

Deep Learning for Visual Computing
Prof. Debdoot Sheet
Department of Electrical Engineering
Indian Institute of Technology, Kharagpur

Lecture – 47
Regional Proposal Networks (rCNN and Faster rCNN)

Welcome. So, today we are going to take on what we had left in the last lecture and that was on activations. So, one of the whole aspect which you understood around with those activations was to find out, where an object is located on an image. And today we are going to take down, another breakthrough which came up and it has some sort of a basis built up on top of your activation maps, and this is called as region proposal networks.

So, what they essentially do I these are networks which while they can classify, and image they can find out or they can classify an image into multiple classes; whether multiple classes of objects are present in the image or not, on the other side of it can also point out exactly where that object is present. Now, this is this might sound something similar to an activation map over there, when a activation pooling where you could actually find out as a hotspot localization on a with different kind of a probability on the whole area over the image where and particular object is present.

Now, if we come down to these kind of region proposal networks; and here what happens is that, you start getting down the region proposal or which is more of as a rectangular bounding box. And these kind of bounding box can be present at different regions of the image. Now, say that one image has a small boy who is holding a banana in his hand, and he is sitting in a boat. Now, you basically have three objects, and this whole image when it is given down for classification and if your system is permit permitting you to do a multi hot classification, or it can give down more than one classes has the number of a class outputs over there.


Then it will basically be giving you all the three classes, so there would be a boy, there would be a boat and there would be a banana over there. Now, if I ask you to exactly find out where is that banana located, where is the boy located, where is the boat located; then that is a problem which comes down into your region proposal networks. Now, these build up heavily on top of the features which are learned on for your classification itself, and use them for doing it out.

(Refer Slide Time: 02:19)

NPTEL ONLINE
CERTIFICATION COURSES
Indian Institute of Technology Kharagpur | Department of Electrical Engineering

Object Localization

Brushing teeth Cutting trees



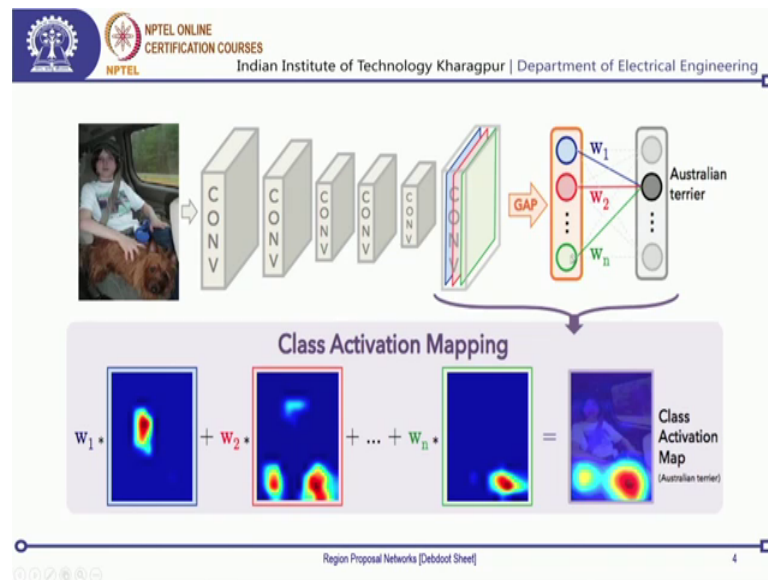
Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, Antonio Torralba, "Learning Deep Features for Discriminative Localization", CVPR 2016.

Region Proposal Networks [Deebdot Sheet] 3

So, let us do a brief recap of what we had learnt about in our earlier discussion with activation maps. Now, the whole idea was that if you have a particular action, or it can be even an object class then where is that object class present.

So, if I was asking you about this action called as brushing teeth, then where which particular parts of the image would signify something, which is associated with brushing of a teeth. Or if I asked down to point down cutting trees, and where what particular actions are, different zones on that image would be something which would be pertinently, referring to a cutting of a tree as an action.

(Refer Slide Time: 20:49)



Now, how it was doing was quite straightforward that you had your image, and then you had your convolutional neural network created down for classification. Now, just before linearizing and flattening out all the convolution layers over there. So, at the last point what you do is, whatever responses come down as convolutional results, you flatten and get them into a linear structure. And then this structure is fed down into subsequent layer layers of a fully connected neural network and to get down an output.

Now, here what we do in this case is that you just do subsequent convolutions, and then at the last layer where you get down different channels coming out of it. You actually average out, each of these channel to come down to an average value, which is represented as the value of this neuron over here. And then you just do a classification over here. Now, with this classification learning part over here, is what will be changing down these different weights and the weights get adapted accordingly.

Now, that as in these weights are getting adapted over there, so what you end up getting is, each channel has some sort of a relevance associated to a particular object which it is classifying. So, as in over here if I am taking this class as an Australian terrier, then certain number of weights will be coming as high; some of them will be coming as low, some even go down as negative weights as well; Now, if I take all of these weights and try weighing down each of my channel outputs, and now if this convolution was, so there



can be two kinds of convolution over there. Either you are reducing the x y or the spatial size of the input image or you are preserving it as it is.

So, if you are doing with strided convolutions or with max pooling kind of an operation, then your spatial size of the image is going to get reduced. However, on the other side of it where you are not going to have any kind of a max pooling operation; or you do not have a multi stripe, you are you are just convolving with the stride of one. And adding appropriate paddings over there; then the resultant x y size of what comes out as the final activation is, going to be the same as the x y size of the image itself.

Now, in either of these cases what you can do is, you can basically weigh down each of these activation maps, which comes out over here; which is the result of convolution. So, there will be n number of convolution kernels, and each of them is going to produce, one 2D map coming out. Now, if you weigh each of these 2D map with these weights, and then combine it out then for a particular classification problem, say this is for an Australian terrier. You find out what region of the image is going to look at it. The then this does not give you a bounding box, that is a major problem which comes down over here; is that you get some sort of a probability given down on the pixels over there.

But if I ask you that, can you give me a bounding box, such that, I segment out only this rectangular region. And I have one object represented only within this rectangular region. So, that is something clearly which does not happen within the activation map, and that is the basis of what we do with this region proposal network.

(Refer Slide Time: 05:32)



NPTEL ONLINE
CERTIFICATION COURSES
Indian Institute of Technology Kharagpur | Department of Electrical Engineering

REGION PROPOSAL NETWORK

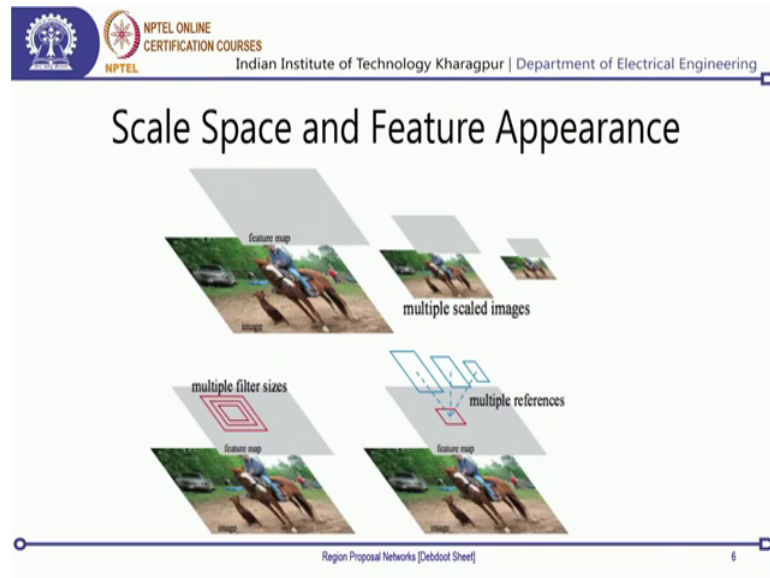
Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." In *Advances in Neural Information Processing Systems*, pp. 91-99. 2015.

Region Proposal Networks [Deepest Sheet] 5

So, this is based on a paper which was published in 2015 in NIPS over there and so, what we are going to do today is a network, which is called as a faster rCNN. Now, there were certain computational tricks applied over there, as well as the model was made simpler. And it was found out that even with the simpler model, it was working out much better and much faster, and that is the faster rCNN model.

So, what it does is it is also two problems together; one is it does the object detection in real time, as well as it also finds out where the object is located or the region proposal over there. So, these are the two things which we will be doing.

(Refer Slide Time: 06:12)



Now, as it goes down and it takes its, basis is something which is historically known as a scale space theory. Now, typically say I can have my image and then if I go down via some convolutions over there, and I would come down to something which is called as a feature space. Now, if you go back to week-1 of our lecture, then you can pretty much recall then we were doing down some kind of feature based learning over there. And what it was essentially trying to solve is that, if you have an image, and I try to compute per pixel certain extra attributes based on the neighborhood around that particular pixel over there. Then that is what is cumulate will be termed as, just a feature coming up.

Now, here one thing can happen is that if I am densely computing the features per pixel over there, then I get down a feature map; whose x y size is the same as the x comma y size of this image over there. Now, this image can also be scaled down which is you can reduce the size of the image over there. So, you can have a scale reduction; on both the directions by half, which will make the whole area or the total number of pixels, reduce by one-fourth. You can go down even further, say if I make all both the directions over there, one-fourth then my total number of pixels or my area wise reduction is coming down to one-sixteenth of my area.

Now, accordingly as I come down to this lower area. I can still be computing out my features, and they are also informative features over there. So, this kind of a mechanism is where say there is a as in you see, a horse and a dog; now the horse and a dog can be

even smaller in size over here, and still how do you figure it out. So, that is something which this scale space here it is with. Now, on the other side of it what I can do is, I can find out another alternative option in which what we do is you have this image, and when you are computing out these features. You always take a you take a variable window size over there ok.

Now, one thing is if I compute it on a pixel by pixel basis, and get my feature map coming up it over there. Then my x y resolution of my feature map is the same as that of the image. Now, if I do a stride of one was a stride of two over there, and try to do it down. Then the x y size of my feature map is going to be a one-fourth of the x y size of my image, now if I compute the same features with a stride of say 4, then it is going to have a area wise reduction, or a number of pixel wise reduction of 16 factor over there.

Now, these are important now as I also look down into my reduction over there. So, if I am taking the multiple strides I would still like to encapsulate, as much as majority of neighborhood information. And that would mean that my window size of my feature detector, or my encapsulator over here also needs to be really wide and really bigger, so that is what we have over here.

So, if you are looking down at reducing our feature map size to smaller and smaller. Then we should have a bigger size window coming down, as and when. Now, this is one side of it where you have feature computations; which are dense at every region over there. Now, the other point over there is what is your multiple reference proposals over there, so what this multiple reference proposal basically says out is that. you can have one neighborhood over there, but then instead of taking only this fixed neighborhood, you can also think of taking a few neighborhoods around this neighborhood.

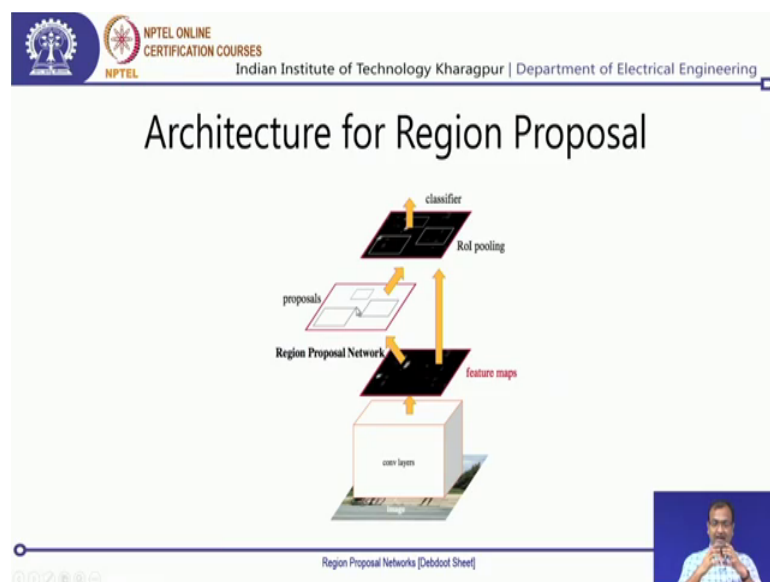
So, if I am taking a small 5 cross 5 region. In order to compute certain feature, then in this earlier case; I would be having a 5 cross 5 region, I will be having a 7 cross 7, I will be having a 11 cross 11 these kind of regions. But here if I take a 5 cross 5 region, now on that 5 cross 5 region; on the top and bottom, I can take a few more regions or I can take some narrower region over there. And then these will act as multiple references which come down.

Now, typically when we solve down a CNN kind of a model, that is something which relies on these kind of approaches over there. So, what essentially we are doing is, you

have your image and then you are convolving, you are doing a max pooling which is reducing the size and then subsequent convolutions and going on. Based on whether you have residual connections, you have dense residual connections you would see these sampling operations, bringing down your features at different level of hierarchy.

But nonetheless this does not look at different neighborhoods, around that region over there. It is it is just a few block of pixels, which gets keeps on getting reduced; and your features are deduced. Now, this other point where you can have multiple reference frames or multiple reference blocks, in order to infer about the object present in that image is, what is the basis for region proposal networks.

(Refer Slide Time: 10:45)



So, let us get into the basic architecture which this particular model says, so what you do is essentially, you have a CNN. So, you start with an image, you have your convolution layers over here; which is within your CNN you take it out.

Now, your CNNs are going to give you certain feature maps ok. Now, this feature maps are nothing other than just activations associated with each channel. So, if your output over there, has a some 6 channels or maybe 16 channels or 256 channel. Then for a given pixel location, if I am considering along this number of channels over there, then any given pixel is represented as, this number of channel times dimensional feature vector.

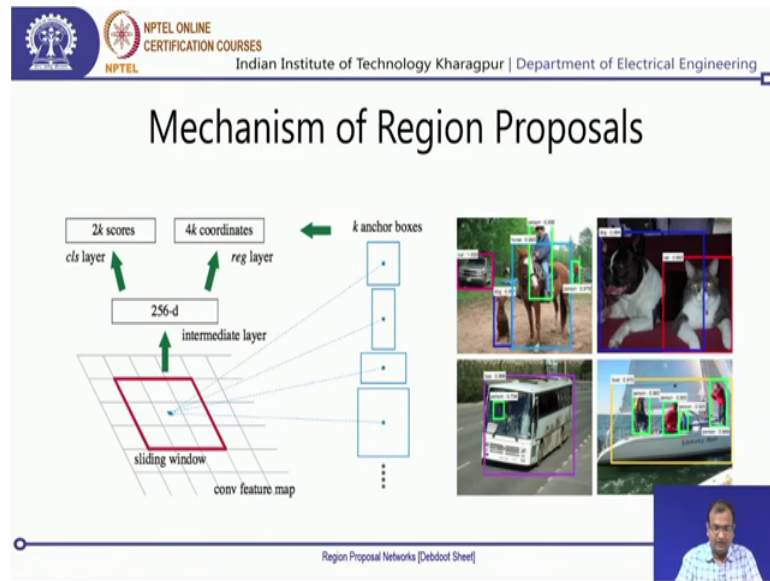
So, say I take down my LinNet. So, my LinNet is going to have 6 channels in the first convolution kernel. So, essentially any pixel gets convolved, over there by 6 different convolution kernels, and that will result into 6 channels of output coming down. Now if I take a given pixel on my channel space, on the activation space of my first convolution layers output; So and that that can actually translate down to a pixel on my image as well. Now, this pixel on the 6 channels over there, is going to be represented in terms of a six-dimensional feature vector, and that is this feature map which comes down over here.

Now, based on this feature map what we try to do is, we try to come up with another mechanism of learning down to get down these kind of proposals generated ok. So, here what is it is going to do is, it will take down a few random guesses initially about some randomized boxes, and then based on the features which are computed within these boxes. So, there will be a bunch of pixels, which are present within this box. Now, it will take out an average value or variance or it can even take down all the features over there.

Now, this dimensional feature vector which is they are present within this box is what is going to say, what is the object present within that box. So, essentially if I have a scene, I try to divide it down to different number of tiled out region; they can be uniform tiles, they can be overlapping, non-overlapping a multiple different different kinds of them. So, that is that is pretty much up to the user how they will try to do. And then within each of these blocks, based on the feature present in this feature map, it is going to say what is the kind of an object present within that block ok.

Now, once it has set out that part. So, what you can do is you can take this region proposals together along with the feature maps, and put it down into another classifier over here, and this is a fully connected network. So, essentially you have a network where your input image goes, you have a few convolutions going down over there; then as a result of this convolution, you chunk out certain part and truncate it into a feature and into a region proposal network, you get some region proposals coming out. You take the other arm from all of your features and then you do a final classification of what is present on the scene ok.

(Refer Slide Time: 13:37)



Now, this is essentially the mechanism of what we are looking down at, so essentially you have your convolutional feature maps. And then if you take one particular window, then you get down say over here if I have 256 channels, on my convolution then it is going to give me a 256 dimensional vector corresponding to any one of these pixels. Now, if I want to take this 3 cross 3 pixel region, then I can take an average value over there; So, that that is also pretty much well and good for my purpose.

Now, that I have this average value of it taken down. I have this 256 dimensional intermediate representation. Now, from here I can go into a classification layer which is just try to classify out, what is present in this image over there; or I can find out these coordinates of my region proposals, and within this particular box what is the particular kind of a proposal, I am trying to generate and give up ok.

Now, if you look into these example images over there. So, on the first one you see that there is a car, and then it gives out a score; what is the probability that it is a car, now it is a car with the probability of one.

So, there was not any ambiguity as I generated. Then there was a dog, but then there was some sort of a confusion. So, it does not generate a probability of a one for the dog, but it generates a probability about 0.967 over here. Then there is a horse, but then there is a person riding on the horse over here, and for that reason the probability of just it being a horse is not that high as one, but its 0.993. So, this is what you essentially see, and even

though you have different objects present over there, some of these objects are overlapping. If you look purely from the bounding box perspective, still it can actually find it out.

So, as in over here that there is a cat and there is a dog. It finds out actually both of them though the cat has that there are certain pixels over here of the cat; which are overlapping with the pixels of the dog as well ok, but nonetheless these are region proposals, so there is not much of a difference which comes out over here. Now, this is an interesting one, where there is one proposal within another proposal; so you have a bus and you have the driver who is driving a bus. So, the driver is a person and this proposal is located within the proposal for the region of the bus itself. Then over here, I have a boat and these people over here, so the people are sitting on top of the board. So, person predictions and it also predicts out the board.

So, this is how mechanism of region proposal essentially works out. Now, from your understanding part it is not too complicated, because the first part of it is still just a simple CNN model which comes out over there. Now, what you need to essentially do is as in the activation mapping act, global activation pooling for finding out activation maps. We had truncated out certain convolutional layers to do it. So, here also you are going to do down, at the terminal convolutional layer. And then from there you eject out and per pixel level feature vector. And then you create some random number of region proposals created over there, which are just rectangular bounding boxes.

Now, within this box whatever are the pixels you find out what is your average value of the pixel and that is a feature, which represents this particular box over there. Now, you need to train another classifier. So, for the ground truth you definitely need your region proposals, some sort of a reference region proposal; and the object in that region proposal. So, you cannot just have a overall classification available for that image. So, this is the first difference which comes down, with respect to when we are looking just at global activation pooling.

However, having said that though it is it is much denser data set which you have to create over there. In the earlier case of an activation pooling it was much easier, but then the only downside with activation pooling is that, you do not get this exact region proposals

or how objects are contextually related with each other which is something which comes down over here.

(Refer Slide Time: 17:20)



Now, based on that let us look into few of these results. So, you can see this result where you have a portion, you have a you have two people. So, one person is standing in front of the horse; one person is riding. So, there is a little girl who is slower riding on the horse, and then you have the horse, and then over here you would be seeing some of these cars present over there. And typically you can see that, the cars aspect ratio is much lesser. So, the number of pixels the car takes is much lesser than the number of pixels all of these objects are taking. So, if I were just doing a plain simple classification, it would have possibly classified this as a horse. And then cars or people would not have figured out.

But, now with this kind of region proposal networks, if you can densely annotate, what all are the different objects present over there, then it can actually classify out and as well as give down a rough location around where. So, this is trying to solve down two problems in one single shot. So, while you are trying to give down, exactly what is the nature of the scene, and also where are these objects present over the scene. So, it is not just what constitutes the scene, but where is the constituent object present on the scene, so that is that is the whole purpose of these kind of one.

So, if you see at a bird then there are this, so on this image there is a bird who is perched on the hand of the person; and then trying to eat something over here. But then there are some birds who are much lower below over there. And for that reason, these birds are of much smaller size. Now, based on these random proposals which get generated, and they are tuned over there it within the network, so you can actually come down to where is that particular object located. And it despite it being really small over that you can still find it out.

Then, there are two cows over here; one is looking down at the camera and trying to really pose out for the image. You, find out this which occupies majority of the image, but then there is also a secondary cow over here, and even these kind of networks, can find out this smaller kind of cows. Then if there are people who are overlapping, and then occluding part of them, it finds out a pretty good definition of a bounding box as you can see. So, although this part of the person in this green shirt, who is stand standing behind is occluded, but still the region proposal box is wide enough to get down. ah

Similarly, happens for this gentleman in a blue shirt that a significant part is occluded between these two people over here, and still you get down a bounding box, which comes out quite efficiently. So, one good thing is that while it gives out just bounding boxes and stuff, and quite unlike the activation map where if you try to do it with this image, you would be getting down certain activation maps, but then all of them would be coming down together. So, counting down how many unique people are there, and where they are exactly located what is the geometric span becomes a tougher job, so that is that is not possible within activation pooling in anyway. So, this is what comes down with our region proposals.

(Refer Slide Time: 20:00)



Now, you can have some of these like really skewed out examples over there, one of them is where there is a big car really big, or you can have a region proposal which can encapsulate the whole image itself; So, for this cat like the whole image is as such the cat. So, the region proposal is itself the whole image over there. So, there can be these kind of examples as well, and it makes it quite easier it is it is robust to occlusions, to the size of the image, to the size of the object over there. And agnostic to a fact that it can be the whole image itself and still the region proposal is fixed out over there. So, these are some of these interesting aspects, which go around with the region proposal network.

Now, based on these we would be solving out. So, in tomorrow's lecture I would be demonstrating out how to get down activation maps and working down. Now, for reducing the complexities involved we are not doing a region proposal network, but there are plenty of resources to implement a region proposal network as such, and work it out, and your activation maps are something, which come down as initial start into how to implement these kind of more advanced networks.

So, with that we come to an end for this lecture stay tuned and.

Thanks.