Modern Computer Vision

Prof. A.N. Rajagopalan

Department of Electrical Engineering

IIT Madras

Lecture-33

Today, we start a new topic what is called low level vision and under this the things of interest are some basic things like filtering which I think you are all roughly familiar with what you have done in your 1D and all, but I still thought there are certain things that are not exactly what you would have found in 1D. So, I thought I will talk about them just differences rest and all I think you can figure it out yourself. And we want to actually head towards extracting features from an image. So, these features are something that kind of stand out for example, you know a corner is actually a very striking feature which is easy to track across frames and so on. So, you want to kind of look at. So, features are interesting points of course, they can also be edges you know they can also be more than a point, but normally this would be some kind of a very local thing which we will see you know where we can apply it and you know how it is kind of useful right, but prior to that let us first start with you know filtering.

So, let us start with image filtering or what is called a 2D filtering. We have already seen in terms of you know. So, filtering is an operation even in CNNs we saw that already right we saw that we could actually we could have filters you know that would work on your feature maps in order to see produce you know output feature maps and so on. And there those filters were not handcrafted right in the sense that given a task you had a network I mean whose job was to figure out you know what might be the right kind of filter and so on.

And we never bothered about what those filters actually in the sense that we never went into their weights to see what they are doing we just understood that you know the initial filters would probably pick up edges and so on the orientation. And then at a mid level something will happen which is a combination and then that whatever the mid level and then the and then the you know final layers something else will happen which is a little more complex in order to solve a task. So, but now right, but now we want to come back you know take two steps backward and sort of look at you know how this filtering was actually you know. So, if you have to do it in a kind of a traditional way right then there are it is very neat and you know one wants to understand right how that is and and like I said right you know it is always good to know both on the one hand you know what one can do with a deep network. And another thing that we should always realize is that it is not like right deep networks can solve our problems right.

Let us understand that you know despite by the way right just one more thing what we have left out out of the deep networks I mean that I think you are lightly encounter is what is called an auto encoder which I think you know which you can easily follow now based upon the background that you have. And there is another thing what is called again a generative adversarial network. So, these are two things that we have left out because we could not cover them, but you know you are in a position now to be able to read any paper you know in deep network and be able to understand it. It might just require little bit of little bit more work if it is something that is not been taught in the class, but but everything you can follow now. So, I think you do not have to there should be no mental block right in being able to read a paper on deep network because right now any paper that you touch will typically have something that is using you know a deep network right.

So, I think you should have that kind of a comfort level now. So, now let us kind of get some comfort level at at a more kind of you know for what you recall you know in the in the kind of say traditional approach. And here right we are going to look at look at basic filtering operations and you know in 2D. And then we will see right how these filtering operations are actually taken forward in order to the other things that I mentioned like you know you might want to extract features you know descriptors and all that. So, how the how all of that happens and and then and then the kind of pipeline is such that you know using those features what else can you do I mean that is when geometry kicks in.

So, we have like low level vision right I mean what we can do there and then how it can kind of it it can feed into a geometry which is which is the one that we will cover next. So, geometry I think we have single and then you know stereo as well as multi view right, but then before going there let just let just do these do these simple things now just to motivate this right I will just I will just show you a few slides just as just to motivate, but then the idea is something like this right. So, so you are filtering right you could you could do it for various reasons and all of these are very simple operations by the way cannot the complicated ones that we have already seen and that is why that is why your I mean this should go in faster. So, you see that right there is an image in the left I mean. So, here what you have is an image and then and then it you want to I think this PDF I cannot edit.

So, what you can do is you know you can actually you can sharpen it if you wish and all these are based upon some kind of a 2D filter right that you want to use and what kind 2D filter will go where that is something that we have to handcraft now because we are not asking a deep network to do to can do it anymore right. Now, we have to figure out what will be the right kind of filter or for example, right you might be interested in in actually edges right something like a Lena I mean this is a very famous image right the very famous lady. So, so if you if you if you apply some kind of an you know edge finder on that image right then you get all the portions where there is a gradient where is a high gradient right and things that are relatively homogeneous like a shoulder and all right you would not see much coming out of there. So, wherever there is image activity right I mean this this kind of a filter should sort of flag it or you might even do a template matching right. So, this is in those days it used to be called as a template matching and the effect was all was always in terms of being able to you know locate a particular template inside an image this is what we do right when you say what is that I spy and stuff like that right.

So, where something is embedded somewhere and then you want to get a pick it up and in those days right people used to talk about how to handle scale how to handle orientation, but these are all taken up as you know individual problems by themselves. Whereas, now I think we know for example, right that there are other ways of doing it also, but a traditional is still very very well grounded. So, it is important to know what goes in there. So, so right this is at a very low level you can actually you know sort of escalate it a little bit more and then and then you know you can kind of talk about things like denoising which is like you know mitigating noise I mean we never say remove noise we always say mitigate or you know reduce the effects of noise. So, you see that in the left right there is this a salt pepper noise right which basically means that we got you know extreme values it is not like a Gaussian noise you can also have additive you know white Gaussian noise, but the example given here is one of salt and pepper.

So, it is like salt and pepper if you had and if you were to sprinkle it right on the image what you would get is a salt pepper noise and it is arbitrary in the sense that you know it is kind of signal independent it can occur anywhere and you know it can and there are ways to deal with it and typically in linear filters do not do well on such kinds of noise and then one has what are called median filters and so on very simple ones what are called order statistics filters and then you know they do a good job they are very simple may not be linear, but they are very simple. So, when you say they are not linear that means there is there is an implication that may be fast implementation is probably not possible and so on because anything that has a Fourier counterpart we know that you know FFTs are available, but the moment you say non-linear right then it sort of hits a kind of a little bit of a roadblock, but that is ok we need not worry about that so much in terms of computation, but I will also hint upon computation somewhere. Then the next picture that you see is one of super resolution. So, here it is like saying that you know how do you kind of if you had a low quality camera and you captured a picture like the one that you see there like that man right and you wanted to actually you want to enhance the quality of that picture and does your phone have this kind of thing these days I think it has more like a zoom tool right I mean. So, you can zoom, but that is all actually interpolation right there is no kind of new information that comes in you have to build upon what you already have optical zoom is not that that is actually different I mean there you actually get the details, but all this soft zoom as they call they are all like you know interpolation you know different interpolation methods, but this is not like that this is actually picking up the details from given a low quality image and there is lot of work in this area, but again right if I talk about filtering right you might wonder for example, you know suppose I say that you know you had to instead of this if you had a blurred picture and suppose you wanted to de-blur it right.

Now, blurring itself is actually a filtering operation it is a smoothing operation. So, you can think of applying some filter on an image and then right arriving at something that looks like a low quality blurred image and then when you talk about you know removing the blur you can even there you can actually talk about what is called you know inverse filtering. So, that is also a convolution operation, but it could be unstable sometimes it is not even you may not even be able to do it depends upon what happens in the to the to the Fourier domain right I mean in the Fourier domain what happens to this function that is acting on that image. So, right if it has zeros in the Fourier domain then you cannot actually invert it right. So, you run into some trouble, but then people have ways something called what is called pseudo inverse filtering and so on which is an approximation and people get away, but again right these have had problems that is why deep networks came in and then you know, but these are all that one of the things that you should notice is that there is no assumptions being made anywhere right you are given just one that one example and then you are trying to do what you can with that there is nothing like tons of data coming in and falling at your feet right nothing like that I mean you have what you have just that one image and you have to we have to worry about what you can do with it.

Then something else which is kind of in painting right. So, here also you can think about those places where there are scribbles and this is just some artificial kind of thing right and and then you might want to say how do I fill in those regions and nowadays right people do what is called out painting which is like you know I give you an image and then you have to tell what could be lying outside the image right for lot of years people are doing what is inside and now they say what what probably how does it extend outside right what what could be the logical extension of what lies outside. So, again right here also you can think of a filter that kind of locally acts right around in a kind of neighborhood and sort of tries to make sense about what might have been there right. One other thing that we should also remember is that you know it is not always true that you know that you have to you have to synthesize these these kind of filters they can also occur naturally it is not like every time we have to sit down and construct a filter to do this. For example, your camera itself right when you have a blurring it is an optical blur when it is out of focus right you are not creating any you are not applying something on the image it comes out automatically right.

So, that is like a natural blur it is a kernel again it is some some sort of a kernel that is acting, but that kernel is coming because of the optics and it is not like you are you are applying something on an image and similarly right when you find motion blur something is moving right you get a smearing effect those are all things that just happen naturally. So, again right one has to one has to be clear that it is not always true that kernel or these filters and all it does not mean they have to be made by us it can also be that they come they occur naturally in the images that you see all right. And so with that right so so the so let me just say right I mean so you might just want to whatever right blur sometimes and can you can you think of an application of blurring I mean why would somebody want to blur something blur sharpen sharpen I can understand why you do not want to blur. Reduce data says. Reduce data says anything else.

Sensitive information. Exactly sensitive information. So, you want to mask somebody's face do not they do it all the time in the paper and all right they do something and then they find they ask can you find out who is this tapestry and they blur it so badly right and and then super resolution in painting right all these all these are exact, but of course, you know the higher you go then these small filters and all do not work, but but till this point to some extent de-blurring right you can you can think about some kind of filtering operation that can that can take you there super resolution is more complicated, but anyway I mean you can still talk about that is again you know a down sampling you know follow I mean that is like an averaging followed by kind of say down sampling. So so the way so the way right we look at this filtering operation is simply a 2D this is in a convolution a convolution we have already seen right. So, so we can say ourselves the time of repeating what we said, but what what we said was this right.

So, G m n this is your 2D image I will just write a discrete case. So, you may have. So, you may have something like an image which is like m prime comma n prime and then and then you know you can have H of m - m prime n - n prime or you can equivalently this is summed over of course, m prime n prime or you can equivalently have H of m prime this is all and most of these things right will be exactly the same as same as what you what you see in 1D right. So, again m prime n prime normally right this is more easy to follow because you know H is typically of a very. So, this is your image and what you want to do to it depends upon what you choose as your H right.

So, for example, it could be an average here in which case in which case you will end up smoothing the image I mean it could be a difference operator or I mean it could lead to a differencing kind of an effect in which case you may get edges and so on from F. And typically these have a very very small support your image could be as big as you care I mean you know 1024 by 1024 or something, but H and all H will typically be like 3 cross 3 5 cross 5 7 cross 7 does not typically exceed that. So, therefore, the boundary effects and all we do not really worry worry so much about that is there in those slides I mean somewhere we have mentioned as to what you should do if you are at the boundary what kind of approximations you can make and so on. But we would not worry too much about you know what happens when that kernel goes to the boundary because rest of the action is happening in so many places which is still large enough that we do not want to I mean harp on what is happening at the boundary.