Deep Learning
Prof. Mitesh M. Khapra
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture-101

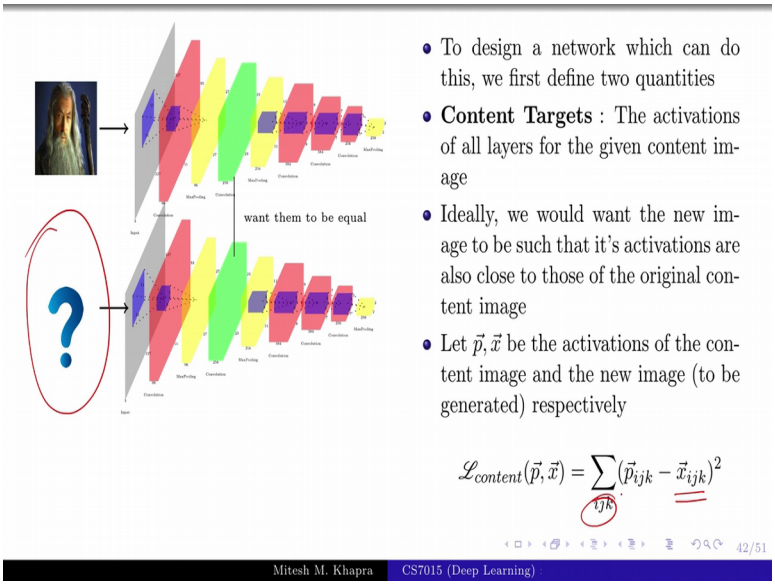
Deep Art

Ok now, we will go to deep art. Now here any questions on that ok.

(Refer Slide Time: 00:19)

So, now here is what, here is a again an IQ test right, so what will happen ok. So, this is deep art ok, someone wanted to try this that if you take natural images or camera images and if you have art from various famous artists and I want to render this original image in this art form. And how can I do? So, I will explain this, the bit of a leap of faith in what is happening here, but just indulge me right. So let us see.

(Refer Slide Time: 00:47)



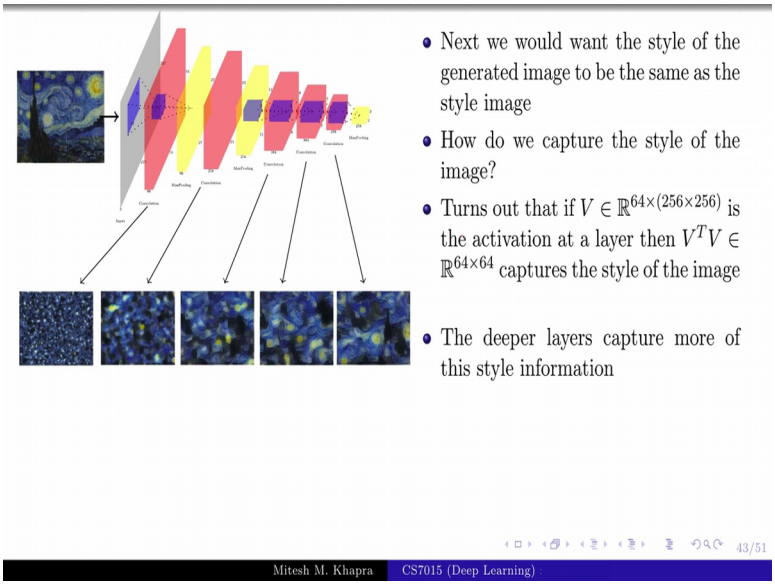
So, to design a network which can actually do this, we design we first define 2 quantities; one is the content targets. So, I call this image as the content image because, this is the content that you are interested in right, I want my final content to look like this. For the content, we would want the following thing that if, I am able to create a new image, when I pass it through the same convolutional neural network; we want these hidden representations to be equal. Right because, that is the assumption here is that, the hidden representation is actually captured the essence of the image which is this face and it is various attributes right.

So, if I create a new image in a different style, still this content should be present in it. And my way of ensuring that or rather the way of the author's way of ensuring this was to make sure that, the embeddings that I learn for the new image and the original image are the same ok. So, I want these to be equal and I have just shown one for illustration, but you could have the same objective function for all the representations right, remember that we learn multiple representations and a convolutional neural network.

So, this is what my objective function would be for the content. I would want that this tensor which is the volume ijk, every pixel or every feature value in the tensor for the original image should be the same as the generated image ok. And again my optimization problem is with respect to what image, I am going to change the image and this is the loss function that I am interested in is that fine.

So, I think x is my original image and p is the new image which I am going to create right.

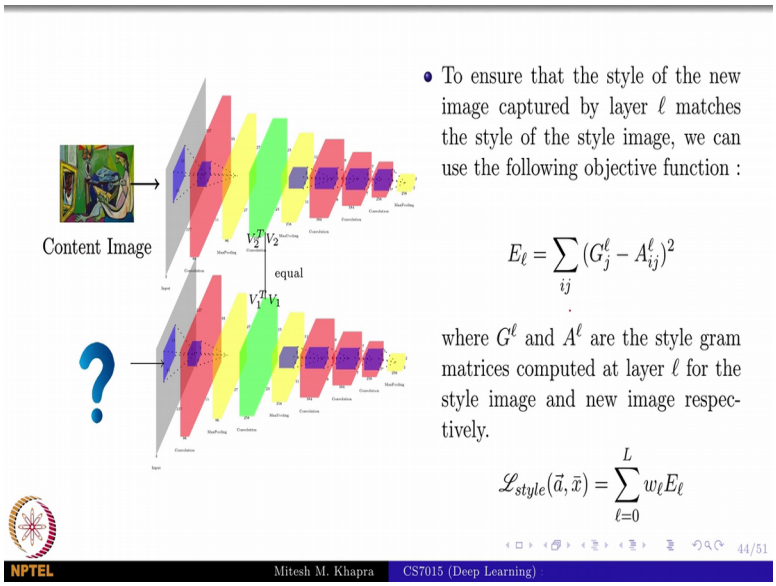
(Refer Slide Time: 02:27)



Now, next and here is where there is a bit of leap of faith. We want the style of the generated image, to be the same as the style image. So, I gave you 1 content image and 1 style image. So, for content the loss function is clear. Now, for style how do you capture the style of the image? So, the explanation given here and I am not very sure about this, but maybe it comes from some traditional computer vision literature, but I just take it on faith that if, you have this volume here, which is say 64 cross 256 cross 256 or any other dimension right then, V transpose V which is a 64 cross 64 dimensional image or matrix, captures the style of the image. So, this is what has been written in the original paper, I am not really dug deep, but my feeling is it comes from some of the traditional literature from computer vision right. So, that is not important, we will just take it for granted that that gives the image and here is the illustration for that. As you go deeper and deeper, so this is if you plot the 64 cross 64 image that you got, then you get different styles as you go deeper and deeper you get a better representation of the style of the original image right. So, that is the argument made in the original paper.

Now, if you assume that this is correct then, can you design a loss function for the style part of it. I want the style of the created image to be the same as the style of the style image.

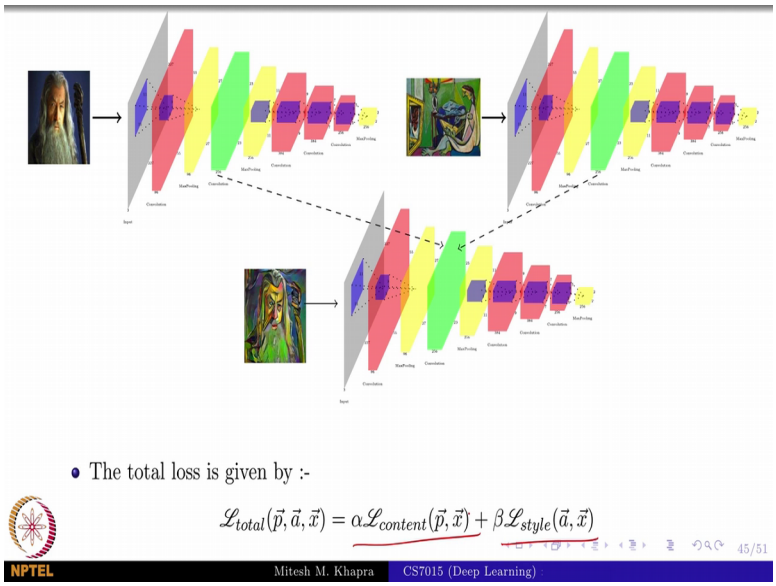
(Refer Slide Time: 03:52)



So, how would I do that? So, this is the content image, this is the actual image oh sorry this is a style image sorry correction ok. So, I would just want that this V transpose V, which captures the style and I could do it for any one of the layers or all layers depending on what I want to do. I just want that this style should be as close to each other.

So, I can have a similar matrix squared error kind of a function right, so that is what this is trying to capture. These are the style gram, so this is V transpose V for the style image and this is V transpose V for the generated image, if I pass it through the convolutional neural network, I want both of these to match.

(Refer Slide Time: 04:37)



So, I want the content to match I want the style to match, so, then what is my total objective function going to be.

Student: sum of these.

Sum of these right, so this is what my total objective function is going to be, I want the content to match and I also want the style to match. So, I will use an objective function which tries to balance between these 2 and alpha and beta are some hyper parameters ok. And if you do this and train the algorithm and try to modify the pixels along with some other bunch of tricks then, you will get this gandalf rendered in this style that you have given right. So, this is again some code is available for this, you can go and try it out. And it is interesting it is in a very interesting idea that you could have taken these two things and now you could be imaginative right, you could do all sorts of things with if you have two different images, how do you want to combine them and so on right. So, that is the basic key idea here.