Digital Image Processing

Prof. P. K. Biswas

Department of Electronics and Electrical Communication Engineering

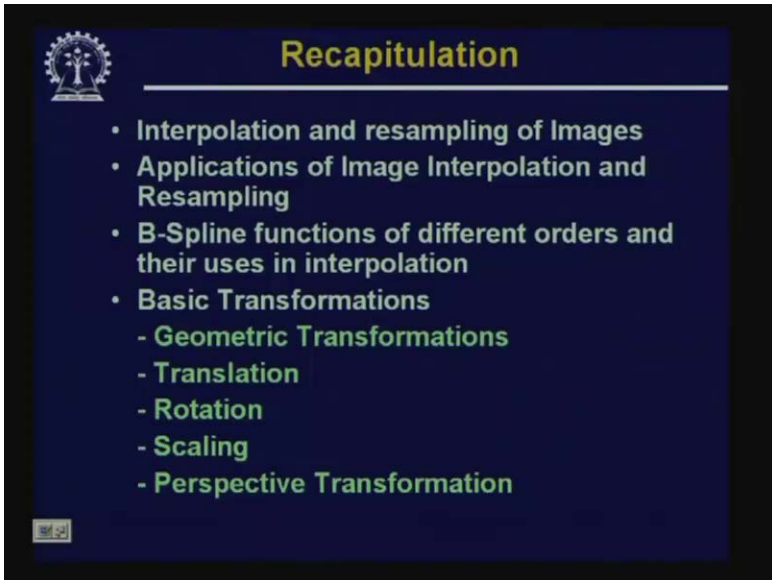
Indian Institute of Technology, Kharagpur

Lecture - 11

Image Interpolation - I

Digital image processing: in the last class we have seen the interpolation and re-sampling operation of images and we have seen different applications of the interpolation and re-sampling operations.

(Refer Slide Time: 1:09)



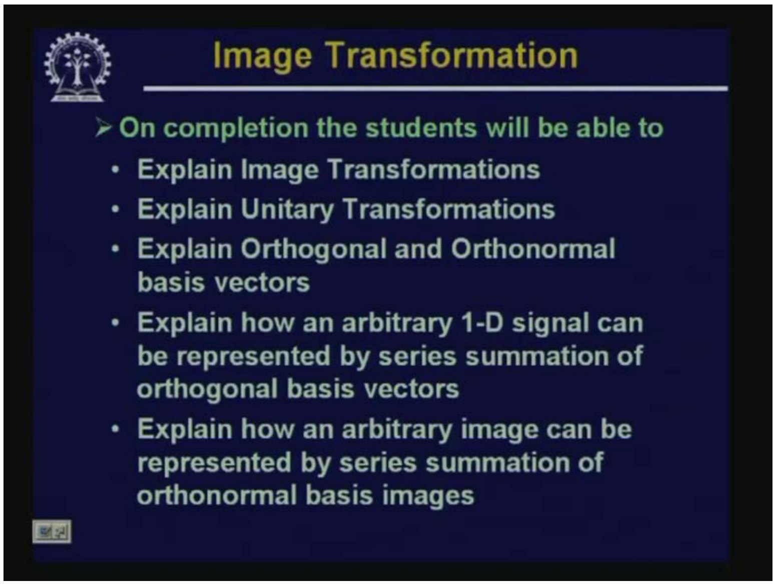
So, while we have talked about the interpolation and re-sampling, we have seen that it is the B spline functions or B spline interpolation functions of different orders which are mainly used for image interpolation purpose and before this interpolation, we have also talked about the basic transformation operations and the transformation operations that we have discussed, those were mainly in the class of geometric transformations.

That is we have talked about the transformation like translation, we have talked about rotation, we have talked about scaling and we have seen that these are the kind of transformations which are mainly used for coordinate translation. That is given a point in 1 coordinate system, we can translate the point or we can represent the point in another coordinate system where the second coordinate system may be a translated or rotated version of the first coordinate system.

We have also talked about another type of transformation which is perspective transformation and this perspective transformation is mainly used to find out or to map a point in a 3 dimensional world coordinate system to a 2 dimensional plane where this 2 dimensional plane is the imaging plane.

So, there our purpose was that given a point or in the 3D coordinates of a point in a 3 dimensional coordinate system; what will be the coordinate of that point on the image plane when it is imaged by a camera? In today's lecture we will talk about another kind of transformation which we call as image transformation.

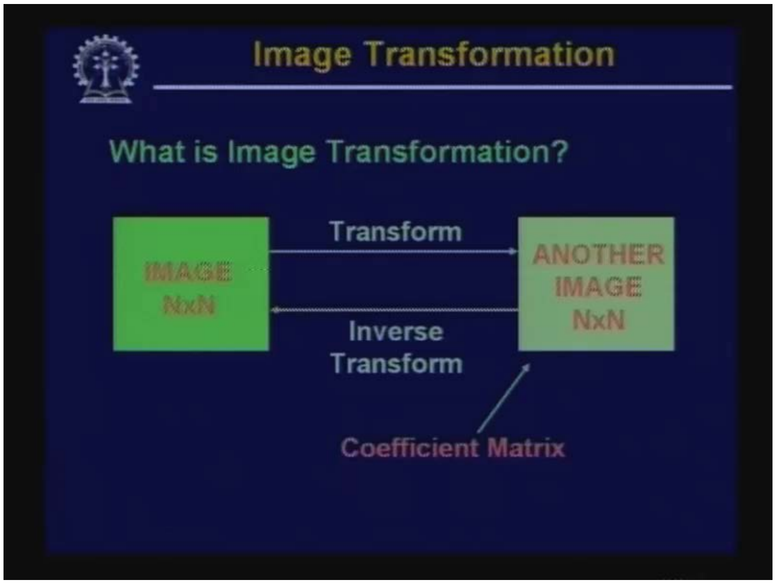
(Refer Slide Time: 2:58)



So, we will talk about or we will explain the different image transformation operations. Now, before coming to specific transformation operations like say Fourier transform or discrete cosine transform or say discrete cosine transform, before we come to such specific transformations, we will first talk about a unitary transformation which is a class of transformations or class of unitary transformations and all the different sort of transformations that is whether it is discreet Fourier transform or discrete cosine transform or hadamard transform; all these different transforms are different cases of this class of unitary transformations.

Then, when we talk about this unitary transformation, we will also explain what is an orthogonal and orthonormal basis function. So, we will see that what is known as an orthogonal basis function, what is also known as an orthonormal basis function. We will also explain how an arbitrary 1 dimensional signal can be represented by series summation of orthogonal basis vectors and we will also explain how an arbitrary image can be represented by a series summation of orthonormal basis images.

(Refer Slide Time: 4:26)

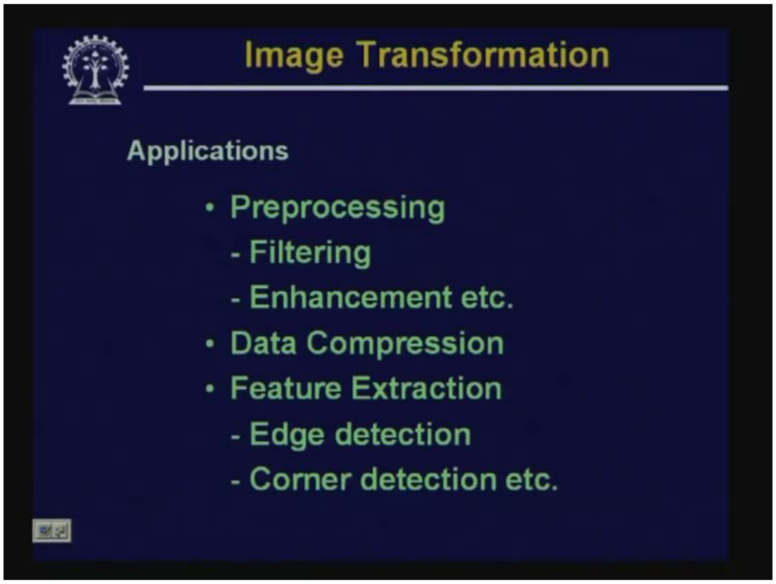


Now firstly, let us see that what is the image transformation. You find that in this case, we have shown a diagram where the input is an image and after the image is transformed, we get another image. So, if the size of the input image is N by N, say it is having N number of rows and N number of columns, the transformed image is also of same size that is of size N by N. And, given this transformed image, if we perform the inverse transformation; we get back the original image that is image of size N by N.

Now, if given an image, by applying transformation, we are transforming that to another image of same size and doing the inverse transformation operation; we get back the original image. Then the question naturally comes that what is the use of this transformation? And, here you find that after a transformation, the second image of same size N by N that we get, that is called the transformed coefficient matrix.

So, the natural question that arises in this case that if by transformation and going to another image and by using inverse transformation, I get back the original image; then why do we go for this transformation at all? Now, we will find and we will also see in our subsequent lectures that this kind of transformation has got a number of very very important applications.

(Refer Slide Time: 5:59)



One of the applications is for preprocessing. In case of image preprocessing, preprocessing of the images, if the image contains noise; then you find or you know that contamination of noise gives rise to high frequency components in the image.

So, if by using some sort of unitary transformation, we can find out what are the frequency components in the image, then from these frequency coefficients, if we can suppress the high frequency components; then after suppressing the high frequency components, the modified coefficient matrix that you get, if you take the inverse transform of that modified coefficient matrix, then the original image or the reconstructed image that we get that is a filtered image. So, filtering is very very important application where these image transformation techniques can be applied.

The other kind of preprocessing techniques, we will also see later on that is also very very useful for image enhancement operation. Say for example, if we have an image which is very blurred that is the contrast of the image is very very poor, then again in the transformation domain or using the transform coefficients, we can do certain operations by which we can enhance the contrast of the image. So, that is what is known as enhancement operation.

We will also see that these image transformation operations are very very useful for data computation. So, if I have to transmit an image or if I have to store the image on a hard disc, then you can easily think that if I have an image of size say 512 by 512 pixels and if it is a black and white image, every pixel contains 8 bits, if it is a color image, every pixel contains normally 24bits.

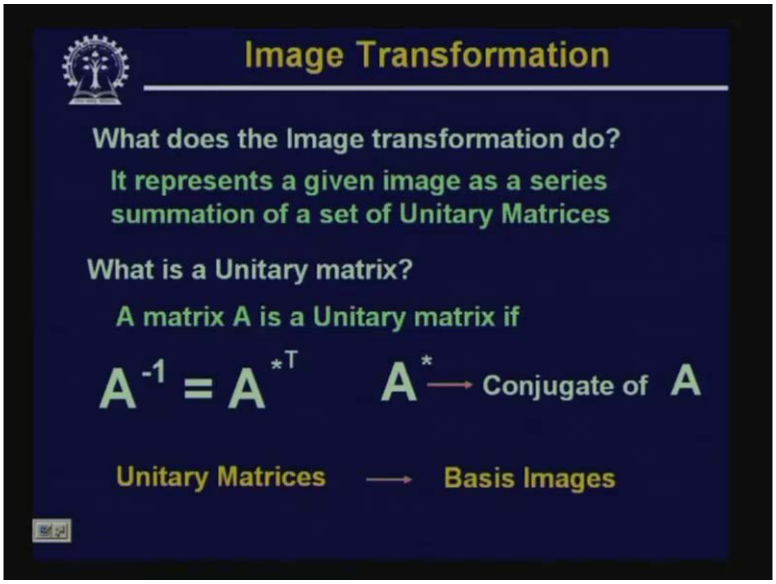
So, storing an image colored image of size 500 and 500 512 by 512 pixel size, takes huge amount of disc space. So, if by some operation I can compress the space or I can reduce the space required to store the same image; then obviously on the on a limited disc space, I can store more number of images.

Similar is the case if I go for transmission of the image or transmission of image sequences or video. In that case the bandwidth of the channel over which this image or the video has to be transmitted is a bottle neck which forces us that we must imply some data computation techniques so that the bandwidth requirement for the transmission of the image or the transmission of the video will be reduced.

And, we will also see later on that this image transformation technique is the first step in most of the data computation or image or video computation techniques. These transformation techniques are also very very useful for feature extraction operation. By features I mean that in the images, if I am interested to find out the edges or I am interested to find out the corners of certain shapes, then this transformation techniques or if I work in the transformation domain, then finding out the edges or finding out the corners of certain objects that also becomes very very convenient.

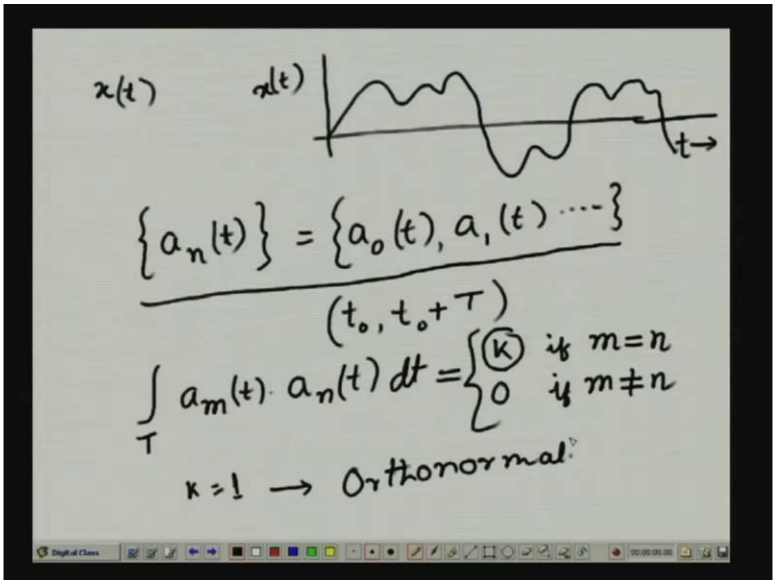
So, these are some of the applications where these image transformation techniques can be used. So apparently, we have seen that by image transformation, I just transformed an original image to another image and by inverse transformation that transformed image can be retransformed to the original image. So, the application of this image transformation operation can be like this and here I have sited only few of the applications. We will see later that applications of this image transformation are much more than what I have listed here.

(Refer Slide Time: 10:19)



Now, what is actually done by image transformation? By image transformation, what we do is we try to represent a given image as a series of summation of a set of unitary matrices. Now, what is a unit unitary matrix? A matrix A is said to be a unitary matrix if A inverse or inverse of A is equal to A star transpose where A star is the complex conjugate of A. So, a matrix A will be called a unitary matrix if the inverse of the matrix is same as, first we take the conjugate of the matrix A then take its transpose; so A inverse will be equal to A start transpose where A star is the complex conjugate of the matrix A that is complex conjugate of each and every element of matrix A. And these unitary matrices, we will call as the basis images. So, the purpose of this image transformation operation is to represent any arbitrary image as a series summation of such unitary matrices or series summation of such basis images. Now, to start with, I will first try to explain with the help of 1 dimensional signal.

(Refer Slide Time: 11:55)



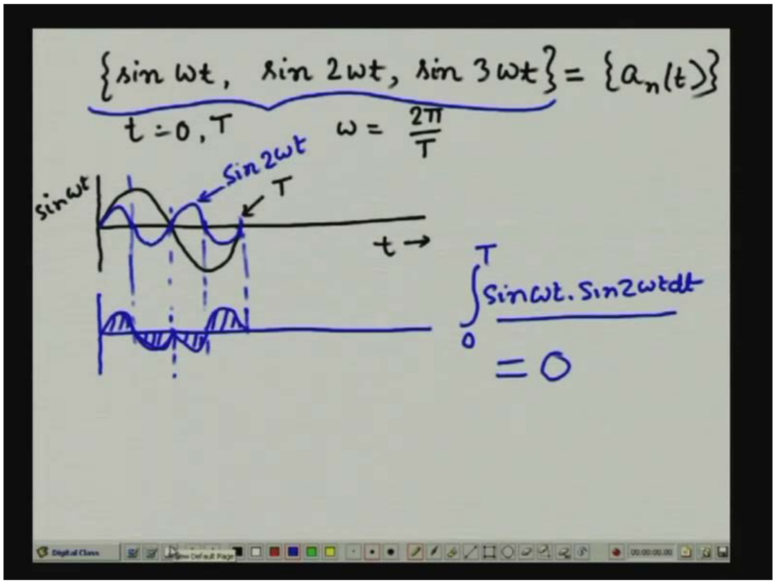
So, let us take an arbitrary 1 dimensional signal. So, I take a signal say x (t). So, I take an arbitrary signal x (t) and you see that this is a function of t. So, this x (t), the nature of x (t) can be anything. Say, let us take that I have a signal like this x (t) which is a function of t. Now, this arbitrary signal x (t) can be represented as a series summation of a set of orthogonal basis function.

So, I am just taking this as an example in for 1 dimensional signal and later on we will extend to 2 dimensions that is for the image. So, this arbitrary signal, this 1 dimensional signal x (t), we can represent by the series summation of a set of orthogonal basis functions. Now, the question is what is orthogonal? By orthogonal I mean that if I consider a set of real valued continuous functions, so I consider a set of real valued continuous functions say a_n (t) which is equal to set say a_0 (t), a_1 (t) and so on.

So, this a set of real valued continuous functions and these set of real valued continuous functions is said to be orthogonal over an interval say t_0 to t_0 plus T. So, I define that this set of ah t inverse real valued functions will be orthogonal over an interval t_0 to t_0 plus capital T if I take the integration of function say a_m (t) into a_n (t) dt and take the integration of this over the interval capital T. Then, this integral will be equal to some constant k if m is equal to n and this will be equal to 0 if m is not equal to n.

So, I take 2 functions a_m (t) and a_n (t), take the product and integrate the product over interval capital T. So, if this integration is equal to some constant say k, when m is equal to n and this is equal to 0 whenever m is not equal to n. So, if this is true, for this set of real valued continuous functions, then this set of real valued continuous functions form an orthogonal set of basis functions. And if the value of this constant k is equal to 1, so if the value of this constant k is equal to 1; then we say that the set is orthonormal. So, an orthogonal basis function as we have defined, this non 0 constant k if this is equal to 1; then we say that it is a orthonormal set of basic functions.

(Refer Slide Time: 16:07)



Let us just take an example that we mean by this. Suppose, we take a set like this, say sin omega t, sin twice omega t and sin 3 omega t. So, this is my set of functions a_n (t). Now, if I plot sin omega t over interval t equal to 0 to capital T; so this will be like this and where omega is equal to 2 pi by capital T. So, capital T is the period of this sinusoidal waveform. Then, if I plot this sin omega t, we will find that sin omega t in the period 0 to capital T is something like this. So, this is t, this is sin omega t and this is the time period capital T.

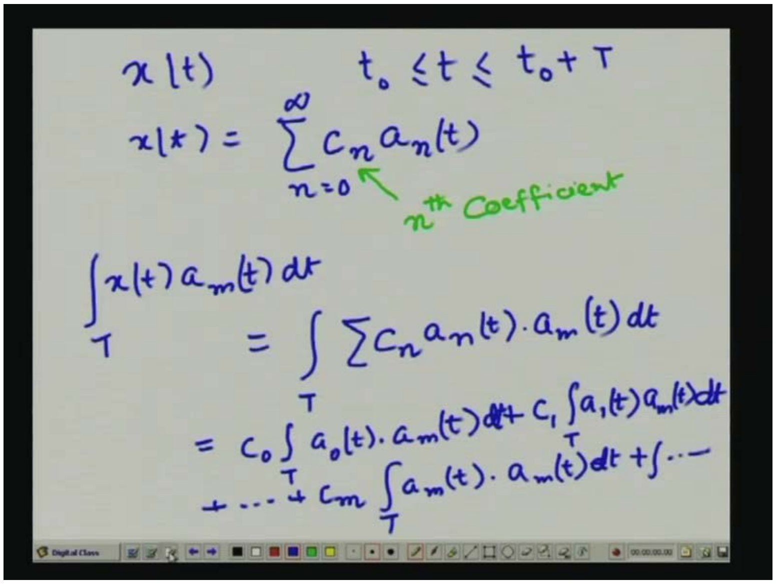
If I plot twice omega t over this same diagram, sin of twice omega t will be something like this. So, this is sin of sorry this is sin of twice omega t. Now, if I take the product of sin omega t and sin twice omega t in the interval 0 to capital T, the product will appear something like this.

So, we find that in this particular region, both sin twice omega t and sin omega t, they are positive. So, the product will be of this form. In this region, sin omega t is positive but sin twice omega t is negative. So, the product will be of this form. In this particular region, sin twice omega t is positive whereas sin omega t is negative. So, the product is going to be like this. This will be of this form and in this particular region, both sin omega t and sin twice omega t, they are negative. So, the product is going to be positive. So, it will be of this form.

Now, if I integrate this, so if I integrate sin of omega t into sin of twice omega t dt over the interval 0 to capital T; this integral is nothing but the area covered by this curve and if you take this area, you will find that the positive half will be cancelled by the negative half and this product will come out to be 0. This integration will come to be 0.

Similar is the case if I multiply sin omega t with sin thrice omega t and take the integration. Similar will also be the case if I multiply sin twice omega t with sin 3 omega t and take the integration. So, this particular set that is sin omega t, sin twice omega t and sin 3 omega t, this particular set is the set of orthogonal basis functions.

(Refer Slide Time: 20:37)



Now suppose, we have an arbitrary real valued function x (t) and this function x (t) is we consider within the region t_0 less than or equal to t less than or equal to t_0 plus capital T. Now, this function x (t) can be represented by a series summation. So, we can write x (t) as summation C_n a_n (t). So, you remember that a_n (t) is the set of orthogonal basis functions.

So, we represent x (t) as a series summation. So, x (t) is equal to sum of C_n a_n (t) where n varies from 0 to infinity. Then this term C_n is called the n'th coefficient of expansion. This is called n'th coefficient of expansion. Now, the purpose is, the problem is; how do we find out or how do we calculate the value of C_n?

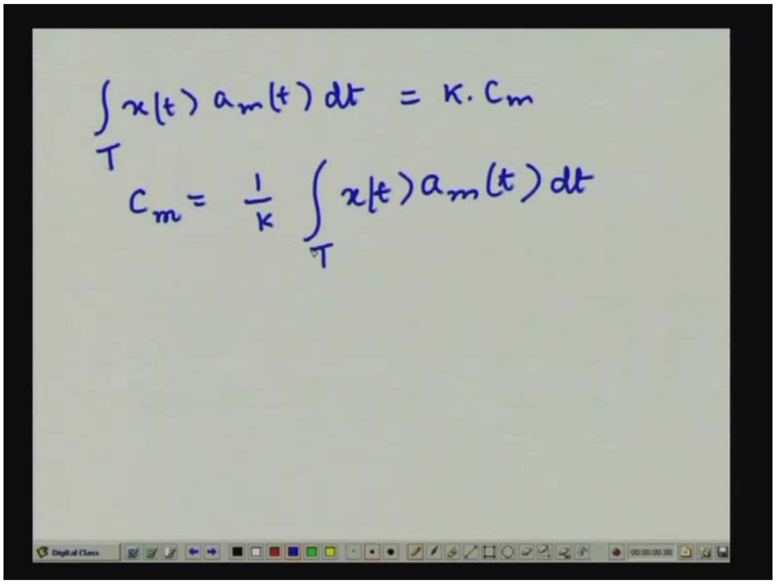
To calculate the value of C_n, what we can do is we can multiply both the left hand side and the right hand side by another function from the set of orthogonal basis function. So, we multiply both the sides by function say a_m (t) and take the integration from t equal to 0 to capital T or take the integration over the interval capital T.

So, what we get is we get an integration of this form x (t) a_m (t) dt integral over capital T, this will be equal to again integral over capital T and this integral of C_n a_n (t) into a_m (t) because we are multiplying both the left hand side and the right hand side by the function amt dt and you take the integral over the interval capital T.

Now, if I expand this, you find that if I expand this; this will be of the form C_0 integration over capital T a $_0$ (t) into a_m (t) dt plus C_1 integration over first… interval capital T a_1 (t) into a_m (t) dt plus it will continue like this, will have 1 term say C_m integral over T a_m (t) into a_m (t) dt plus some more integration terms.

Now, as per the definition of the orthogonality that we have said, that a integral of a_n (t) into a_m (t) into dt that will be equal to some constant k if and only if m is equal to n and this integral will vanish for all the cases wherever m is not equal to n.

(Refer Slide Time: 24:49)

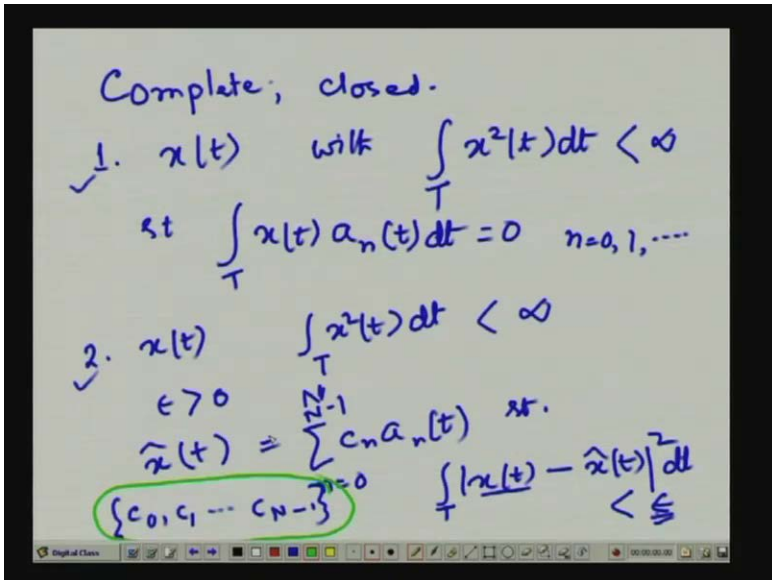


So, by using that formula of orthogonality, what we get in this case is we simply get integral x (t) into a_m (t) dt this integral over capital T. This will be simply equal to constant k times C_m because the right hand side of this integration that we have said, this right hand side, all these terms will be equal to 0 only for this term a_m (t) into a_m (t) dt the value will be equal to k.

So, what we get here is integration x (t) a_m (t) dt is equal to the constant k times C_m. So, from this we can easily calculate that the m'th coefficient C_m will be given by 1 upon k integration x (t) a_m (t) into dt where you take the integration over the interval capital T. And obviously, you can find out that if the set is an orthonormal set, not an orthogonal set; in that case, value of k is equal to 1. So, we can get the m'th coefficient C_m to be x (t) c a_m (t) dt integrate this over the interval t. So, the value to term k will be equal to 1.

So, this is how we can get the m'th coefficient of expansion of any arbitrary function x (t) and this computation can be done if the set of basis functions that we are taking that is the set a_n (t) is an orthogonal basis function. Now, the set the orthogonal basis the set of orthogonal basis functions a_n (t) is said to be complete.

(Refer Slide Time: 26:53)



You say that this orthogonal basis function is complete, if this is complete or closed if 1 of the 2 conditions holds. The first condition is there is no signal say x (t) with integral x square t dt over the interval capital T less than infinity. So, this means the signal with finite energy. So, there does not exist any signal x (t) with x square t dt less than infinity such that integral x (t) a_n (t) dt is equal to 0.

This integration has to be taken over the interval capital T for n equal to 0, 1 and so on. And the second condition is that for any piece wise continuous signal x (t); so x (t) is piece wise continuous and with the same condition of finite energy that is x square t dt integral over capital T must be less than infinity and if there exist an epsilon greater than 0, however small this epsilon is, there exists an end and a finite expansion such that x hat t is equal to C_n a_n (t). Now, n varies from 0 to capital N minus 1 such that integral x (t) minus x hat t square dt taken over the same interval capital T must be less than epsilon.

So, this says that for a piece wise continuous function x (t) having finite energy, there must be an epsilon which is greater than 0 but very small and there must be some constant capital N such that if we can have an expansion that x hat t is equal to summation of c_n a_n (t); now this n varies from 0 to capital N minus 1 for which this term x (t) minus x hat t square dt over integral over capital T this is less than epsilon.

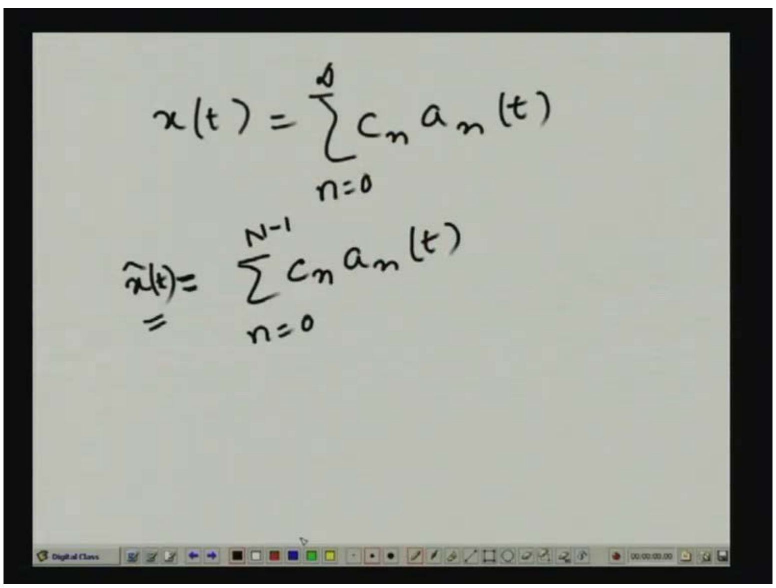
So, you find that this x (t) is the original signal x (t) and x hat t, earlier case we have seen that if we go for infinite expansion, then this x (t) can be represented exactly. Now, what we are doing is we are going for a truncated expansion. We are not going to take all the infinite number terms but we are going to take only capital N number of terms. So obviously, this x (t), it is not being represented exactly but what we are going to have is an approximate expansion. And if x (t) is of finite energy that is integral of x square T dt integration over capital T, S less then infinite; then we can say that there must be a finite N, capital N, the number of terms for which the error of the reconstructed signal, so this x (t) mines x hat is square dt this is nothing but the energy of the error signal, of the error that is introduced because of this truncation which must be limited, it must be less then are equal to epsilon where epsilon is a very very positive small value.

So, we say that the set of orthogonal basis functions a_n (t) is complete or closed if one of these conditions hold, at least one of these conditions hold. That is the first condition or the second condition. So, this says that when we have a complete orthogonal function, then this complete orthogonal function expansion enables representation of x (t) by a finite set of coefficients where the finite set of coefficients are C_0 C_1 like this upto C_N mines 1. So, this is the finite set of coefficients.

So, if we have a complete orthogonal function, set of orthogonal functions; then using this completes set of orthogonal functions, we can go for a finite expansion of a signal x (t) using the infinite number of expansion coefficients C_0 C_1 upto C_N minus 1 as is shown here. So, I have a finite set of expansion coefficients.

So, from this discussion what we have seen is that an arbitrary continuous signal x (t) can be represented by the series summation of a set of orthogonal basis functions and this series expansion is given as x (t) is equal to C_n a_n (t) where n varies from 0 to infinity if I go for infinite expansion or this can also be represented as we have seen by finite expansion, finite series expansion.

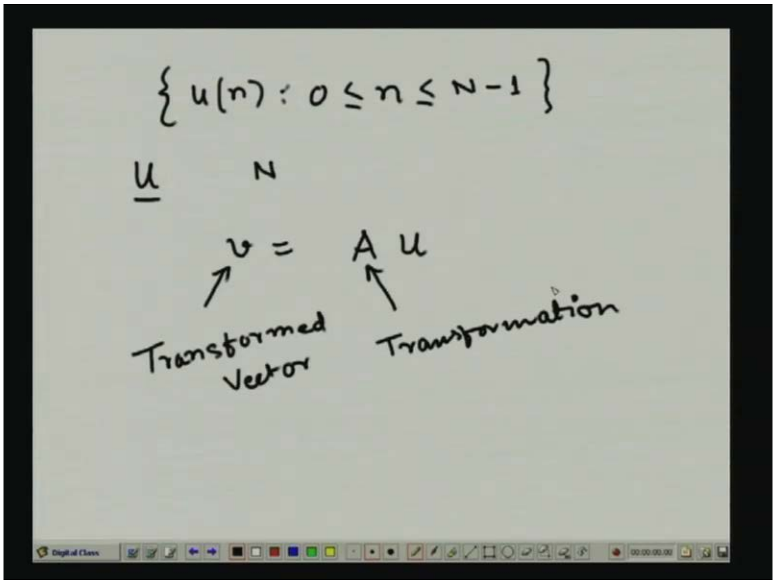
(Refer Slide Time: 33:05)



In this case, this will be represented by c_n a_n (t) where n will now vary from 0 to N capital N minus 1. So, this is x hat t. So obviously, we are going for an approximate representation of x (t) not a complete expansion, not the exact representation of x (t). So, this is the case that we have for continuous signals x (t). But in our case, we are not dealing with the continuous signals but we are dealing with the discrete signals.

So in case of discrete signals, what we have is a set of samples or a series of samples. So, the series of samples can be represented by say u (n) where 0 less than or equal to n less than or equal to capital N minus 1.

(Refer Slide Time: 34:19)



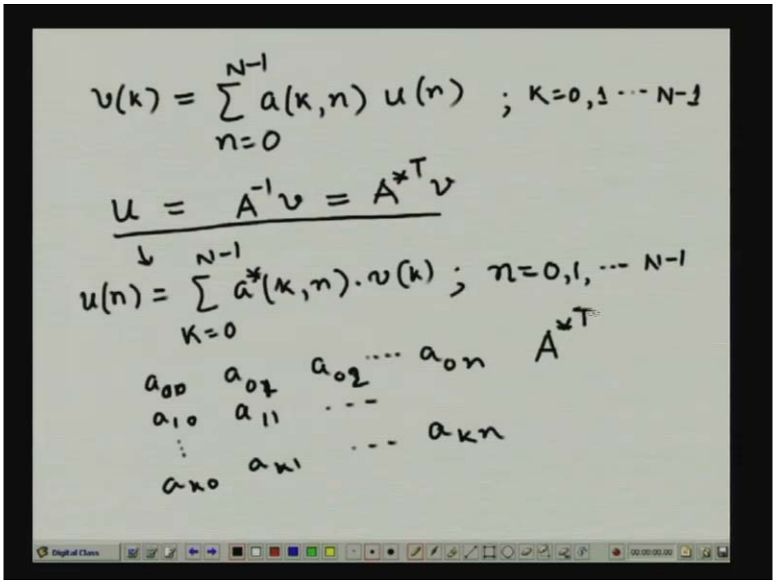
So, we have a series of discrete samples in n. So, in this case, we have capital N number of samples. So obviously, you can see that this is a 1 dimensional sequence of samples and because it is 1 dimensional sequence of samples and the sample size is capital N that is we have capital N number of samples; so, I can represent this set of samples by a vector say u of dimension capital N.

So, I am representing this by a vector u of dimension capital N and for transformation, what I do is I pre multiply this vector u by a unitary matrix A of dimension N by N. So, given this vector u, if I pre multiply this with a unitary matrix capital A where the dimension of this unitary matrix is n by n; so you find that this u is a vector of dimension N and I have a matrix a unitary matrix of dimension N by N. So, this multiplication results in another vector v.

So, this vector v we call as a transformed vector or transformation vector. This is transformed vector and this unitary matrix A is called the transformation matrix. So, what I have done is I have taken an N dimensional vector u, pre multiplied by pre multiplied that N dimensional vector u by a unitary matrix of dimension n by n. So, after multiplication, I got again an N dimensional vector v.

Now, so by matrix equation, this is v equal to A times u. If I expand this, so now what I do is I expand this matrix equation.

(Refer Slide Time: 36:51)



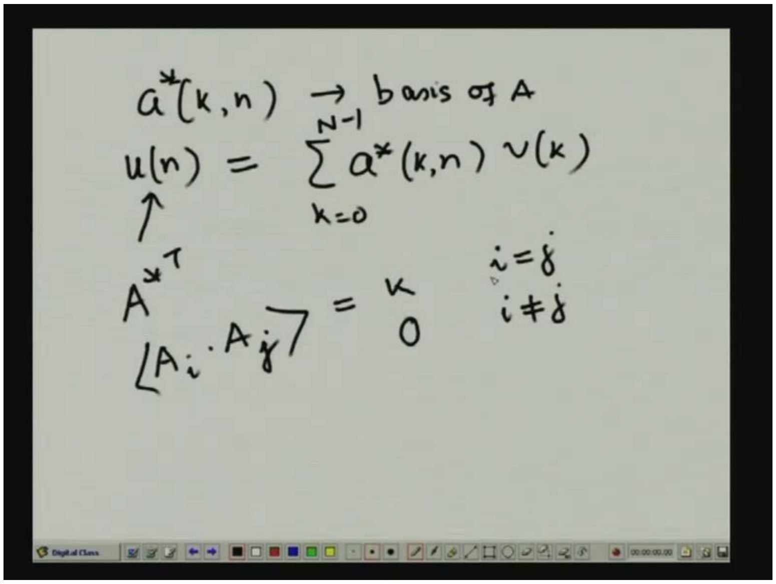
So, if I am expand this matrix equation, this can be represented as a series summation which will be given by v (k) is equal to a (k, n) into u (n) where n varies from 0 to capital N minus 1 and this has to be computed for k equal to 0, 1 upto N minus 1. So, I get the all the N elements of the vector v (k). Now, if A is a unitary matrix, then from vector v, I can also get back our original vector u. So for doing that, what we will do is we will pre multiply v by A inverse.

So, this should give me the original vector u and this A inverse v because this is an unitary matrix will be nothing but A conjugate transpose v and if I represent the same equation in the form of a series summation, this will come out to be u (n) is equal to a star (k, n) times v (k) where k will now vary from 0 to N minus 1 and this has to be computed for all values of n varying from 0, 1 upto N minus 1.

Now, you find that what is this a star (k, n)? Now, if I represent this a (k, n) or if I expand this matrix a (k, n), this is of the form a_{11} or a_{01} a_{02} a_{03} like this a_{0n} a_{10} sorry this is a_{00}, a_{01}, a_{02} upto a_{0n}, this will be a_{10} a_{11}. So, it will go like this and finally I will have a_{k0} a_{k1} like this, I will have a (k, n).

Now, find that in this expression, we are multiplying a (k, n) by v (k) or a (k, n) star which is the conjugate of a (k, n) into v (k). Now, this a (k, n) star is nothing but the column vector of matrix A star. So, if I have this matrix A, this a (k, n) this a (k, n) star is nothing but a column vector of matrix A star. So, this column vectors or column vectors of matrix A star transports.

(Refer Slide Time: 40:18)

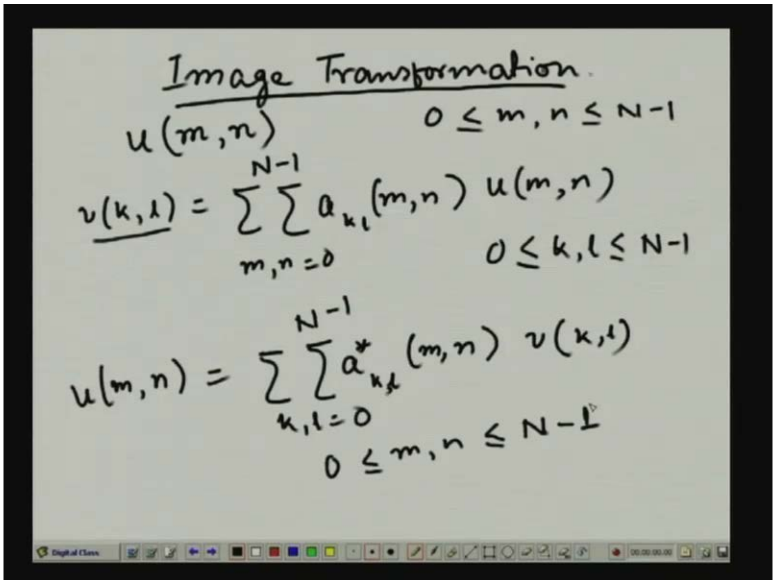


So, this column vectors are a (k, n) star a (k, n) star, this column vectors are actually called the basis vector of the matrix A and you remember this matrix A is an unitary matrix and here what we have done is the sequence of sample u (n) or vector u (n) has been represented because we have represented an u (n) as summation of a star (k, n) into v (k) for k equal to 0 to N minus 1. So, this vector u (n) has been represented has a series summation of of a set of basis vectors.

Now, if this basis vectors has to be orthogonal or orthonormal; then what is the property that it has to follow? So, if we have a set of basis vectors and in this case, we have said that the columns of A star transpose, this forms the set of basis vectors. So, if I take any 2 columns and take the dot product of these 2 those 2 columns, the dot product is going to be non zero the dot product is going to be 0 and if I take the dot product how the column with itself, this dot product is going to be non zero.

So, if I take a column say a column i and take the dot product of A_i with A_i or I take 2 columns Ai and A_j and take the dot product of the these 2 columns; so this dot product will be equal to some constant k whenever i is equal to j and this will be equal to 0 whenever i is not equal to 0. So, if this property followed, then the matrix A will be a unitary matrix. So, in this case, we have represented the vector v or vector u by a series summation of a set of basis vectors. So, this is what we have got in case of a 1 dimensional signal or a 1 dimensional vector, vector u.

(Refer Slide Time: 42:53)



Now, we are talking about image transformation. So, in our case, our interest is on image transformations. Now, the same concept of representing a vector as a series summation of a set of basis vectors can also be extended in case of an image. So, in case of an image, the vector u that we have defined in the earlier case, now it will be a 2 dimensional matrix. So, u instead of being a vector, now it will be a 2 dimensional matrix and we represent this by u (m, n) where m and n are row and column indices where 0 is less than are equal to m, n and less than or equal to n minus 1. So, see that we are considering an image of dimension capital N by capital N.

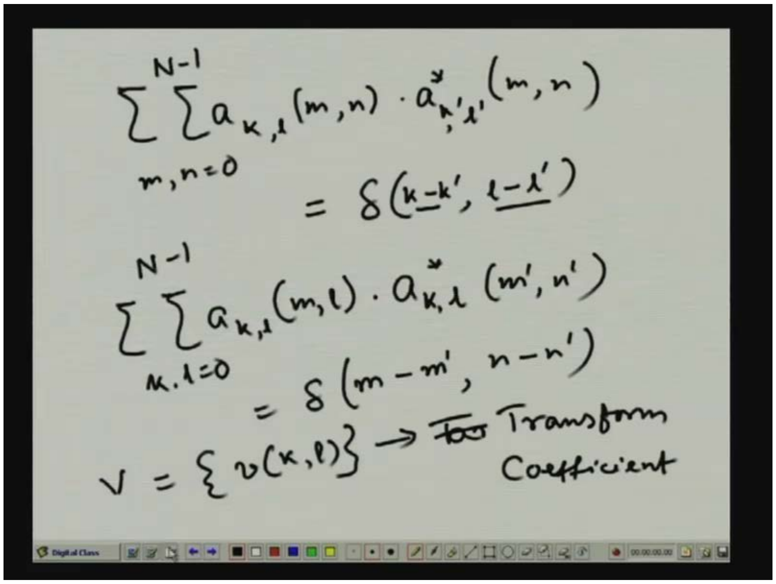
Now, transformation on this image can be represented as v (k, l) will be equal to again we take the series summation $a_{k\,l}$ (m, n) into u (m, n) where m and n vary from 0 to capital N minus 1. So, here you find that $a_{k\,l}$ is a matrix again of dimension capital M by capital N but in this case, the matrix itself has an index k, l and this computation v (k, l) has to be done for 0 less than or equal to k l less than or equal to capital N minus 1.

So, this clearly shows that the matrix that we are taking this is of dimensional capital N by capital N and not only that we have capital N into capital N that is N square capital N square number of such matrices or such unitary matrixes. So, this $a_{k\,l}$ (m, n) because kl k and l, both of them take the values from 0 to capital N minus 1; so I have capital N square number of unitary matrices and from this v (k, l) which is in this case the transformation matrix, I can get back this original matrix u (m, n) by applying the inverse transformation.

So, in this case u (m, n) will be equal to again double summation a star k dash l dash into v sorry a star kl into let me rewrite this. We will have u (m, n) will be given by double summation a star$_{k\,l}$ (m, n) v (k, l) where k, l varies from 0 to N minus 1 and this has to be computed for 0 less than or equal to m, n less than or equal to capital N minus 1.

So, you find that by extending the concept of series expansion of 1 dimensional vector to 2 dimensions, we can represent an image as a series summation of basis unitary matrices. So, in this case, all of akl or is that all of $a_{k\,l}$ (m, n) will be the unitary matrices.

(Refer Slide Time: 47:40)



Now, what is the orthogonality property? What is meant by orthogonality property in case of the matrix? The orthogonality property says that for this matrix A it says that $a_{k\,l}$ (m, n) into a $_{k\ dash\ l\ dash}$ (m, n), if I take the summation for m, n equal to 0 to capital N minus 1; this will be equal to a Kronecker delta function of k minus k dash and l minus l dash.

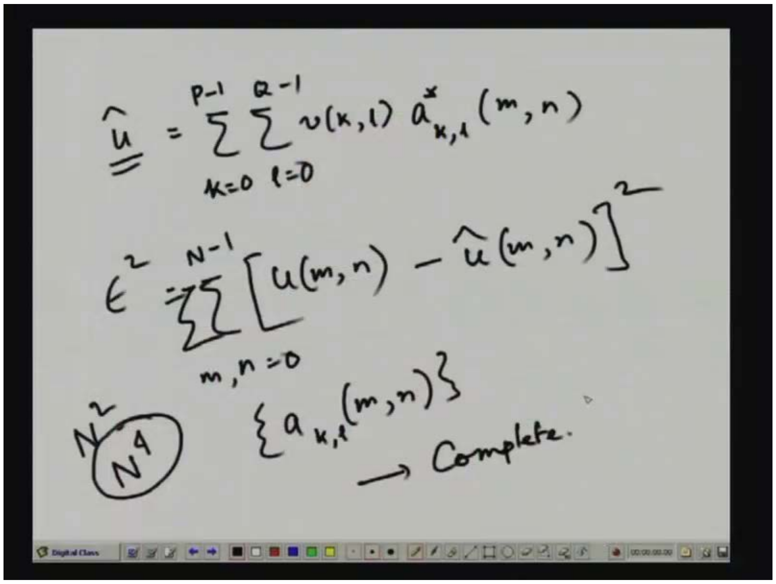
So, it says that this functional value will be equal to 1 whenever k is equal to k dash and l equal to l dash. In all other cases, this summation will be 0 and the completeness is that if I take the summation $a_{k\,l}$ (m, n) into $a_{k\,l}$ sorry this should be a $_{k\ dash\ l\ dash}$ star. So, $a_{k\,l}$ star m dash n dash. Summation is taken over k and l equal to 0 to capital N minus 1. This will be equal to Kronecker delta function m minus n dash and n minus n dash.

So, it says that this summation will be equal to 1 whenever m is equal to n dash and n is equal to n dash. So, the matrix by applying this kind of transformation, the matrix v which we get which is nothing but set of v (k, l), this is what is called the transformed matrix or the transformation coefficients. So, this is also called the transform coefficients.

So, you find that in this particular case, any arbitrary image is represented by a series summation of a set of basis images or a set of unitary matrices. Now, if we truncate the summation; so in this case, what we get is we get the set of coefficients and the coefficient size the same as the original image size. That is if we have m by n image, our coefficient matrix will also be of m by n.

Now, while doing the inverse transformation, if I do not consider all the coefficient matrices, I consider as a sub set of it; in that case, what we are going to get is an approximate reconstructed image and it can be shown that this approximate reconstructed image will have an error, a limited error if the basis matrices that we are considering, the set of basis matrices or set of basis images that is complete. So, this error will be minimized if the basis image that we considered is complete basis images.

(Refer Slide Time: 50:48)



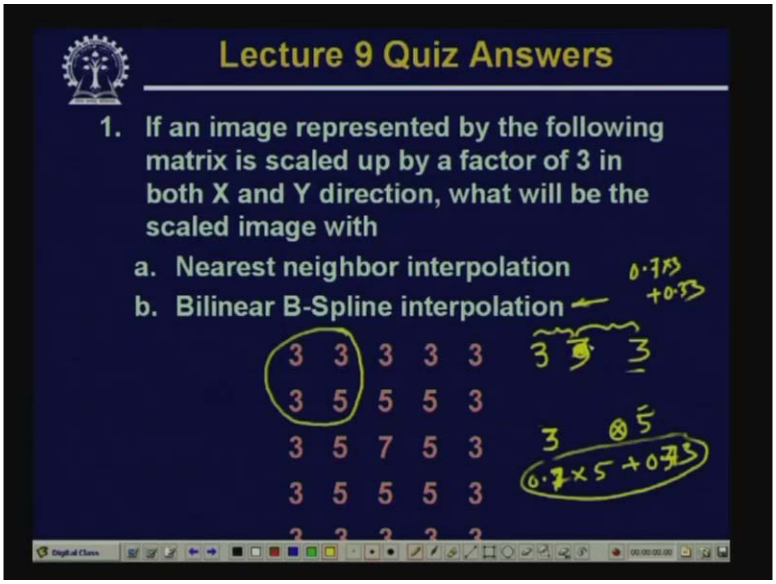
So in that case, what we will have is the reconstructed image you had will be given by double summation say v (k, l) into a star$_{k\,l}$ (m, n). Now suppose, l will vary from 0 to Q minus 1 and say k will vary from 0 to P minus 1. So, instead of considering both k and l varying from 0 to N minus 1, I am considering only Q number of coefficients along l and p number of coefficients along k. So, the number of coefficients that I am considering for reconstructing the image or for inverse transformation is P into Q instead of n square. So, this P into Q is using this P into Q number of coefficients, I get the reconstructed image u hat.

So obviously, this u hat is not the exact image. It is an approximate image because I did not consider all the coefficient values and the sum of squared error in this will be given by epsilon square equal to u (m, n) that is the original image minus u hat m, n which is the approximate reconstructed image, square of this and you take the summation over m, n varying from 0 to N minus 1 and it can be shown that this error will be minimized if our set of basis images that is a$_{k\,l}$ (m, n), this is complete.

Now another point that is to be noted here; if you compute the amount of computation that is involved, you will find that if N square is the image size, the number of computations or amount of computations that will be needed both for forward transformation and for inverse transformation will be of order N to the power 4. So, for doing this, we have to have we have to incurve tremendous amount of computation.

So, one of the problem is how to reduce this computational requirement when we go for inverse transformation or whenever we go for forward transformation. So, we will continue with the discussion of this unitary transformation in our next class.

(Refer Slide Time: 53:44)



Now, today let us consider the solution of the quiz questions that we have given at the end of lecture 9. So, you remember that at the end of lecture 9 or during lecture 9, we were discussing about image interpolation and image re-sampling and one of the applications of this we have said that if I scale an image or scale of an image or if I rotate an image or if I want correct an image; there will be many cases where the information will not be available at regular grid points.

So, interpolation and resembling aims to fill up those grid points where those where no information is is available in the transformed image. So, here we have given a problem where this image will be expanded by 3.

So, for solving this particular problem if I consider this particular part, the same will be applicable for the entire part. So, if I expand this or if I scale it up by a factor 3, you will find that I will get a value 3 here. Then for 2 subsequent locations, there will not be any information. I will also have a 3 here; here again, for 2 subsequent locations, there will be no information. I will have a 3 here, here again for 2 subsequent locations I will not have any information, here I will not have any information, here I will not have any information, here I will have the value 5.

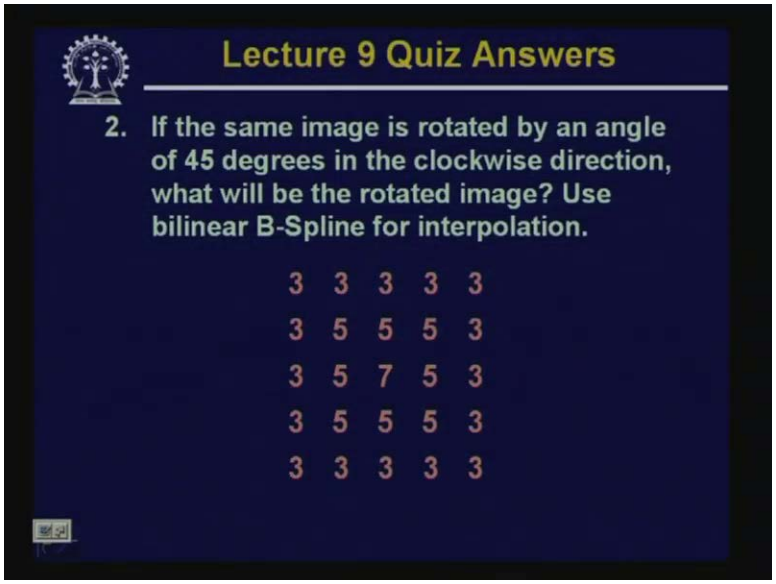
Now, the purpose is I have to fill up all these blank spaces. Now, you find that if I go for the nearest neighbor interpolation, the nearest neighbor interpolation says that you fill up this location with the value of the pixel which is nearest to it. So, if I go for that nearest neighbor interpolation; this location will be filled up by 3, this location will be filled up by 3, this location will be filled up by 3, this location will be filled up by 2, this will be filled up by 3, this will be filled up by 3, this will be filled up by 3, this will also be filled up by 3, this will be filled up by 3 whereas these 3 locations will be filled up by value 5.

So, this is what we have in case of nearest neighbor interpolation. If I go for bilinear B spline interpolation, then what i have to do is when I want to fill up these bank locations; so here I had the value of 3, here I had the value 5. So, whenever I want to fill up this particular bank location, I have to take a linear combination of this pixel and this pixel.

This will get a weightage of the distance of this pixel from this particular point and a weightage of this pixel, the weight is given by distance of this pixel from this particular point. So, we will find that this particular point will be filled up by a value which is equal to 0.7 times 3 plus 0.3 times 3 which in this particular case is also equal to 3. But when I come to this particular pixel, the value that will be assigned to this particular pixel will be equal to 0.3 times 5 plus 0 0.7 times 3 sorry it is the reverse. This is 0.7 times 5 plus 0.3 times 3.

So, if I compute this, I can find out what will be the value at this particular location. So, using similar procedure, I can find out what will be the values at all the points where i do not have any information in the transformed image. So, using similar procedure, you can find out the values at all the positions, all such blank positions where there is no value.

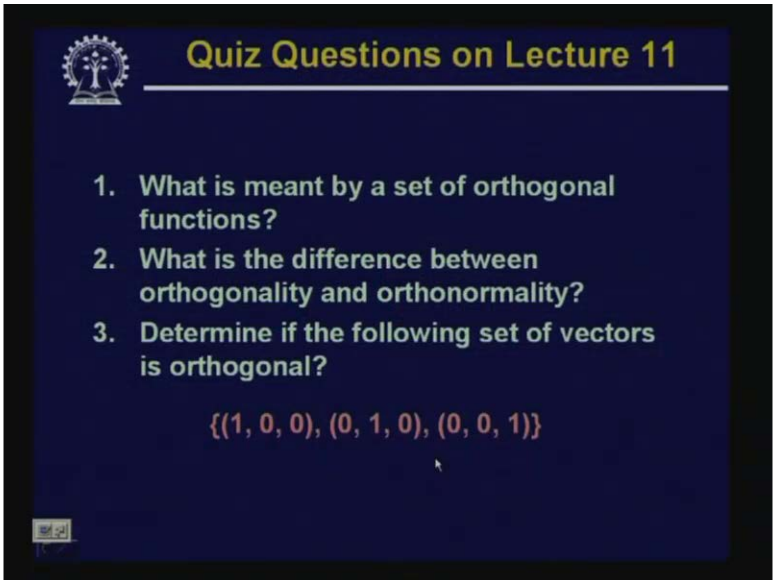
(Refer Slide Time: 57:54)



The second one is also of the same type where the image is to be rotated by 45 degree and here again after rotating the image by 45 degree; we will find that there are some locations where there is no incensory value. So, for those locations, what you have to do is you have to go for inverse geometric transformation and then you find that to which particular point in the original image, this point lies. And you have to interpolate the value at that particular point using either Bi - linear transformation as is given in this problem following the bilinear interpolation that we have discussed and the value that we get there; you replace that in the corresponding location in the transformed image.

So, following this similar procedure, you can also find out what will be the rotated image, rotated interpolated image. Now, coming to today’s questions.

(Refer Slide Time: 58:48)



So, there are some quiz questions on today's lecture. First one is what is meant by a set of orthogonal functions? What is the difference between orthogonality and orthonormality? The third problem: determine if the following set of vectors is orthogonal or not. The vectors are (1, 0, 0) (0, 1, 0) and (0, 0, 1).