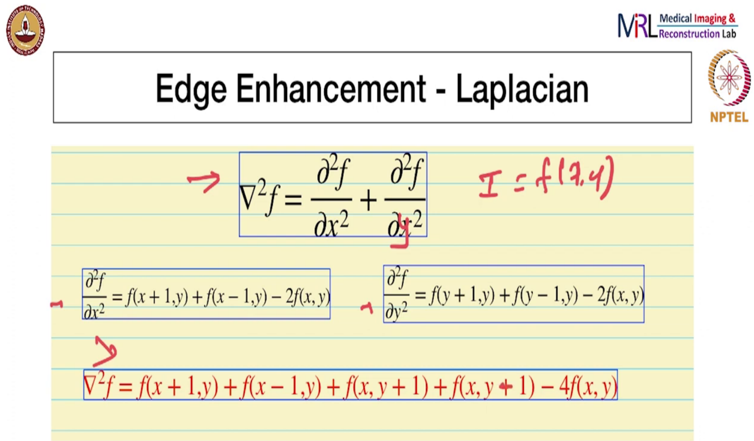
Medical Image Analysis
Professor Ganapathy Krishnamurthi
Department of Engineering Design
Indian Institute of Technology Madras
Lecture 09

Edge Enhancement – Laplacian

(Refer Slide Time: 0:15)



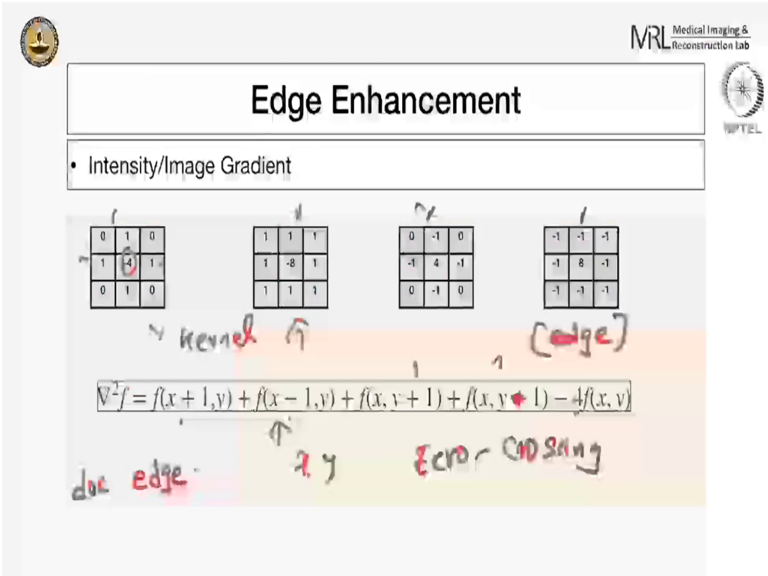
Hello and welcome back. So, in this next video we are going to look at some edge enhancement operators or edge detection operators specifically I will just say we look at the laplacian of an image as well as how we define gradients of an image. So, the first operator that we are going to look at it is called an operator that is a technical term for it is called the laplacian which is again defined for continuous functions f.

So, if you treat the image f I we can say we see we can say that I=f(x,y) and f(x,y)is of continuous function and the laplacian of f is given by this expression with there. And so, in order to make this work for images now images are discrete they are not continuous functions they are so, image is basically a 2d matrix in most cases.

The idea is how do we define this laplacian for images. So, for in discrete form we with our two directions. So, $\nabla^2 f$ where two del square f in terms of $\frac{\partial^2 f}{\partial x^2}$ as well as $\frac{\partial^2 f}{\partial y^2}$. So, the expressions for $\frac{\partial^2 f}{\partial x^2}$ and the $\frac{\partial^2 f}{\partial y^2}$ are both given here again this emerges from finite different schemes start off with del f del x series expansion and then from there you can derive this.

I am not going to derive this but just going to show you what this expression for are for. So, the del square f then is the sum of l square f by del x square and del square f by del y square sorry there is a mistake here this is y. So, then this is the final expression for the laplacian for an image. So, this is implemented using convolutions. So, what do you mean by using one and verify that I have the I made a mistake here. So, it is if you look f of y minus 1. So, this there is a minus sign here rather than the plus sign. So, I missed that one.

(Refer Slide Time: 2:32)



Now if you look at it specifically I will tell you how to interpret this. So, del square f is given by this expression. So, if you I look at this formula carefully there is f of x plus 1 y comma y which means if you are considering x comma y to be the current pixel you are at and you are trying to evaluate the laplacian at that point then you look to the which is x plus 1 and its coefficient is 1.

So, that is what we have here coefficient is 1 look to the left here that coefficient is also 1. Similarly, top and bottom there is a mistake here one of them should be f of x comma y minus 1 and there is a minus four times x comma y which is here right there. So, there are various so, this is the kernel. So, this is the what they call the filter kernel.

$$\nabla^2 f = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

So, this you can convolve with the image the convolution with the image and to get the output which is the laplacian of the image. Now there like I said there are various versions of this which I have indicated here some of them have the signs flipped some of them are perhaps slightly higher values and they have diagonal terms also.

So, for instance this does not have diagonal terms this has diagonal terms once again here the sign is flipped in this one here this here our sign is flipped and it has diagonal terms. So, there are different variations possible. So, let me go but kind of carried on over here. So, it is so, this is the laplacian kernel and it is usually convolved with the image to get the output. So, what is it used for. So, the laplacians detects what are called zero crossings.

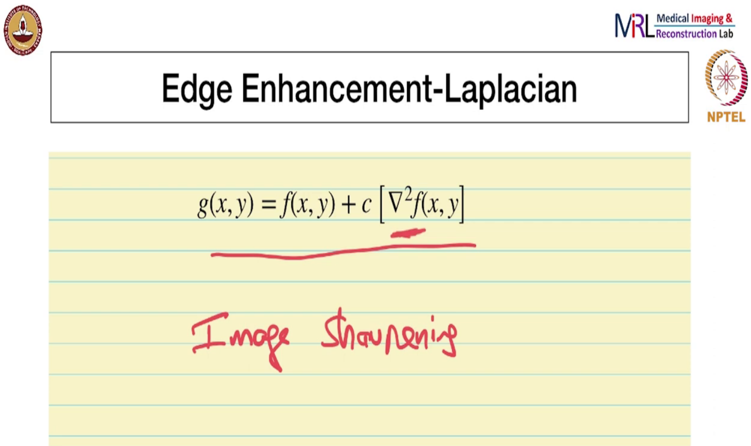
So, the zero crossings are actually used to detect edges. So, whenever there is an abrupt change in the intensity in the in an image. So, which happens when you are at the edges of structures the center of the transition region where it goes from one value to another that can be detected using the laplacian.

So, it is basically an edge detector it also is sometimes used for enhancing edges like basically you calculate the laplacian of the image multiply it by a constant and then add it back to the original image. So, that kind of processing is also sometimes done it is called a lab running matlab you can use any order of the kernels here as input to this matlab function called edge.

So, you can log into matlab look for a doc you can type this command doc edge it will show you the documentation of it. So, one document one of the modes of this function allows you to it is called under zero crossing argument you can use, you can only input one of these kernels on an image and gives the nice edge map of course remember that since images are somewhat inherently noisy always there will be a lot of false edges.

So, you will have to do a thresholding to remove the false edges and will work with the good ones. So, that much is done. So, please write this function out in matlab class and edge enhancement.

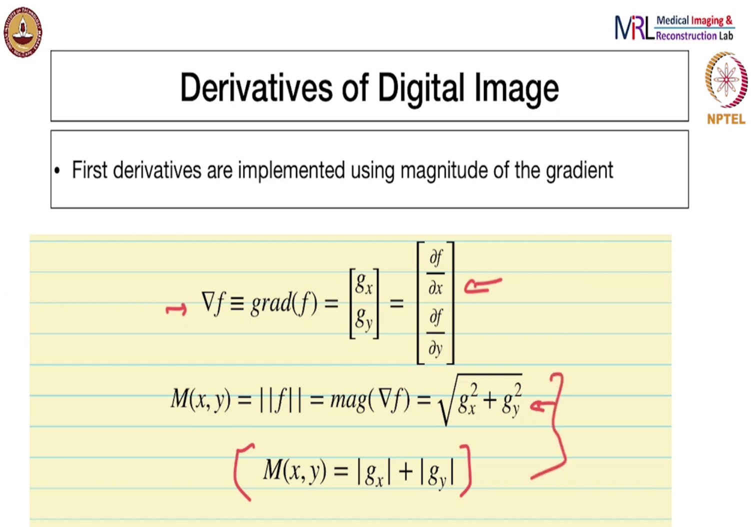
(Refer Slide Time: 5:53)



So, that is one of the typical processing that you do. So, what it does is highlights the edge of the images and so it is called image sharpening if you can want to call it that it is called image sharpening. So, you can use laplacian image sharpening to do that. So, the definition of laplacian we saw and how to compute it for a discrete image and what it is used for.

So, typically if you do the laplacian on an image you can use any one of the kernels here. So, for instance if you are okay. So, the image sharpening of course you can do this. So, when you do this this laplacian is of course the threshold version otherwise there will be a lot of noise.

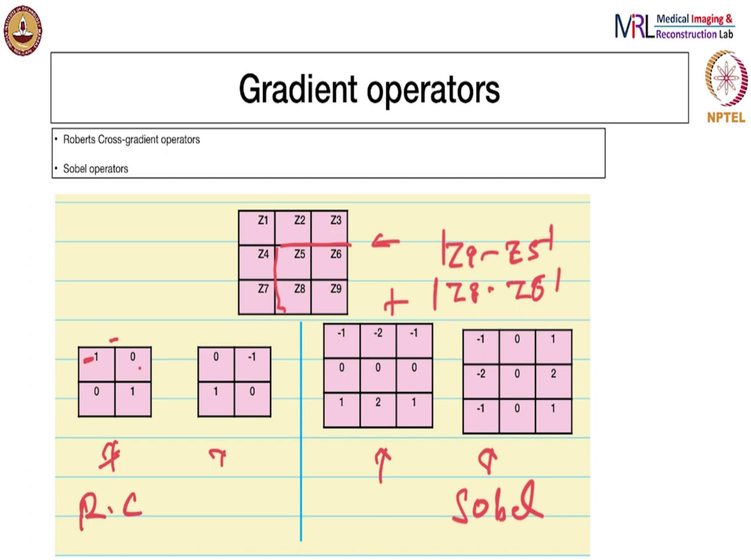
(Refer Slide Time: 6:34)



One other way you can do that if you want to use that laplacian sharpening is you can smooth the image before you do that and then that will work better. The another operator that is typically used often is the derivative, derivative operator derivative of digital image. So, again typically the gradient as it is called gradient again is a vector is actually in two has two dimensions especially for an image gx and gy and they are defined as such the magnitude of the gradient is nothing but in an euclidean vector here the data which it is like gradient metric gx square plus gy square.

And once again you can also define the magnitude in this fashion modulus of g of x plus modulus of g of phi. So, that works all of these are equivalent in a sense the magnitude these are equal definitions but in the general sense when you are talking about gradient of f you would calculate both these components.

(Refer Slide Time: 7:30)



So, here are some of the more commonly used operators. So, if there is an image here. So, the robots cross gradient operators are given by this expression these two kernels one of them is for one component the other for the other components. So, gx and gy similarly these are sobel operators these two are Sobel. So, these are these are the Roberts Cross gradient R.C I am going to call them these are the Sobel operators.

So, you can apply these onto these images and write down the expression for the output in terms of z's. So, for instance if you have used the Robert's Cross operator you want to use the Roberts Cross operator on any of these x maps what will be the output. So, how will you how will you do that? So, for instance if you want to let us say you place this kernel somewhere it is one well typically Robert's Cross gradient operator it is minus I think I made a mistake in the there is a there is a minus 1 here I am sorry.

So, if there is a minus 1 instead of plus 1. So, that that is why it has to be symmetric. So, z if you want to let us say calculate the edge around this region you would place this operator on top. So, your output is actually z 9 minus z 5 so on and so forth. So, similarly you can do and then similarly you can place this operator in the same place and then you will get z 8 minus z 6. So, then you can calculate the magnitude of the gradient as modulus of this plus or modulus of that or you can square each of them and add them and take square.

So, that is one way of doing it you can do similar operations with the Sobel operator also. So, these operators you will you will use typically in gradient in edge detection most of the time that is where it is used it is used in edge detection quite a bit in order to let us say in many cases some of the earlier image segmentation routines used edge detection. So, you detect the edge and connect all the pixels in the edge and then you have a contour.

So, that is and this is also a preliminary step on many algorithms. So, you first detect some integrated 10 x to initialize a contour for instance. So, this can be used and also for let us say edge enhancement. So, you want to sharpen the image you can use these gradient operators to detect the edge and then subsequently sharpen the image.