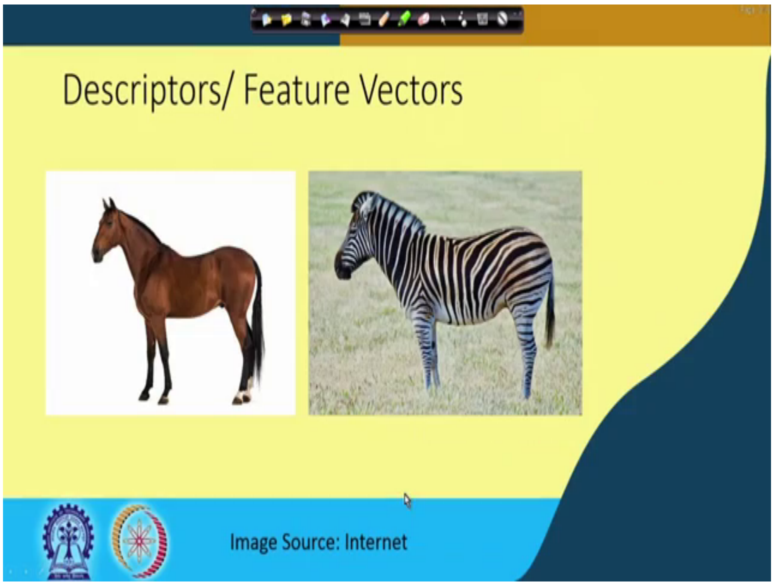
Deep Learning
Prof. Prabir Kumar Biswas
Department of Electronics And Electrical Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture - 03
Feature Descriptor - II

Hello, welcome to the NPTEL online certification course on Deep Learning. In the previous lecture, we have discussed about the ways in which we can obtain the boundary descriptor or boundary features of any given shape which tells you what is the shape of the boundary or captures some information of the shape of the boundary.

(Refer Slide Time: 00:56)



So, in the previous class, I showed you these two figures one of the horse and one of the zebra and we have said that though the shape of those two animals horse and the zebra are similar. So, these shape information is not really sufficient to distinguish between a horse and a zebra. So, to distinguish between these two animals horse and zebra, we need the description of the shape which is obtained from the boundary of the figures.

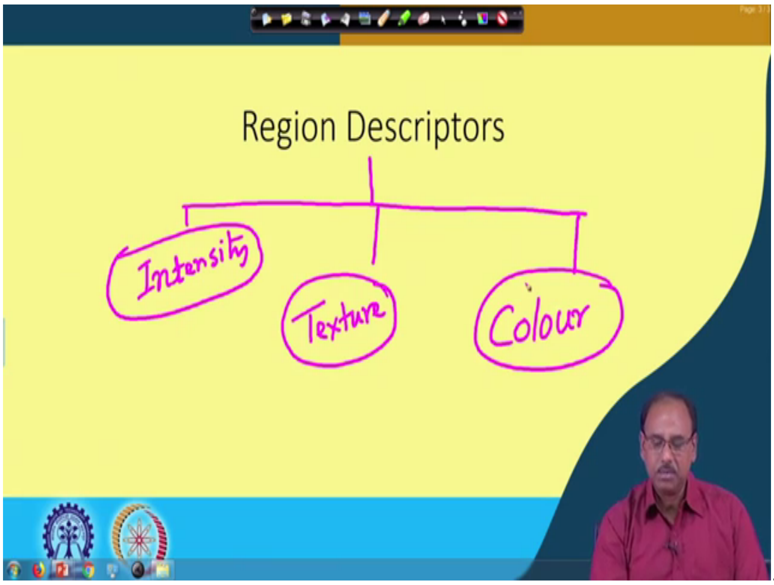
In addition we also need information of what is the color, what is the texture and what is the intensity. So, these are the descriptors or the features which are known as region descriptors or region features.

(Refer Slide Time: 01:51)

So, in the previous class we have talked about the boundary features how we can obtain the boundary descriptors or boundary features from any arbitrary shape. In today's lecture, I am going to discuss about the region features. So, how the intensity, the texture and the color that description can be obtained what are the techniques for that? In addition, as we said in the previous class that machine learning or deep learning not necessarily is concerned about only the visual signals it is also concerned or applicable for understanding the audio signals.

So, for that I also need to understand, how we can extract that discriminating features or the discriminating descriptors from the audio signals using which I can different I can have different applications like speech car identification, speech to text conversion and all that.

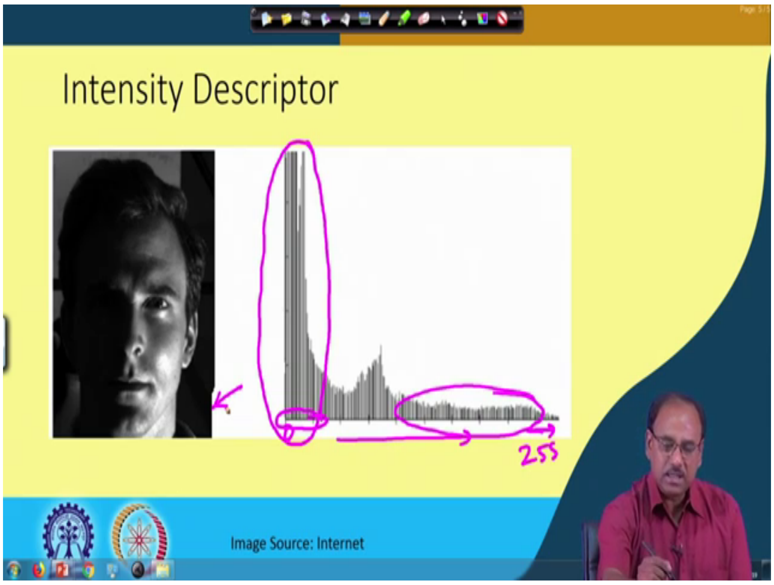
(Refer Slide Time: 03:00)



So, firstly I will talk about the region descriptors, how we can obtain region descriptors. So, as we said before that when we talk about region descriptors, I am concerned about extraction of three types of information one of the information that I want to obtain from the figure is what is the intensity or the intensity profile. So, this is one of the information of the region that I want to capture or I want to obtain. The second kind of information that I will also be interested in is, what is the texture?

So, texture information is also important to distinguish between two different figures or among different figures. Similarly, the third kind of information that will be interested to obtain is what is the color of the particular object or the particular region. So, in region descriptor extraction, we are mainly concerned about these three quantities the intensity, the color and the texture. So, we will talk about what are the different ways in which we can obtain these three different types of informations. So, I will go to them one by one.

(Refer Slide Time: 04:27)



So, how do I obtain the intensity, you find that over here I have shown a picture and on the right hand side what I have is what is the histogram or intensity histogram. So, before going into that let me define what is meant by intensity histogram.

We told earlier that if I have a black and white image or a grayscale image like this then at every point or every pixel in the image the intensity is quantized by a 8 bit binary number. So, because it is 8 bit number so I can have an intensity at any pixel varying from 0 to 255. So, the minimum intensity level or very dark pixel is having an intensity value 0 and a white pixel or having an intensity which is maximum the intensity values 255.

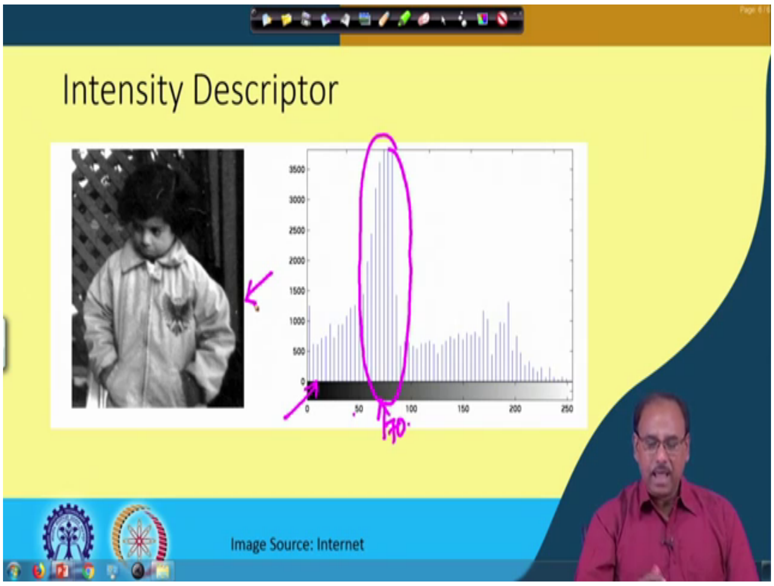
So, what I say the way a histogram is defined is like this. If I have take an intensity value say i then histogram h i tells me that how many times this intensity i appears within the given image or in other words h i tells me the number N i that is how many pixels within the image have intensity value h i.

If I normalize this histogram; that means, if I put it as h i is equal to N i by capital N where capital N is the total number of pixels in the given image and N i is the number of pixels having intensity value i. So, this actually tells you that what is the frequency of occurrence of intensity level i within the given image or this is nothing, but what is the probability of a pixel having an intensity value i.

So, when I normalize a histogram, the normalized histogram also gives me the intensity probability distribution. So, here you find that in this particular image the intensity values most of the pixels are having intensity values which are on the lower side. So, on this side I have intensity 0, on this side I have intensity 255 that the maximum intensity value.

And in between the intensity values changes accordingly as per this scale. So, this histogram says that most of the pixels as over here in the h i value for i within a small region over here is quite high compared to the frequency over here where the intensity is high, here the intensity is low. So, for lower intensity the h i is high that indicates that most of the pixels within this image have intensity value which are very low that is the image is very dark and that is quite obvious by looking at the nature of this image.

(Refer Slide Time: 07:58)



Now as against to this, if I take the next image here. Here you find that the histogram the intensity values over here may be something around 70 or so it is in between 50 and 100.

The probability of occurrence or the frequency of occurrence of intensity values around 70 is very high compared to other two. This is as against to the previous one where the intensity values or maximum, maximum probability at intensity values near about 0 and you see that what is the effect of that. This image appears to be brighter than the previous image that we have shown. So, that clearly says that this histogram gives you a very important information about the intensity distribution within the given image or within a given region.

So, as we have extracted the shape information or the shape descriptors in our previous class. In this class, we are going to describe discussed about the region property extraction and this intensity distribution is one of the important region properties.

And it is clearly shown that this intensity distribution is captured who is in this captured within what is known as intensity histogram. So, based on the nature or shape of the intensity histogram, we can estimate the different descriptors, we can find out in different intensity descriptors. So, a very important approach for intensity distribution extraction is using the histogram shape.

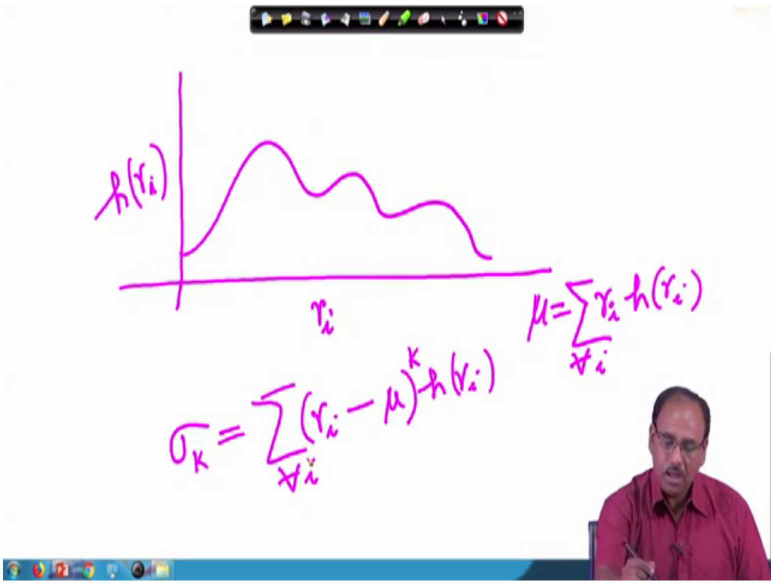
(Refer Slide Time: 09:48)

So, similarly over here if I have a colour image as we said earlier that for colour images we have three different planes right. One of the plane is a red plane the other one is green plane and the third one is blue plane. So, here in this particular image, you find that the colour image which is given over here if I take the histogram of the red plane the histogram shape is something like this. So, this tells you that how strong or what is the distribution of the strength of the red colour component within this image.

This is the histogram of the green component so this tells that, what is the distribution of the strength of the green component within this colour image. Similarly, the third one gives you the information of the distribution of a blue component or strength of the blue component within the given image.

So, as we have seen in the previous example with the black and white image or the grayscale image that the histogram shape of the histogram gives you important information about the intensity distribution within the given image. Similarly, these three colour histograms gives you important information about the colour distribution, the nature of the colour of the given image and by analyzing these histograms I can update important information important descriptors of the intensity distribution and important descriptors of the colour distribution. So, how we can do it?

(Refer Slide Time: 11:57)



So, the way I will do it is I assume that the histogram say my intensity is intensity level I put it as ri and the histogram is given by h of ri.

And this histogram is having different types of shape different depending upon the intensity distribution if it is the histogram taken from and intensity image or colour distribution where I will have three such histograms for each of the colour planes.

Now, once I had this histogram and if the histogram is normalized; that means, it gives you the frequency of occurrence of an intensity value ri which is indicated by h ri. So, in that case this shape information can also be captured through different statistical moments that we have said earlier. So, to capture the different statistical moments firstly what I have compute I have to compute is what is mu that is the mean of the intensity values which is nothing, but r i times h of ri where h ri is the frequency of occurrence of the intensity value r i take the summation of this over all i.

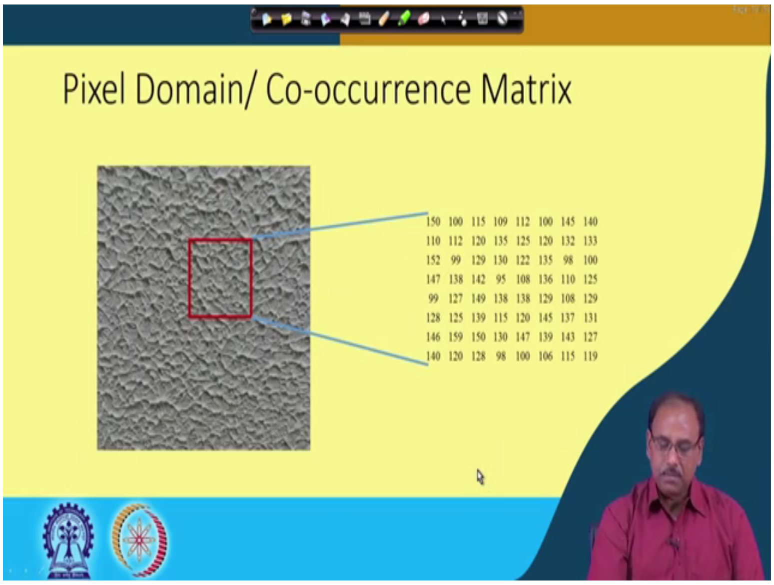
So, that gives you what is the mean intensity value and once I have this mean intensity value then I can compute the statistical moment of order k which I write as sigma k which is given by ri minus mu to the power k times h of r i, take the summation of this over all i. So, this is my statistical moment of order k and as we said earlier, this statistical moment of order k gives you different shape information, information about the shape of the histogram or how the intensity varies what is the variation of intensity or what is the variation of different colour components within the given image.

And we said earlier that if k is equal to 2 this is nothing but what is variance the variance tells you what is the spread of the histogram whether it is a wide or a narrow histogram, if k equal to 3 it tells you the skewness of the histogram; that means, whether the histogram is symmetric about the mean or it is a symmetric about the mean if it is a symmetric, how the symmetric is which is captured by the third order moment.

Similarly, fourth order fifth order all higher order moments gives you some information about the shape of the histogram and the shape of the histogram if it is histogram of an intensity image or black and white image tells you useful information about the nature of intensity distribution and if it is for colour image; obviously, for colour images I have to compute this histogram for all the three components the red component, green component and blue component and all of them tells you useful information about how the green component varies within the image, how the red component varies in within the image or how the blue component varies in the image.

So, I can have different colour information from that. So, this in histograms or moments from the shape of the histogram tells you useful information about the intensity of the region, it tells you useful information about the colour within the region. The next type of region descriptors that we are going to talk about is the texture descriptors this is also; obviously, region descriptor along with intensity and colour. So, what is a texture descriptor?

(Refer Slide Time: 19:56)

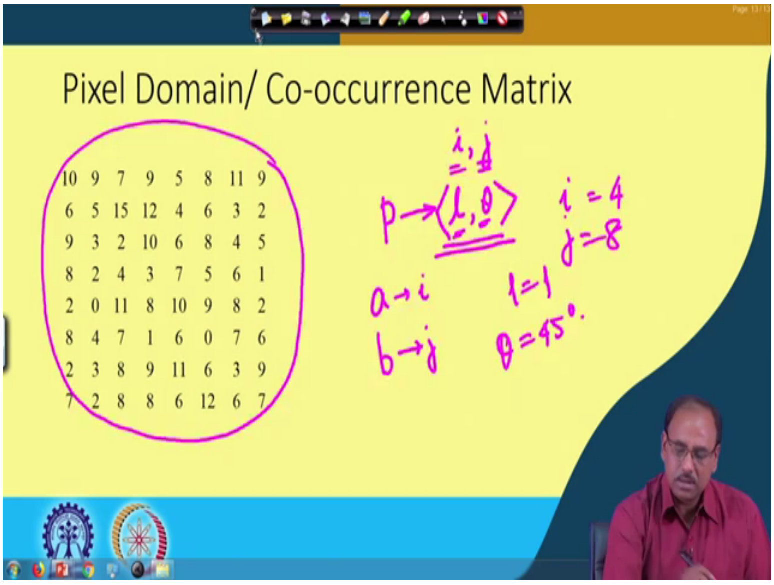


Let us look at this figure, the image which is shown on the left this is a texture image though we do not have a solid definition a convincing definition of a texture, but it tells you the way the intensity varies within the given image and it follows certain pattern and it may not always be possible to represent this pattern through a formal definition.

However, earlier we have said that an image is nothing, but a two dimensional array of integer values where normally in an image and intensity is represented by an 8 bit binary number. So, an intensity value varies from 0 to 255. So, given that if I take a small rectangular or square area within this texture image this is nothing, but a two dimensional array or a matrix of integer numbers which is as shown over here.

So, given this and texture image is also nothing, but a matrix of two dimensional matrix of such in integer numbers. So, given this matrix I have to compute or I have to extract the texture features where the texture is nothing, but variation of intensity values act a regular or semi regular manner.

(Refer Slide Time: 17:23)



So, how we can do it? So, one of the ways in which this texture information can be obtained in the pixel domain is by using something known as a co-occurrence matrix. So, let us see that what these co-occurrence matrix means. Co-occurrence matrix says that given two intensity values say i and j, i is one intensity value j is another intensity value.

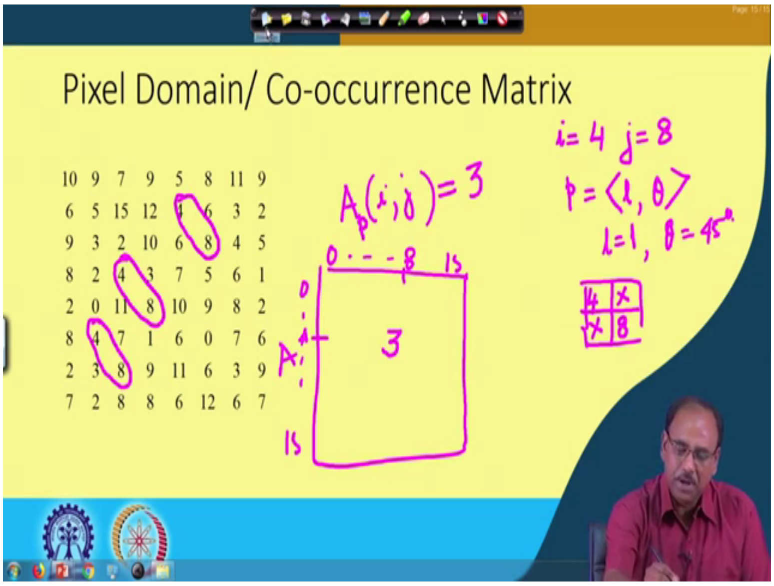
So, it says that how these two intensity values co-occur within the given image. So, when I say co-occur that is two intensity values i and j, the pixels having intensity values i and j also has to follow certain geometric or locational constant. So, I put this location information in the form of a vector say P or in the form of a parameter P where this P will consist of two components one is l and the other one is theta.

So, by this what I want to say is, if I have two pixels say a having an intensity value say i and another pixel say b having an intensity value j then the pixels a and b will be separated following this positional parameter; that means, the distance between the pixel a and b will be l and the orientation of the line joining these two pixels a and b will be theta with say horizontal axis.

So, given this and an image of this form which; obviously, I have shown by a very small square matrix having intensity values in this case intensity values from minimum intensity value is 0, maximum intensity value is 15; that means, it is basically a 4 bit quantization.

So, intensity values of every pixel that is from 0 to 15, I am not considering up to 255 because they are my co-occurrence matrix size will be very high and I want to find out the co occurrence matrix for a particular l theta pair and for this case let me assume that value of l is equal to 1 and value of theta is equal to 45 degree and I also put say i intensity i is equal to say 4 and intensity j is equal to 8.

(Refer Slide Time: 20:14)



So, given this what I want to compute is, this co-occurrence matrix I will put it as a matrix a. I want to compute the element value A i j where i may vary from 0 to 15 as I have said that the minimum intensity value in this image is 0, the maximum intensity value is 15. Similarly, j is also vary from 0 to 15. So, effectively I have a co-occurrence matrix A of dimension 15 16 by 16 varying from 0 to 15 for each of i and j.

So, as I said earlier that I assume value of i to be is equal to 4 and value of j to be is equal to 8 and this follows a positional constant P, what P is given by l theta pair having l is equal to 1 and theta is equal to 45 degree. So, basically what I want to compute is, I want to compute the number of occurrences of two pixels, the first pixel having intensity value 4, the second pixel having intensity value 8.

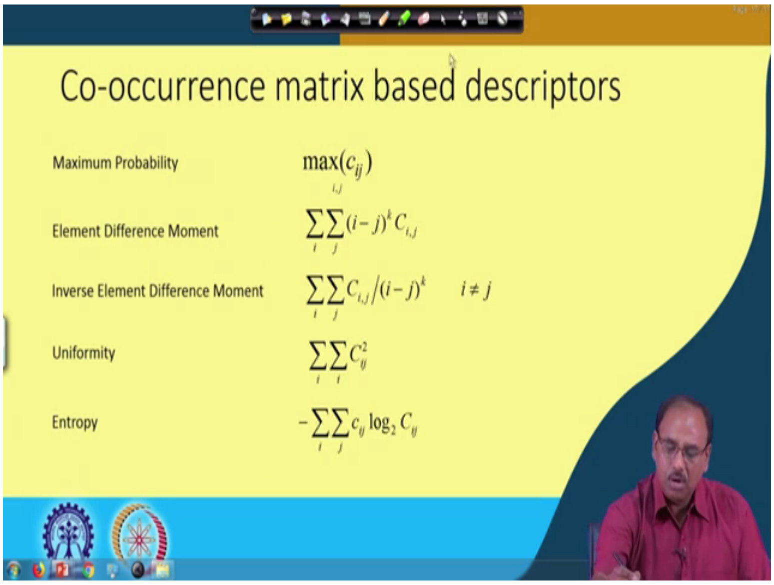
These two pixels have to be at a distance of 1 and the line joining these two pixels will be at an orientation of 45 degree with the horizontal axis or in other words, if I have such a 2 by 2 window I want the pixel at this location to have a value 4, a pixel at this location to have a value 8 irrespective of what are the values at these two locations, how many such occurrences I have within this given image.

So, if you scan this image, you will find that this is one such occurrence. If you look at this is another such occurrence, do I have any more? This is one such occurrence; I think that is all right. So, you find there are three occurrences of pair 4, 8 intensity pair 4, 8 following this positional constant P within the given image.

So, the content at location 4, 8 will be equal to 3. So, my co-occurrence matrix is a matrix A having 16 elements starting from 0 to 15 horizontally, 0 to 15 vertically and at location 4, 8 the value is 3 indicating that this pair of intensity values 4, 8 following this positional constant P occurs three times in this given image and that I compute for all values of i and j.

So, effectively what it gives me is the number of occurrences of different pairs of intensity values how many times they occur within the given image for a given P. So, if I vary p; obviously, I will have different number of such co-occurrence matrices. Now, if I normalize this coherence matrix then what I basically get is a joint distribution right, that is that what is the frequency of occurrence of i jth pair of intensity values following this positional constant p occurring in the given image. So, it is the frequency of occurrence which is nothing, but the probability distribution of the pair of intensity values. So, once I have this probability distribution then I can compute different types of features from such a matrix.

(Refer Slide Time: 24:33)



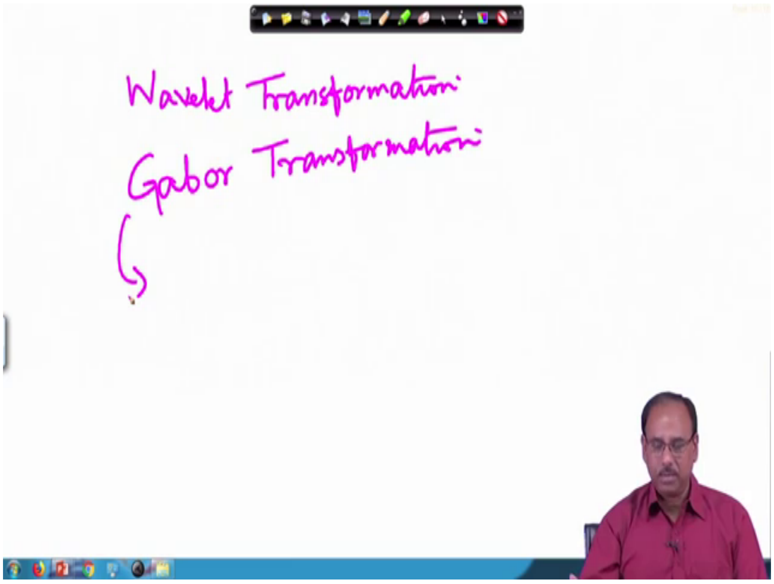
One of the features is what is the maximum maximally occurring pair of intensity values? So, that is what is the maximum probability. So, this cij represents the normalized matrix A i j. So, one of the property that I can extract is what is the maximally occurring pair of intensity values ij. The other one that I can compute is what is known as element difference moment of order k which is given by i minus j to the power k cij take the summation over all values of i and j.

What is the significance of this? The element difference moment, you find that if the larger values of c ij are concentrated around the diagonal around the main diagonal of the co-occurrence matrix or normalized co-occurrence matrix c ij then the value of an element difference moment will be very low because of this factor i minus j to the k. Similarly, I can have the opposite one which is inverse element difference moment. So, which is c ij upon A i minus j to the power k. So, this will have an effect which is just opposite to the element difference moment. So, when element difference moment is large inverse element difference moment will be low, I can also have a measure of uniformity.

So, you find that the sum of c ij square will be maximum if all the elements of this co-occurrence matrix c ij they are equal that tells me what is the uniformity. Similarly, it also tells me that what is the entropy right. So, this entropy will be maximum if the content c ij are maximally random; so, all this different types of features or the properties that can be computed from the co-occurrence matrix that we have just computed.

So, this is what you obtain from that raw image, the raw texture image itself. We can also obtain the features in the frequency domain; these are the features that we can have in the pixel domain in the raw image. So, if I want to compute features in the frequency domain, there are different the transformations that can be applied on the given texture image to transform into frequency domain and I can compute the different types of descriptors in the frequency domain.

(Refer Slide Time: 27:37)



So, for texture images, the number of transformations the different transformations which have been very popular to extract texture features are one is Wavelet transformation and the other one is Gabor transformation.

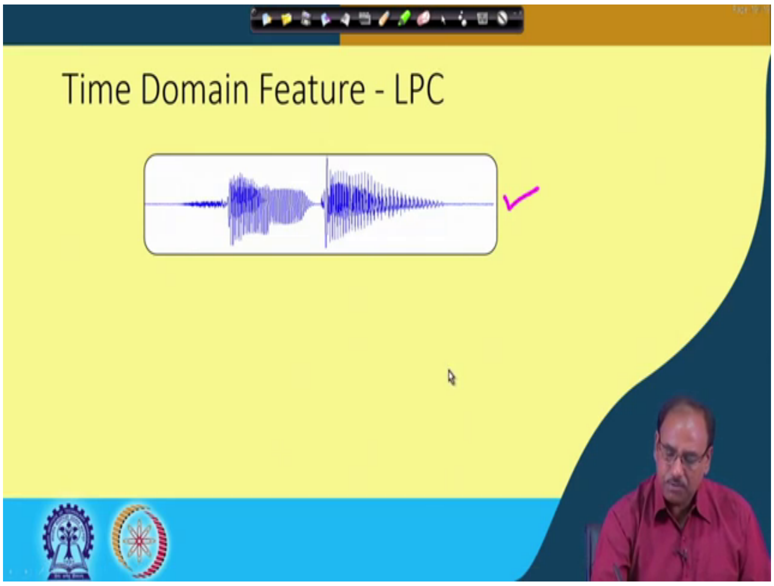
So, what you do in case of Wavelet transformation is, this breaks the original signal into different frequencies of lines and then you compute the energies in the different sub bands and put those energies of different sub bands as defined features. In case of Gabor transformation, this is very interesting Gabor transformation is nothing, but a filter which is cosine modulated Gaussian envelope and that can be oriented in various directions.

So, here as it is as it is cosine modulated Gaussian. So, I have two parameters over there, one is what is the variance of the Gaussian that tells you about the scale and because it is cosine modulated what is the cosine what is the frequency of that cosine signal and the other parameter that I get because it can be oriented what is the orientation ideal.

So, I can have three different parameters for Gabor filters frequency, scale and orientation and I can have different filter coefficients by varying these three parameters. So, I can generate an array of filtered signal outputs and for each of these outputs I can compute the energy and put the energies in the form of a feature vector. So, that is what can be done in the transformation domain as well. So, I can have components in the frequency in the spatial domain, I can also have features extracted in the transformation domain or in the frequency domain.

So, all these what I have done is for the visual signals like images or what we can see, but as we have seen said before that the frequency extraction or feature extraction is not only required for the visual information it is also for the audio information.

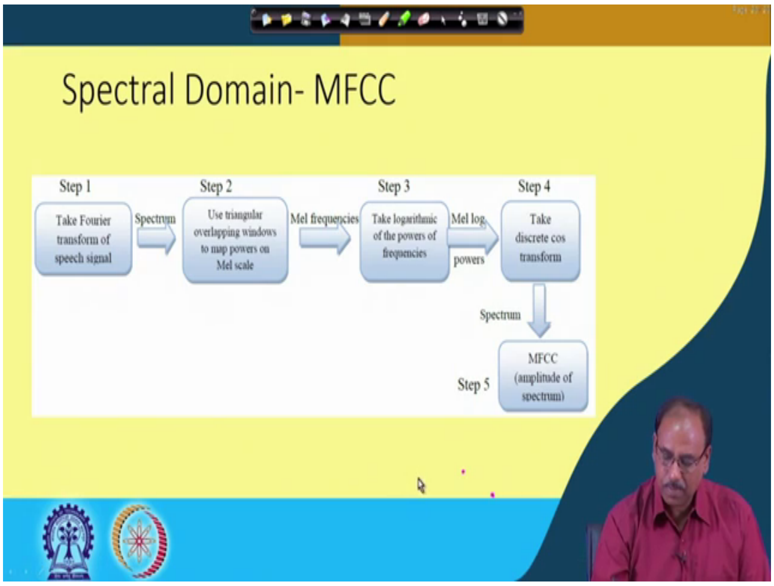
(Refer Slide Time: 30:20)



So, in case of an audio, an audio signal whatever I am speaking audio signal is generated by capturing this information by a microphone and if you check with a CRO what is the output of the microphone? That is nothing, but a signal of this form something like this; this is the output of the microphone. So, for audio signal feature extraction what is done is this output is passed to our digitizer. So, through a digitizer I get a sequence of samples.

And from the sequence of samples, I can compute different types of features right and all those features can be used as descriptors to be used for speech recognition, for speech recognition, for speech to text conversion and all that. So, even an audio signal can be represented in the form of a vector right.

(Refer Slide Time: 31:20)



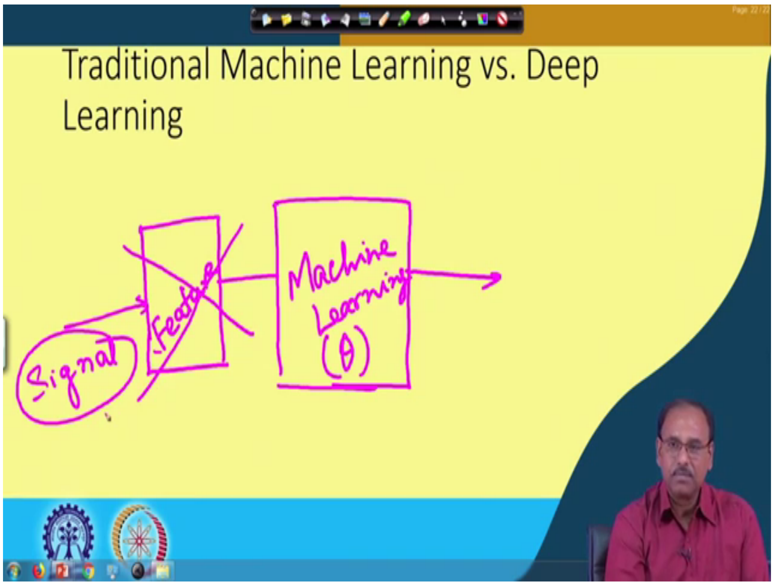
So, the different types of vector representation that we can have is one of them is a well known Linear Predictive Coding or LPC coefficients, the other kind of feature vectors which has become very popular and very powerful is what is known as MFCC or Mel Frequency Cepstral Coefficients.

So, this shows what are the steps for computation of MFCC? First is you take the Fourier transformation of the signal samples that you have our speech samples that you have then you convert these frequency coefficients the Fourier coefficients into Mel frequency coefficients, this is required because our auditory system is not that sensitive to high frequency components, but it is very sensitive to low frequency components right.

So, that is what is known as this conversion to Mel frequencies then after that you take the logarithm of Mel frequency coefficients, why do you need logarithm? Because again our auditory system is not that sensitive to loud signals, but it is very sensitive to the signals which are not so loud. So, you get you have to perform a logarithmic transformation and the output of this log operation, you take the discrete Fourier transformation of that and the discrete Fourier transformation coefficients are nothing, but your MFCC coefficients.

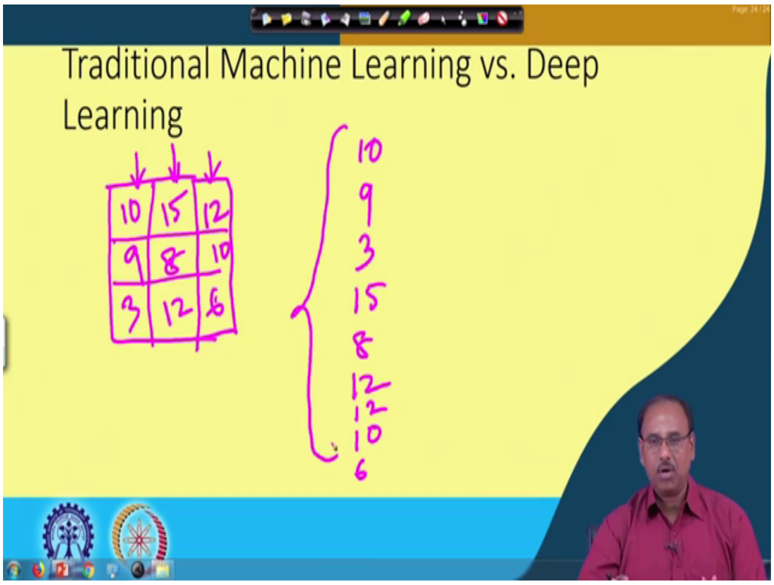
And using this MFCC coefficients, I can discriminate among different speakers I can identify the different spoken words ok. So, all these that have talked about is regarding or trade traditional machine learning approaches. So, what is that?

(Refer Slide Time: 33:18)



In traditional machine learning approach, the system is like this I have input raw signal which passes to a feature extractor block and these features are fade to the machine learning algorithms for this machine learning algorithm has some parameter set of parameter say theta and using this machine learning algorithm, you take decision on this input signal what it is, but when you talk about today’s deep learning algorithm this feature extraction block is absent.

(Refer Slide Time: 34:30)



So, what you have is I have this machine learning algorithm again with some set of parameters theta and raw signal is fed directly to this machine and output of the machine is the decision that you take from the signal. So, even for deep learning, I also need to represent the input signal as vector or set of vectors. So, how do I represent that input signal as a set of vectors?

So, for that let me assume again from the visual domain that I have an image and I take a small segment of the image or 3 by 3 image and this image may be the pixel intensities maybe 10, 15, 12, 9, 8, 10 again 3, 12, 6 something like this. So, I need to convert this image itself into a form of vector, how we can do it? Take each column of this image and concatenate all the columns together to form a single vector. So, what you are doing is from a matrix, I am converting this to a vector. So, in this case my vector will be 10, 9, 3, that is the first column then I take the next column it will be 15, 8, 12 then I take the third column again it will be 12, 10, 6.

So, you find that from this 3 by 3 matrix, I have made a one dimensional vector having nine different components and this vector is fed to the machine learning algorithm whether it is during training of the machine or during the testing, testing phase and that is what you do in case of deep learning algorithms, where we expect in case of machine learning traditional machine learning, the features are to be decided by the user or the human being accordingly I have to have feature extraction algorithms and then the machine learning comes into picture. In case of deep learning, we expect that no we will not tell what features to analyze for taking a particular decision, but later the machine learn the features also.

So, during training phase what we say is you feed the input raw signal in the form of vector to the machine and tell the machine what this signal is and based on this. So, that is basically you are trying to supervise the machine to learn the features from the signal and not only the features it also learns to take the decision or to learn to describe those signals or those features.

So, starting from feature extraction to decision making this end to end application is done by machine and that is what is done in deep learning. So, you see that in today's lecture, we have talked about the different types of features both from the visual signal domain as well as from audio signal domain, how to extract those different feature vectors in case of traditional machine learning, in case of deep learning given a raw signal in the form of a raw image or this is also applicable in case of audio signals where given audio signal inputs in the form of sequence of samples I can form vectors out of it following window operation.

So, given a raw signal how we can obtain or how we can vectorize that raw signal. So, next class onwards we will not discuss about the descriptor or how do you obtain a descriptor of a given signal, I will assume that given any signal it is represented as a vector. So, the input to my machine will be a vector and everything that we do whether it is learning or testing, decision making everything has to be done on that vector.

Thank you.