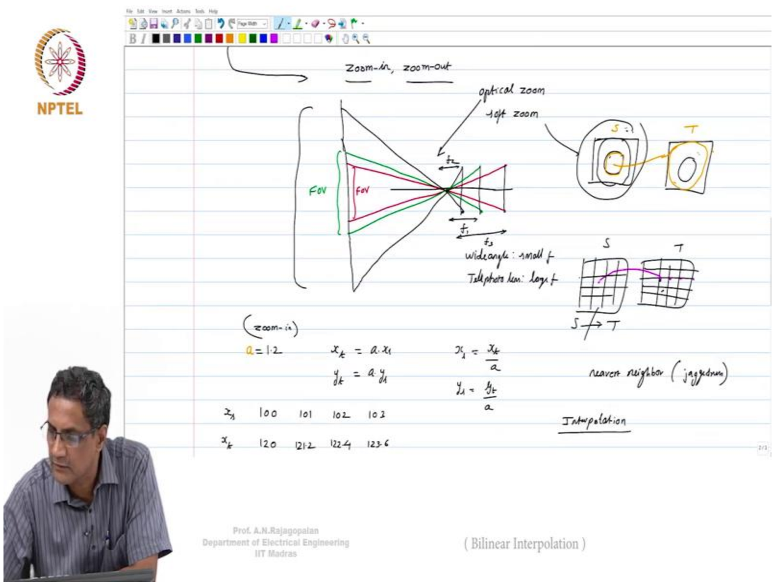
Image Signal Processing

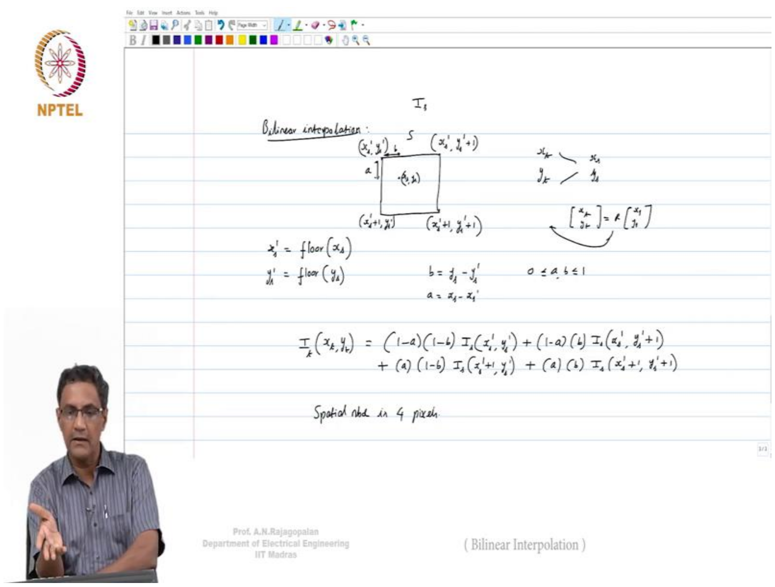
Professor. A.N. Rajagopalan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture No: 8

Bilinear Interpolation

[Refer Slide Time: 0:16]





Just as you have Linear interpolation in one d, this is this Bilinear, because it is linear long both you know X and Y directions. So, it is called a bilinear interpolation. Now, this bilinear interpolation what it tries to do is? Is tries to make use of not just one pixel value but then tries to make use us in fact four neighbors in order to be able, because when you land in that box, you have not just 1 neighbor, you have 4 neighbors.

And you want your want your final assignment to be controlled by, control by how near this vary of land and that how near it is to the pixel on to the left whatever left top, right top, right bottom, left and then right bottom right.

Now, imagine that now I am going to draw a bigger box here. So, suppose you have landed here, suppose suppose you have suppose you have done some inverse mapping, and this is that 1 pixel box. So, so so imagine that we are imagine that we are somewhere here. So, in this box that I mention how do you push this up. So, imagine that this box that I shown here, that is a box that I am that I am expanding there.

It is it is that box and suppose I suppose let us say write suppose so it means that I have assumed some xt. And I am assumed some yt and from here, I get my xs ys. So, if it is in plane rotation, I will do R transpose on on on xt yt, you have suppose you had your you know xt yt as R times xs ys. Then you will get your x sys are transpose, because R inverses to the R transpose, this is all orthogonal matrix, any rotation matrices are in fact orthogonal matrix.

So, so when you bring this here, it become become the R transpose, and that will multiply multiply xt yt. So, you so you choose an xt yt multiply it by suppose it was rotation or else depending upon whatever it has transformation, that you have, that relates it and target and source you want it use inverse transformation. And you will get a for a given for a chosen xt yt, you will some get some value for xs and ys. Suppose suppose that xs and ys is this, here is where you have land it. And this coordinate is xs ys, for some xt yt in the this one target grid.

Now, this grid, because because this is all this is all within a within a unit pixel box, this this is one pixel, one pixel on the all 4 sides. So, suppose we call this is let me just give some notation here. And so that we can stick to one kind of notation what do, I xs dash ys dash. So, suppose we call this is xs dash ys dash then clearly xs dash is equal to this case floor of xs, whatever be xs, xs dash will be because you want a name this is xs dash ys dash, because that is the that this is to left up and left up an xs and ys. And ys dash is equal to floor of ys.

So, suppose now again again right this here again I am here assuming that assuming that you know when you travel down xs and ys are increasing, it increasing to the right and then this is increasing downwards. Suppose that kind of a convention then this will be floor of xs and and this will be floor of xy.

And this grid will then what rotation I am using here xs dash, ys plus 1, ys dash plus 1, then this will be xs dash plus 1 comma ys dash anyway, what I am saying is in the lab suppose we give the other way, suppose we ask you to take this xs and the ys and you should just according the change. That would not change the logic, just like the convention might just change. Sometime because this is a way I think Mathlab has it like row then column. But then when we are try this convention this R rotation matrix that I wrote for a clock wise I said xs this way.

So, X is column and Y is row, so here there is a small sort of drifts on that, that is okay. Now, it would not really change your logic in any way. So, xs dash plus 1, and then this is ys dash plus 1, correct. And let us say that what is some A and B if you want let us call from here to here, this A, from here to here this is B where this so here B is like or no let us say B is like ys minus ys dash, and A is xs minus xs dash. And 0 less than equal to A, B less than equal to 1.

So, the max value the minimum value in A and B and take a 0 in which case I will just land here, remain we are both 0 and x sys will land exactly at xs dash ys dash, maximum that you take is 1, that means you will land here at xs dash plus 1 and ys dash plus 1, in between they can take any value, that is a range of A and B, with this sort of a convention.

Now, when you do a when you do a bilinear interpolation. So, what you what you what to do is? You want to be able to assign a value to your target It at xt yt, because it is for that xt yt that you got a this xs ys that is brought you here. And since write sitting somewhere inside this box in a bilinear interpolation, we do not want simply bunk on one of the intensity we want to we want to use as many as we can in this case the nearest neighbors are these 4 guys we would like to know all 4 of them.

So, in this case what it will be mean and then and then you would want to wait those intensities depending upon how close this xs ys to those pixels correct. The closer you are you want to give a higher weightage for that location, or the intensity at that location. What does it means that you will have something like 1 minus a into 1 minus b I source. So, I source means this one, so this image is like I source.

So, intensity here is Is of x dash comma y, xs dash comma ys dash, intensity at this location Is of xs dash comma ys dash, plus 1 and so on. So, Is at so 1 minus a correct and xs dash ys dash plus with respect to this grid it will be what? 1 minus a into b Is of xs dash comma ys dash, plus 1 plus when you talk about this grid here that will be a into 1 minus, 1 minus b Is of xs dash plus 1, ys dash plus then for the last it will be simply a into b into Is of xs dash plus 1 ys dash plus 1 correct all these are the source intensities.

So, what is actually means is that? If you had a situation where an a is equal to point 5, and b is equal to point 5, that means you landed exactly in the middle of the grid, middle box. Then it will be mean that all these values then will be it is point 2 5, everything will get scale 1 by 4, which is okay, which mean all of them will get will get uniformly weighted.

Because it equally near from all of them, just suppose you a was 0 and b was 0 then it means that all these grids are will drop out this will drop out, this will drop out, only the first term is remain a is 0 and b is 0, a is 1 , b is 1 there only the last term will remain everything else will go out, naturally because it is you know it is land of that pixel location itself.

But if a is some point 1 and b some point 2 then accordingly whichever it is nearest to so for that case this grid will get will get a higher wattage this intensity this will get relates. So, so so this kind of a relating weighting where you where you just check how far how where you are from your 4 neighbors and then assign intensities.

Now, this way this way bilinear interpolation is done and this actually reduces that jagged appearance in the box, because you are not banking on just 1 intensity using 4, you can off course on a go for go for go for interpolation that can use a larger special neighborhood that here you have used a neighborhood of neighborhood of 4 pixels. So, here we have special neighborhood that is called a consist of nbd a special neighborhood is 4 pixels.

But then you can go higher you know 9, 16 and so on, but then that will mean you will have to do more what you can have. So, those are like (())(9:38) cubic plane interpolation all which you have must heard of. So, those are all kind of higher order, higher order operations some of them involve even a even I say even sort of a derivative of the intensities and so on depending upon which of those interpolation algorithm you use.

But simplest of all is bilinear interpolation where you can simply just take 4 nearest neighbors because that seems to be the most straightforward thing to do. And now you can clearly say that you will fill up a target grid without any wholes and also you would be able to reasonably take care of a jaggedness and so on, which is simple operations like a bilinear interpolation, yes?

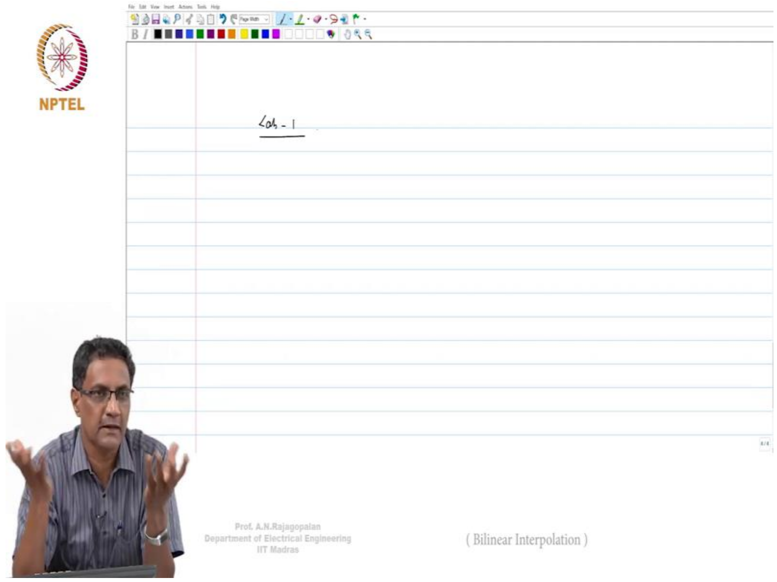
Student: (())(10:19)

Professor: No, no no it does not mean it will automatically result in space a when you do some kind of an averaging, if you do a blind average, and if you do a uniform averaging then what you are saying is true, that is what cost nearing. But then if you if you if you if you rate such that as you go outer and outer you are going to you are going to give lesser and lesser influence to those then smearing on that.

So, you can still use a larger neighborhood but only thing is intuition say that something that is further influence less my intensity here then something which is right next to me. So, so so as long as as long as you take care of that it is okay to use use even, even a larger neighborhood does not mean it will automatically smear.

What we are saying is something like an averaging, operation where you also doing weighted but then your weights are not are not in sink with you know where each location is. Then that can cause smearing but these these will take care of that just as just as even here we are take care of that, we are making sure that the grid that is closes will influence the intensity the most.

[Refer Slide Time: 11:23]



Ideally right, see this was only to only to only to make you realize that you know doing a doing a transformation at what is a den tail and if you ask you know why is this useful, why is such an operation why should I care? I am able to rotate, I am able to translate and if you ask why do I need that, then simplest example is an zoom in which I said you may had an image and you may want to zoom in you could use thing like this.

A more important than that is a when let us say people want to want to check what exactly has happened in an area over over lesser period of time, like for example imagine that IIT madras hit 40 years ago what kind of tree, tree sort of density that IIT madras have verses what is has now.

Suppose somebody would ask this question, then one way to do would to kind of you know get some photograph that was taken in 40 years ago look at the time from some aerial view of IIT madras. And then you should take an aerial view now and these the point you cannot exactly go to the same location from where it was taken earlier, you take from somewhere say nearby.

So, you might have taken it from here this is let us say the IIT madras campus you have taken a picture and then you come here and then you take one more picture you could be you may not be the exactly there a same location at all. Now, when you want to compare these 2 you cannot directly subtract them, because what will happen is your orientation is going to say has changed.

So, what you have to do is? You have first make sure that these 2 are aligned. So, for example in these 40 years let us say no tree had grown this is that right nothing had happened then if you subtract will get absolute 0, you will say oh nothing has change. But you know for a fact that some trees has been replaced by some buildings and may be some places that were earlier empty have grown trees there all this is happened.

So, first thing that you wanted to do is bring one of them in alignment with the other. It does not matter which one, let us say let us say I take the image 2 and then I try to align suppose image 2 come up like that. And then the first thing I need to do is alignment with respect to image 1 correct.

So, which then means that first of all I need to know what transformation will take me from here to there, and once I know the transformation then I have apply the inverse of that on this image. Because if it is like this and I can just write need to make a like. Now, that is a transformation that you talking about, within this itself I have a source which is this image 2 but I want to create an image 2 prim, which will be in sink with this image 1 in terms of alignment, when I want to do that that is exactly this thing that we are talking source to target.

So, there your target will be this realignment image, your source will get this image 2 that you have captured. And your image 2 prim will then align with respect image one that you can slap you know one on a top of other and then you can take as a whatever you can do a differencing, whatever you want to do. And then and then you can start concluding, what changes have happen, which areas of have become now more trees or so on.

So, so, that is a very, very important problem even for a surveillance point of view people are always worried you had satellites, satellites going around and there are times let us say these are all looking other the other country. And there are there are certain kind of sensitive zones where they know that if those military trucks begin to come out then you know something is going to happen.

So, the idea is as long as region is quit you know you may not worry, but at the moment you see a big activity there that it means or something is going to happen. So, again so a change a simply finding a change is such an important problem even in an airport think about it, to be able to say that something is now laying unattended that is a very very important problem.

So, somebody comes leaves a hand baggage and then if you know there is a hand baggage and for the next 2 minutes nothing has happened, nobody has come near nobody has picked up that means your change is 0 there should be an alarm they look because first of all you should know that somebody left a hand baggage there.

And then you should be able to tell that the next 2-3 minutes nothing happened again there problem is little simpler because a camera is typically effects and then you would only have to worry about may be someone else coming into the area and so on. Where is typically these things are used when you cannot restrict a camera to a particular location, you allow a camera to go to where it once you may seen and then come back and then you want to compare.