Modern Computer Vision

Prof. A.N. Rajagopalan

Department of Electrical Engineering

IIT Madras

Lecture-88

So, this is these are architectures for object detection. So we have already seen the recognition task right, where you have something like let me just open. So for example, we have seen that now you can do the recognition or what we call as a classification task right, where you have for the entire image you have one label right. So you say that you know whether this image contains a tiger or whether it contains what it contains. Then the even next level is when you want to classify, so this is like a classification problem. Then the next one right that you can escalate it is by saying that you know classify + is a localized right, which basically means that you know you also have to put this on a bounding box right around the object of interest.

Then the next thing right that you might ask is what is called a detection object, this one a detection which is what which is what which is what is of interest to us today. And here what it will mean is that is that you know not just one object that you could have actually multiple objects, some of them could be same. For example, you could have 2 dogs right within the same, but then you need a bounding box around each one of them and you also want to be able to tell what they are right. So you see a multiple boxes + labels + labels okay.

And then right you can go further up where you can have you know this one a segmentation okay where it is not just a bounding box right, you do a pixel level sort of you know a segmentation and this itself you can actually escalate it further, this is what we saw as semantic segmentation right, this is what we call as semantic segmentation where you know instead of instead of label for the entire object right we are doing label at actually a pixel level and this is semantic segmentation. Then there are further improvements over this you know so what is called instant segmentation then a panoptic segmentation and all that okay. Our interest is here object detection okay and a traditional approaches right that have been around I mean you know of course you know these are sort of now these are not so important anymore right given the fact that so much has happened with the use of deep networks right. And traditional approaches typically will be like you know one of the famous ones is actually what is called you know Viola Jones okay, this is a Haar wavelet based kind of you know a detector works very well for actually faces, it can also do for humans and so on but actually very works very well for faces.

Then in fact you can also use your SIFT in a particular way in order to be able to do object detection then you can do hog histogram of you know this is an oriented gradients.

So all of this is possible but then these are all still very sort of you know what you call handcrafted and you know and they have issues in terms of robustness and all of that. And therefore the trend right in a sense has been in the last 10 years or so right the trend in yeah mostly maybe in the last in fact 7, 8 years has been to use actually deep networks for this task of you know object detection. Now one of the ways right you know in which you can think about it is suppose I have an image and I have some objects right some lying here, some lying here okay there could be many for that matter. But thing is right if you the problem is right we do not know where those objects are right it could be that you know it does not have anything of interest or it could be that it has objects of interest but then we do not know where they are. Therefore one way right which is also a traditional way right to do is for example use what is called a sliding window right.

So what that will mean is that we keep a sliding window keep on examining what is happening under it and then right somewhere when the sliding window right encompasses this object at that time right you will know that I mean you know right that there is something interesting out there and then you could say that well you know there is a particular object class there and then maybe right when the window comes here right you could have a same sort of you know situation and so on. But all of this is too slow I mean you cannot take a window and then go all over town right if I try to figure out you know where is my objective interest. Therefore deep networks have taken sort of a different approach and there is a set of 3 papers that came almost from the same set of authors. So these are called RCNN, fast RCNN and faster RCNN okay and these are all they have that R because these are all like region convolutional networks in the sense that right they do not go through the sliding window based approach one of the key things that they do is generate what are called actually proposals. A proposals which basically means that you know given an image a proposal would be that something that is likely of interest and let me also tell that right I am not going to write much because everything is available okay I am going to only explain basically how these things work and this in this particular lecture right we are not going to be focusing too much on feature map sizes and all that right it is the philosophy that is more important the network details and all are available okay.

What you want to what I would like you to understand more is in terms of how these things work okay and so this is the idea is to idea is to sort of flag in a sense right regions that are that could potentially be of interest it does not mean that every region proposal will have an object under it but then these are region proposals in the sense that these are proposals that are likely to be interesting that are likely to contain an object of interest it does not tell which object it is. The region proposal is not there to tell what is the object it

only tells that you know here is a region of interest and then you can have a subsequent network right which can work on those regions alone on those regions that have been flagged as proposals and it could just act on those and these proposals are typically rectangular shaped boxes and they can come in various sizes and various shapes by shapes I mean something could be tall some could be square some could be flat and elongated whatever right. And the idea is that these rectangular boxes can then be plugged out because these are actually these each one of them is a kind of a potential proposal and therefore can be sent through a network in order to find out if it actually contains anything of interest okay that way you cut down on time in a big way but still this is still not the fastest but then right that is the idea okay. So now the first network that came is called you know RCNN right I mean you know which sort of created waves and that is the one here okay. Now if you see right how does it work so for example see there is a proposal stage a region this one a proposal stage which basically means that right this is something okay which is supposed to flag like I said which are all these which are all the interesting regions in my image.

Now this is not a part of part of RCNN this is a module that comes from outside this is not this is not something that you can train for this network okay this is not trainable okay it comes from outside and in the sense right the way it works is you know there is a there is actually there is something called a selective search which is actually a module okay which came this is actually a paper in IJCV 2015 okay and those authors right this is more kind of a traditional approach where you kind of generate what are called super pixels based upon color, texture whatever right shading commonality whatever right when you want to call shape, size right based upon all those things right what you do is given an image right you try to so for example you know so here is an example. So what it will do is you know so it will try to try to merge so you have seen what is sub pixel right this is just the opposite way you go to what are called super pixels that means you actually merge pixels that kind of seem to actually look alike right and the idea is not to go into the selective search algorithm but selective search algorithm is something that can actually that can identify blobs or regions right that things could all be these pixels that have something in common. The commonality could be in terms of like a size, color, texture, shape whatever it is right and based upon that right this selective search will flag and it is not needed it is all there in the paper okay if you go to the paper you will find all this information but the key thing is to understand it how this thing works. So this region proposal stage is something which will actually flag regions of interest and typically given an image it actually outputs of the order of 2k proposals that means it is about say 2000 you might wonder you know is in kind of 2000 a large number actually it is not so large compared to doing something like a sliding window imagine we had a 500 cross 400 image and if you did a sliding window at every location looking at 500 to 400 that many that many proposals right compared to the 2000 is still a very small number and these proposals

come in various shapes okay they are not uniform okay some could be square, some could be rectangle, some could be like this but they are all actually rectangles of different shapes and sizes okay. Now all these come now the problem is what happens is right now the RCNN what it uses the backbone is actually is actually an say AlexNet okay now the Alex so the idea is that right I mean this is actually you know a pre-trained right this is an AlexNet pre-trained on let us say the you know typically you know it is pre-trained which is on your image net database which has you see 1000 classes and then actually a million images and all that which we have seen before.

Now what it is trying to do is and in fact this is also the reason why this work is one of the most important ones because this is the first one to show that you can use a pre-trained network which was meant for a different task for example that AlexNet can never localize right you can really tell you know whether it can really give you a label for the image right it does not tell where that object is it does not know anything about localization but this was the very first work to actually take up a pre-trained network something like an AlexNet which is which has a very good feature representation. Again why do we pick AlexNet because that right that has been trained on millions of images for of course a classification problem which is typically a 1000 object class 1000 class you know sort of a classification problem but then it has learnt a representation which is very strong. What they do is you know they actually so what they what these people do is they do a fine tuning on top of that because their classes right could be something else right because it does not mean that you know whatever is there in image that is what is your class of interest. So what you have to do is right you will have a bunch of classes right that you are interested in okay. So let us say right you got you know n number of object classes which is your data set and that you want to kind of you know build this RCNN for your data set and you want to be able to localize right where those objects are inside an image.

So what it will do is you know if you have n number of object classes it will create a vector which is n + 1, 1 is for the background okay. So n is for the object classes, 1 class is for the background. So now what it will do is this output layer right where you where the final layer for the output classification will come right that 1000 class thing that you had in AlexNet is replaced by this guy which is simply an n + 1 kind of a class and the you know entire thing is trained. But to train one of the things that we realize is that a CNN right along with the AlexNet along with the fully connected network at all expects a fixed size input right which is typically 227 cross actually right you know 227. That is why you have a warping stage here warping function now from here onwards actually the whole thing is RCNN by the way but then the region proposal stage is really an outside module which is being invoked right.

So this is coming from outside which is being invoked and then you have an affine image

warping which will actually take all these like I said that these proposals could be of different shapes it will warp them all to the same right 227 cross 227 which is what the AlexNet can take as input and then and then right all of this will go all the way and then what happens right so for the time being ignore this whole part okay ignore this entire part and just kind of see right look at what is happening here right on this arm. So on this arm right what is happening is you have one network right that is being independently trained with respect to this you see n + 1 object classes. Now one of the things right that I have to mention is that when you want to train right so the batch size is actually 128 okay so the batch size is actually 128 out of which so it means that these boxes right that you have you actually 128 of them okay in every batch size I mean so you got like you see 2000 boxes with you right so one batch size is like you see 128 of them and out of which right 32 are supposed to be the I mean you know supposed to be positive and whatever the right rest of them right and okay 128 then what is this 96 huh so 96 these numbers you can probably write check in the paper but then the idea is this. Now negative and let us say right a positive right what do we actually write mean by that now you know that you know that right given my image what do you know for example if I have an object somewhere right then I know that there is a ground truth bounding box for that right which I know correct because this and all is completely supervised right so one of the things right which I know is that this is that right around at this annotation somebody has done okay. So for example if you take a data set that is meant for this kind of a detection problem right then you will have these annotated data sets and what they will have is actually a ground truth bounding box right which will typically hug the object closely right as closely as possible so that it is nicely sits inside and then you will have a bounding box for some other object you will have a bounding box for some other object and then you also know the label of that object okay.

Now when you have a proposal network right which is trying to bounce off all these proposals so what it might have done is right about this point also when it found something interesting right it would have sent right one proposal like this another proposal like that another proposal like that it would have sent so many of them. Now when you say that there is a certain number of these positive sort of you know a proposal right what you really mean is that the bounding box right that you have for those right you know for those examples these bounding boxes are such that if you compute their intersection over union with the corresponding ground truth box sitting there right around that region you will get at least an IOU which is I think greater than 0.5 do you follow this? So what it is saying is that so for example if I have a ground truth box and then I flagged a proposal here a bunch of proposals I will take each one of them find the IOU right with respect to that only if it is greater than 0.5 then I know that probably right this proposal could imply this ground truth object okay in which case I also have a label for that then and I also know that this ground truth box is for a particular label right and therefore right you could have several

such objects and some of those specific proposals could also come from the same object 32 positive does not mean 32 different objects okay 32 positive proposals that means that you could have multiple proposals coming from the same object you could have multiple proposals coming from the same object sitting right in let us say various different places does not matter okay. Now that is what you send us a batch right and that is your batch and then what you are doing is you are actually training this kind of you know a classifier where again right I mean you know basically right what is the so every example goes in for which you know that the output vector should be this one hot vector because I know that right it should have this label and therefore right you will say that for this bounding box right this should be the object label okay that is how you would train it.

Now this does not take care of the fact that whether your bounding box is hugging the ground truth box very closely or something it only means that it has a good IOU beyond that it does not say much okay. Now the now this entire thing is actually fine tuned what that means is you do not really you know you know churn the entire weights and all you just use a very slow learning rate so that it adapts to your to this to this object class right that you have and once that training gets over then what is done is the step following that is something like is something like learning a classifier now. Now this is not the way you would typically do the you know deep network right typically you we always you know whenever we argued about a deep network we always used to argue that it should be end to end the features and the classifier should be should be together so that right for that classifier we know which are the most ideal features to learn this is been our slogan but in this case that is not the way it happens okay this is also its weakness. So what it does is now once this whole thing is trained right then what it will do is after the so just before the before the final output layer right the fully connected layer that you have I think it is a 40404096D right now that is that is the dimension feature that you get now what it will do is it will pass all so once the training is over right then then it will take every image for which it has about you know 2000 of these proposals but then right it will again you know pick those that have a good IOU okay only those proposals now what it will do is I mean you know what it will do is it will actually pass them through this network so as to be able to get actually feature maps for each one of them now that the training is over and what it will do is it will take all this feature maps and then now it will try to learn you know right this one support vector machines right which has this is SVMs. So the support vector machines are kind of sitting outside and what it is doing is it is doing one versus rest so it is like saying if you have n object classes you will have in fact in this case it is n + 1 right so you will have like you know you will have basically that many SVMs so for one SVM you train it for one object class the second SVM you train for the second object class so what this means is that when you pass examples right or feature maps to this for training so what you would do is it will be like you know one against the rest so the one will be of course you know certainly will consist of those proposals that come from that object right

that you know but then the ones which are actually negative or not simply the background everything else is a negative even other objects are actually right negative okay.

So right which then means that which then means that when you when you want these negative proposals you just look at look at all those IOUs right which have a value less than 0.3 right with respect to these proposals which are positive you take you take right all of them to be actually to be those proposals that can be considered to be the rest of the category so each SVM is learned like that it is like one versus the rest one versus the rest right and that is how you learn the these so that is why we call them class specific SVMs right so each SVM is trained for one object class. Now this is actually a lot of work right if you think about it but you have to store all these feature maps right each one is 4096 dimensional you have got 2000 proposals per image you got probably right you know 10000 images or something so you have to store all this to train the SVM so which means that there is a lot of memory error you need during training at least during training you need all of that you need to store them somewhere pull them out right train the SVM. Now and then the bounding box right we are still not done because the bounding box right could still not be hugging the actual ground root box very well right so what is done is then you train you train actually you know actually a separate right I mean you know this one sort of this one a regression this one a network right whose job is to simply take in so right this is supposed to be a 4D vector by 4D right what we mean is if you have a bounding box right so what they typically do is take the corner coordinate and then here the height and the width let us say 4 unknowns therefore the ground root box right we know has a corner coordinate and then has a particular width and the height and therefore what you do is so that 4 dimensional vector is what you will use an L2 loss. So you will say that this bounding box which I have got and of course you know there is something called actually a non-maxima suppression also that happens right which we have seen earlier there was a so we want to make sure that the bounding box right you know is learnt well okay and I was talking about this NMS right non-maxima suppression so this non-maxima see what can happen is you know so once you have a bounding box right I mean you can I mean around an object proposal you could have several bounding boxes right all of which all of which could look like a potential box but then around one object if you put 4 boxes right it would not look nice right therefore what you do is you do what is called a non-maxima suppression.

By non-maxima suppression what you mean is that among the boxes right that are trying to among these boxes that are kind of trying to flag a particular ground truth label you pick the one which has the highest IOU and then what you do is you find the IOU of this box with respect to the others and whichever has an IOU with respect to this box which is greater than 0.5 you remove all of them which very means that right that they are all very close to each other right all those all those proposal boxes which seem to be also right

flagging the same object you kind of you take the you take that intersection over union with respect to the box that has a maximum IOU and then if they have a number greater than 0.5 right you throw them all out okay that is that is how you do this non-maxima suppression which means that finally right typically you will be left with only one box there okay against that object because otherwise right it will not look nice if you put right two three boxes right around an object. And then the box itself right the actual where the box should be lying there is a coordinates of the box so that when you actually fit it I mean it should fit very well on the you know on the object itself that means that with respect to the ground truth box you have to do some kind of a regression and that actually right I mean that kind of a regression happens here. So, so really speaking right it has got it has got four different things there is a proposal thing which is coming from somewhere then there is one network which is being trained you know independently which is for the for these classes then it has SVMs which are then right coming in and then they are getting trained and of course you know we are not even looking at what is the best feature for these SVMs right whatever we learn okay from the from the network we throw them as feature maps into the SVM which is not the most ideal thing to do and then finally you have another network right which is actually which is actually doing this bounding box regression but RCNN was still was still way ahead of the right ahead of the rest in the game right at that time I think this is 2014 or something or 2014 right and therefore it caught the attention of people the main complaint was that it is very slow it takes about 47 seconds for one image of course you know in those days with traditional approaches also it was very common to run into seconds therefore it probably did not matter so much but then right people started thinking you know why should it be why should it take why should you have a network that it takes so long to do to do this you know object detection.

Now let me show you some of these outputs so right this is how you get your RPs region proposals and then right here is where here is what I meant okay with respect to that network right that actually does the does the classification so right so basically here is where so right here is where your final okay now here is where you have a detection class labels and here is where you have your final output layer that will that will have the number of classes that you want and then this 4096 and 4096 is standard right you know that is already there in your you know AlexNet rest of it is all actually AlexNet only okay and then if you look at it right so in this case right what it has flagged are you know some of those boxes are being shown and because it expects a constant size input right so each one of them will have to be warped right so you can see that you know this warping is actually not really a good thing to do but then right because of the fact that you are using an AlexNet you are using fully connected layers all the way to the end therefore right you are forced to input a certain size right that means you have to warp right so each one of these will get warped right through the warped region and then right this is what will happen if you do not do non-maximal separation and after you do non-maximal separation you will get a

courtesy bounding box around the object of interest and then right you can also have a class label I mean you can of course you have the class label along with the score right which tells how sure it is right about that particular object so here it is a dog right and then there it is a ball and so on okay. Now one of the things right that I said right really slows down right the RCNN okay is really this point that you have to actually generate these feature maps one thing is right it is not end to end right because you are actually training SVM separately you have a classifier right which is being trained you know sort of separately the feature maps are coming from somewhere and then of course the bounding box itself is again you know being done separately and so on therefore right it is not really the most ideal way to actually set up a network and then the second thing right that you want to actually look at is the fact that the amount of this one right like I said you have to store so many feature maps and all of that right so the same set of authors right they came up with what is called the fast RCNN same set in the sense that at least one or two people were common I do not know whether all the authors are common whatever right you can check that up. Now what do you think right you would have done I mean okay suppose I do not show this right now suppose I tell you right what would be the way out suppose I ask you right what would you do right if you wanted to come out of this mess if you did not want to sort of okay the region proposals let us assume they are still coming from somewhere okay. Now after that right the way the way the way right see look at the look at the number of times is the I mean CNN is invoked right in this case every this one a proposal comes you have to send it through the CNN you have to get a feature map if another proposal send it through the CNN get the feature map right it is like doing so many times instead of that what else right could we have done that could have kind of shortened this whole thing what I am saying is the way right you are actually extracting the feature map can you do something there what would you do there is one smart thing that you have to do. No what I am saying is instead of having to pass every proposal through the convolutional network to get a feature map is there a better way to given a proposal can I access the feature map in some other way that is what the fast RC and N does okay.

So it is like this right okay let me show you know a picture like this yeah right this is actually better see here is my image okay and then right and then of course this is my selective search which is an outside module right as I said which is flagging N of these region proposals something here right I mean you know something here and all right it is flagged. Now earlier right what were we doing right I mean you know we were sort of you know tracking each one of these proposals warping them right pushing them through the CNN to kind of get the feature map but if you really think about it right if you take let us say of course in this case they actually changed right AlexNet to actually you know a VGG net. Now if you actually look at the max pool layer right before the fully connected layers okay if you take the VGG net and look at the last max pool layer right before the fully connected layers begin you will have actually a convolutional feature map right I mean

after that it is all the fully connected feature map which is what we used earlier but now just step backwards a little bit and then you have the fully sorry not fully connected the you have a convolutional feature map. What is the advantage with respect to convolutional feature map? In the convolutional feature map right if I pick a point in the convolutional feature map I can tell exactly from which part of the image it has come because there is an operation that I have done no I have come in a particular way I can always right go back that way to trace from where it came with a fully connected layer you cannot do this because right everything comes in therefore at the whole image you are looking at the whole image there is nothing like a locality but the moment you talk about a convolutional feature map I can actually trace things backwards. Now if you think about it right this convolutional feature map has all the information right because I have taken the entire image right on the left I have taken the entire image and then for this I have a convolutional feature map right which is this whole whatever volume that I have.

Now if you give me a proposal here I can actually find out which is that volume that is sitting inside this which actually maps to that region correct you guys get this right that is the key in the sense that you know it is already there I mean you know so you do not have to go elsewhere to kind of look for where is that feature map that feature map is sitting as a right a cuboid right cube is all sides equal I always get confused so it is actually a cuboid right. So it is a cuboid and this and this cuboid is what is the feature map right with respect to that region proposal and if you pick another region proposal the shape of this cuboid can change in the sense that it is the depth cannot change but then the size right the kind of whatever the I mean if you think about it right a 2D side that you see right other than the depth that shape can change because what will happen is your region proposals are not of the same shape you have sometime a rectangle that is longish you have some rectangle you know a rectangle that is like this sometimes you have a square therefore if you try to locate where that where that cuboid is the cuboid shape can continuously keep changing depending upon your region proposal what do you do then if that is a problem what is the standard thing that we have learnt if we have if you run into some trouble like that what would we do? We did something very recently right did I not talk about SPP spatial pyramid pooling what did we do there right what is the motivation for that right we said that if I have feature maps right and you know and basically and then this is what I talked about last time right if you have an input size would size keeps on changing even if you have a convolutional network right the output size will keep on changing but then if you want everything to boil down to one size you do what is called spatial SPP right that means you basically divide the divide this into a fixed number of regions for larger image right you will still have a 4 cross 4 a smaller image you will still have 4 cross 4 regions except that in the larger guy the each block will be will be larger in a smaller image each block will be smaller that is all but at the end of the day you do a max pooling max pooling on each of those blocks you get like you know one value one value and then you get actually

sixteen values whether you have a large image whether you have a smaller image it does not matter right that is the idea behind a pyramid pooling but then we call it a pyramid pooling because you have you know different like we saw last time we had sixteen then we had eight then we had four then we had two then we had one right but you can have this on a pyramid now this is a special case they call it as ROI pooling this is a spatial pyramid pooling you can think of it as SPP but with then with a single with a single pyramid level I mean you do not have you do not have multiple levels so what they will do is you know so this so this volume that you have right they will squash it squash it all of them into into basically one one fixed size which you can do right it is exactly the same idea okay if it is of a difference it does not matter so for example if I had if I had one cuboid right which was like this okay whatever okay and then and then and then if I had another cuboid which is smaller okay then what I mean in both cases right if I am going to say that this should be divided into let us say 3 cross 3 this should also be divided into 3 cross 3 that is done right so now I do not have to worry about worry about the size and all right that is exactly the exactly this ROI this one pooling so that at the end of the ROI I mean I am not looking at I am not going to go through the dimensions at all those are all available the key idea is that right because that is one problem that you will encounter you will know where the feature maps are but then you will not know how do I handle now because after that I have got fully connected layers right which means that everything should go to that in a fixed size right I cannot change my size anymore but then that you can actually fix irrespective of with whatever be the shape of the RP which is a region proposal then you have a nice thing the other nice thing about this is that after that it has got two sibling layers right two sibling layers in the sense that one does a classification another would do a bounding box state of you know a regression that means you have only one loss there is one loss one which is a cross entropy loss that is for the whether it is for the object classification + a bounding box regression this is simply that 4 cross 1 sort of a you know a vector whereas right this is again those kind of see labels and so on and again that even here you need to say NMS and all of that right because you can still get too many proposals coming around the same thing but then the but then the key idea is this right the very very key idea is that the you do not have to store all this learn a separate SVM learn a different class everything can be done end to end except that this guy is still something that is coming from outside you do not have a network that can actually flag them for you it is coming from somewhere that is okay but then it is a smart move right all that see the nice thing about these things when you look back right at that time probably right it did not occur to many but if you think about it you know it looks like I am probably it is the most obvious thing to do perhaps right but at that time look at it right there is only few people that could actually think about it that is why always and then finally it is all over right I mean otherwise do you think that this is such a great idea that right one of us could not have could not have kind of right you know applaud our head or does not right what is it does not something fun I know so out of the earth thing exactly this is a kind of thing

pooling well we know how that happens right we know all that we need is where to go for the feature map I mean it is all sitting there and somebody had to see it. Sir, if the diagram do a transfer in. Yeah all of these all of these where you are doing this fine tuning. How do they realize that this won't be negative transfer.

Well no that is why that is why right what happens is right in this case in this case the idea is that the feature maps are already good right because it is already you know a pre-trained network the fine tuning is very is done very little fine tuning is all that happens you are not see I mean here what I am saying is just a fine tuning I mean even the learning rate is very low they do not want to do you see too much because it already works well and as long as it works on your set of object classes they are fine with it they are not they are not they are not going to reuse this network back in the image net or something right this network they want to use for their problem and what they have done is they have just taken a pre-trained whatever a VGG net in this case fine tuned it right end to end but then with a slow learning rate so that you do not really turn all the weights and all you do not want to make right you know too many changes and most cases right this is not the only time I mean right many times people do that I mean you just take the feature map that is already been learned you know through a sort of you know through a good sort of this representation but then if you are talking about continual learning at all this is not continual right continual learning at all if you say right then there are issues and in the sense that I have already learned with some n object classes now I want 5 new object classes to come in right and then and then you know and then you know I do not want to retrain I do not want to retrain with respect to the old classes maybe I do not even have those examples with me and then I just have these have these you know few new classes that I want to throw in but then when I train this it should not happen that it starts forgetting what the other n were right those kind of things but here it is not it is not that complex at all I mean just a simple case just take the take the feature representation already know that it works very well right it is been used for so many other problems just using it for this problem and in fact in most cases these object class these objects are far fewer in number limit right it is not like it is not 1000 or it is far fewer in number okay maybe you let me know what exactly you mean by this negative transfer learning right and then maybe right okay then we can think about it okay so that is what this fast R-C and N-S right alright. So then ROI pooling right max pool within each grid cell so that so that you get you get a kind of you know constant size feature map that you can then use.