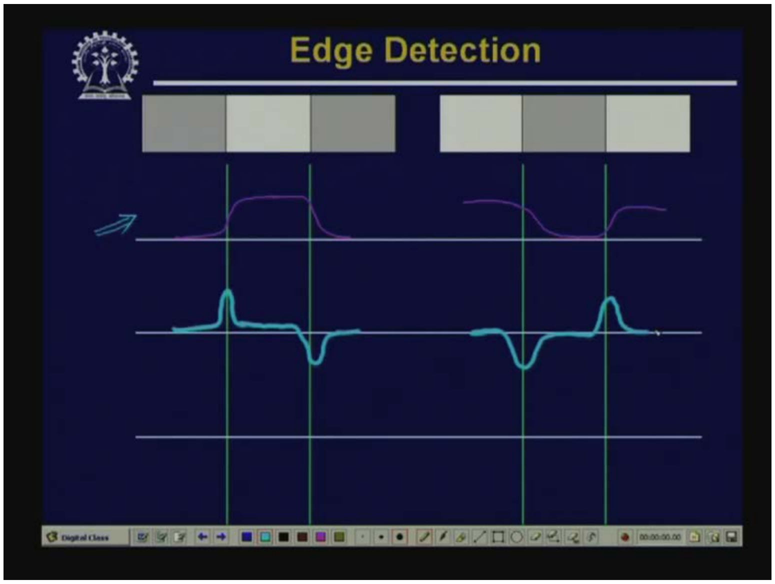


So, here you find that on the top, we have taken 2 typical cases. In the first case, we have shown a typical image region where we have a transition from a dark region to a brighter region and then again to a dark region. So, as you move from left to right, you find that you have transitions from dark to bright, then again to dark and in the next one, we have a transition as we move from left to right in the horizontal direction, there is a transition of intensity levels from bright to dark and again to bright.

So, these are the typical scenarios in any intensity image where we will have different regions having different intensity values and an edge is the boundary between such regions. Now here, in this particular case, if I try to draw the profile, intensity profile along a horizontal line; you find that here the intensity profile along a horizontal line will be something like this. So, you have a transition from dark region to bright region, then from bright region to dark region whereas in the second case, the transition will be in the other direction; so, bright to dark and again to bright.

So here, you find that we have modeled this transition as a gradual transition, not as an abrupt transition. The reason is because of quantization and because of sampling; all almost all the abrupt transitions in the intensity levels are converted to such gradual transitions.

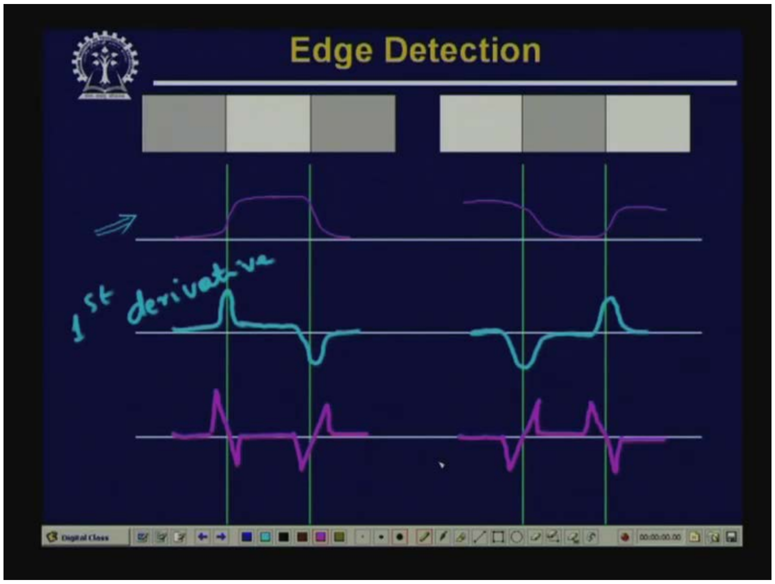
(Refer Slide Time: 26:51)



So, this is your intensity profile along a horizontal scan line. Now, let us see that if I differentiate this, if I take the first derivative of this intensity profile; then the first derivative will appear like this. In the first case, the first derivative of this intensity profile will be something like this and the first derivative of the second profile will be something like this.

So, you find that the first derivative response whenever there is a discontinuity in intensity levels that is whenever there is a transition from a brighter intensity to a darker intensity or wherever there is a transition from the darker intensity to a brighter intensity.

(Refer Slide Time: 27:54)

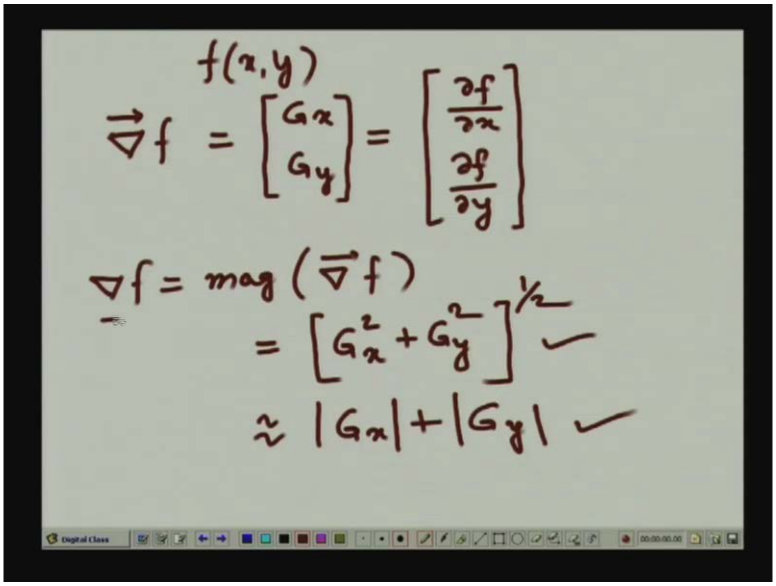


So, this is what we get by first derivative. Now, if I do the second derivative, the second derivative will appear something like this and in the second case, the second derivative will be just the opposite; it will be something of this form, it will be like this. So, you find that first derivative is positive at the leading edge whereas it is negative at the tailing edge; similarly here and you find that the second derivative if I take the second derivative, the second derivative is positive on the darker side of the edge and it is negative on the brighter side of the edge and that can be verified in both the situations that the second derivative is becoming positive on the darker side of the edge but it is becoming negative on the brighter side of the edge.

However, we will appreciate that this second derivative is very very sensitive to the noise present in the image and that is the reason that the second derivative operators are not usually used for edge detection operation. But as the nature says that we can use these second derivative operators for extraction of some secondary information that is we can use the sign of the second derivative to determine whether a point is lying on the darker side of the edge or a point is lying on the brighter side of the edge and not only that, here you find that there are some zero crossings in the second derivative and this zero crossing information can be used to exactly identify the location of an edge whenever there is a gradual transition of the intensity from dark to bright or from bright to dark.

So, it clearly says that using these derivative operators, we have seen earlier that the derivative operators are used for image enhancement to enhance the details filled in the image. Now, we see that these derivative operators operations can be used for detection of edges present in the image. Now, how to apply these derivative operations? So here, you find that if I want to apply the first derivative, then first derivative can be computed by using the gradient operation.

(Refer Slide Time: 30:46)



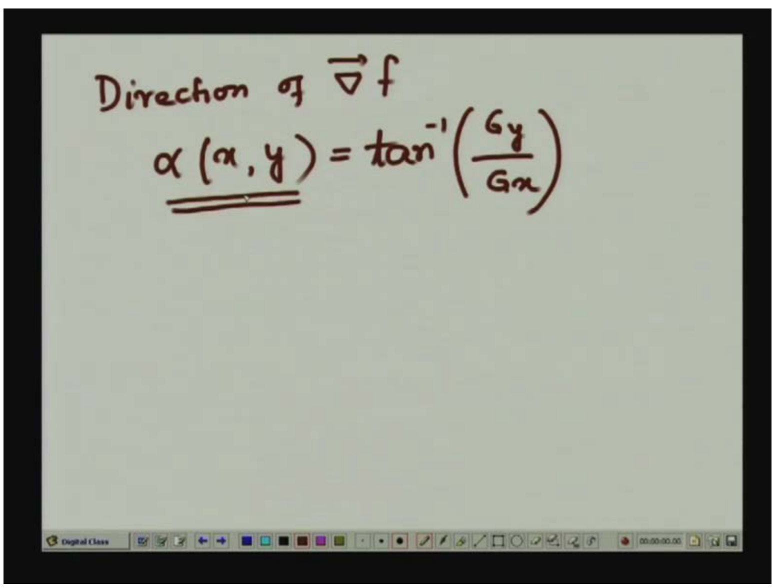
So, when I have an image say, f (x, y), I can define the gradient of this image if (x, y) in this form. So, gradient of this image f will be G_x G_y - a vector; obviously, the gradient is a vector, so it will be G_x G_y and this G_x is nothing but del f del x and del f del y which is the G_y. So, G_x is the partial derivative of f along x direction and G_y is the partial derivative of f along y direction. So, we can find out the gradient of the image f by doing this operation.

Now, for edge detection operation, what we are interested in is the magnitude of the gradient. So, the magnitude of the gradient; we will write like this - grad f which is nothing but magnitude of the vector grad f and which is nothing but G_x square plus G_y square and take the square root of it and you find here that computation of the magnitude involves squaring the 2 components G_x G_y adding them and then finally taking the square root of this addition.

Obviously, squaring and computing the square root; these 2 are computationally intensive process. So, an approximation of this is taken as magnitude of the gradient to be sum of magnitude of G_x that is gradient in the x direction plus magnitude of G_y that is gradient in the y direction.

So, this magnitude of the gradient whether I take this or an approximation that is to be this; this tells us what is the strength of the edge at location (x, y), it does not tell us anything about what is the direction of the edge at point (x, y). So, we have to compute the direction of the edge that is the direction of gradient vector f.

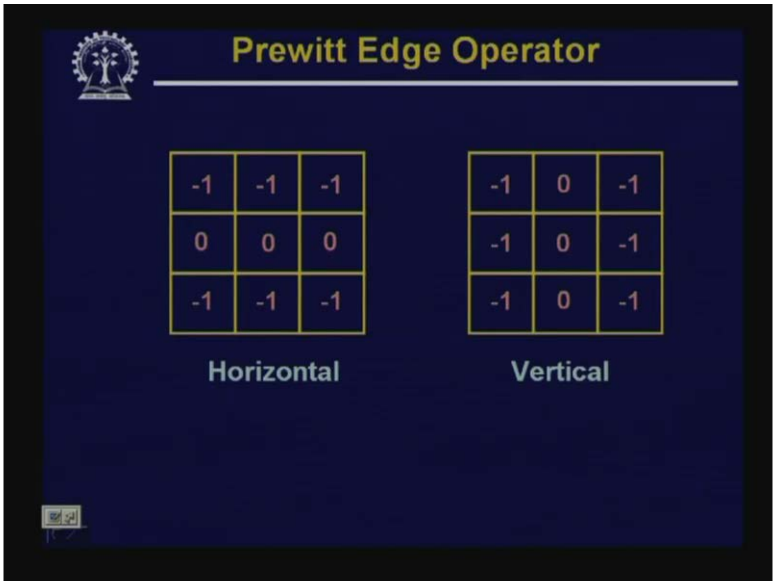
(Refer Slide Time: 33:24)



And, the direction of gradient vector f at location (x, y), we can write it as alpha (x, y) is equal to tan inverse G_y by G_x where G_y as we have said that it is gradient in the y direction and G_x is the gradient in the x direction. Now, you find that this alpha (x, y), it tells us what is the direction of gradient f that is a vector. But actually, x direction is perpendicular to the direction of the gradient vector f.

So, we have the first derivative operators or the gradient operators and using that gradient operators, we can find out what is the strength of an edge at a particular location (x, y) in the image and we can also find out what is the direction of the edge at that particular location (x, y) in the image and there are various ways in which this first derivative operators can be implemented and here we will show some operators, some masks which can be used to compute the image gradient.

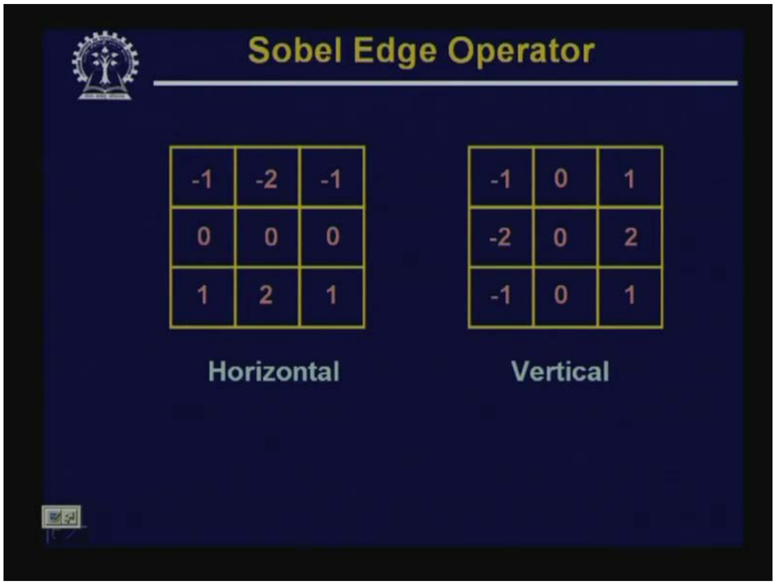
(Refer Slide Time: 35:02)



So, the first one that we are showing is called a prewitt edge operator. You find that in case of prewitt edge operator, we have 2 masks; one mask identifies the horizontal edges and the other mask identifies the vertical edges. So, the mask which finds out the horizontal edges that is equivalent to having the gradient in the vertical direction and the mask which computes the vertical edges is equivalent to taking the gradient in the horizontal direction.

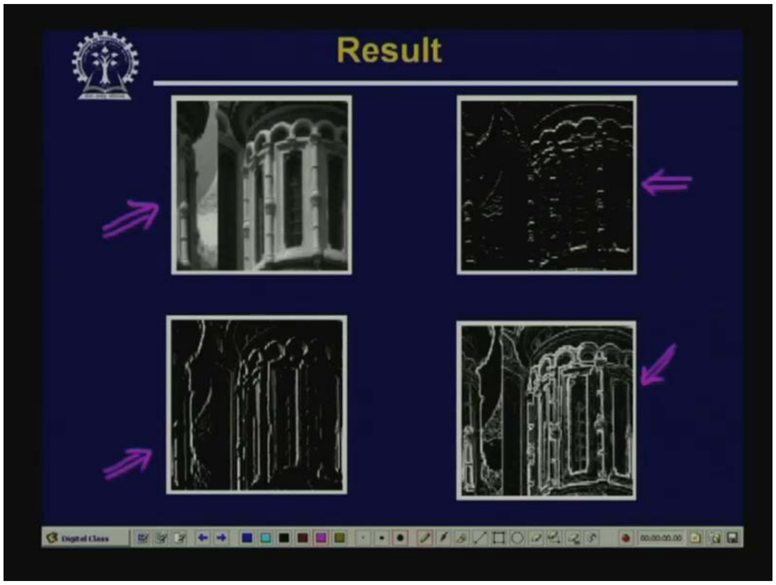
So, using these 2 masks, by passing these 2 masks over the intensity image, we can find out the G_x and G_y component at different locations in the image and once we compute the G_x and G_y, we can find out what is the strength of an edge at that particular location and what is the direction of an edge at that particular location.

(Refer Slide Time: 36:06)



The second mask which is also a first derivative mask is called a sobel operator. So, here again, you find that we have 2 different masks; one mask is responsible for computation of horizontal edges, the other mask is responsible for computation of the vertical edges. Now, if you try to compare this prewitt operator, prewitt edge operator and sobel edge operator; you find that this sobel edge operator gives an averaging effect over the image. So, because this sobel edge operator gives an averaging effect, the effect due to the presence of spurious noise in the image is taken care of to some extend by the sobel but it does not taken but it is not taken care of by the prewitt operator. Now, let us see that what kind of result we can use, we can have by using these edge detection operators.

(Refer Slide Time: 37:10)



You find that here we have shown result on a particular image. So, this one is our original image on the top left, this is our original image; on the top right, it is the edge information which obtained using the sobel operator and the edge components in this particular case are the horizontal components. The third image is again by using the sobel operator but here the edge components are the vertical edge components and the fourth one is the result which is obtained by combining this vertical component and the horizontal component.

So here, you find that if you compare this image with your original image, you find that different edges present in the original image, they have been extracted by using this sobel edge operator and by combining the outputs of the vertical mask and the output of the horizontal mask, we can have the edge components, I mean we can identify the edges which are there in various directions. So, that is what we have got in the fourth slide in the fourth image.

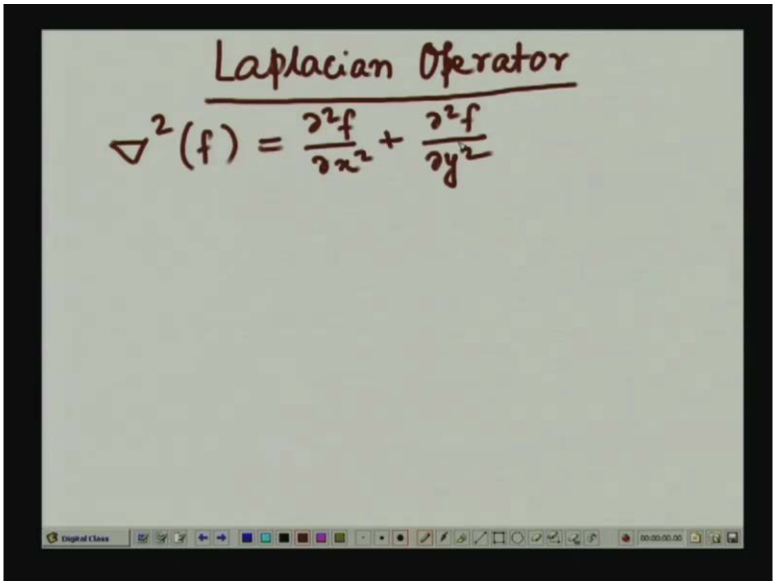
So, this prewitt operator and the sobel operator, as we have said that these 2 operators are basically first derivative operators and as we have already mentioned that for edge detection operation, the kind of operators derivative operators which are used are mainly the first derivative operators and out of these 2 - the prewitt and sobel operator; it is the sobel operator which is generally preferred because the sobel operator also gives an smoothing effect and by which we can reduce the spurious edges which can be generated because of the noise present in the image and we have also mentioned that we can also use the second derivative operator for edge detection operation but the disadvantage of the second derivative operator is it is very very sensitive to noise.

And secondly, as we have seen that second derivative operator gives us double edges. Once for every transition, we have double edges which are generated by the second derivative operators. So, that is these are the reasons why second derivative operators is not normally preferred for edge detection operation. But the second derivative operators can be used to extract the secondary information.

So, as we have said that by looking at the polarity of second derivative operator output, we can determine whether a point lies on the darker side of the edge or the point or a point lies on the brighter side of the edge and the other information that we can obtain from the second derivative operator is from the zero crossing, we have seen that second derivative operator always gives a zero crossing between the positive side and the negative side and the zero crossing points accurately determine the location of an edge whenever an edge is a smooth edge.

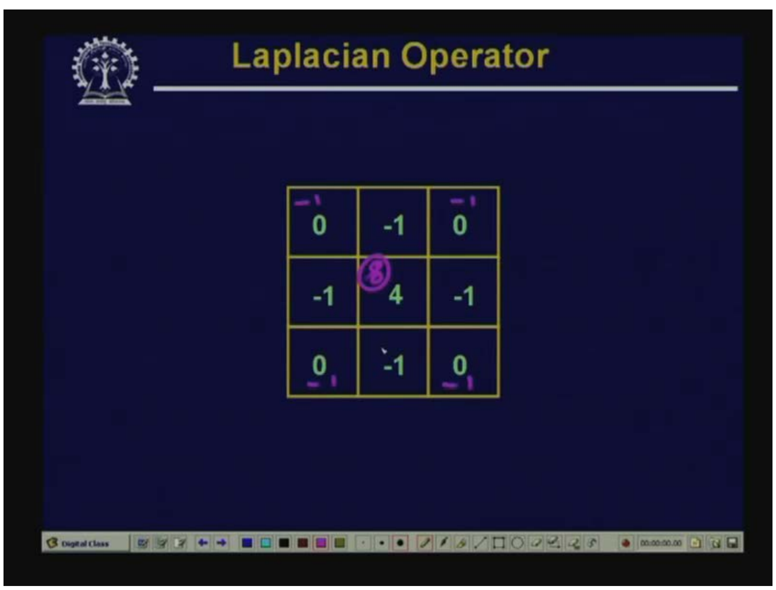
So, those second derivative operators are not normally used for this detection operation but they can be used for such a secondary information extraction. So, one such second derivative operator is what is called the Laplacian operator.

(Refer Slide Time: 41:00)



We have seen the use of Laplacian operator for enhancement of image details. Now, let us see that how this Laplacian operators can be used to help in edge detection operation and as you already know that the Laplacian operator of the function f is given by del 2 f del x 2 plus del 2 f by del y 2 where del 2 f del x 2 is the second derivative on an x direction and del 2 f del y 2 is the second derivative in the y direction.

(Refer Slide Time: 42:08)



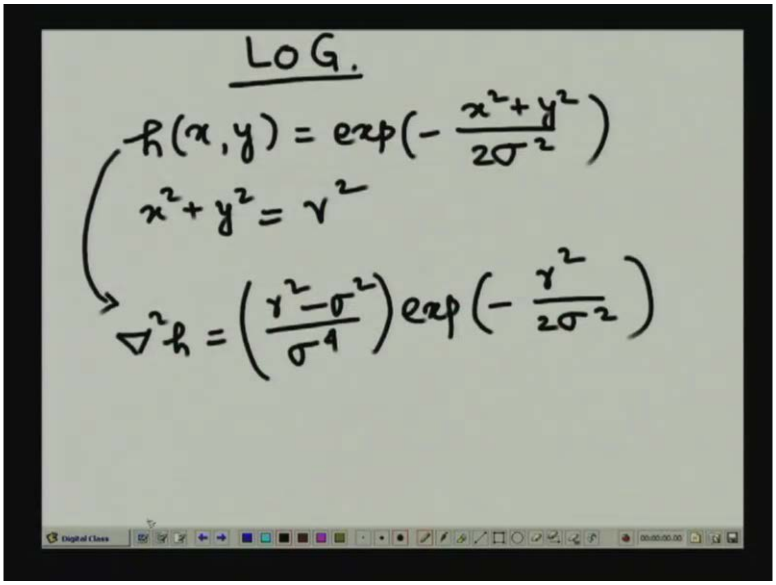
And we have also seen earlier that a mask which implements the second derivative operator is given by this where we are considering only the horizontal direction and the vertical direction for computation of the second derivative and we have also discussed earlier that if in addition to this horizontal and vertical directions, we also consider the diagonal directions for computation of the second derivative; in that case, the center element will be equal to 8 and all the diagonal elements will also be equal to minus 1.

So, this is the one that will get if we consider in addition to horizontal direction and vertical direction, the diagonal directions for computation of the second derivative and we can also have the inverse of this where all the negative signs will become positive and the positive sign will become negative. So, this is how we can have a mask for computation of the second derivative or computation of Laplacian of function f.

But as we have said that this Laplacian operator normally is not used for edge detection operation because it is very very sensitive to noise and secondly, it leads to double edges at every transition. But this plays a secondary role to determine whether a point lies on the bright side or a point lies on the darker side and it is also used to accurately locate or to accurately find out the location of an edge.

Now, along with this Laplacian operator, as we said that the Laplacian operator is very very sensitive to noise; to reduce the effect of noise what is done is the image is first smoothed using a Gaussian operator and that smooth image can now be operated by this Laplacian operator and these 2 operations can be used together to have an operator something like something which is called a Laplacian of Gaussian or LOG operator.

(Refer Slide Time: 44:34)

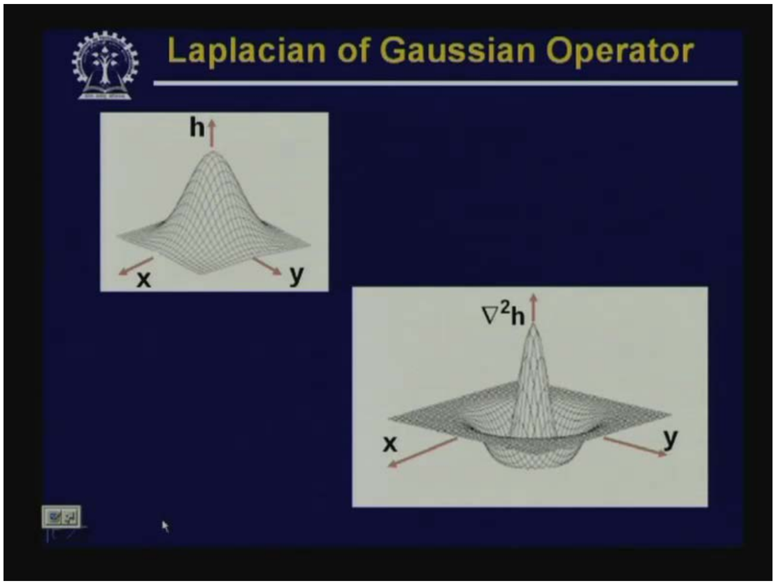


So, the essence of LOG or Laplacian of Gaussian operator - LOG that is Laplacian of Gaussian operator; we can have a Gaussian operator, the Gaussian can be represented by this say, h (x, y) is equal to exponent of minus x square plus y square upon twice sigma square. So, this is a Gaussian operator which is having a standard deviation of sigma.

Now here, if we let, x square if we set x square plus y square equal to r square; then the Laplacian of this h that is del square h can be written in the form - r square minus sigma square upon sigma to the power 4 into exponential of minus r square divided by 2 sigma square. So, as we said that our operation is firstly we want to smooth the image using the Gaussian operator and that smoothed image has to be operated by the Laplacian operator and if these 2 operations are done one after another, then this reduces the effect of the noise present in the image.

However, these 2 operations can be combined to have a Laplacian of Gaussian operation that means we can operate the image with the Laplacian of a Gaussian. So, Laplacian of a Gaussian operation on the image gives us an equivalent result.

(Refer Slide Time: 46:38)

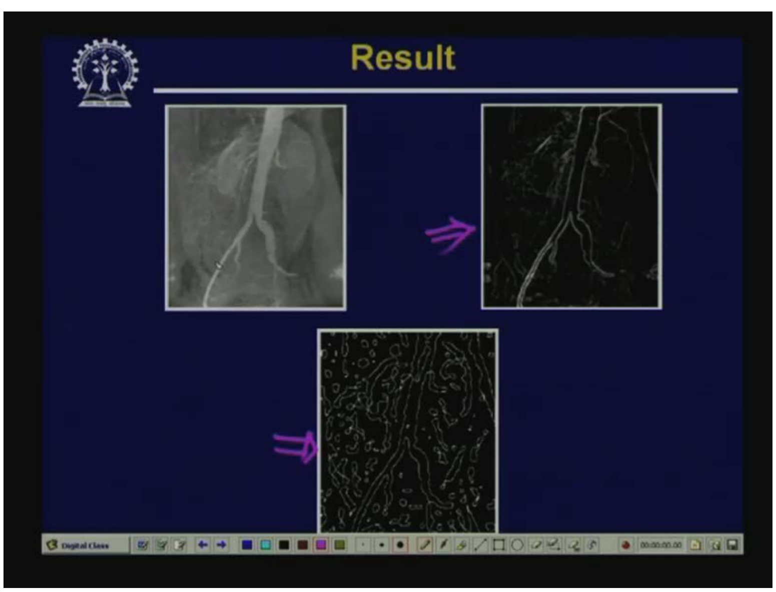


Now, we find that in this slide, we have shown that this is our Laplacian operator, this is a Gaussian mask in two dimensions and if I take the Laplacian of this; the Laplacian of the Gaussian will appear as shown here. Now, this Laplacian of Gaussian can again be represented in the form of a mask which is called a Laplacian of Gaussian mask. So, if I represent this Laplacian of Gaussian in the form of a two dimensional mask, the Laplacian of Gaussian mask appears like this.

(Refer Slide Time: 47:19)

So, here you find that our Laplacian of Gaussian mask or LOG mask that we have shown is a 5 by 5 mask and if you compare this with the LOG - the Laplacian of Gaussian expression on the surface; you find that here it says that at x equal to 0, LOG - the Laplacian of Gaussian is positive, then it comes to negative maximum negative, then tries to move towards a value 0.
And, the same is obtained using this particular mask that here you find that at the center, the value is maximum positive which is 16. Just away from this, it becomes minus 2, then it goes towards 0 that is it is becoming minus 1. So, if I apply this LOG - the Laplacian of Gaussian mask on an image, I can detect the location of the edge points.

(Refer Slide Time: 48:48)

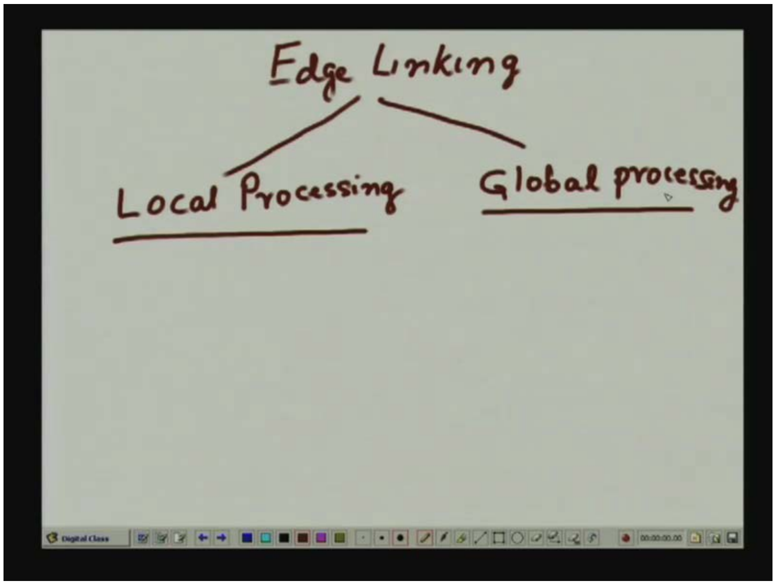


So, the location of the edge points, you find that here in this particular image, we have shown an image and on the right hand side, we have shown the output that is obtained using the sobel operator. So, this is the output which is used using the sobel operator and the bottom one, shows the output of the LOG operator.

So here, you find that all these bright edges, these are actually the location of the edges present in the original image. So, this establishes as we said earlier that LOG operator - the Laplacian of Gaussian operator can identify can determine what is the location of an edge present in an image. Now, whichever operator we use for detection of edges, whether these are the first derivative operators or the second derivative operators; as we said that the second derivative operators is not normally used for edge detection operation because of other problems but it is used to extract the secondary informations but the first derivative operators like sobel should ideally give us all the edge points that is any transition from a bright region to a darker region or from a darker region to a brighter region.

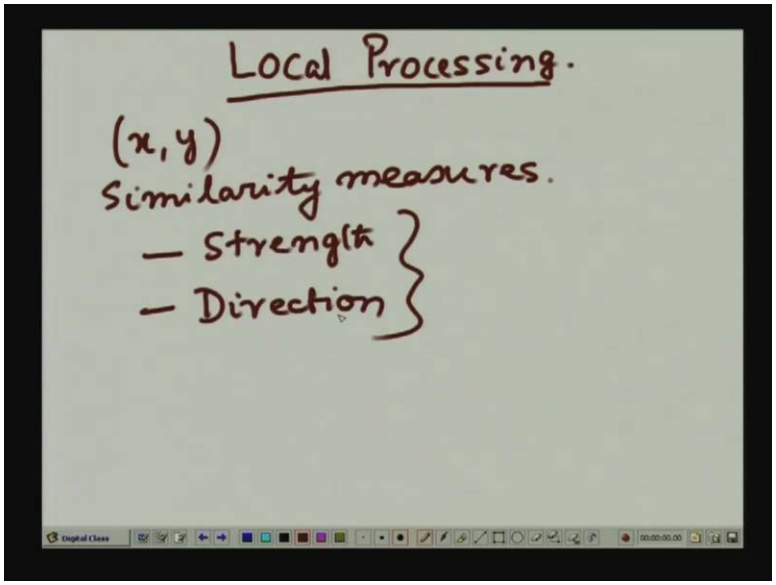
But you find that when you take an image, maybe it is because of the noise or may be because of non uniform illumination of the scene; when you apply the sobel operator to an image, the edges are not always connected, the edge points that you get, they are not always connected. So, what we need to do is we have to link the edge points to get some meaningful edges to extract some meaningful edge information. Now, there are usually 2 approaches in which this linking can be done.

(Refer Slide Time: 51:04)



So, for edge linking, we can have 2 approaches; one is called, one is the local processing approach and the other one is global processing approach. So, our aim is whether we are going for local processing or we are global processing, we are going for global processing; our aim is that we want to link all those edge points which are similar in some sense so that we can get a meaningful edge description. So first, we will talk about the local processing approach for edge linking.

(Refer Slide Time: 52:00)

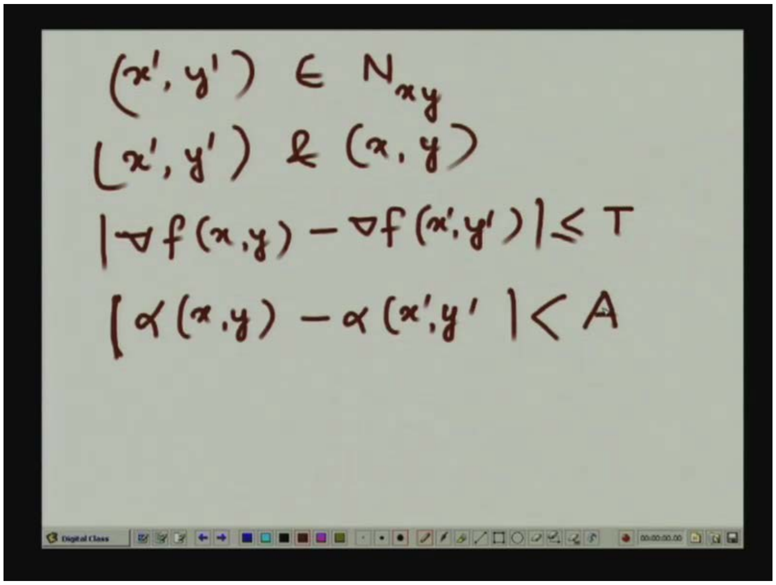


So first, let us talk about the local processing technique. In local processing technique, what is done is you take an image which is already edge operated. So, for edge operation if I assume that we are using the sobel edge operation; suppose the image is already operated by the sobel edge operator, then we consider say every point in that edge image if I call it as an edge image, I consider each and every point in the edge image.

So, I consider; let us take a point (x, y) in the image which is already operated by the sobel edge operator. Then, we will link all other points in that edge image which are in the neighborhood of (x, y) and which are similar to (x, y). So, when I say that 2 points are similar, we must have some similarity measure. So, you have to have some similarity measure.

So, for this similarity measure, what we use is the first one is the strength of the gradient operator and we also use the direction of the gradient. So, these two together are taken as similarity measure to consider whether will say that 2 points are similar or not.

(Refer Slide Time: 53:48)

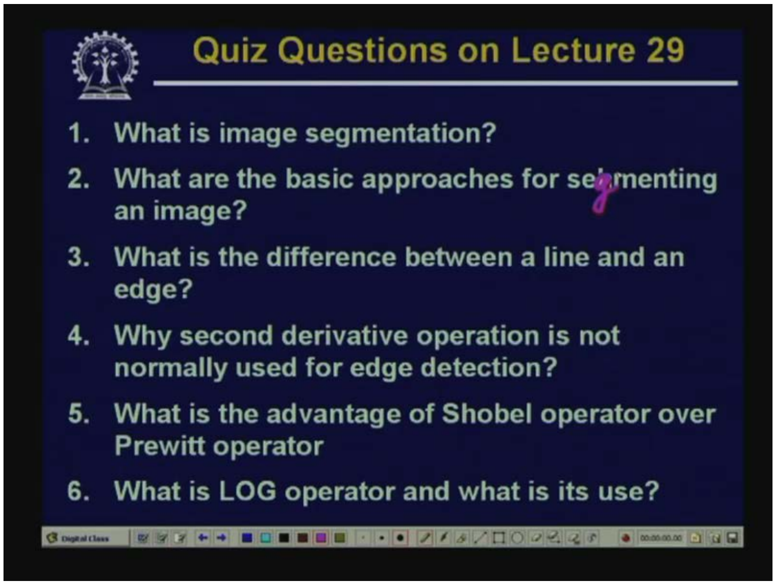


So, our operation will be something like this that we take a point say (x dash, y dash) which is in the neighborhood of some point (x, y) in the image and we say that these 2 points (x dash, y dash) and the point (x, y), they are similar if grad f (x, y) that is the strength of the gradient operator at location (x, y) and grad f (x dash, y dash), they are very close. That means this should be less than or equal to some non negative threshold T.

And, we also said that the directions should also be similar. That means alpha (x, y) minus alpha (x dash, y dash), this should be less than some angle threshold A. So, whenever we have a point (x dash, y dash) which is in some neighborhood of (x, y) and the points are similar that means they have the similar gradient magnitude value and the similar angle for the edge orientation; we say that these 2 points are similar and those points will be linked together and such operation has to be done for each and every other point in the edge detected image to give us some meaningful edge description.

So, let us stop our discussion at this point today. We will continue with our discussion in our next class. So now, let us see that some questions, some quiz question on today’s lecture.

(Refer Slide Time: 55:40)



The first question is what is edge what is image segmentation? Second question, what are the basic approaches for segmenting an image? sorry this should be segmenting. Third question is what is the difference between a line and an edge? Fourth question, why second derivative operation is not normally used for edge detection? Fifth question, what is advantage of sobel operator over prewitt operator? And the last question, what is LOG operator and what is its use?

Thank you.