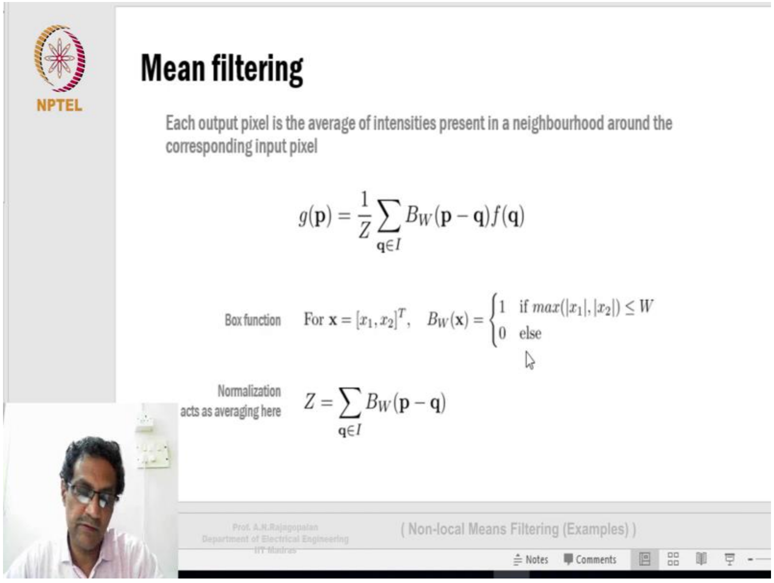
**Image Signal Processing**
**Professor. A.N. Rajagopalan**
**Department of Electrical Engineering**
**Indian Institute of Technology, Madras**
**Lecture No. 67**
**Non-local Means Filtering (Examples)**

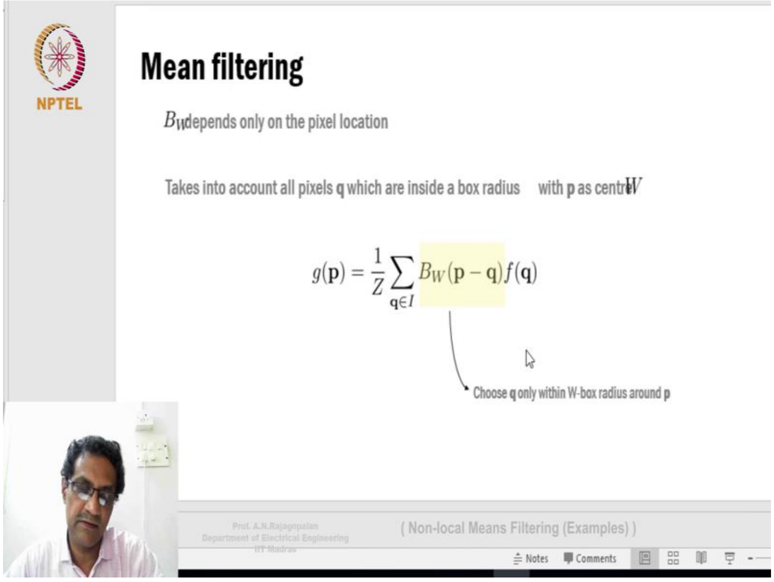
So, we have already seen how the non-local means filter works endow, I will show you some of these examples which we will try to illustrate how well this filter works let us compare to your non-averaging, averaging kind of filters which are all very-very local, they do all, they are all do local averaging including the mean filter and the Gaussian filter, and so on. Now, some of these slides are self-explanatory because, we gone through modally by navigation I am going to go through each one of them.

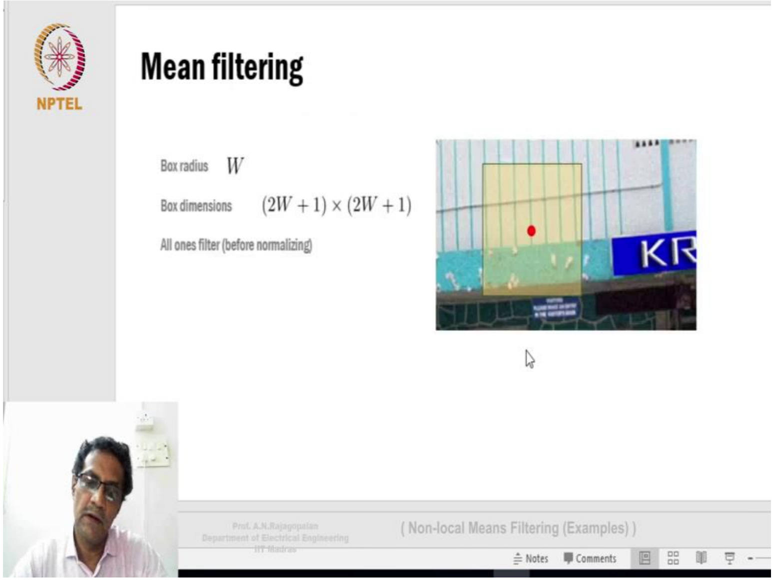
(Refer Slide Time: 0:45)




So, if you see so, mean filter we have already seen these equations like a box filter has all uniform weights.

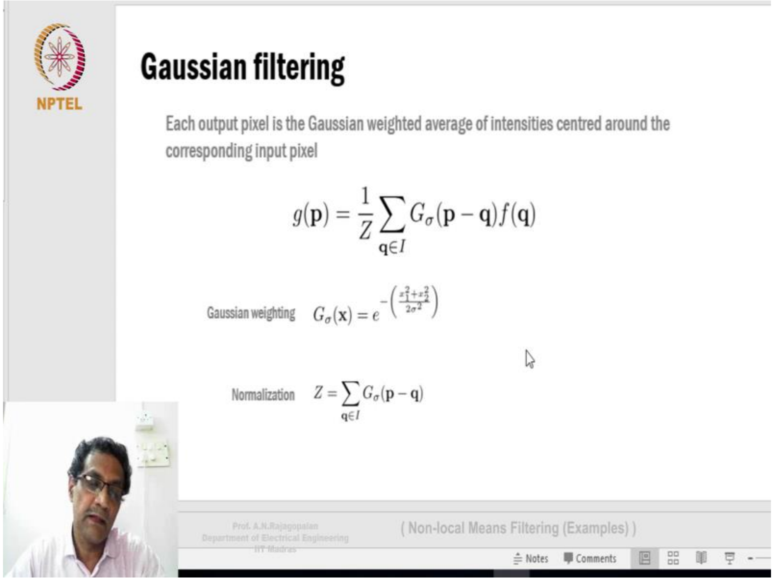
(Refer Slide Time: 0:50)



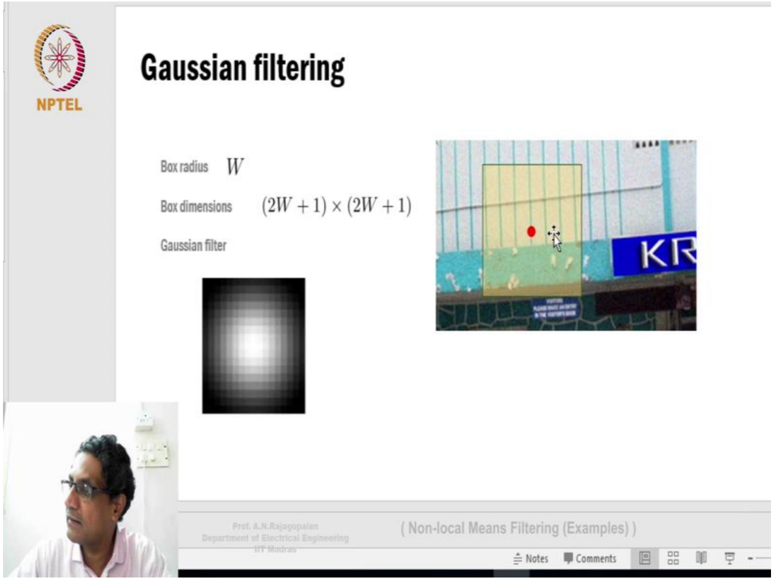

And then and the kind of the kind of filter that you get out of this is actually is what I would say specially variant because, of the fact of the weights remain the same irrespective of where you are in the image, irrespective of which location which location you have chosen the weight remain the same.

(Refer Slide Time: 1:11)




The Gaussian also very similar, just have to know that the weighting is not uniform but, otherwise there is also spatial invariant filter, wherever you go you will get the same set of weights and therefore, both mean and Gaussian filter can be implemented using Fourier transform because termalution holds.

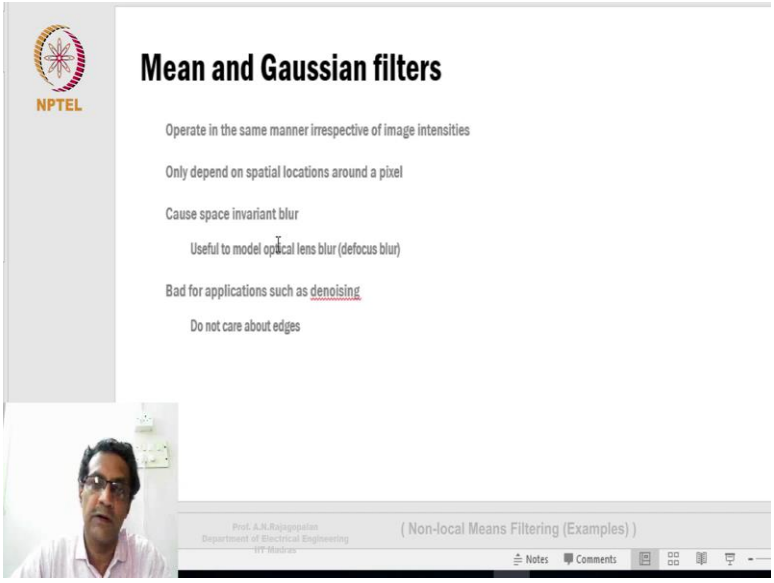
(Refer Slide Time: 1:26)




But then then the issue of course here is Gaussian filter you can see that the shape of this filter, it has a maximum in the middle the weight is maximum in the middle, and as you go outside and then the, here is an image so, that is a pixel that you are looking at that is been shown in red dot and then we are trying to filter it.

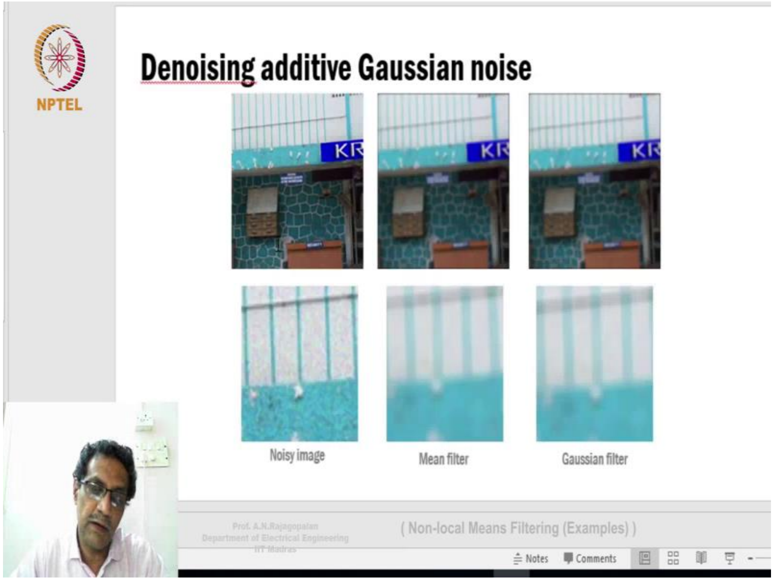
Off course, your filter size will typically not be as big as what is being shown in this image, it is only an illustrative picture. But it will be like 6 sigma plus 1 cross 6 sigma plus 1 which maybe 13 or 15 or something, typically is not bigger than that normally.

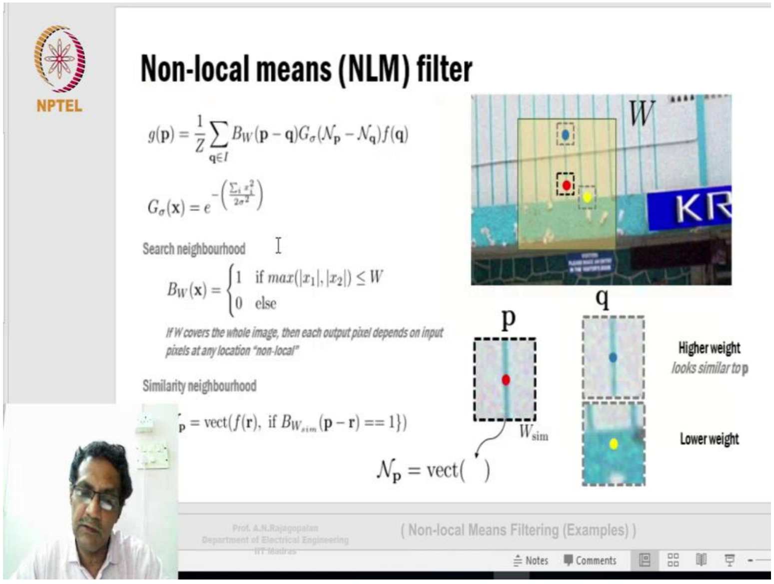
(Refer Slide Time: 1:58)

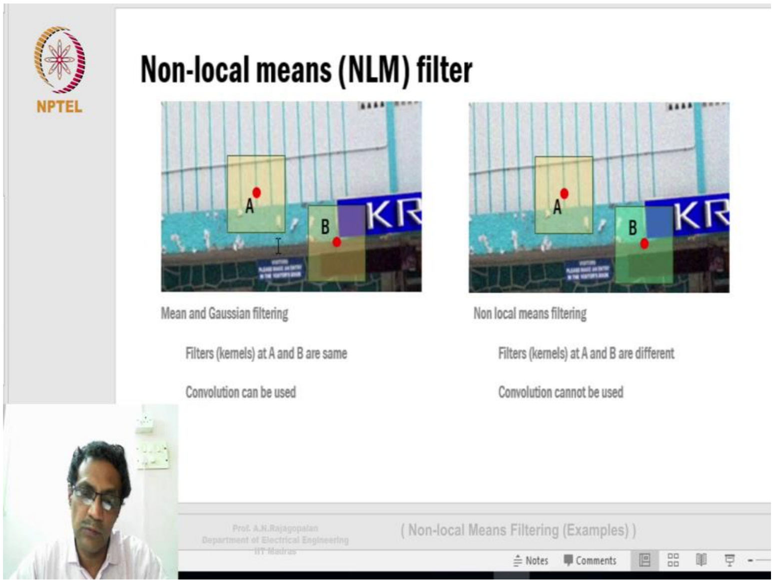



But then the point is whenever we use mean or this Gaussian filter it will end up with the weights that are constant but, then this also causes something like a space invariant blur because, any any averaging operation will introduce some amount of blur and then because it is weighted or weighted averaging or uniform averaging is constant across the entire image. So, it causes what we called is space invariant blur. This is bad for application is denoising because of the fact that it cannot retain the edges for reasons that we have already seen before.

(Refer Slide Time: 2:29)

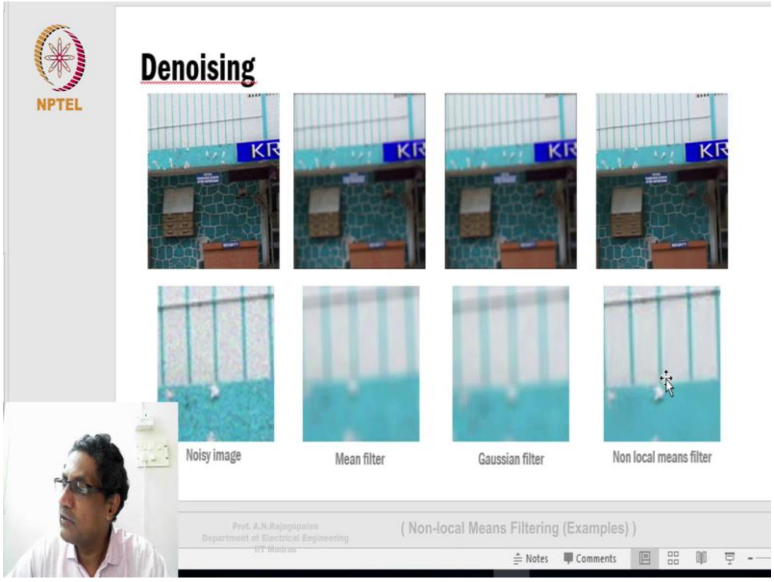


Then dinoising additive, now I will skip this non-local means filter I have already explained to you how this works.

(Refer Slide Time: 2:35)



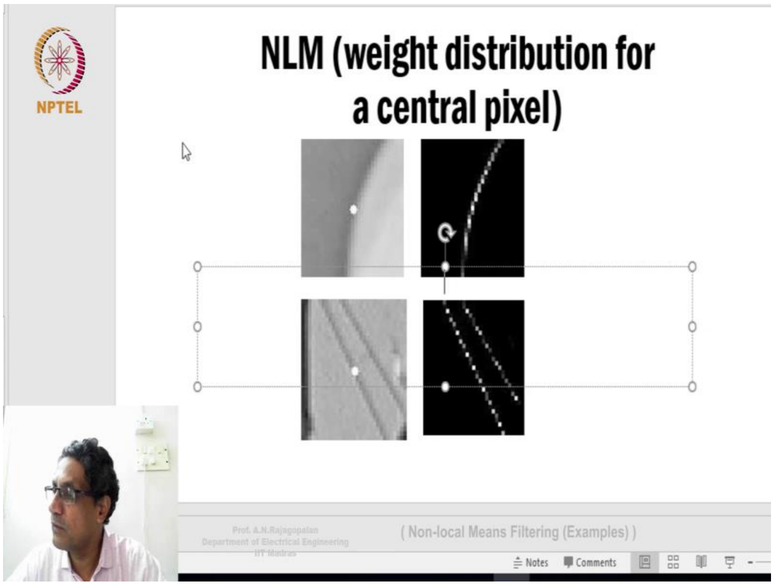

And I know we will just see as to what we output, what kind of output you get ow, for example it is noisy image, it is a noisy image it is another input noisy image and then when you actually inputted through the filter if you look at the mean filter, the output looks very smeared, the vertical edges are all blurred out. Similarly, for example, if you look at this horizontal edge, it is supposed to be sharp here. In the noisy image, it is sharp but here it is all blurred out.

In the Gaussian filter, maybe a shear better than this but there maybe also is not spectacular where there is a non-local filter is considerably better than both. You can see the both that the edges are well preserved, they are not shamed out, even the horizontal edges very good, even those small block you see here, it is dot and totally smeared and the other two. It looks it looks quite sharp in the output image.

(Refer Slide Time: 3:20)


Now, as you would expect, you can another way to interpret this another non-local mean filter is sort of look at this patch with this p, which is being taken on the edge and now suppose, we try to see weight will q1 get what weight will q2 get and what weight will q3 get. And by looking at this image, it is obvious that q1 and q2 should get a higher weight as compared to q3 because q1 and q2 where the patch and all on q1 and q2 look very similar to around p that is the one at q3 it looks like something else.

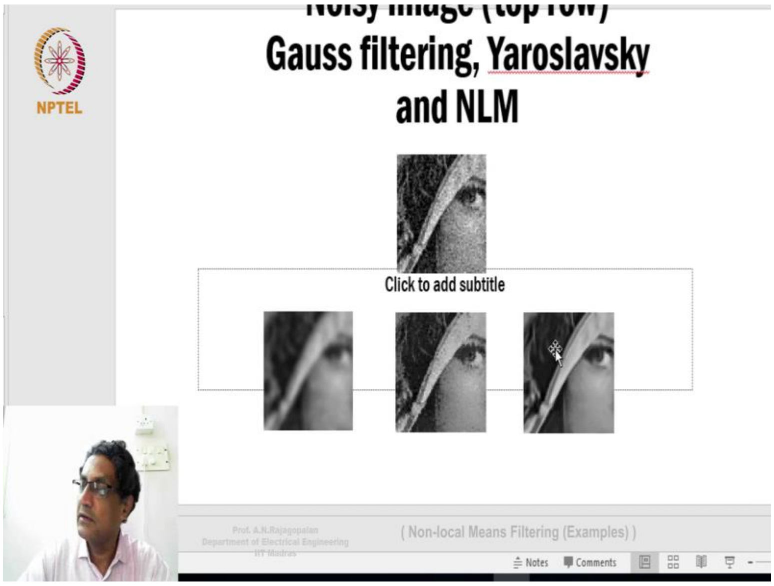
(Refer Slide Time: 3:50)




And therefore, look at this what exactly happens. We traverse on this hat and if you try to see what kind of weight we get from this pixel at p, then you see that over the entire image, it is only over the hat that you get in the weights that are significant everywhere else, the weights are almost 0.

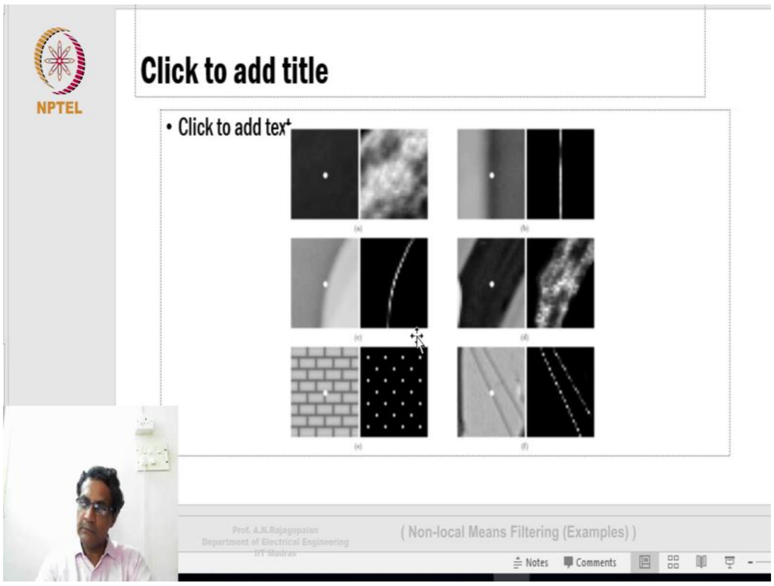
Similarly, if you want to take this pixel out here on this edge, then you see that on the weights you get from all the patches from around, from this entire image, what you find is that here all you get only significant intensity is along along these two lines which are also similar looking edges that is those weights are very weak. Furthermore, you can kind of go to let us say one more case.

(Refer Slide Time: 4:30)




One more situation here is Lena that is being filtered. Now, Yaroslavsky filter is a filter that works very similar except that it looks at individual intensity does not look at patches. And therefore, you see that the Gaussian filter does not do a good job, the Yaroslavsky filter gives you the an output we can say is better than the Gaussian output but, nothing great whereas the NLM filter looks really-really good. If you to be able to come from there to this is really appreciable.

(Refer Slide Time: 4:58)




And then you can see a few more examples where this kind of see shown you the weights. So, for example, if you look at this, this is texture image which is kind of a brick wall, you can see that if, you take a p that is at the intersection of these three lines, then all the weights come from similar locations. It is kind of distinct nothing that stands up, whereas if you took a patch around which there is no great structure, then you see that the weights get smeared out.

So, something is like is really-really not nice but, something like this is excellent. So, all these examples serve to illustrate that NLM clearly beats other filters specially this non-local averaging filters and of course that further improvements over NLM, there is something called a BM3-D filter which is actually, block which is called block matching and the 3-D filtering, it is kind of a collaborator filtering that is the most recent one which came out seven in 2007.

That is (())(5:53) is another filter which if you are interested, you can actually go through. It uses a unitary transform kind of an approach, you will not to be able to which is also patch based but, it uses some kind of here, unitary transform filtering in order to be able to reduce the effect of noise. For those of you are interested, you can actually read the BM3-D also. Now, that we have all this background which will illustrate forward BM3-D.