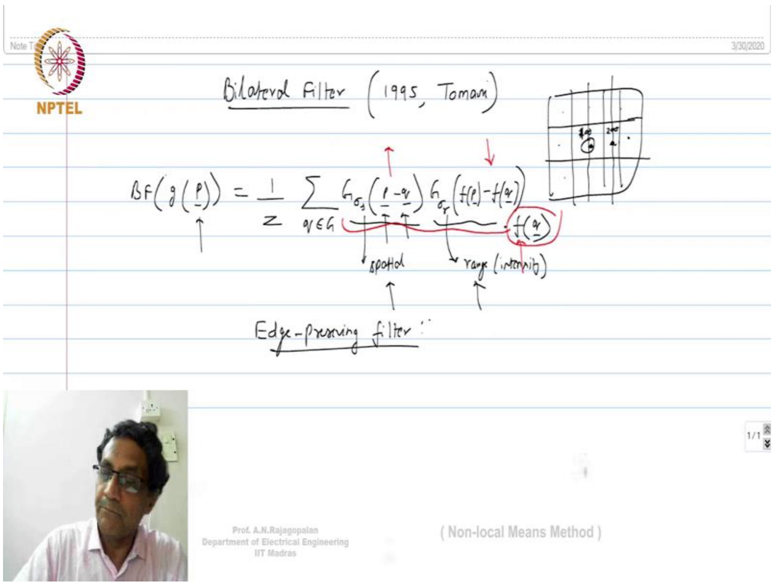
Image Signal Processing
Professor. A. N. Rajagopalan
Department of Electrical Engineering
Indian Institute of Technology, Madras
Lecture No. 66
Non-local Means Method

So, we saw that how a Local Spatial Averaging Filter works among, those we saw how a uniformly weighted filter works wherein all the same they all of course added to 1. Then, we saw a Gaussian Filter where we said that you know the weights will not all be the same the central pixel gets maximum weight and then the weights begin to taper off as you go outward from the central pixel.

(Refer Slide Time: 0:52)



Now, as you notice it was not taken into account the intensities that all the spatial averaging filters just kind of look at whether a pixel is close to a pixel under consideration or not, if there is the pixel that you are trying to filter intensity at this pixel then you simply look at the spatial locations of the other filters and based upon how far or how close the other locations are you assign a weight, the further pixels get the lesser weight and so on.

But, then the issue with this is that when you do something like this and what really happens is when you actually operate in this mode what is called the local averaging more than what happens is if, I have a pixel that is farther of and that is intensity which is also very unlikely to be similar to this intensity then what I am going to do is or for example, even if something is not too far away image that I have something with me right next to it and soon right next to this pixel.

Then suppose, this intensity is not similar to this which can also happen at an edge for example, if you have an edge read you may have a different intensity here and then you have a different intensity there and this could be 200, that could be or for example this will be 100 and that could be 200.

Now, if that is the case then what will happen is when you because, of the fact that this is specially closed you are going to assign the high weight to this and then what you will go is you will take an average even, if is not even as Gaussian weighted still because, it is very close to it will have a reasonable weight and therefore it will happen is you will actually end up smoothening the edge.

So, the point is when you go here you will take this intensity in account when you come here, you will take this intensity account and then at both pixels you will get some average value here and here an average value that will come here and that will blur this edge. This is this is unwanted, this is undesirable, we do not like our edges to get lost like that simply because, we want to do noise filtering.

So, one way out of this is an advancement over the over the over the spatial spatially averaging filter is what is called a Bilateral filter. I will just talk about it in brief and then kind of move on to is filter that is much more recent. Bilateral filter came in 1995 it was Script proposed by Carlo Tomasi and his students and it is called bilateral, it is called bilateral filter because, the fact that it takes up a form like this I am going to first write down how it looks like.

So, all these things now have the same interpretation that we had earlier except that z is a normalization constant all these things are the same and here what happens is here is something like G sigma s p minus q so, if you will notice I am going to indicate this as s but, I am also going to indicate some like G sigma r f of p minus f of q minus there is a minus let, me write it a little away if, a p minus f of q and then the whole thing multiplied by f of q.

Now, what this means is that so, this filter is still a spatial filter. So, at actually it looks at the locations p and q and based upon so, you are looking at filtering a pixel at p therefore, it looks at whether q is sufficient is close or far and depending on that it assigns a weight but, now you find that there is one more function that is sitting here which is also a Gaussian but, then this has a subscript r, this s stands for spatial location, r stands for range or in other words intensity, range or intensity.

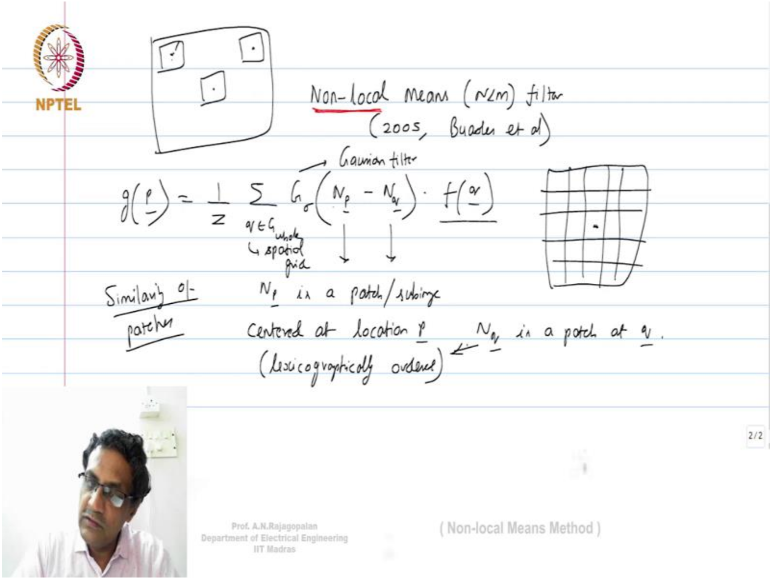
So, here what you see is that clearly you have inside not p minus q but, rather f of p minus f of q, f is the intensity in the noisy image at p, q is the intensity in the noisy image at f of q is the noisy intensity, is the intensity in the noisy image actual location q. Now, what this is trying to do is so, now it is called bilateral because, it got not just a filter in the spatial domain but, you also have a filter that is also looking at the intensities.

Now, if you revisit this example that we saw earlier now, what will happen is now, even though this location is close to this, suppose, let us say this is your p and then that is your q even though q is close to p just one pixel away if, you had just used a spatial filter it would have given it a high weightage and then you would uptick simply take an f of q as 200. Now, what will happen is this is weighted by this other filter which is also looking at the intensities norm.

So, this g sigma r is going to compare these two intensities and then only of these intensities are close will this weight be high otherwise, this weight will be low. Therefore, therefore we say net I mean effective weight so, the effective weight which is a combination of the two that means the spatial weight is further related by the by the by the by the by the closeness of intensities and that is going to be the net weight that will apply in x of q. Therefore, even if this weight is high but, if the intensities are very different this is high because, the pixels are close but, if intensities are very different, then basically this will weight down.

And therefore, the net weight that will give for 200 is going to be much less than what it was before this is also called an Edge preserving filter. So, that is how we are able to say preserve edges. So, edge preserving filter such a filter is called edge preserving filter and in such filters when you try to filter noise out, you will actually naturally see a much better improvement over what you saw with just the other kinds of with the very simple filters like that like the mean filter or let us say weighted Gaussian and so on. That is the idea bend in the bilateral filter.

(Refer Slide Time: 6:38)



Now, let me come to a more important filter or the more popular filter what is called Non-local Means or what is called NLM filter, this was introduced in 2005 I think, non-local means filter I think this came around if, I am if mistaken this came around in 2005 and this was proposed by Buades Etol and the idea right behind this filter is that as a name itself suggests its non-local see bilateral is also local, you are only looking at immediate neighborhood just as you are doing with respect to the mean and averaging filter.

So, even here if you go back this is still local filter except that its bilateral it is still kind of local the way you are not going far away from the actual p, this one clearly says this is non-local which means that it is willing to go travels all over the images, it was willing to go wherever it wants in order to be able to find the similar intensity. So, if you are here this is your p, then this is willing to go anywhere, how does this filter operate?

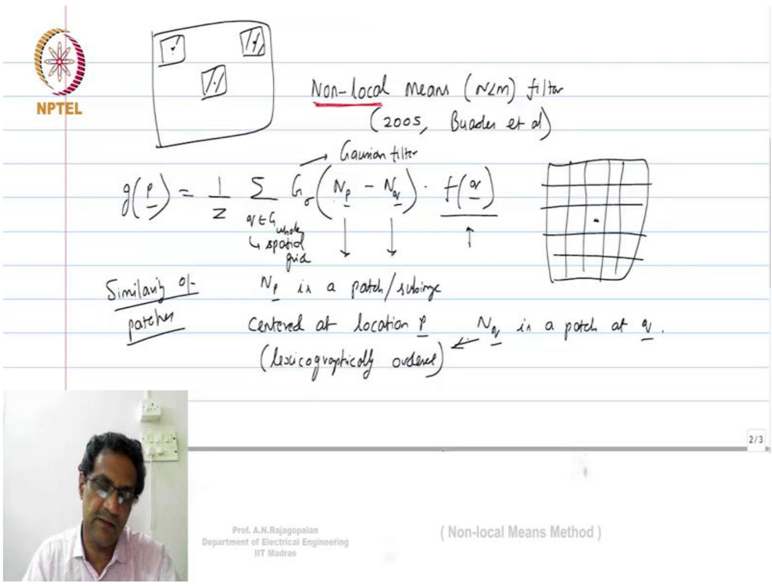
So, again if you go by g of p where p is spatial location and g is the filter image. This given is 1 by z in summation u belonging to g again read everything as before if, you notice there is a small difference now, g sigma nut then what I am going to write is N p minus N q, this is a subscript by the way it is not N into p or N into q this is q the subscript into f of q. Now, what is this N p and what is N q? So, these are patches so, N p is a patch or a sub image as it is called around or off-centered at centered at P, at location P and q similar to N p is so, N q is similarly patch at q, at around q, centered at q.

So, what this means is that you can so, if you think about it so, you have a big image so, what it is saying is I will pick a patch here let us say at p now, I am going to going to look at patches anywhere in the image which could be situated anywhere that is why it is called non-local at q and these patches I am going to compare the patches and this is and N p and N q these are lexicographically ordered that means you have to convert them into vectors, lexicographically ordered, lexicographically ordered therefore, what it means is you got two vectors.

So, these patches is a lexicographically ordered into a set of a column vector and you are trying to compare these two and this way it comes out with the similarity of patches so, what you are examining is not similarity of intensities but, rather similarity of sub images or patches. So, the whole idea is one of patch similarity. Now, if you find that a patch here is similar to the patch here, then that intensity at q again not the whole patch, this intensity setting at q is given a higher weight.

That is if you have another patch sitting at q but, this patch is not similar to this, then this intensity at q will be actually weighted down by this G sigma which is of course a Gaussian filter, this is a Gaussian filter but then this now kind of looks at patches. So, unlike all it we saw till kind of comparing scalar intensities that means you are just looking at one location and comparing the intensity in non-local means that is not the case. So, here when you are going to be comparing and here you know what I mean you can actually go with the entire grid so, entire spatial grid. So, there is no notion of no notion of all spatial grid, you can go anywhere then limit so just strictly speaking completely non-local.

(Refer Slide Time: 11:06)



Now, this G sigma has a kind of let me just write down now something about it. So, if you look at the fall of this G sigma. So, let me say that G sigma of I will call this as x, where x is now because, of the fact that it is a patch it is sub image patch. So, I am going to write this as x1, x2, all the way up to let us say, xm or m square if patch sizes m cross m, the patch size is m cross m, then this vector is of length m square, G sigma of s has the form given by e raise to minus summation and the z will still be there.

So, z that is a normalization constant so on, e raise to minus summation xi square upon H Square. Now, the summation xi square is goes on going all the way from 1 to m is of course obvious is to rate what it is going so, it is simply comparing the intensities at every location I mean within the patch and then it is complicated is coming out with some error. Now, this h so, the h is something that will actually decide the sensitivity so, let me just write that down.

So, h is h controls controls sensitivity to to dissimilarity of sub-images, dissimilarity of sub-image that means, what it means if, h is low, if h is low, then then the patches have to be have to be very patches must should have to be have time be very similar looking to get to get a weight to get a reasonable weight. On the other if you if you actually increase on the other hand if, you make h high it means that even if, two patches do not look so very similar in terms of their intensity values, you would still give them some weight.

So, this h really controls how many patches really end up now how many of these patches really play a role while, you are computing this weighted average and always remember what is being average is just the intensity at the central pixel in these patches, not the whole patch, the patch is only used to compute the weight and the intensities there that are averaged or all these or all these locations about which the patches are centered, intensities the locations about which the patches the center.

Now, that is how a non-local means filter work and then a strictly now, what typically done is because, of the fact that this can be computationally very heavy so, what is typically done when you search do not really go over the entire image. So, what you would typically do is if you have a whole image and suppose let us say, your p is here then what you would now, let me just maybe draw it a little centered.

Now, suppose I am here and what I would lose and I will choose some window size about which and typically this will be like a 31 cross 31 kind of window search window so, you do not really strictly speaking, theoretically speaking, you go all over the image in order to be able to compute the patches that are similar but, normally we stop within a search window so 31 cross 31 search window is what we would normally look for.

But, this again can be changed. It is of flexible hyper parameter and the patch size itself is typically can typically range from can range from let us, say 7 cross 7 to about 15 cross 15 max not more than this because, because too bigger patch will also not really help. Now, that is what I have to say about the non-local means filter, h is of course hyper parameter. This is hyper parameter should be chosen by you appropriately. This should be this is something that like you have to choose depending upon the example on hand.