Modern Computer Vision

Prof. A.N. Rajagopalan

Department of Electrical Engineering

IIT Madras

Lecture-28

Now, the final one that I wanted to talk about was actually a ResNet. So, this ResNet is when did this come 2015. So, you can see lot of activity in that space of time. And ResNet means a residual net, ResNet, ResNet right. So, this is actually, so ResNet means a residual net. So, actually, so ResNet is actually an interesting idea right.

So, if you go and see, what where is this ResNet, let us go straight away jump to that here. Again, again right we just have to look at what is actually a key idea. So, the key idea is this block, a residual block. Now, you have what is called DenseNet, now you have residual dense block, now you have residual in this residual dense block, I mean after this read so many have come up again.

I mean, so that is why read this subject beyond a point right, we just seem to lose control over it. Because, read people just keep on adding things and then something works and then you just have to accept it right. But the, but then right this idea is neat. See what it says is that, I mean if I had something like you know HFX to learn, if I had to learn this mapping from X to HFX right, one way is, one way is I just put a lot of layers learn that mapping. What these fellows did, one of the first thing that you observe is that this group, this is Microsoft at that time, now this guy I think is with Facebook.

Now it is 152 layers right, that means you cannot even think about it, 22 layers where was 22 and this guy talks about 150 layers. So, first of all you first think vanishing gradient right, how do you even handle that. Then one way to think about is just as in the other one right, you had an alternate path right, you would actually think that if you give, if you gave an alternate path maybe you know, maybe you could still you know make sure that the gradients do not tie down. That is one thing. But then the point is right, what was actually found was, see I mean if you just used a network right without this kind of a residual block and all, if it is simply trained of 152 you know sort of a layer this one network right, it would simply fail.

One of the reasons is of course you know definitely vanishing gradient. The other thing right which you might think is probably right with such a with vanishing gradient I mean definitely with depth right, this is a problem vanishing gradient. But then the second one which you think might be a problem is what, if I have such a deep network what other thing can happen. Large parameter. Large parameter, no which will then mean that you are trying to write over fit right.

I mean you have 150 layers and it is like saying that I have a simple data that I need to fit and then I am putting a very complex function through it right, which means that exactly to that over fitting problem where I am using a ninth degree polynomial to just you know fit a few points right, something like that can happen. So that is what they thought probably is happening. It is over fitting and therefore it is not working. But then what they realized was it was not working even on the training data. See if it is over fitting then on training it should do well.

That is when you say it is over fitting right. It is fitting very well to a training data and just fit so well, it fits so well that you know the moment you come out of it and give an example which is a little kind of say different, it completely fails. That is over fitting. That was not what was happening right. What they found was it was not an over fitting problem because the training error itself it was doing badly.

Now, now another reason right what they were trying to offer was well why should it not work because if I had 152 layers and suppose let us say right imagine that actually 40 layers is all is all you need. So one argument was why cannot the other layers be a simple identity right. I mean you know why cannot this network just put 70 I mean not like in consecutive thing but then in between it they just become identity mappings. So that if at all it needs only 40 layers maybe it should just use that 40 and then right rest of them should be just identity. Now all these arguments were there.

So, so, so just look like then why, why does it not work right. But then this vanishing gradient right is actually a first problem. That is definitely one of the problems. The other problem was what to learn. See right this, this thing is very important.

What should you, what should you expect a network to learn because some other time right when I give you an application problem and I will actually talk about what it means to sort of right even ask what should I learn right. When it cannot be like every time I have a bunch of data I just throw it into a block and then hope that hope that it learns what should it learn right. If you have a better clue of what to learn then it will it will surely help your cause. In this case right what happened was so, so, so this group right came up with this idea of a residual block. So what this meant is if you had to learn h of x.

Ok now they know the reason right the explanation that was offered to not being able to learn probably an identity mapping is because they said that you know there could be so many solutions possible and probably the network does not automatically know that hey here is a very simple answer to this right. It is not obvious to it. It may be obvious to you that hey just just put identities in between. But then it is not obvious to it. To the network it is not obvious when it is solving such a big complex problem.

It is not obvious to it. It looks like it is not obvious that it can find out the other answer which could be simply in a use simple identities apparently right. That is again I mean these are all

these are all explanations right that are offered to sort of say as to why why this network rate was not working maybe the reason is that it does not know that the identities could simply be used in between. But what was actually clear was was this notion of identity which which emerged from this from this idea. It is 152 layers in between if you had identity right would it have worked.

So the so that word identity right came from there. And then the other thing which came was actually a residue. Now this this is a residual thing was actually well known in the in the other community when you do a compression right image compression when you go from one image to another right. So what you do is when you find out what has changed you do not code the whole image it does not make sense because you know if the if the scene has not changed then why code the whole thing because it is exactly the same scene. So you would only only only find a change.

This idea was was kind of well known right in let us say compression community and so on. But here they do not mention about that by the way right in the paper they never say that this idea came from there and all. This is their argument the identity part. Then what they said was if you wanted to learn you see h of x which is a mapping that you eventually want what they said was push and push push a skip connection this is called a skip connection. That means you offer an alternate path straight to this output and then whatever is the output that is coming out of your network add it to this.

Then what it effectively means is you know if this is some f of x right that you are learning effectively your h of x becomes you see f of x plus x right that is what your h of x is. Therefore what you are learning so effectively what this network is now learning is actually f of x right that is what that is what this portion is. So f of x is simply the original h of x minus x right that means a residue a residue between between between the actual x and then the h of x that you want if you just learn sort of a residual it is enough rather than learning learning this entire h of x right. So h of x I mean you will anyway get it but then through through a residual learning. So this so this notion of residual learning came from here.

So you just learn h of x minus x that is far more easier than learning h of x and this skip connection right gave it so this skip connection was given as an argument that if you wanted to propagate the gradient right there is now an alternate path for them to flow right because if they had to come in one line right then then probably it is not at all possible because it has to go through 150 layers but now there are there are say alternate paths right which can which can which can allow for the gradient to flow right. Of course you know this and all they eventually showed that all of this work right it is all easy to talk about it once somebody has done it is always easy right but then you know but for somebody to come up with that idea and say that right residual learning makes a lot of sense and then and then you know this kind of identity. So skip connection came from there so the idea skip connection was first introduced right you know in this kind of a network ok and this one with 152 layers right they could make it work it was 3.5 percent right top 5 error so like down from 6.4 or

something that you saw for you know so for Google let and you also see that right it is it is swept all challenges in fact that that period of time in 2015 segmentation detection whatever wherever you throw this ResNet it would win and in fact this this is a ResNet also has a kind of another is a I think ResNet 34 again right it has again depending upon the number of layers right there are various kinds of kinds of say ResNet for your information I thought I will put yeah ResNet 34 ResNet 50 ResNet 101 ResNet 152 ResNet 152 is the one that went to do that challenge.

So again ok these ResNets you can again use them you can you can the feature maps that come out of them have been used heavily for a for you know for you know other tasks ok then finally right there is something called actually a DenseNet so this and already we will see ok this is all what I said already ok this is a residue then I don't know whether this has a DenseNet ok well it doesn't have well you know a DenseNet right looks like looks like this the final thing right that I want to just talk about was actually a DenseNet ok with this right we will be done and then a DenseNet right looks like ok I thought I would have a figure right to actually tell ok so it goes like this right so you have an input ok and then let's say right you have a conf ok you have a conf block I will just take two minutes to finish this just to get you to know the idea then you have a concatenator you concatenate then you again have a conf block something like this ok this is how a Dense block would look like then you have a concat then let's say you have a conv and so on ok and then this goes on then you have a concat ok now what it does is right I mean what so what it does is every so you know a DenseNet right it's called DenseNet because it what happens is so this input right you actually concatenate with let's say every other every other every other output ok so it goes from here it goes and concatenates with here again right the output from here for example, after the first layer right this will actually concatenate where is this this conf right so this will concatenate with this this will also concatenate with that and similarly right this will go right and then this will actually concatenate with that so the idea is that right you have ok these arrows are all like this pointing in that sort of a direction so but you notice that it's not an addition ok it's not a residue in that sense it's a concatenation and it's called Dense because right I mean you know it's going all over the place right from so wherever you are at the input right you have a skip connection to read every output block then from the first output block you have a skip connection to every every every other output block and similarly you go on right till the end and this is called a DenseBlock now this DenseBlock so you know a network constructed like this is actually called you know a DenseNet and this is 2017 ok this is by Huang this is like you know 2017 and this again right has done excellently this is again another network I mean all these architectures right that these people have sort of you know been able to be able to arrive at and this DenseNet again right has been shown to in fact you know it's a 50 layer ok and it outperforms ResNet this is just 50 layers by the way this is a 50 layer network and it outperforms 152 layer ResNet on the same challenge again ok and and then right following this there have been ideas where let's say people have used a residual dense block right this is like a dense block that is like a residual block I mean what do you think a residual dense block will be like you will have concatenation plus the addition ok you will have a residual learning you will have concatenation both kind of put together is actually residual dense

block what are called RDBs then people came up with the residual in in this one residual dense block ok I mean you can just go on and on and on but I'm just saying that you know well one doesn't have to go into all that in a proper deep learning you know if you had the whole course for this we could have done all that but just just we just wanted to understand that that right these are the these are the main architectures which you will encounter there will be there will be things that that that that let's say people built on top of these but really right these are the ones ok DenseNets ResNets GoogleNet one cross one Inception module one cross one convolution right all max pooling right all these ideas are the ones that kind of led to various architectures ok so I think with this whatever we had to cover under CNN right under this sort of a quick recap right we are done and next class we will do RNN just one class and after that we switch we switch to a traditional computer vision ok so I will see you on Monday.