Modern Computer Vision

Prof. A.N. Rajagopalan

Department of Electrical Engineering

IIT Madras

Lecture-81

Now, in order to handle this right. So, we can so we can kind of right clearly see that well if you have a larger aperture perhaps right we can we can we can do things better right that is exactly what this Lucas-Canade Lucas-Canade method right does method ok. So, what does this Lucas-Canade method do? So, it introduces a spatial coherence constraint spatial coherence constraint in addition to the brightness constancy assumption that we had. Now, these people bring in a spatial spatial coherence constraint what that really means is that within a small window you can assume that your optical flow is roughly the same for all the pixels which is not which is not such an unreasonable assumption. This is where I said smoothness smoothness smoothness right I kept saying in one of the earlier classes also read local smoothness local smoothness c this is again local smoothness right. What this means is that within the window as long as the window small enough you can assume that all the pixels are probably moving with the same u and v if that is not satisfied of course, you know you will have some errors, but wherever it is true it should be ok right.

So, then what this means is if you take a window of size let us say write m by m ok. So, basically write around a pixel and you are trying to compute of course, optical flow at that pixel, but now you have, but now you are assuming that around it right you have the u and v so long as m is small. So, m light you can think of something like 3 m equal to 3 or 5 right not probably more than that. So, it has to be very very local region ok.

Then then then this then this equation right then what we can do is you know then we can actually repeat it for all the pixels here right then what you will get is let us say i x 1 i y 1 then I will have like i x 2 i y 2 all the way up to let us say i m square no i x m square right I mean there are m square pixels here right and then maybe i y m square with that many pixels I have and then I have u v right because because all of them will actually satisfy the same equation, but then here right I will have whatever i t of i t of x 1 y 1 because because because the temporal motion need not be the same for these pixels right and then i t of x 2 y 2 all the way up to i t of x m square y m square. So, in effect right what you have is is a matrix of size m square by 2 2 cross 1 and this will be like m square cross 1 and all this we can of course, estimate just from the image data. So, so right this is available. So, let us call this as a u vector is equal to b right and now we know kind of right how to do that.

So, you can do right a transpose a u is equal to a transpose b or in other words where you can do u is equal to a transpose a the whole inverse a transpose b right and and a transpose a is clearly right a 2 cross 2 matrix this is just this is just just a just a 2 cross 2 matrix, but if, but if, but if you think about what does a transpose a is right.

So, what is a transpose a transpose is this i x 1 whatever right i x 2 blah blah blah then i y 1 i y 2 blah blah blah and then this guy right same thing ordered this way i x 1 i y 1. So, if you do this right. So, this is like like the 2 by m square like m square by 2. So, you get actually actually 2 cross 2 matrix whose entries are simply gradients squared along x then i x i y then whatever i y i x which is same as i x i y and summation i y square right that is what it is and what is we have seen this matrix before when did we see this gradient coming in and at and you know 2 cross 2 when did we see this matrix Harris corner detector right there it came no. Now, the same kind of situation that is what we have here this is simply a 2 cross 2 matrix right it will have 2 Eigen values the only catch is here right now you have to find its inverse right which one means that means the right it has to be kind of reasonably stable right by stability what we mean is right it should not have like Eigen values that are kind of see crazy in the sense that right both Eigen values being kind of see very small will also mean that the invertibility is right will not be good.

Too much of a spread in the Eigen values right is also bad for example, 1 Eigen value being very high and the other Eigen value being very low is also bad. So, the best situation is to have Eigen values which are both reasonably good which is when which is when you can definitely say that the condition number have you guys read about condition number at all its like right sigma singular value of max singular value by the min singular value anyway right I mean not. So, important right now, but yeah, but the invertibility of A transpose A right invertibility of A transpose A right depends on the depends on these Eigen values of depends on the Eigen values of of A transpose A. So, λ 1 λ 2 right let us say let us say let us say too small right I mean λ 1 some less than less than some epsilon λ 2 less than epsilon epsilon very small not good what this means is that the variance is probably coming because of some noise and not not really because of some actual motion. Then λ 1 much much greater than λ 2 right not good either why does this happen when does this happen we have seen this before at an edge right we have seen this know at an edge what will happen at an edge you will have you will have right no no no the the the along the edge your Eigen values are very small right, but across the edge is where the is where the maximum action is right.

So, there this we already saw when we did a corner strength and all right similar thing here. So, we know that we know that right. So, which is like like right example is an edge this example at where kind of right this can happen is over a smooth region or low sort of texture smooth region where where the only action that is happening is because of some noise there it is not really that that that there is a big change in intensity or anything it is happening simply because there is a low texture there ok. Edge is where is where right this situation will occur and the ideal situation is λ 1 right λ 1 λ 2 both both let us say greater than some alpha where alpha is kind of significant as that means right these are both kind of significantly high right that is when that this is like high texture region. So, if you.

So, if you actually compute the optical flow right this is what you will actually observe that that that it is very reliably computed in those places where there is a lot of activity where there is a lot of texture and that is also true with the with the with the human being right suppose I showed you showed you a blank wall and suppose I suppose I moved it in front of you and then if I tell you how did the wall move you cannot tell because right you cannot make out what moved right because everything looks so smooth right only when there is texture you will know something happened you rotated it you translated it you will say something right. So, same thing. So, in optical flow when you want to kind of analyze motion patterns there should be something that should happen that is seen should satisfy. So, very low texture is bad you cannot make out much because all that you see is probably you know the effect of some noise out there and you know and that can be random and have an edge then again you cannot you cannot tell exactly because because like I said right along one direction right you do not have any information at all and the places that are good are any place where there is actually high texture and this is what is shown in the in the next this one let us go and then see some cases like that yeah let us see this situation. So, for example, right.

So, if you have. So, see this pixel here that is an that is an edge right that is actually sitting on that on that edge of the roof right and therefore, right we know that we know that one of one of the Eigen values is going to be very large and another is going to be very small comparatively and therefore, right that is not really a great place to kind of compute optical flow. This is also not a great place to compute optical flow simply because it is so homogeneous right that very difficult to make out what motion is going on there. The ideal place it would be kind of so we would basically write things like these where there is a lot of texture here here all these places these are all textured regions. Therefore, at all those places you will get kind of you know reliable estimate of the optical flow.

I think I will stop here today.