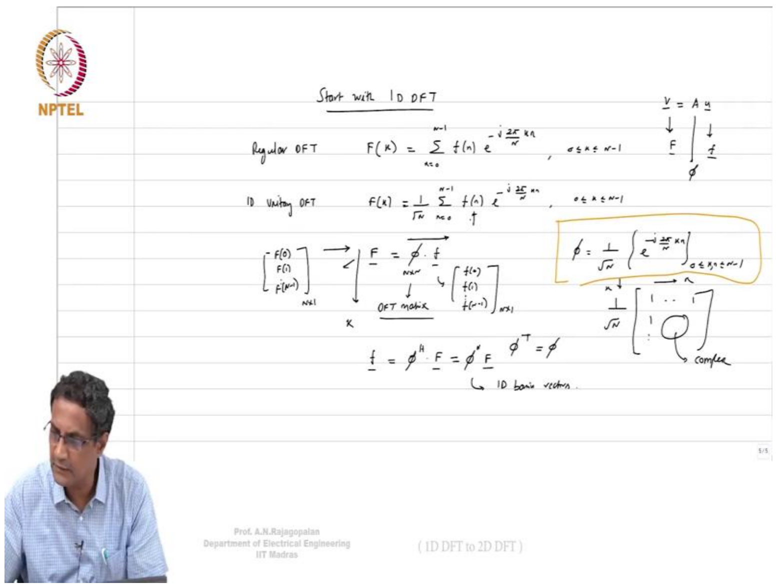
Image Signal Processing
Professor A. N. Rajagopalan
Department of Electrical Engineering
Indian Institute of Technology, Madras
Lecture - 45
1D DFT to 2D DFT

(Refer Slide Time: 00:16)



We will start with actually 1D DFT and then, we will start will 1D DFT and we want to get a seamlessly go to really a 2D DFT and we do not want to strain our limbs. So it should be like, it should be like a straight forward thing.

Now, a 1D DFT, so if you, if you see a regular DFT, this is what I think your MATLAB computes, and so it is like, suppose I mean, now I will not use V and, well, I will use something like a capital F and small f, just because it is a Fourier transform. I will say capital F of k, that is what we normally use, is equal to summation f of n at e raised to minus j 2 pi by N kn. Then what is it? So this sum goes from n is equal to 0 to n minus 1, and this goes from k, goes from 0 to n minus 1.

So a regular sort of a DFT is this but the unitary DFT, a 1D unitary DFT, 1D unitary DFT if you look at it that will have a scale factor here, which is 1 by root N summation n equals 0 to n minus 1 f n e power minus j 2 pi by N kn. All else remaining the same.

So F of k is equal to this, then so if you want to look at a DFT matrix phi, so the same operation, we would also want to think about it as a matrix-vector, vector kind of (multi) because as far as possible, we do not want to dabbling with summations and all because when you look at matrices then there is a lot more structure to all of this.

So we would want to look up on this as a vector F, which is like F of 0 so this has all the coefficients of, this is all the, all these transform coefficients F of 0, F of 1, F of n minus 1, this is your F and you want to be able to express it as some matrix phi times small f. So small f is this guy, is this sequence f of 0, f of 1, all the way up to f of n minus 1. So this is like N cross 1, this is N cross 1 and therefore, of course, this guy is N cross N, phi; and this matrix is what we call really as a DFT matrix.

Now, if you go back to our unitary DFT thing, so what we had? We had V is equal to A times u. So V, we have replace by F; u, we have replace by small f and then all else remain is, and then A we have, of course, replace by phi now. So we are saying that if you wanted a 1D DFT then simply replace A by phi, where this phi consists of entries such that, phi is a matrix, of course, N cross N consists of entries 1 by root N e power of minus j 2 pi by N kn 0 k comma n going from n minus 1.

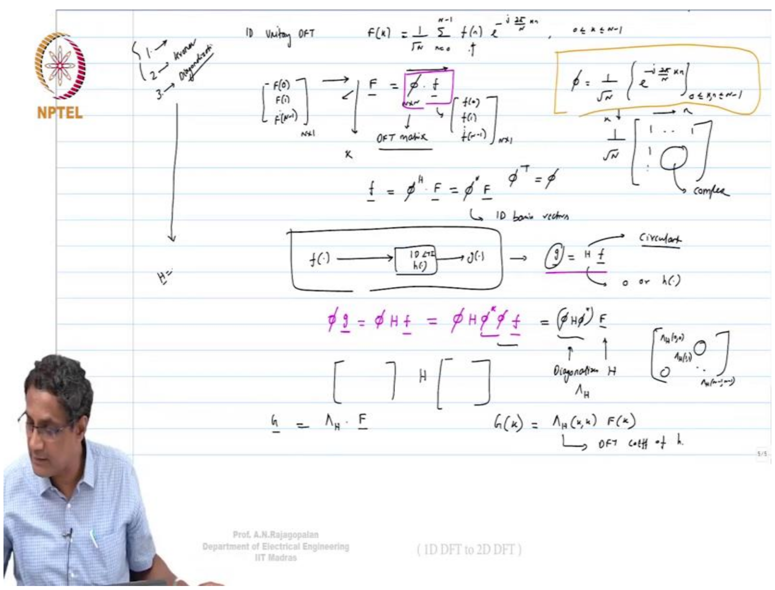
So if you look at it, it will have like 1 by root N and then, and as you know n varies in this manner, k varies in this manner. So n equal to 0 to n minus 1 as we walk along the columns, k equal to 0 to n minus 1 as you walk along the rows because that is a way it is structured, F; so if you go down F, you get, you have a variation of k; if multiply phi with F you have a variation in n because a row multiplying f is like a change in n.

So the first entries will all be 1, because either k is 0 or n is 0, and then inside, of course, you will have all kinds of e power minus j, whatever depending on the size of the matrix. So these entries are typically complex unless you take the 2 cross 2 like we did yesterday, that was real but in general, it is complex, these entries are all complex and therefore, we would, of course, expect that similar to the case that we had. So we can, we should be able to, of course, invert this guy and we should be able to say that is phi Hermitian F.

And it turns out that for a DFT matrix because it has a certain structure, DFT comes like this, this is not, this is not kind of any old A. I mean this is, this has the form, therefore, what do you think you can say about phi transpose? phi transpose is phi. phi transpose is phi and therefore, you can even, you do not even have to worry about the transpose sitting on this, you can simply say this phi star F.

So it shows, shows that how easy it is to get on to, get on to the inverse. Just take the original phi, take the complex conjugate, and then there you have and then the columns of phi Hermitian will now be or whatever phi star is. This will be, this will be the basis now, this will be the 1D basis. So these columns of same thing, whatever we did for general A, all those things will hold now. So 1D, 1D the basis vectors are all sitting as the columns of phi star or phi Hermitian, as you would want to call it.

(Refer Slide Time: 05:50) End Time:



Now, the point is, now suppose see, until now we have only like as I said, there were actually three points. Again you kind of go back to those three points that I introduced in the first class on unitary transforms. The first point was, was trying to establish a relation between data-independent transforms and dependent. Hey, by the way, data-dependent transform, SVD is one more thing that I forgot, KLT, I think I mentioned, SVD, I forgot to mention. Next day I was going to remember to tell you, I am sure I did not mentioned that.

So SVD we will take it up as sort of a dependent transform. Because the SVD basis come from the image itself. So that is, I should have mentioned that I just said KLT and I did not mentioned SVD. The second one was about, was about using a Kronecker thing as a vehicle to go from lower to higher dimensions. The third one which, so until now we have kind of hovered around these two, we have been jumping across those. But the third one, which is about you see, diagonalization, which we have not seen it till now.

So the diagonalization has two kinds of implications, especially when you looking at actually DFT, because the way some at the undergrad level when they actually teach signal and systems, they talk LTI systems and then they kind of then they would switch to Fourier, Fourier series, Fourier transforms, and so on. So if you have wondered even why that is so, this is how, those have very strong sort of a coupling between the two, and that coupling I thought we will try to see that here.

There also, I think it is fairly obvious, but here it will become even more obvious so as to what this is affinity of this DFT matrix, what kind of an affinity it has for certain systems, like when we say LTI system. So it has a natural affinity and that affinity, that is why in any signals and systems book, it is always like LTI system, then Fourier, Fourier, Fourier all the way.

But here, of course, we do not want to stop with Fourier, we also want to go forward go look at other transforms but then to sort of look at this third property which we have not examine, we will try to make a small move towards that. Assume that you we have a 1D system, which is, of course, LTI and all of that, a 1D LTI system, and let us say it has an impulse response h. And imagine that your input sequence is some f and then the output is some g. And we know that we saw that when you, we saw that g is equal to, so if we implement liner convolution via cyclic convolution, it say that g is equal to H times f.

This we, this we saw before and other time I had sort of left it here and I had said that we will revisit later. So this H had a structure to it, we called it a circulant matrix. This had a circulant structure. Now, for example, if you see this as an LTI system with an input and an output and an impulse response, we know that the entries of H are either 0 or it contains H in it. The matrix H, system matrix has only zeros and a small h in it sitting right inside it.

Now, the point is when I want to go to the Fourier domain, I know that I have my sequence g which is in the, it is in time domain. Now, what I, what should I do know? If I want a Fourier of this, now we saw just now that to get the Fourier transform, we should actually multiply by a

DFT matrix, that sequence of an appropriate length. So what this means is that I should multiply phi with g, I should do, I should operate phi on g.

But if I do that and because of the fact that this equation, this is the equation that is valid so this also mean that I should might have multiply phi on the right. So it makes it phi H f. Now, what do I do next? It still does not seem to be in a form that I would like to see it as. What should I do next? Some small trick I should play.

Student: Replace f with phi Hermitian.

Professor: How? How do I do that? But then it will actually change that equation. I cannot simply arbitrarily make f as phi Hermitian.

Student: phi g becomes capital G, phi h, phi Hermitian becomes some (())(10:37).

Professor: Okay. So you, I thought I did not hear that correctly. So what he saying, is correct. So what you should do, phi H, now we know that phi star, phi is in fact, identity. phi star phi is identity and you have f. So your equation is still balanced, that is no, it is no issue, phi star phi is identity. But now what happens is if you look at phi f that is F, that would be a DFT coefficients of the sequence f, and you are left with phi H phi star.

Now, this should be, this should be something you see, special. This should have a special affinity for H because it is like because you see normally what would, what would expect? If you did a convolution in time domain, the Fourier domain, it should become a product. So we have convolved, in time domain we have convolved H and f.

So on the one hand, I have here f, wherein the DFT coefficients of say, f are sitting. Now, this phi H is, phi H phi star it should then do something. What do you think it should be basically effectively do?

Student: Diagonalize.

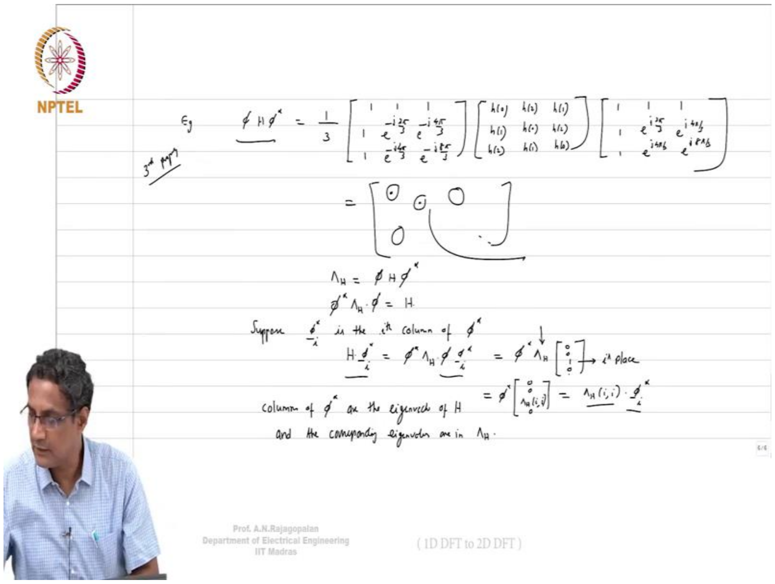
Professor: Diagonalize. Diagonalize H. Which is what it does. But then it does this diagonalization not because, not because it is some arbitrary H, it is because H is a circulant matrix. A DFT has a special affinity for a circulant matrix, it does not mean that we can take any matrix and then do phi A phi star and then we will see a diagonalized A and all, that would not happen.

Because of the fact that your H is actually a circulant matrix, it turns out that if you pre-multiply by phi and then post-multiply by phi star, then this diagonalizes H. Now, in order to show that, I mean, I am not going to show it, you can actually take an example. Write a simple MATLAB code where you can assume a certain size for phi and take your H to be a circulant matrix and then multiply that with an again phi star. You try this, I do not want to spend time showing all of that here. This is just straight forward, you should be able to show this.

Now this, suppose I indicate this as diagonal H, then what this means is that I have got like G is equal to diagonal H times F. Or in other words, I will have like G k, suppose I am looking at the kth coefficient that is diagonal H k, k F k. So if you look at this diagonal entries of this guy, this diagonal, it will be all 0 and just that you will have like diagonal H 0 comma 0, diagonal H 1 comma 1, all the way up to n minus 1 comma n minus 1. That is, that will be that matrix.

So naturally, this should be something, this cannot be, 1 is, of course, it is Eigenvalue of H, but then more than that it has a relation with small h, what is that? It is, in fact, the DFT coefficient of H. So for example, if I, because I have used a capital H already here, suppose I had written h as suppose I computed DFT of h, are you, are you following this? Because I can show this, I mean, you want me to show this then it will be okay. Now, you know what, I can take actually a 3 cross 3 example, which is what I am searching for whether I have. See, for example, I will just, maybe, I will just show you.

(Refer Slide Time: 14:39)



Suppose you had like a example, the 3 cross 3 example then what will, the way it will look like is phi H phi star will look like 1 by 3 1 1 1, you can go and verify this, 1 e power minus j 2 pi by 3, e power minus j 4 pi by 3, then this will be 1, this will be e power minus j 4 pi by 3, this will be e raise to minus j 8 pi by 3 and then you will have your circulant guy h 0, h 1, h 2, h 2, h 0, h 1, then h 1, h 2, h0 and then you will have phi star which is 1 1 1, 1 1 then e j 2 pi by 3, e j 4 pi by 3, e j 4 pi by 3, and e j 8 pi by 3.

So if you actually multiply this, so this diagonal matrix that you get, you can show that if you, say expand this then each of these diagonal entries is but the, is but the, say DFT. So the first entry will be the zeroth DFT coefficient of h. The second is that is diagonal h 1 comma 1 is like the first, not first, the second DFT coefficient. So h of 0, h of 1, h of 2, if you think about it till h of n minus 1, so each of this is the but the DFT coefficient of small h. Which is what it has to be it cannot be anything else because you are doing a product, you are doing a convolution in time domain that has to give you a product in the Fourier domain.

But normally, we do not see it this form, in a matrix form we do not see it often. We tend to write integrals and so on and then that is how we see but then it is nice to see that in a kind of a matrix-vector form, it is nice to see the circulant matrix coming in, sorry, DFT matrix coming in action with respect to a circulant matrix.

You can also show that, so for example, I mean so to show that these are the Eigenvalues and what are the Eigenvectors, so what we can do is, so we have a diagonal H is phi, H phi Hermitian strictly speaking, but let us say we call it as phi star now. Then it means that if I want H then I should multiply by, on the left by phi Hermitian, this guy; and then on the left, I should, no, phi star, no idea why am I writing, phi star delta h phi. So that is equal to H.

So now, suppose, suppose let us say, suppose I call phi I star, suppose is the ith column, is the ith column of phi star, this matrix. Suppose it is the ith column then you can see that what will happen, if I do, if I act H on phi i star, then that is a same as phi star diagonal H phi and then this vector phi i star. Phi and all is a matrix, phi i star is vector and because we know that phi phi star is identity, therefore when you multiply every row of phi with phi i star, what will happen? So you will get all zeros expect one in ith entry.

So, this will be like phi star diagonal H and then you will get a vector, it should be like 0 0 0 and then in the ith place because the ith row of phi when it multiplies phi i star that is when you will get 1 here. So this is like the ith, what is that, ith place, ith entry and then again zero. Now, since you have a diagonal matrix here, so what will this be equal to? phi star, again all zeros expect that here you will get, what you will get?

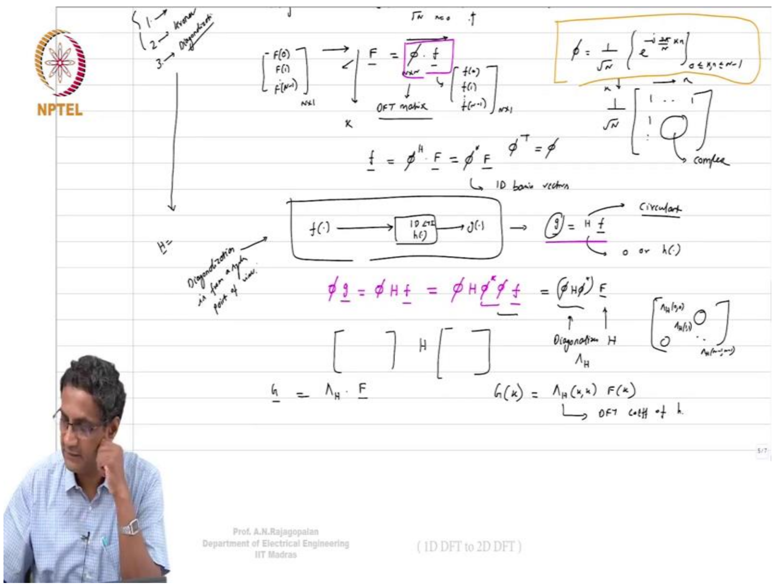
One entry you have to make, what will that entry be? Delta H is diagonal, delta H i comma i; diagonal i I mean, so calling it or you want to call it as lambda whatever you want to call it delta H i comma i and then 0. And then this we know is the same as a 0 times the first column of i star plus 0 times a second column plus, plus, plus, but then delta H i i comma i times the ith column of i i star.

So this is delta H i comma i phi i star. So you will see that H phi i star is some scalar times phi i star. So which means that the columns of, the columns of phi star not phi, the columns of phi star are the Eigenvectors of H and the corresponding Eigenvalues are given by and the corresponding Eigenvalues are in delta H, delta H i comma i and so on. I mean this is a kind of a fairly obvious thing but still, I just thought I will show this.

We are still in 1D. Now if you go further, so one way to look at this the action of, so now this is that third property. Third property where I said that we will look at diagonalization. And you know if you go back and see, I had said that diagonalization, both from a system point of view as well as a data de-correlation point of view. There were actually two things I had mentioned. So a system point of view is this for at least a DFT, because of the fact that Fourier transform it has this convolution product kind of a nice relationship not all transforms have it.

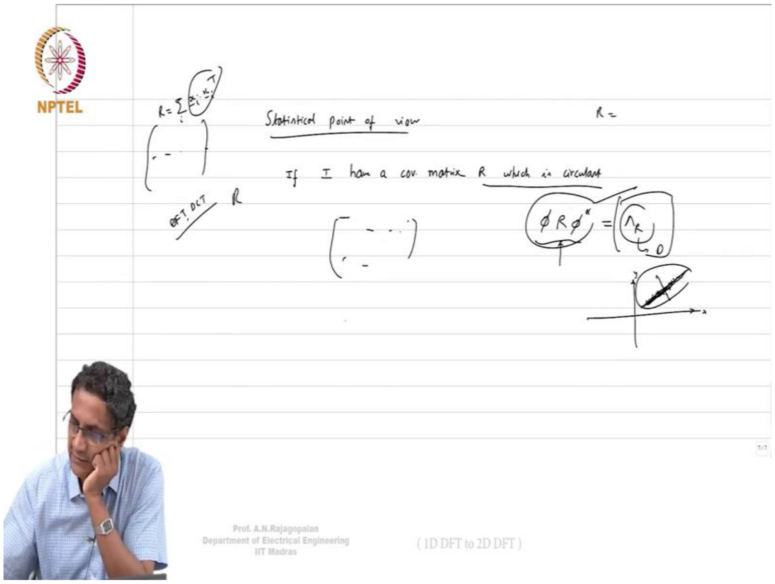
Now the other one, which is a common thing in fact across transforms that is what is called a statistical point of view. So I would rather call this as really a system point of view.

(Refer Slide Time: 21:22)



So this diagonalization is from really, really a system point of view. This is this diagonalization that we saw just now, it is from a system point of view, from a system point of view.

(Refer Slide Time: 21:42)



But the next thing that we will, that we will see is really a diagonalization from a statistical point of view, statistical point. Statistical in the sense that a data, a covariance, and so on, a statistical point of view.

So, from a statistical point of view this is actually interesting that if I have a covariance matrix R, if I have a covariance matrix R which is circulant and this, and of course, if you ask when does it happen if you have a periodic random sequence that will a R as circulant and, in fact, many times a Toeplitz matrix itself, one can show that asymptotically a circulants and Toeplitz looks the same as in, so as the size begins to become larger and larger. So in that sense, making a circulant assumption even for a Toeplitz is okay for large dimensions.

So, if you have a covariance matrix which is actually circulant, but then when you look at covariance then it brings in more structure. See, if you saw the H matrix that we had earlier, H, for example, we said it is circulant but then it need not to be symmetric and all. You can have a Toeplitz matrix but then there is no guarantee that it should be symmetric, which is why when you look at the coefficients, when you diagonalize you can get entries which are actually complex. Which is why we say that the DFT coefficients are in general complex, we would not expect them to be real.

But when you talk about a covariance matrix R, this is not same as really a system matrix now. This is a covariance of some data, so how do we calculate that? Let us say, I mean I give a lots of examples. One way to estimate is, so you sort of you take the average of let us say, if you dissolve 0 mean then you will do xi xi transpose and submit over all the examples. That is how you will estimate covariance because many a time, you have data but then you will have to estimate the covariance, nobody will directly gives you an analytical R.

But if you have an analytical R also, it is fine, sometimes, R is analytical also. If it is not then you will learn it from the data whatever it is. But then, when you have a covariance matrix it automatically, because of the way it comes a symmetry is automatically there. Look at the fact that expectation x x, I mean x x transpose already means that symmetry is there. Which then means that if you now try, because like I said, a DFT matrix has a special affinity for circulant but then it does not care whether that circulant is coming out of a system matrix or somewhere else.

So now, if try R suppose I do phi, R phi star the same operation that I did there, there I did it on H. Now instead of H, which is from a system point of view, now I come at look at a data covariance matrix R and suppose I try the same stuff, phi R phi star then what happens, this will diagonalize R. Like I said, as long as it is circulant, it will kind of say diagonalize it. But this diagonalization has a now a big implication because now it is like saying that if R has a strong correlation, if R is matrix and suppose, for example, if R is not diagonal you have got lots of off-diagonal entries that means, that means there is correlation.

Now this guy, when you take such a matrix and then if you try to do this operation, this will actually de-correlate the data. Decorrelation and this is something that we will see when we will do KL transform. It is simply means that it is reorienting from your standard basis to another basis in which things look.

So it is like saying that if you had a standard basis like these are your X and Y these are your standard basis, but then suppose I had some data like that you are very kind of see closely something like that. Now, as I increase X, I can see that Y is also increasing. So it means that there is a correlation, which is strong. Suppose we are slightly noisy, this is not exactly on the line, some are below some are above, something like that. It is kind of a noisy data which is around a line.

Now, if you are going to think about this line, it looks like as X increases Y is also increasing, that means there is a correlation. And so, if you compute a covariance matrix for this that will show up an R, that correlation will show up an R. But now, if you say that instead of this axis, suppose I take one of the axis to be this and other axis to be that, suddenly you see a de-correlation because now it looks like even if I do not transmit the one, which is, so for example if you look at the variance, the variance is very high along this but then the variance along the other in this one, direction of this data is very small.

So you might say that I might capture most of the information by just sending this and not even sending that, which may not be exact and all what error you incur in a mean square sense that we will see later, but I am saying. So this decorrelation is about all of this. So instead of a standard basis, you reorient with respect to a some other basis, and typically it is your Eigenvectors, that is where your KL transform PCA and all come in. Because the best basis that you can have that will de-correlate is in fact your say, Eigenvectors for that R.

And therefore, so here the implication is something else. Here it is not about the DFT and all of that. I mean, here, it is not about the convolution or anything, it is about taking a covariance R and if you had a covariance R that is circulant, then you could apply. Now, what normally happens is, sometimes if you had a choice, let us say I had DFT, I had DCT, and all of this, I had all of this, then what can happen is, it may happen that, it may so happen that none of them is really is able to diagonalize.

Suppose I have some R, whatever be it. I am not assuming it is circulant if it is circulant, so here is where somebody has said that there is a, there could be a bridge between a data-independent and actually a dependent transform. So this is one such case where a data-independent transform diagonalizes R without in the sense that, in a sort of a perfect sense, a perfect sense.

So in that sense, this is now the optimal transform, even though this is a data-independent because this phi basis is coming from nowhere, we have already fixed it. But if R turns out to be circulant, then the best transform that you can use to de-correlate the data is in fact, a DFT and in fact, DFT is in fact, the KLT.

So the bridge that I said, there could be bridges between data-independent and data-dependent transform that comes about in this case. But sometimes what may happen is none of them may be able to diagonalize it exactly. In the sense that after you, after you do this operation you may not get really a diagonal R, you may still have few things left here and there.

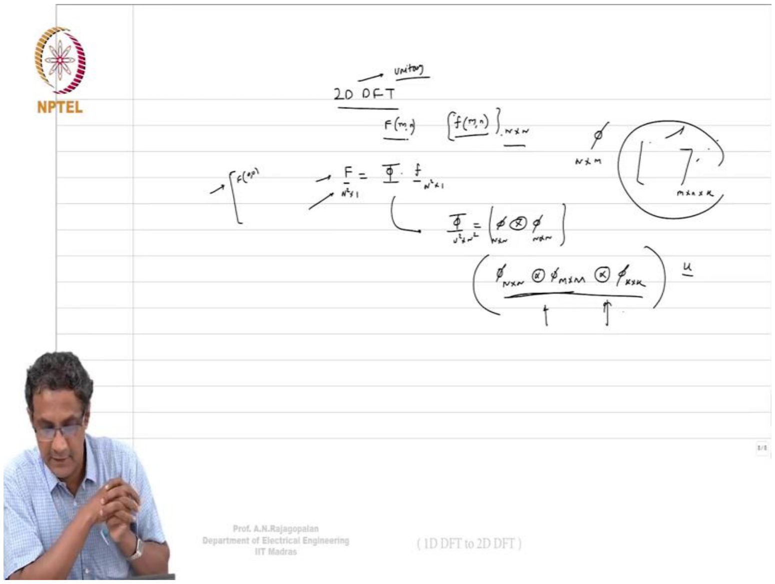
If that happens, then you might want to ask which transform should I then use for that kind R. So which is the reason why the statistical part is more, is more involved. The convolution part is only for LTI and then DFT is the best and then we stop there, but then when you talk about data de-correlation this is a third point that diagonalize ability that has more implications. Because any time you have data, because it is all about data, today, look at the outside world everybody talks about data analytics, data mining data, data, data, it is all data.

So if you construct covariance matrices out of all that data, and if you want to make effective sense out of them then all these guys are sitting out there and then people, what they do is effectively try to find out which one works the best and so on.

But this is a very simplistic thing, what they do is, what they do is far more complex but I am just saying if you just stop at second order, in the sense that if you stop at covariance, which is like a second-order statistics, if you just stop there, which means that you are sort of underlying, you are assuming that it has Gaussian, the underlying law is simply a Gaussian, it may not be true there might be higher-order statistics and all of that, but if you just stop that and you say that I treat all my data strictly Gaussian, then R is all that you have. Your sufficient statistics is basically mean and R.

So, with that this diagonalization property then it starts to assume importance and then you start worrying about which transform is best to use and so on. In this case, turns out that this will be a perfect diagonalization and this is what we mean by really a decorrelation, data decorrelation. Still in 1D. Now the move to actually 2D is very straight forward, we do not, we do not have to do much.

(Refer Slide Time: 30:26)



A 2D DFT now, if you ask and then if you know how to do a 2D DFT, then 3D DFT, a 10D DFT is all fine. A 2D DFT, what will we do, we will just go by whatever we did. So we will say, I mean if I had an image, suppose I call that as as f, f of m comma n and then if I want to compute my DFT coefficients, 2D DFT, from now on I will call this is as unitary. Automatically I will assume that I am talking about unitary DFTs only.

And then suppose you call that as, what is that, so F, a capital F of m comma n, then it means that in order to compute a 2D DFT of this, all that I will do is what? You will, you can all write this down now. F is equal to, so for example, if it is square and suppose this of size N cross N then you will simply take a capital Phi times f, where f is N square by 1, this F is N square by 1 and all these coefficients, which are your transform coefficients are all sitting as, so for example, if you look at the first entry, that will be like 0 comma 0, and then F of 0 comma 1, then 0 comma n minus 1, then 1 comma 0 all the way. They are all sitting in this N square by 1 sort of vector.

And this Phi and please note that I put it like sort of a capital kind of Phi, I am not writing it, so in 1D DFT, I wrote it like that, for 2D DFT I want to, I want you to be able to separate the two. So I am writing it as like this. So what is this guy? This has to become as phi then you take a Kronecker with phi. So whereas this phi is N cross N, this phi is N cross N with it corresponding to 1D and then you do this Kronecker, this will give you a Phi which is N square by N square as we know and that is what we will multiply this F. Straight forward in one shot.

Now, if instead of F of m comma, now if, let us say if this was actually a rectangular image what would you do? If this was m cross n, what would you do?

Student: (())(32:45)

Professor: Yeah, m cross m. Sorry, m cross m, not m cross n; n cross m, see this is a point. So, for example, if you had 2D image, instead of 2D, if I had a video, if I had some k number of frames then if you kind of look at those whole 3D object, what will that be? That will be like m cross n cross k.

Now, if wanted to, wanted a 3D sort of a DFT, what would you do now? You would extend it as what, phi. You would add one more simply a Kronecker, which will be phi K cross K. So this guy will actually multiply. Now u, now u will be what size? m into n into k by 1. So you will have to convert this whole thing into one vector, one long vector.

You may not implement it like this, I mean, nobody implements like this but then this is just that insight that going from a lower dimension towards say, a higher dimension, this is a straight forward. We just, so which is why I said that a Kronecker product is guy that kinds of sits and then now you can imagine if you go to a four-dimension and whatever, it should all make sense, one simply does not take it just for a heck of it, but if it makes sense to have a four-dimensional object whose Fourier transform you want to take, 4D DFT, and then again, you can do all of that.