Modern Computer Vision

Prof. A.N. Rajagopalan

Department of Electrical Engineering

IIT Madras

Lecture-86

So, we saw right optical flow, we saw that there was this Lucas-Canard method right that can that can deal with large shifts and there was one for a small shift and then we had a variant for that which could handle large shifts and so on. Therefore, it automatically there is a lot of interest rate in terms of you know having deep networks that can save this earth that can solve the same problem. And right I am going to I am going to talk about 1 or 2 1 or 2 1 or 2 most sort of say relevant works. This is this is you know a 2015 paper by the way I mean right both these both these works right which I am actually talking about are roughly by the same bunch of authors just 1 or 2 have changed but really right. So, and one of them is actually Thomas Brocks who has done some very good work in optical flow in terms of classical method before. So, it is not surprising that he took the plunge and came into the you know deep network domain to see what he could contribute there right.

So, you see this there is a flow net S and then flow net C. So, S typically stands for simple and then C is for actually complex but C is also correlation kind of flow net and you can also think about C as core flow net core sometimes it is written or flow net C. So, the way this works is actually it is completely supervised okay and you might ask kind of how does one do optical flow in a supervised way. So, the way they they actually do is they actually created a flying chairs sort of a data set and what that meant is you know they kind of assumed assumed a chair right inside a inside a room and then they would apply affine motions different kinds of affine motions independent of affine motions on the independent because you are kind of thinking of a camera moving around and then the objects flying around which was called flying chairs right.

So, they apply a different affine motion on the chair a different affine motion on the on the scene and then because they know exactly what and then because they have a they have a 3D model right. So, they can tell exactly where what is going in fact in fact you know that is whether it is one of their one of their sort of main this one the contributions other than the C network and so on. You would create a data set like that where you can handle occlusions because what happens is when you actually apply an affine motion right what will happen is something that this fellow is hiding before it will will start to appear in the in the next frame because you moved, but then what was originally appearing here will start to not you would not see in the next one right. So, they are going to take care of

all of that. So, they kind of handled all of that and that is why right it is called the I think these days they call it the chairs data set and and it is actually pretty simple right and one of the things that you want to do is this kind of a down sampling.

So, they have something like a like you know a contraction part which is which is which is up to this entire thing is kind of a contraction and then there is a there is actually a refinement which is the which is the expansion part and why do they want to do contraction expansion the I mean same thing right you want to be able to one is you do not want to kind of take the high resolution image all the way inside okay because the computation is going to be costly. Second thing is that you want to be able to get aggregate I mean you want to aggregate right what is around around right in terms of the context. So, the context that you want to get as much context as you can and that is one of the reasons why you down sample right. Of course, you can do convolution max pooling right this is what they do okay. So, like a con max pool con max pool, but then this is this is one of the original papers.

So, what they did was when they when they came to that sort of you know end of that end of that contraction part right they had to go back to the original image size because optical flow is exactly the same as the image and then you have to find out what is u comma v for every one of those pixels. The other ground truth with u, but then this it has to actually predict at that at that higher resolution because you have come down in terms of resolution. So, this is the refinement part right they did something right. So, this is the refinement part right actually looks like this. So, what they do is in the refinements of the last layer right that that kind of see comes from comes from that comes from that you know down sampling sort of you know blocks.

So, what they will do is. So, they have the they. So, what they do is they I mean if you simply interpolate that right back to the original size it will look very smoothed out right it would not look nice. So, what they do is they actually they actually take the take the take the input or take the output of the output of the of the next previous convolution block. So, it is like con 5 must have been must have been the earlier one then con 4 is even earlier than con 3 and then all the way to the input right that is how you have to think about this.

So, what they do is they actually up sample this con 6 by a factor of 2 because they know that they come down by a factor of 2 which means that it will match the say dimensions of con 5 output and then they will actually fuse the 2 then go and then go to the next level and then where they will bring in con 4 because con 4 will then match the size of size of the output here right. So, that will match the size of the output here and then and then right when you go like this you go like this and so on and then and then and then right and then eventually hit the highest resolution which is which in this case is some say 96 cross 128

or whatever and that is where you actually do the that is how you do this is a refinement. Then then they had also this other network right in the same paper which they called as flow net core. Now if you look at here the input was stacked it in the sense that both the input frames for which you want to compute the optical flow they were stacked together and sent whereas in flow net core or flow net c right what they did was they had actually identical sort of arms okay in both in the sense that they wanted to they wanted to pick up feature maps that are that are independently independently you know that you can independently get for get for each image okay. Then they had a correlation they have a correlation block here which is actually which is which you do not have to train it is a usual correlation and this correlation is done right among the patches I mean so you have these 2 feature maps that coming out of the left and the right and not left and right the first and the second frame and then they would actually look for the correlation among the patches right as a function of function of how far away you know are they from each other and that correlation information then they pass through this kind of a flow net s I mean after this whatever follows is roughly the roughly the same as what they had what they had earlier the claim was that using this correlation actually helps you to you know improve improve the optical flow.

Again these are I mean you know deep networks as we always say right it is very hard to sort of write pin it down to exactly unlike unlike the unlike the analytical approaches right where where the where the where the where the inside is way higher right then you can get out of a deep net but yeah that is how right people people people the idea in deep networks is to be able to come up with something that works. So why you did that and then as long as you have something reasonable some reasonable explanation going right people accept it and and right now so here is that here is that kind of a synthetic sort of a data set which is called the chairs data set of course some of those look look very unrealistic because you do not have a chair right flying around and across the building and so on but then they could do that because it was all simulated and and then and then right so they had some augmentations and so on and then the the kind of then this is the this is the this is the optical flow which is actually a color coded optical flow then they also compared it with some epic flow and so on. And and then then they had a following network which was in you see 2017 this is called flow net 2 the earlier version was like flow net 1 and here the main thing was this right so they wanted to handle large shifts the earlier one could only handle small shifts so they wanted to build something that can do you know larger shifts right across the 2 frames. Similarly following just like what we did for Lucas canada we said for small distance displacement first and then how do we handle larger displacement same line of thought but now but now what you see is right they have a flow net c followed by a flow net s followed by flow net s and there is another network here which is meant for meant for small shifts alone. So, what they do is so they again right flow net c as you know right takes actually 2 of them into into independent arms that

is how it takes it whereas, flow net s we know that it takes a concatenated set of feature maps except that it is modified a little bit in order to take the warped version of image 2 then image 1 then the optical flow computed out of this large sort of a displacement network and then they also take the see take the you know brightness error.

So, the entire thing is this kind of given as a concatenated set of feature maps to the flow net s which in turn let us see predicts the flow which flow is then used to warp image 2. So, the idea is that as you keep keep keep right going inside the the optical flow sort of a sort of a difference will reduce right that means optical optical flow vector will itself go down in value because because you are already warping and sending warping and sending and then the idea is that for and then this network that they have here this is s d means small sort of a displacement there what they say is that I mean you know if if if in reality if you are if your frames did not have a large displacement then having this helps in the sense that because these are meant for large displacements. But then this one right it does not do it does not influence anything if there is a large displacement, but then when there is a small sort of a displacement then they say having this helps in addition right it does not have any influence when there is a large displacement, but for small displacements it is useful. Then finally, right the features coming out of this are fused with the features coming out of the flow net s d and then you get the final flow the complete completely it is completely supervised there are there are a couple of works you know that are unsupervised, but then right I am not even going to unsupervised things because those are way it is still a long way to go there. So, even supervising networks are still struggling and here are some more kind of see results and then you can kind of you know take a kind of look at this paper for those of you are interested and then mainly that you have to kind of look at the time also because when optical flow you want things to run in this real time.

So, you are looking in milliseconds right the flow rate too takes a little bit more, but then you can see that right the way it comes out with the with the movement it is very nice right. So, so there is a look around truth look at this and look at this right whereas, these are still struggling PCA at all is way away and this one is way away in time. So, all of that matters right time architecture the parameters the accuracy and so many things. Now, as far as semantic segmentation is concerned right. So, by segmentation what do we mean right I mean we really mean by semantic segmentation you want to we want to classify objects right I mean in the scene.

So, in this case right we want to give kind of read one label to the tree and then a label to the road right I mean. So, here is an automatic driving car or something and then what did you see right in front of it right seeing all. So, you have segmented the people out and then you have segmented whatever the road out and so on right this is what you want to do ok ideally and of course, you know you can kind of go back to all those segmentation

techniques that we talked about, but then really if you wanted to do it without with kind of a deep network then what it means is right you want to be able to label every sort of a pixel with a class label law right. I mean this is not like you know a recognition or a kind of a classification problem right. Here it is a kind of classification, but at the level that every pixel you have to tell right which label it should belong to.

So, you already have an idea right let us say you have got some you know 10 classes or something and then every sort of a pixel in the image you want to tell did it come from class 1 or class 2 or something and then once you do that then you have a segmentation map. Now, the you know the right the deep convert the deep networks that have been around right are the are somewhat like this. So, the initial works were on were on FCNs which are fully convolutional and again at the same idea this kind of down sample and then up sample this all goes back to the same argument that in terms of the resolution right you do not want to be operating at a high sort of a resolution one thing right because because you know that it makes it computationally expensive. Then second thing is in terms of the so, one of the things is the things is the things is you see what you call is the receptive field right this is this is very very important. So, you want a receptive field that is actually very high right because you need context.

So, for example, if you see a tree and then right if you see in the context and there is a road next to it then that you want to you want to get a see pick up right. So, that so, that next time you see you see an image with the tree then there is a context right that you have learned before that the tree right possibly appears with the with the road somewhere around it or with some soil around it and all of that is what is what we mean by the context right or if there is a person then may be right very likely that that somebody is typically next to him or for example, if you are inside a room then may be right there could be a table. So, all these things are actually right we are not we are not right we do not get us we do not get directly teach right explicitly, but then implicitly right it is supposed to learn and the only way it can learn is by kind of looking at the whole thing when it when it wants to segment a pixel should also be looking at the kind of global sort of a context right. If it looks very locally right then it is going to go to miss out on whatever else that is around it which can actually aid in the say segmentation task right. So, that is the reason why you typically have this kind of a down sampling layers and then of course, you know one of the things that you should remember is that how do you how do you think your final layer will be like I mean you know suppose I do this right I do some convolution I do some down sampling right I do some max pooling all then again go back to the original resolution because the segmentation has to come at the original image resolution right.

Now, how do you think will that final layer look like now? I mean when you when you did the classification and all right there was a there was a there was a certain way in which

you did it right. Here also it is a classification problem, but then it is actually a per pixel classification. So, how would your how would your output layer look like? It will yeah it will be it will be actually a 3D I mean not I mean 2D it will be a volume. See if it is 2D right then it is just one plane one feature map a 2D feature map, but in that pixel in that every pixel you want to be able to tell which class it belongs to right. So, if you have just one you cannot do that right.

So, you will have multiples of how many as many as the number of classes right as many as if you have 10 classes you will have to you will have 10 feature maps and for every pixel you will say able to you know with what sort of a probability right could that be in class one then what is the probability that it is in class 2 class 3 and so on and that you would do for you know every pixel. So, when you do a cross entropy loss or for example when you even do the softmax it will be it will be a per pixel softmax now. Along the depth. Along the depth right it will be it will be a per pixel softmax now right earlier we had only only only only this one right we had we did not we had a situation where we had only one vector it was like a one dimensional case now now you have like you know right vectors all over the place. So, you have to do a per pixel per pixel softmax followed by a per pixel c across entropy because because for that vector it will have a one shot right you will have for that for that particular pixel you will have a one shot vector is a ground truth which will say that 0 0 0 and then oh this is that class right to which it should belong 1 0 0 0 that is your y right y i right and then you have a y hat right which is which is what is the which is what is the networks output and you and you do that whatever it minus summation y i log y i hat right and that and that is that is your cross entropy, but it will be at a pixel level.

So, this you need to do for every this one pixel and you simply average these losses, but normally what is done is right people compute what is called in order to find the accuracy right people compute what is called the what is called the you know what is called the mean intersection of union. So, it is like saying that saying that right. So, it is like mean it is just the average over over over all classes, but for one class right imagine that you are you may know the intersection over union is like is like a intersection b over a c union b. So, so what this means is that may so a is really a target which means that I know for example, which of these pixels are supposed to be supposed to be from that class b is actually a predicted a predicted output. So, what it is saying is if so ideally IOE should be 1 in the sense that a intersection b should be equal to a union b if not if not right it will start to reduce if you are doing very badly then IOE is going to be close to 0 right and and mean IOE means over all the classes you need to kind of do this right that is that is kind of a way to kind of compute the accuracy because you cannot do like mean square error and all here right that does not make sense I mean if you want to know how well did I do overall I did I completed the training.

Yeah right I have completed my training and I need to know how well did I do. So, the way to so that metric that they use for segmentation is what is called mean mean mean intersection mean intersection over over over union value and here right and as you see right it kind of goes down and then and then afterwards right again of course, you know you need to do kind of some kind of an up sampling and and you know which which these fellows. So, yeah. So, how they how they up sample is again is again somewhat similar to what we saw in the optical flow paper right.

So, what they will do is. So, you have come down all the way to kind of pool phi right this is like max pooling layer and then after pool phi what they will do is. So, one of the things will be like a will be like a full blown a 32 x right which is your final resolution ok and then what they will do is they will and then and then what they will do is they will they will they will then scale this up by in the ok where is this 2 x up sampled and 16 x 1 minute. So, so what they will do is you know. So, for example, right.

So, here here. So, so, so, right this map right you actually up sample it by 2 x and then you actually fuse it with this with this you see pool 4 layer which will have the same sort of resolution as this as this up sampled output this is what you saw in the optical flow also right which means that which means that the that the output is now 16 x up sampled and then and then right this one you take and then and then up sample it by actually 2 x and and I read that that output right will come from here. So, from this pool 3 right you will take this you will actually fuse it because these 2 are at the same sort of a sort of resolution and then and then and then right I mean that is how you go right in order to be able to arrive at the arrive at the final sort of a segmentation map which you will compare with the with the some ground truth. So, this called the skip layer concept this is not the best way to do things I think you know this must have probably come from the idea of the way right in optical flow papers the way they did it I do not know which one came first and which one came later, but the ideas look very similar. So, what happens is right. So, so, if you do one skip connection then right it is not so good.

So, so 2 skip means you are going further back 3 skip will mean you will go further back and then integrate information from the from those pooling layers and the ground truth of course, you know. So, you can see that this still not so good right because you have still not got the hands and all properly segmented there is a lot of smoothing smoothing going on right because you have lost lost this detail here you have lost some details here and there right. So, not not a not a great output, but yeah at that time I think it was still considered good. Now then came this kind of you know deconvolution network.