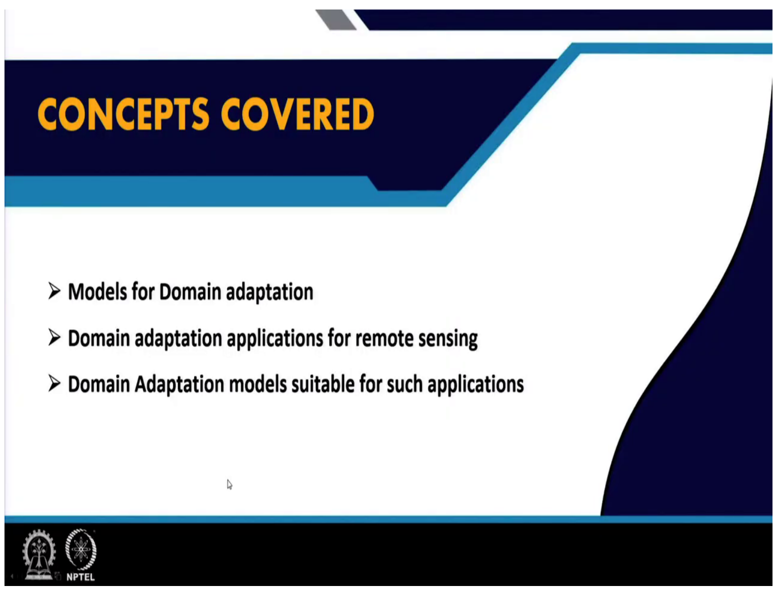
Machine Learning for Earth System Sciences
Prof. Adway Mitra
Department of Computer Science and Engineering
Centre of Excellence in Artificial Intelligence
Indian Institute of Technology, Kharagpur

Module - 04
Machine Learning for Earth Observation Systems
Lecture - 33
Deep Domain Adaptation for Remote Sensing

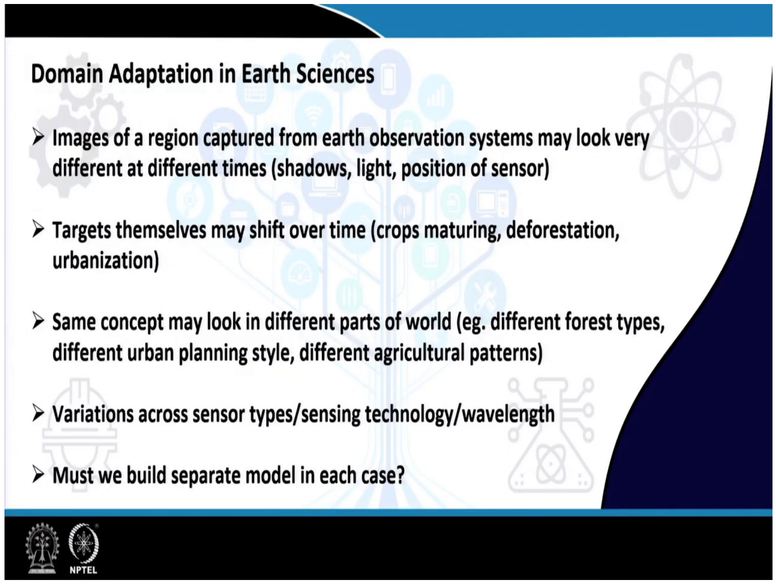
Hello, everyone. Welcome to lecture 33 of this course on Machine Learning for Earth System Science. The we are in currently in module 4. In fact, this is the last module of module 4 where we are discussing the various applications of machine learning in earth observation systems. The topic of this lecture is Deep Domain Adaptation for Remote Sensing.

(Refer Slide Time: 00:48)



In this lecture, we are going to discuss various models and concepts related to domain adaptations. Next we will see some applications of remote sensing where domain adaptation becomes important, and we will also see how the domain adaptation models can be tailored to such applications.

(Refer Slide Time: 01:07)



So, first of all why like why do we need domain adaptation in earth sciences? So, like now when we capture the images of a particular region using the earth observation systems, we have already discussed about different kinds of earth observation systems, about the different imaging technology they may employ and so on .

Now, when the particular region is being captured by different like devices or different kinds of sensors now the regions may look very different at different times. The I mean by times I mean the times in a day like in that like morning it may look one is in like it may look in a particular way, in the afternoon or mid day it may look another way, in the night it will be different and so on.

So, this happens due to the various shadows that are present, the light also the position of the relative position of the sensor corresponding to the object. So, like all these are factors which make sure or which cause the images of the same region captured by different sensors or even the same sensor at different times of the day different from each other.

Additionally, like over time with the targets then the target region itself may change. Say for example, if it is an agricultural region the like when it is barren that is before the crop cultivation or just after the seeds have been planted, it will look in one way. But when the crops mature the region will look very different especially when viewed from the top.

Then like there may be a forested region which gradually gets deforested, as a result the original or the early images of that region may look very different from the later images after deforestation. Similarly, for urbanization also. So, the this is to say that the in the first case it is the target remains the same, but it may look different in different circumstances. In this case we are saying that the target itself may change over time.

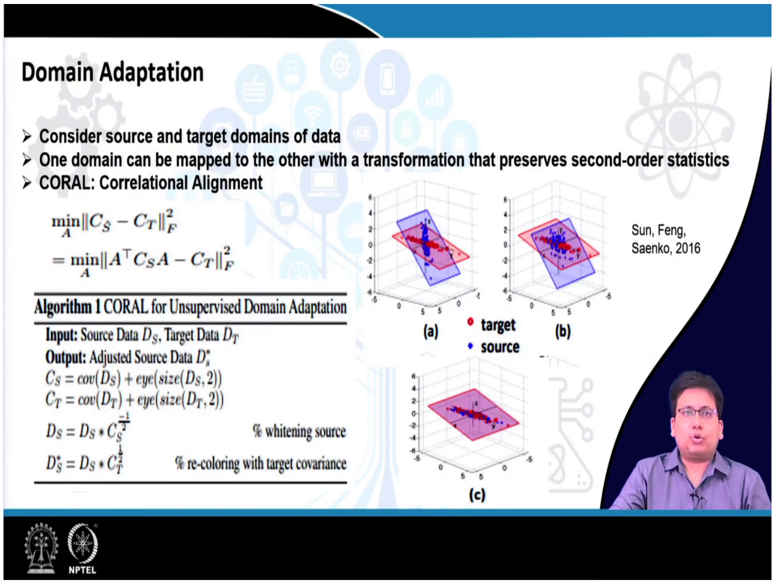
Also like we may develop the model for some remote sensing purpose for and train it with data from one part of the world, but that model may not work in another part of the world because the targets may look different there. Say for example, we will we have in some of our earlier lectures we have talked about LULC changes, we have talked about building detections in cities and so on.

Now, like suppose I train a model for detecting buildings in a city. Now, that I may train it on some cities which may be developed according to a particular urban plan. Now, suppose we deploy that same model on another city where the urban plan is significantly different. Let us say one has mostly tall high rise buildings, the other has short buildings which I mean or maybe less tall buildings which span a wider area and so on.

So, in that case the buildings may look very different in the 2 cases and the building and model to detect buildings in the first case may not work that well in the second case. Apart from that there may also be variations due to the different sensor types the sensing technology or we have discussed about the different wavelengths at which the images can be captured. So, all these are issues.

Now, so, like since there is so much variation. It is very unlikely that the same model is going to work in all the different cases. Like as I said suppose I build an LULC classifier or a building detector and so on and I train it using the data from one part of the world like there is no guarantee that it is going to work in other parts of the world or even at other times in the same region.

(Refer Slide Time: 05:20)



So, in that case should we always keep on building separate and new model? So, the answer is no and this is possible because of the id of this concept known as domain adaptation.

So, domain adaptation what it says is that like they let us say that there are 2 sources of data or let us say 2 domains of data. Let us say the source domain and the target domain. So, if you look at this image so, like the data the blue data points let us say they have come from one particular source and red data points they have come from they let us say they belong to the target domain.

So, as you can see that there is a difference in the like alignment of the data that is it seems that the source domain and the target domain they are characterized by a different hyper planes. Now, of course, for the like for viewing we have showed it in 3 dimensions, but in general such data will of course, be extremely high dimensional that we cannot visualize, but the basic concept remains unchanged.

So, what is the idea? The idea is that we will match the or we will map the source domain to the target domain in such a way that the variance or the second order statistics are preserved. That is to say I so, let us say the covariance matrix of the features in the source domain is C_S and that in the target domain is C_T. So, the covariance matrix as we have discussed earlier also, they basically encode the second order statistics the covariance's between pairs of features.

So, these let us say like each point which you are saying seeing here is a point in the feature space. So, they will have certain features. Yes and now that aim is that I like this C_S the covariance matrix of the source domain we will map to $\hat{C_S}$ which should be in the like as close to the covariance matrix of the target domain and that kind of transformation is to be achieved by this kind of like pre multiplying and post multiplying by the same matrix A, ok.

So, what is this is expected to achieve is like we basically the idea is that the source data points are being which are currently in this blue plane they are being projected onto this red plane of the target so that every source point will have a corresponding like transformed point in the target domain.

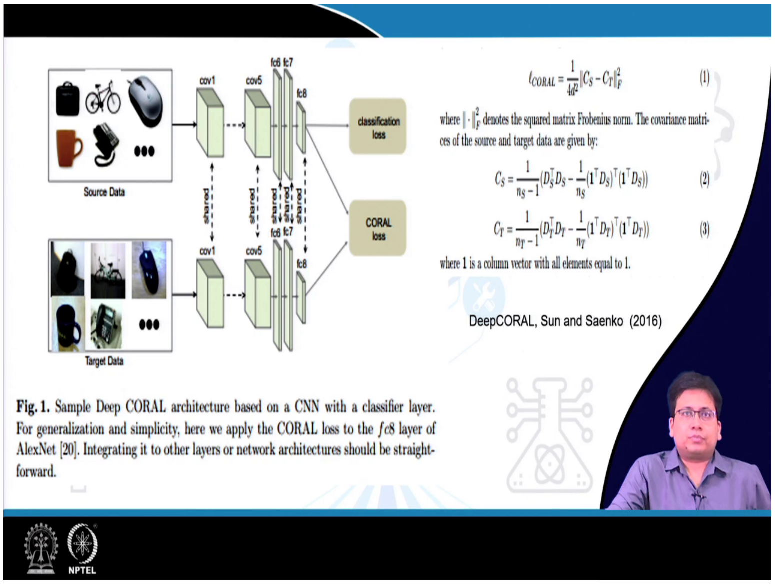
However, this has to be done like this cannot this kind of projection obviously, cannot be just like random projection some like it has the salient properties of the source domain the target domain they need to be maintained. So, that is achieved by making sure that the in that is with the basically trying to match the second order statistics.

So, I am the claim here is that like of course, like this is just the basic principle, but like if you look at the paper by Sun, Feng and Saenko in 2016, this is the this is known as CORAL or which stands for like Correlational Alignment like this has been mathematically proved that like if this kind of a transformation is done then the then certain properties of the source of the source domain is going to be preserved even after the transformation.

So, this is how the algorithm works. As you can see it is a really straightforward algorithm you just have the C_S the covariance matrix of the source domain and we have C_T the covariance matrix of the target domain and you just carry out this kind of transformation of both the like of the. So, that is you the D_S stands for the data points in the source domain.

So, you just do this kind of matrix multiplication and you get the mapped source or the projection of the source domain. And, then you can like once you have mapped all the source data points into the target domain then whatever model you want to learn you can learn it in that transformed source domain.

(Refer Slide Time: 09:46)



Now, this can also be achieved with the help of deep learning. So, there is a deep CORAL. Just for illustration purposes let us say that the source data is like looks like this and the target data looks like this. So, as you can see it is the same things the same objects, but seen in a different way.

In this case each of the source images like each of them focus on only one single object, but here that object is present along with certain background also. So, it is the same source I mean the source and target are like they are basically the same, however, the target has certain transformation namely the addition of the background. So, like it.

So, they suggest this kind of a deep learning architecture the basic idea is that you do not have to learn separate models for the source domain and the target domain. If you learn a model for the source domain like you can actually share most of it is weights with the target domain also.

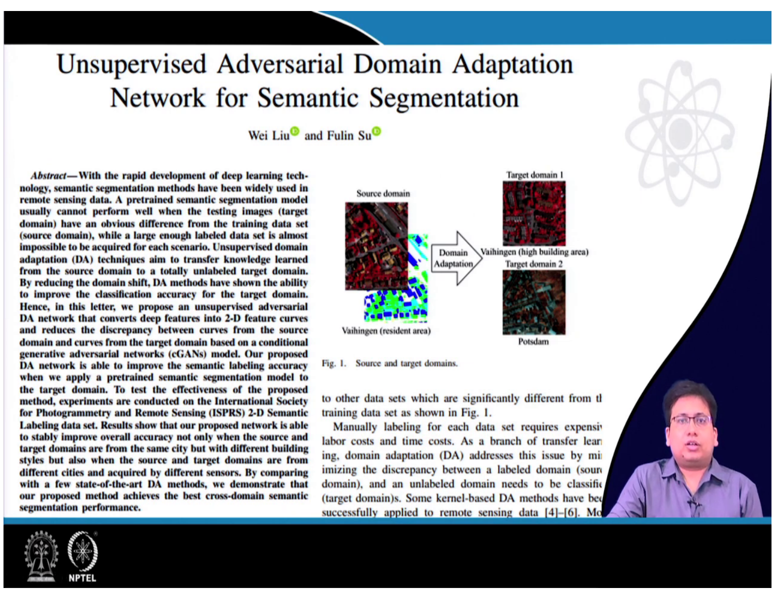
So, like we can say that we are building 2 neural networks one for the source domain and the one for the target domain. Now, the neural both like almost all layers of the neural network will share their weights except that we will have an apart from the usual classification loss which is I mean this is of course, an object classification problem.

So, there will be a classification loss apart from that and an additional loss function will be added that is called the CORAL loss and this CORAL loss is actually nothing, but the difference between the covariance covariance's of the different features the same thing that was done here. So, here also we tried to minimize the difference between the covariance matrices of the transform of the target domain and the transform source domain, here also we want to do pretty much the same thing. So, that thing is like this term is now included as a loss function.

So, basically the weights of this neural network will be learnt in such a way that not only improves the classification, but also helps in alignment of the 2 of these 2 domains. And, alignment in what sense? Alignment as say as discussed in terms of covariance matrix. So, like this so, let us say this initially the source domain and target domain were misaligned now like in 2 steps we align them to the same domain.

So, like we can say that like here the correlation the original correlation which was present in the source domain that is destroyed and then next the projection happens the original data points they are projected into the this target domain according to this particular property.

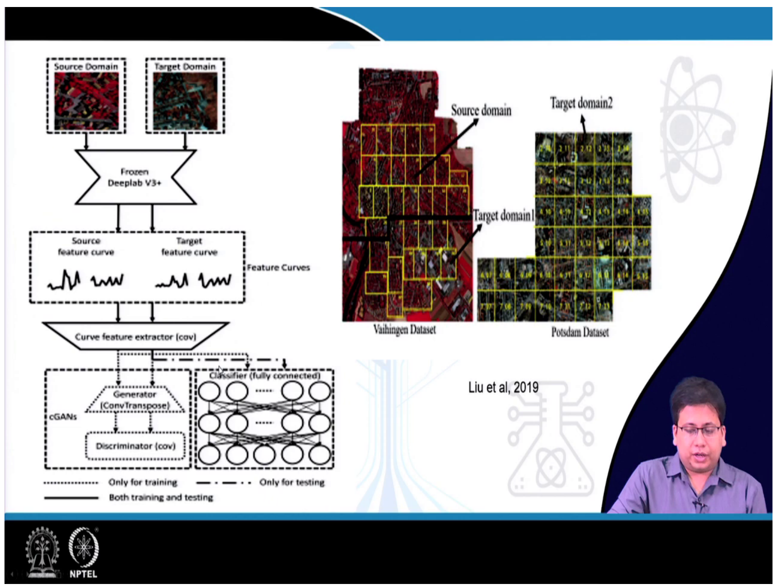
(Refer Slide Time: 12:45)



Now, let us see a few applications of this. So, let us first discuss this paper on supervised adversarial domain adaptation network for image segmentation. So, image segmentation we have discussed earlier. We have seen it is various applications in remote sensing and earth system sciences.

So, like here the idea is quite straightforward they have developed a model that is an image segmentation model for one region or one city. And, now they want to deploy that to our different to another target region which might be another city or which like. So, it may have like the buildings there may have different characteristics and so on.

(Refer Slide Time: 13:28)



So, for that they like the image segmentation approach that they use is called Deeplab this is also like a deep learning based image segmentation somewhat similar to the image segmentation models which we have already discussed along the lines of Unet and so on.

So, like as you can see like this is frozen that is to say like here the same thing it does like this might be trained in the source domain, but it will be used in the target domain without any further changes. So, like once that is done like we will get certain features from both the both of these domains.

And, next, we will have something known as a curve feature extractor that is like. So, basically what it is doing is from the it is basically calculating the covariance among the different features as was discussed in the like in case of this CORAL. And, then like using the machine learning the deep learning framework called conditional GANs like the mapping is done from the source domain to the target domain.

So, like GAN is a Generative Adversarial Network. It is a state of the earth topic or a state of the earth model in deep learning. So, we have not discussed this in great detail earlier. So, roughly this has a generator component and a discriminator component. The aim like the aim is that we will take an image or like we will take some kind of an input and then it will and pass it into a generator neural network.

The generator neural network will transform it into a data from our like and then there will be a discriminator component which will try to like classify whether the generated data point or like whether it is generated by the generator or does it come from a particular like a particular data set that has been on which the discriminator has been trained.

So, and this so, you like if the discriminator is able to successfully understand that or is successfully able to separate the generated images and the data set images; that means, the generator needs to improve. The generators task in this case is to produce samples which are near identical to the data set in like we are talking about. So, and it will like it will just keep testing it is ability to do that against the discriminator.

If the aim of the generator is to fool the discriminator and it will keep on training and retraining itself until it is able to fool the generator the I mean the discriminator. That is, the discriminator given a particular data point is unable to say whether it has been generated by the or it has been synthesized by the generator or it has just been taken from the data set.

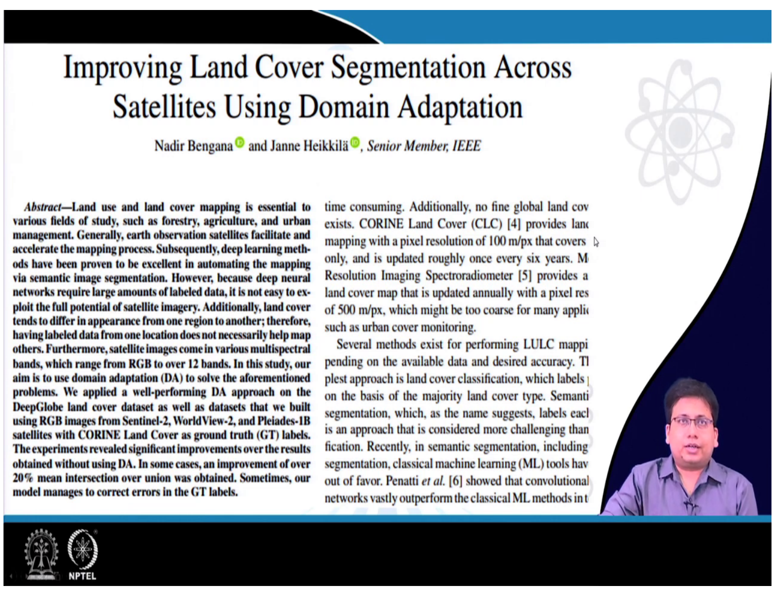
So, in this case also the same idea is used. The generator function in this case takes an image of the from the source domain and tries to map it to the target domain and then there is a discriminator which supervises how good this mapping has been by taking every image and classifying it whether it belongs to the target domain or it is like merely a transformed source image.

And, that generator will keep on training and retraining itself until it is able to fool the discriminator. When the discriminator is fooled that basically means that the generator can now transform any source image into the target domain so that the like after transformation the source image looks near identical to any target domain image. I mean it will not there will not be a one to one mapping that is not desirable also.

That is to say one image from the source domain will not be mapped to another image of the target domain, but rather it will be transformed in such a way that there will be no way to I say that this is this does not actually belong to the target domain it has merely been obtained through transformations. So, this concept of generative adversarial networks this is very cleverly used here in the task of domain adaptation.

So, here you can see the results. So, the like this is these are some images from of like of some remote sensing over a particular city in Germany. So, that is the like as you can see here like it has a particular structure and now, the and this is the target domain. So, like so that so initially the model is trained on source domain then it is deployed on target domain one which is a different part of the same city and then it is deployed on target domain 2 which is a different city.

(Refer Slide Time: 18:40)



Now, this is one application of namely image segmentation. Now, let us come to another paper which focuses on a different application. So, here the aim is improving land cover segmentation across satellites using domain adaptation. Here the keyword is across satellites. Let us say in this case it might have been from the same satellite, but over 2 different cities; in this case we are talking about across satellite that is the 2 satellites may use a different sensing technology and so on.

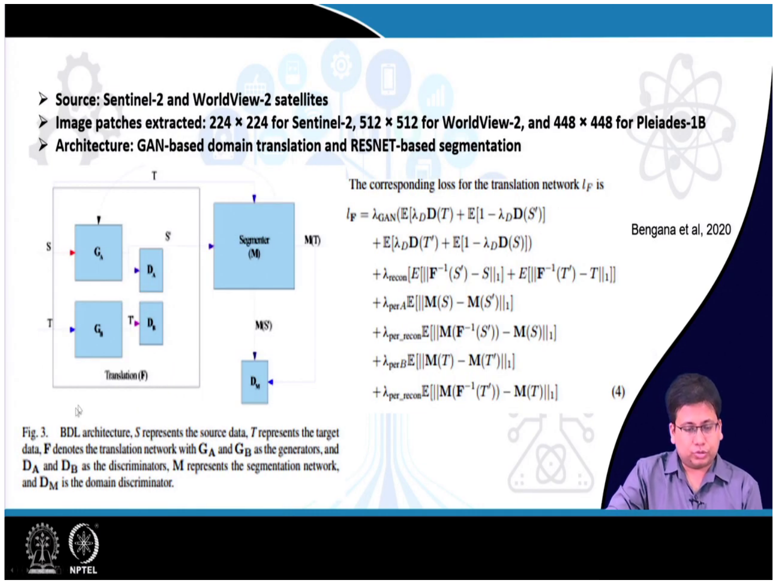
So, land use and land cover mapping is essential to various fields of study such as forestry, agriculture and urban management. Generally, earth observation satellites facilitate and accelerate the mapping process. Subsequently, deep learning methods have been proven to be excellent in automating the mapping via semantic image segmentation. However, because deep neural networks require large amounts of labeled data, it is not easy to exploit the full potential of satellite imagery because of the absence of labeled data. I mean data is present in plentiful, but what about the labels.

Additionally, land cover tends to differ in appearance from one region to another. Therefore, having labeled data from one location does not necessarily help map others. Furthermore, satellite images come in various multi-spectral bands in which range from RGB to which range from RGB to over 12 bands. In this study, our aim is to use domain adaptation to solve the aforementioned problems.

We applied a well-performing DA approach on the DeepGlobe land covers data set. So, I guess in one of our lectures about like segmentation we have talked about this DeepGlobe data set. So, like where they have like basically land use images from different parts of the world and the it has it defines certain challenges like identifying different land covers, extracting the road and water bodies and so on like land cover data sets as well as data sets that we built using RGB images from Sentinel-2, WorldView-2 and Pleiades-1B satellites with CORINE land cover as ground truth labels.

The experiment reveals significant improvements over the results obtained without DA. In some cases, an improvement over 20 percent mean intersection over union was obtained. Sometimes our model manages to correct errors in the ground truth labels also.

(Refer Slide Time: 21:18)



So, like basically this is the approach. So, like there is the so, in this case so, unlike here if you can see the segmentation takes place. First the segmentation takes place and then the mapping takes place. That is it is really the segmented image which is mapped from the source domain to the target domain. In this case, it is the opposite here the translation happens first and only then we put it to the segmented image.

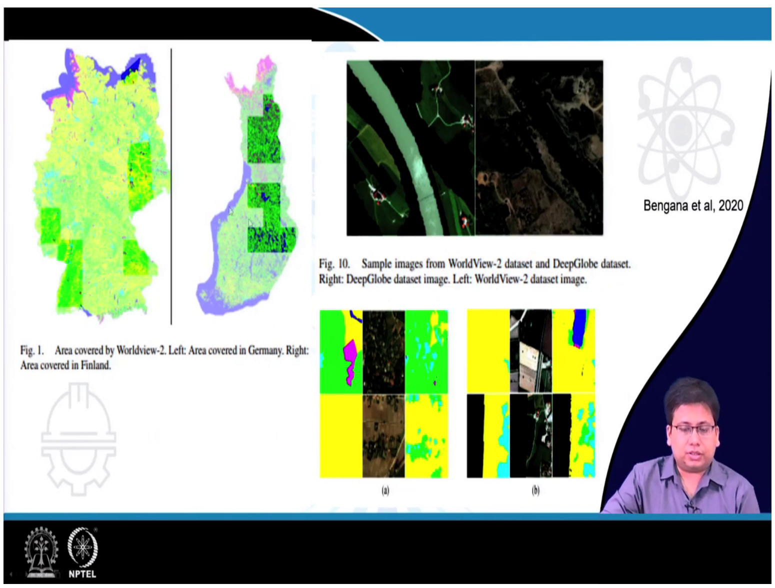
So, like the different image patches that are extracted they are of different sizes in this case depending on which satellites is being used. Now, and the architecture is the there is this domain translation this is done again using a generative adversarial network which we discussed just now and after that like I like the like using the pretty much the same idea that is the source domain images are mapped to the target domain images in such a way that like the discriminator is unable to distinguish between them. So, that is the translation part.

And, then the this translated data they are passed on to the segmenter which is based on the concept of RESNET that is basically a network with residuals with skip connections and so on and, finally, we get the like the segmented image. So, like they have this D_M which is which stands for basically the domain discriminator which is like given any image it informs like which domain it belongs to the source domain or the target domain.

So, like so, in this case the discriminator is used in like once in the translation phase and once after the segmentation phase also just to add another further level of security. And, when like when the aim is of course, to fool the discriminators that is these generators as well as these each of these generators try to fool the corresponding discriminators and the model itself tries to fool this domain discriminator.

And, the and a loss function is defined accordingly like this with the GAN loss that the reconstruction loss and the then for like for every each is like for the source as well as the target domain there are these all these loss functions and so on which we will like basically they indicate the how good the segmentation has been and all that. So, we will not go into the details of all these things, but this is how what the some of the results look like.

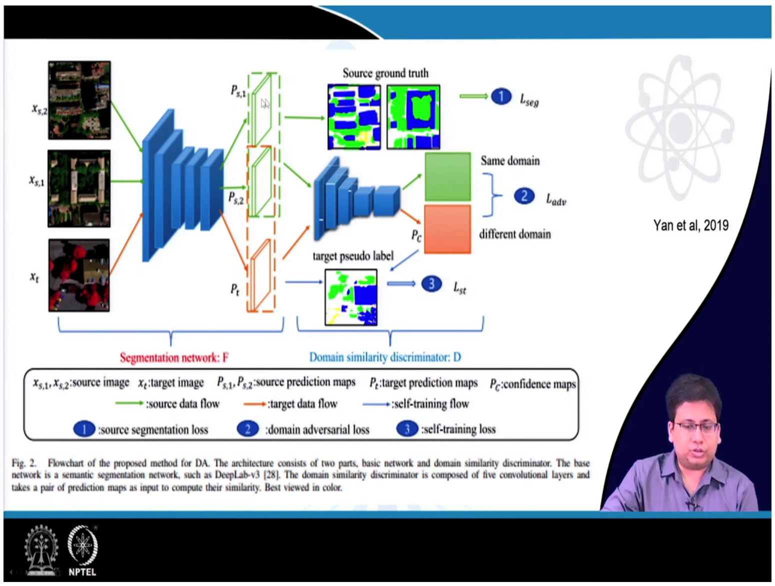
(Refer Slide Time: 24:01)



So, we have so, like the like for example, these are some ground truth images. This is one image from the WorldView data set and this is another image from the DeepGlobe data set. And, then after segmentation this is so, like here you can see like these are basically what we have on the left hand side these are the ground truths the ground truth segmentations and these are the segmentations which we get like after the like using the model. So, we can see that in some cases the segmentation results which we get like agrees with the ground truth in other cases it does not.

Next like we move on to another paper. Here this is based on the triplet adversarial domain adaptation for Pixel-Level classification of VHR that is very high resolution remote sensing images. So, here this is also like we can say this is also related to the image segmentation only. They are doing pixel level classification that is to say each pixel in the image is to be classified into one of different classes.

(Refer Slide Time: 25:21)



So, for this once again they use this kind of a like a deep learning framework. So, like here they have. So, there are 2 image sources here or let us sorry, I mean there are let us say there are 2 images from the source domain and there is a target image. And, then like they are there is a these are this neural network which maps all of them to these maps to the prediction maps. So, these are the source prediction maps and the target prediction maps.

And, next the aim here is like the source image is of course, compared to the ground truth and like we get the some kind of segmentation error that is. So, this basically it can the map which I am talking the prediction map which I am talking about is basically a segmentation. So, this is so, this is a segmented image. So, that so, like the source segmented image is compared with the source ground truth and we get the segmentation error.

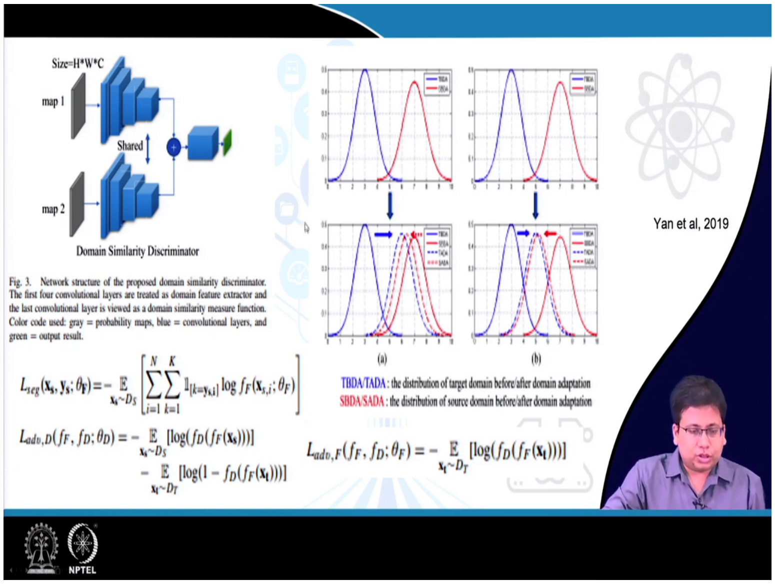
Similarly, the target segmented image they are also like mapped to the or they are compared with the target ground truth and, then after and some error is obtained based on that also. Now, for this part where here what happens is basically the alignment the domain alignment. The source domain images I mean the source domain image after segmentation and the target domain image after segmentation like basically like an alignment is done between them using this neural network, their labels are also aligned.

And like and accordingly we get this kind of loss and, we like. So, this is again so, you can say this is once again this is adversarially trained in the same way as we were talking about earlier and like that is the aim of one is to fool the other and so on. And, yeah so like so basically what this is trying to do is this is trying to get the source segmented image and the target segmented image and just trying to map or trying to and like see which you are trying to distinguish which is the source and which is the source domain and which is the target domain and accordingly it is trained.

So, finally, we get the like we you can say the result is that the target image is oil like once this network has been properly trained the outcome will be that the target image will be segmented so accurately as to like as the source images themselves. Now, for the source images of course, we have the ground truth. So, the so like we can hope that the source images are going to be segmented perfectly.

Now, the target images are also going to be segmented like with equal efficiency even though we it has not been trained with the target ground truth images. Note that here there are source ground truth images are present, but no target ground truth, instead we have the target pseudo labels which are I mean by pseudo levels I mean the labels which are obtained from the network itself and so on and so forth .

(Refer Slide Time: 28:46)



So, the basic idea here is as follows. So, it is so like so, this part is actually shown up in a bit more detail here. So, like so, this is the network structure for the proposed domain similarity discriminator this part is called the domain similarity discriminator. What it tries to do is like it takes the segmented target image and the segmented source image and tries to like find the difference or discriminate between them.

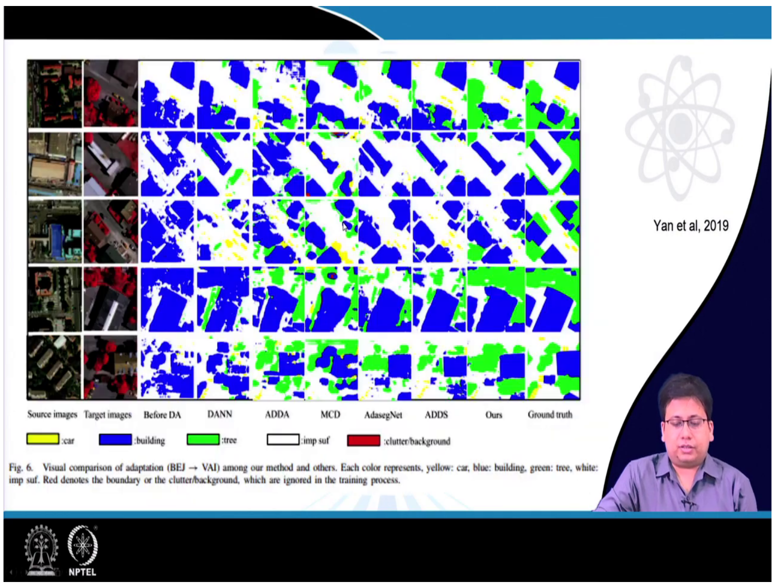
So, like the source the 2 maps both of them are passed through the neural network of the same way like using having shared weights yeah like there are 4 convolutional layers which can be called as domain feature extractor extractors and like finally, they like this here the 2 things are somehow they are compared with each other as the and yes so, and so on.

So, the basic idea behind this approach is that like let us say so, let us say that there are 2 distributions or the features. This is the feature distribution in the. So, let us for example, assume that the features are just one dimension. So, in the featured in the source domain this is the what the feature distribution looks like and in the target domain this is how it looks. So, clearly there is some mismatch between them.

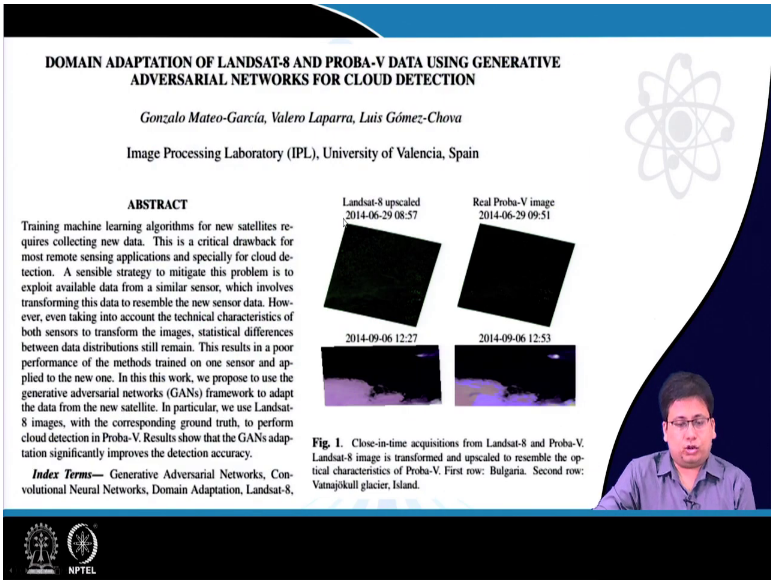
Now, one way we can do is we can just shift the source domain to the target domain as we had been trying to do in the previous CORAL and such approaches. So, that is what we are trying this is shown here that is this data set like this distribution is somehow like it is preserved, but it is translated to closer to the target distribution, right.

But, in this case it is like both the source and the target they are both are being translated is to some kind of a common distribution. So, this is what is actually achieved by the or attempted to be achieved by this kind of a by this approach unlike say some of the earlier approaches like CORAL.

(Refer Slide Time: 31:01)



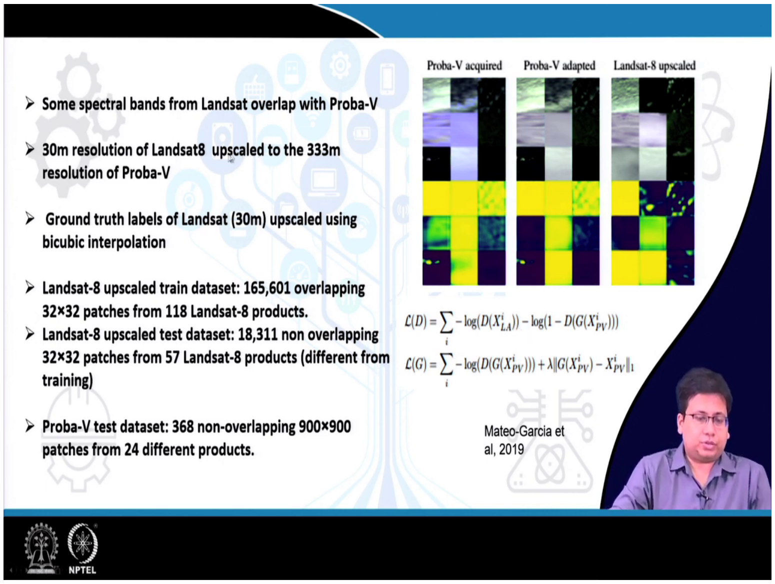
(Refer Slide Time: 31:03)



Here is one more application here the idea is the like I. So, are some of the other papers which we are discussing like they are from different cities or from different satellites and here and so on. In this case it is like there are 2 so, 2 different image sources which might be using different technologies for sensing.

So, one is the Landsat and the other is Proba and the aim here is to like align the images obtained from both the sources. This time also.

(Refer Slide Time: 31:37)

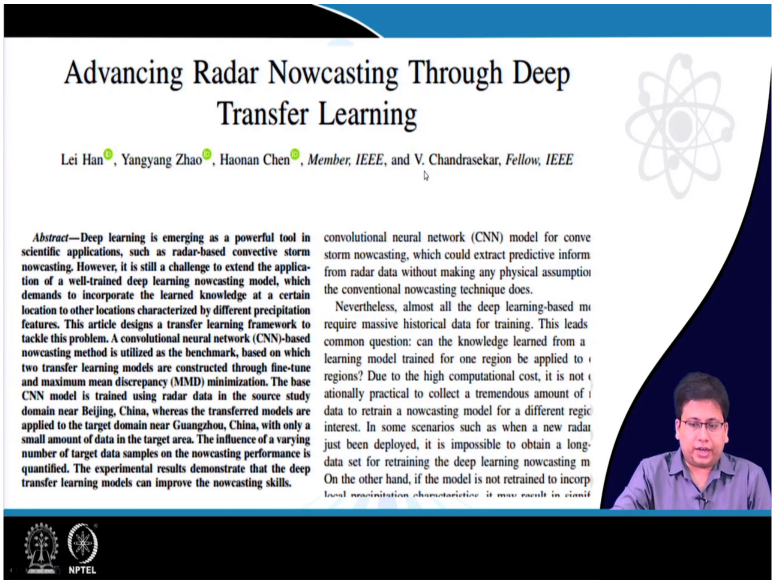


So, these are some of the specifications of the images obtained from the 2 cases. So, like here some of the spectral bands from the Landsat, they overlap with the Proba and then like. So, basically for that now for the now to as pre processing some amount of upscaling is done the also the ground truths of the Landsat image they are also upscaled to the other image to the Proba image because the Proba this Landsat this has a resolution of that is.

So, I am sorry the Landsat has higher resolution that is about 30 meters per pixel while Proba has lower resolutions that is about 333 meters per pixel. So, some kind of upscaling or coarsening of the Landsat images has to be done. So, that and they are the grounds truths also have to be upscaled accordingly. So, for that some kind of bicubic interpolation is used and then the I will.

So, basically we get for every region we get 2 images – one from the Landsat and one from the Proba. And, then the idea is to like so like and so, these 2 images are first up scaled and like brought to the common resolution and so on. After that like both are fed into a neural network to obtain which is again based on this the concept of a the adversarial network with a discriminative and generative component with the result being that we get the some kind of a common representation which cannot be distinguished either from either of the domains.

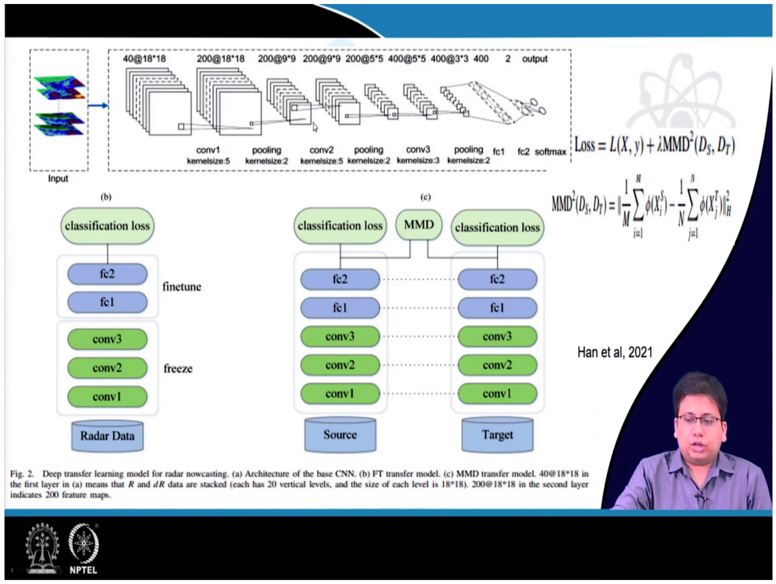
(Refer Slide Time: 33:22)



And, finally, we come to one last application so, advancing radar nowcasting through a deep transfer learning. So, here the so, we have already discussed in some of the previous lectures that how these radar images can be used for nowcasting of rainfall and like other like extreme weather phenomena storms and so on.

So, the idea here is that like there is this now like a radar-based nowcasting algorithm, but that is trained on the radar imagery obtained from one region namely Beijing in China.

(Refer Slide Time: 34:00)



And, now they want to like map it to a different region. So, for that they once again use the same idea of domain adaptation. So, like so, this is the original thing, the original algorithm with so, there are input images coming in from the radar. So, they are they undergo a sequence of convolution and pooling steps as happens in a normal CNN and so, the output in this case is just binary the aim is to just predict after a certain number.

So, this let us say these images were acquired at a particular time T the aim is after a time window will an extreme event happen or not. So, that is a binary classification problem. So, accordingly this the network is trained. Now, the same network is going to be deployed in the target domain also.

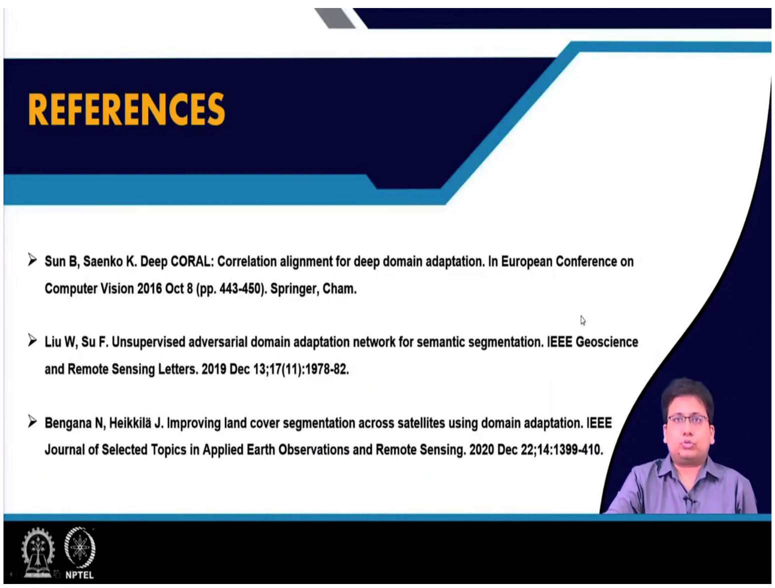
However, it will be like most of the edge most of the like for deployment in the target region most of the things will the weights etcetera will be shared. It is not going to be a retraining or anything only some in the last 2 layers there might be some reorientation of the weights and that will be done according to this loss function.

So, again this what this loss function does is, it simply maps the source images to the to that I mean the source radar images to the target radar images I based on their feature representation. These feature representations are obtained from the neural network itself from the first few convolutional. So, this what this phi basically means is the feature maps obtained from the convolution layer.

So, basic like what they are doing is they are mapping the feature space of the source images to the feature space of the target images and then the like. So, this is a an extra loss function which they are adding apart from the usual classification loss function and these 2 fully connected layers their weights are going to be like tuned so as to minimize this loss function so that the features of the target domain are mapped to the features of the source domain.

And, then the once this is achieved then this module can be deployed in the target domain also with equal efficiency as the source domain.

(Refer Slide Time: 36:24)



(Refer Slide Time: 36:28)

So, these are the different papers that we covered today. So, this brings us to the end of Module 4 and from next lecture we will begin Module 5.

So, till then, bye.