Multidimensional Signal Processing

Mark Eisen, Alec Koppel, and Alejandro Ribeiro

Dept. of Electrical and Systems Engineering
University of Pennsylvania
aribeiro@seas.upenn.edu
http://www.seas.upenn.edu/users/~aribeiro/

March 15, 2018

It's all oh so simple

- Once and again, things are invisible or obscure in time domain
  - ⇒ But they become visible and clear in the frequency domain
- Even when clear in time, they are easier to understand in frequency
- Literally a new sense to view things that are otherwise invisible

"On ne voit bien qu'avec le coeur.
L'essentiel est invisible pour les yeux."

The Little Prince

- One sees clearly only with the frequency
  The essential is invisible to the eyes

Signal representation

- Why a new sense? $\Rightarrow$ We can write signals as sums of shifted deltas

$$x(n) = \sum_{k=1}^{N} x(k)\delta(k-n) \tag{1}$$

- Conceptually, the same as writing signals as sums of oscillations

$$x(n) = \sum_{k=1}^{N} X(k)e^{j2\pi kn/N} \tag{2}$$

- Only difference is that we sense time but we don't sense frequency
- We say we change the signal representation or we change the basis
- It all hinges in our ability to represent the signal in a different domain

Moving forward

- If something is obscure in time but also obscure in frequency
  $\Rightarrow$ Change the representation $\equiv$ Change the basis
- Images $\Rightarrow$ multidimensional DFT, Discrete cosine transform (DCT)
- Stochastic processes $\Rightarrow$ Principal component analysis (PCA)
  $\Rightarrow$ Eigenvectors of the correlation matrix
- Signals defined on graphs $\Rightarrow$ Graph signal processing
  $\Rightarrow$ Eigenvalues of the graph Laplacian

Images

Signal representation

Images

Two dimensional discrete signals

Two dimensional (2D) discrete Fourier transform (DFT)

Two dimensional (2D) inverse (i) discrete Fourier transform (DFT)

Energy conservation (Parseval's theorem)

Convolution in 2 dimensions

Applications

Discrete Cosine Transform

2D Discrete Cosine Transform

JPEG image compression

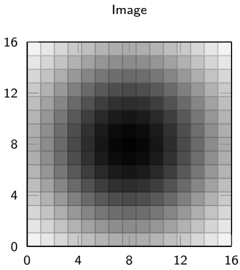
Images

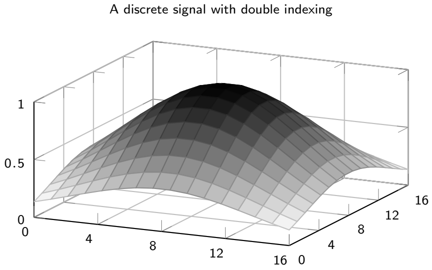
- A grid of pixels. Values define the luminescence of the point
  - $\Rightarrow$ In a black and white image
- In a color image we record multiple channels for different colors
  - $\Rightarrow$ E.g., red, green, and blue (RGB). Or Yellow Magenta Cyan blacK

- Not unlike signals we studied except that defined over two indices

Images as signals

- An image on the left and a signal on the right
  $\Rightarrow$ These are just different ways of visualizing the same information





- Can we perform DFT of image? $\Rightarrow$ Yes, vectorize the matrix
- Vectorization records nearby pixels far away $\Rightarrow$ 2D signal processing

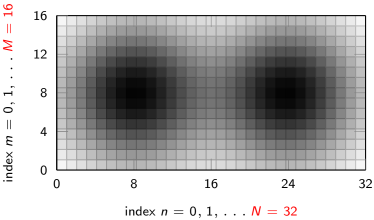
Two dimensional discrete signals

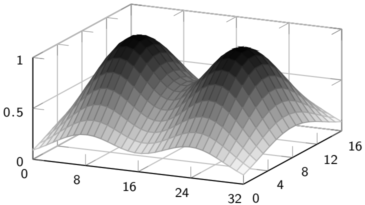
Two dimensional discrete signals

- Two dimensional (2D) discrete signal indexed by two indices (m, n)

$$\begin{aligned} m &= 0, 1, \ldots, M-1 = [0, M-1] \\ n &= 0, 1, \ldots, N-1 = [0, N-1] \end{aligned}$$

- M rows and N columns. A total of MN different indices





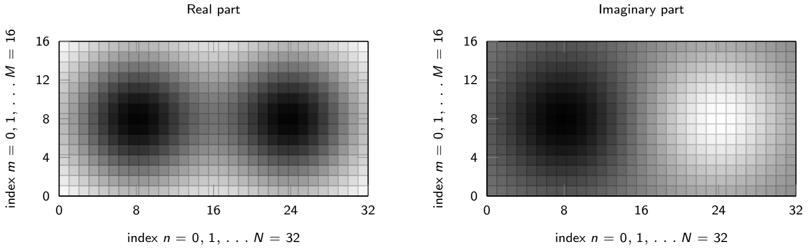
- 2D signal formally defined as map $x : [0, M-1] \times [0 : N-1] \to \mathbb{R}$
- The value that the signal takes at indices (m, n) is $x(m, n)$

Complex 2D signals

- As in one dimensional case, may want to define complex signals

$$x : [0, M-1]x[0 : N-1] \to \mathbb{C} \quad (3)$$

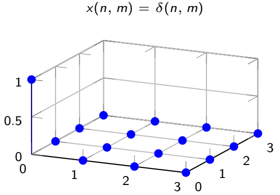
- Space of $M\times$ 2D signals = space of $M \times N$ matrices $\mathbb{C}^{MxN}$ or $\mathbb{R}^{MxN}$

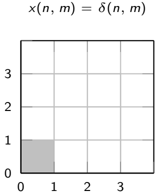


- Because, unsurprisingly, we are going to define two dimensional DFT

- 2D delta function $\delta(m, n)$ is a spike at (initial) position $(m, n) = 0$

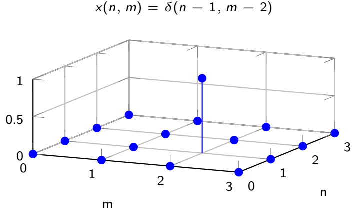
$$\delta(m, n) = \begin{cases} 1 & \text{if } m = n = 0 \\ 0 & \text{else} \end{cases} \tag{4}$$





- Shifted delta $\delta(m - m_0, n - n_0)$ has a spike at $(m, n) = (m_0, n_0)$

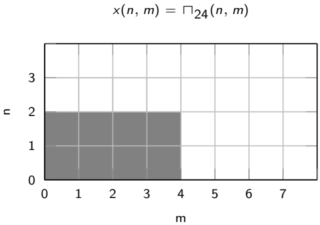
$$\delta(m - m_0, n - n_0) = \begin{cases} 1 & \text{if } (m, n) = (m_0, n_0) \\ 0 & \text{else} \end{cases} \tag{5}$$



Rectangular pulses

- Rectangular pulse of N_0 rows and M_0 columns $\sqcap_{M_0 N_0}$ is defined as

$$\sqcap_{M_0 N_0}(m, n) = \begin{cases} 1 & \text{if } m < M_0, n < N_0, \\ 0 & \text{else} \end{cases} \tag{6}$$



- If $M_0 = N_0$, rectangular pulse is said square. Denote $\sqcap_{N_0 N_0} = \sqcap_{N_0}$
- Can consider shifted pulses $\sqcap_{MN}(m - m_0, n - n_0)$

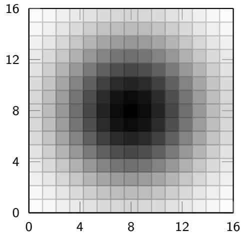
  $\Rightarrow$ Shifts must satisfy $m_0 < M - M_0$ and $n_0 < N - N_0$

Symmetric Gaussian pulses

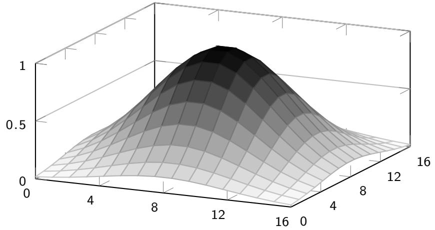
- A 2D Gaussian pulse of mean μ and variance σ^2 is defined as

$$g_{\mu\sigma}(m,n) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{(m-\mu)^2}{2\sigma^2} - \frac{(n-\mu)^2}{2\sigma^2}\right] \qquad (7)$$

Gaussian pulse, mean $\mu = 8$, variance $\sigma^2 = 16$



Gaussian pulse, mean $\mu = 8$, variance $\sigma^2 = 1$



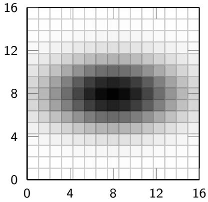
- An actual bell shape. The pulse is symmetric centered at (μ, μ)
- Variance σ^2 controls how fast the pulse decays

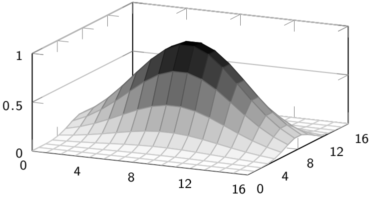
Generic Gaussian pulses

- Different centers in each coordinate and different variances
- Define coordinate vector $\mathbf{n} = [m, n]^T$. Just a variable
- Define center vector $\boldsymbol{\mu} = [\mu_1, \mu_2]^T$. Center coordinates
- Define covariance matrix $\mathbf{C} = \begin{pmatrix} \sigma_{11}^2 & \sigma_{12} \\ \sigma_{12} & \sigma_{22}^2 \end{pmatrix}$
- Diagonal controls stretch in each direction. Off diagonals rotation
- The 2D Gaussian pulse of mean $\boldsymbol{\mu}$ and covariance $\mathbf{C}$ is

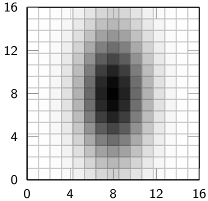
$$g_{\mu\sigma}(n, m) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{1}{2}(\mathbf{n} - \boldsymbol{\mu})^T \mathbf{C}^{-1}(\mathbf{n} - \boldsymbol{\mu})\right] \qquad (8)$$

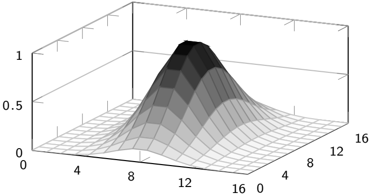
- A Gaussian pulse skewed in the *m* direction $\Rightarrow \mathbf{C} = \begin{pmatrix} 16 & 0 \\ 0 & 4 \end{pmatrix}$





- A Gaussian pulse skewed in the *n* direction $\Rightarrow \mathbf{C} = \begin{pmatrix} 4 & 0 \\ 0 & 16 \end{pmatrix}$





Inner product

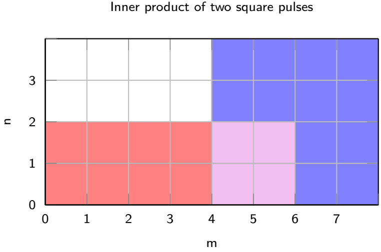
- Given 2D signals x and y define the inner product of x and y as

$$\langle x, y \rangle := \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m,n) y^*(m,n) \tag{9}$$

- It has the same properties of other inner products we encountered
  - $\Rightarrow$ Is a linear operator $\Rightarrow \langle x, y+z \rangle = \langle x, y \rangle + \langle x, z \rangle$
  - $\Rightarrow$ Reversing order entails conjugation $\Rightarrow \langle y, x \rangle = \langle x, y \rangle^*$
- It also has the same interpretation $\Rightarrow$ How much x looks like y
  - $\Rightarrow$ Positive = Positive correlation = same direction
  - $\Rightarrow$ Negative = Negative correlation = opposite directions
  - $\Rightarrow$ Null = Uncorrelated = Orthogonal = Perpendicular

Inner product of two rectangular pulses

- The inner product of two square pulses is the number of pixels in which both pulses are active (both are one)



- In the inner product sum $\langle x, y \rangle = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m,n) y^*(m,n)$ only the terms in which both pulses are not null count

Norm and energy

- The norm of the 2D signal x is $\Rightarrow \|x\| := \left[\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} |x(m,n)|^2\right]^{1/2}$

- We define the energy of the 2D signal x as the norm squared

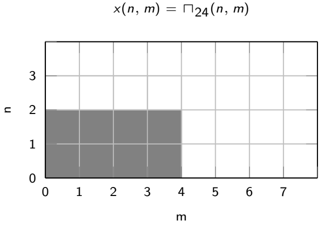
$$\|x\|^2 := \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} |x(m,n)|^2 = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} |x_R(m,n)|^2 + \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} |x_I(m,n)|^2 \tag{10}$$

- We can write the energy as self inner product $\Rightarrow \|x\|^2 = \langle x, x\rangle$

Energy of a square pulse

- Rectangular pulse of N_0 rows and M_0 columns $\sqcap_{M_0 N_0}$ is defined as

$$\sqcap_{M_0 N_0}(m, n) = \begin{cases} 1 & \text{if } m < M_0, n < N_0, \\ 0 & \text{else} \end{cases} \quad (11)$$



- To compute energy of the pulse we just evaluate the definition

$$\| \sqcap_{M_0 N_0} \|^2 := \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |\sqcap_{M_0 N_0}(m, n)|^2 = \sum_{m=0}^{M_0-1} \sum_{n=0}^{N_0-1} 1^2 = M_0 N_0 \quad (12)$$

- The energy is the number of pixels $(M_0 N_0)$ in the square pulse
- Can normalize by $1/\sqrt{M_0 N_0}$ to obtain pulse of unit energy

Two dimensional (2D) DFT

Signal representation

Images

Two dimensional discrete signals

Two dimensional (2D) discrete Fourier transform (DFT)

Two dimensional (2D) inverse (i) discrete Fourier transform (DFT)

Energy conservation (Parseval's theorem)

Convolution in 2 dimensions

Applications

Discrete Cosine Transform

2D Discrete Cosine Transform

JPEG image compression

Definition of 2D DFT

- 2D signal x With N rows and M columns. Elements $x(m, n)$
- We will focus on signals with $M = N$. To simplify notation.

- Signal X is the 2D DFT of x if its elements $X(k, l)$ are

$$X(k,l) := \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m,n) e^{-j2\pi(km+ln)/N} \quad (13)$$

- As in 1D we write $X = \mathcal{F}(x)$.
- X may be complex even for real 2D signals x. Focus on magnitude.

- Argument k is horizontal frequency and l is the vertical frequency

The 2D DFT and the (regular, 1D) DFT

- Separate terms in the exponent and regroup factors to write

$$X(k,l) := \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} \left[\frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(m,n) e^{-j2\pi ln/N} \right] e^{-j2\pi km/N} \qquad (14)$$

- For fixed m, the term between parentheses is the DFT of $x(m,\cdot)$
- We then take the DFT of the resulting DFTs with respect to m

- The 2D DFT of x is the column-wise DFT of the row-wise DFTs
- Or the row-wise DFT of the column-wise DFTs. Just the same

- Useful to know. Not a new computation

Discrete complex exponentials

- 2D Complex exponential of horizontal freq. k and vertical freq. l

$$e_{klN}(m,n) = \frac{1}{N} e^{j2\pi(km+ln)/N} = \frac{1}{\sqrt{N}} e^{j2\pi(km/N)} \frac{1}{\sqrt{N}} e^{j2\pi(ln/N)} \quad (15)$$

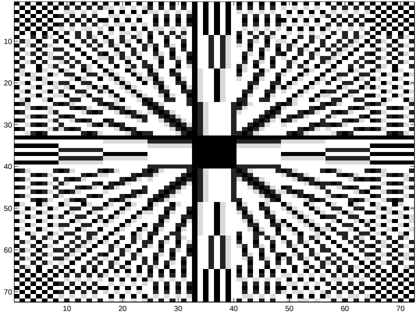
- Separate the exponential into two factors to write

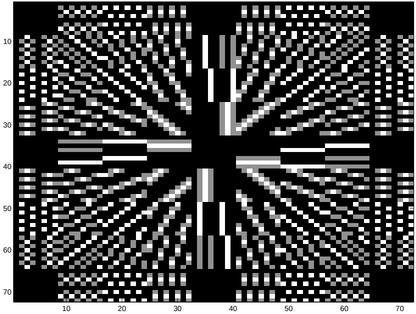
$$e_{klN}(m,n) = \frac{1}{\sqrt{N}} e^{j2\pi(km/N)} \frac{1}{\sqrt{N}} e^{j2\pi(ln/N)} = e_{kN}(m) e_{lN}(n) \quad (16)$$

- 2D complex exponential is product of two 1D complex exponentials

What 2D complex exponentials look like

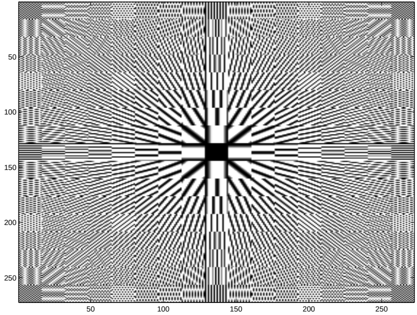
- Signal length $N = 8$. Total of $N^2 = 64$ different exponentials

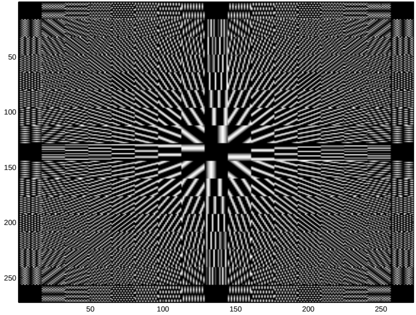




- Horizontal / Vertical frequency $\Rightarrow$ Horizontal / Vertical variability
- Diagonals $\Rightarrow$ diagonal variabilty $\Rightarrow$ Directionality also important

- Signal length $N = 16$. Total of $N^2 = 256$ different exponentials





- Horizontal / Vertical frequency $\Rightarrow$ Horizontal / Vertical variability
- Diagonals $\Rightarrow$ diagonal variabilty $\Rightarrow$ Directionality also important

DFT elements as inner products

- Rewrite 2D DFT using definition of 2D complex exponential

$$X(k,l) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} x(m,n) e_{(-k)(-l)N}(m,n) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} x(m,n) e^{*}_{klN}(m,n) \tag{17}$$

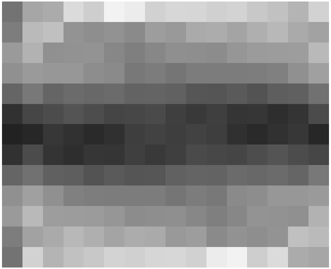
- From definition of inner product we have $\Rightarrow X(k,l) = \langle x, e_{klN} \rangle$
- DFT element $X(k,l)$ $\Rightarrow$ Inner product of $x(m,n)$ with $e_{kl,N}(m,n)$
- How much x is an oscillation of horizontal freq. k vertical freq. l
- 2D DFT contains information on rate of change as the 1D DFT

  $\Rightarrow$ But also on the direction of change

2D DFT of an image

- This is 256 × 256 image. We rarely do DFTs of full images
  - ⇒ Separate in 256 patches, each with 16 × 16 pixels

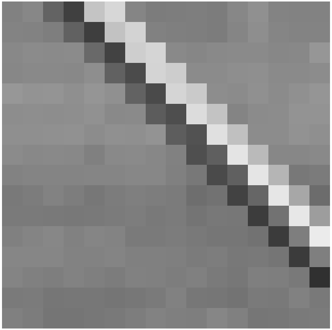
- Image patch on the left, 2D DFT coefficients on the right

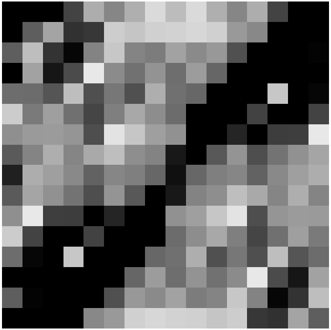


- Signal mostly constant in vertical direction
  - ⇒ Large coefficients concentrated at low vertical frequencies
  - ⇒ Row frequencies more variable due to last column

A patch and its 2D DFT

- Image patch on the left, 2D DFT coefficients on the right

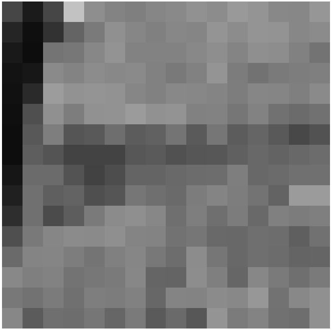


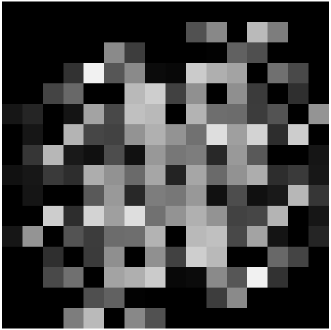


- Signal changes diagonally from top left to bottom right
  - $\Rightarrow$ Large coefficients on diagonal axis from top left to bottom right

A patch and its 2D DFT

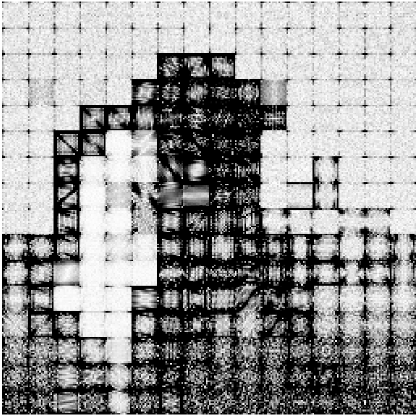
- Image patch on the left, 2D DFT coefficients on the right





- Signal shows variability in many different directions
  - $\Rightarrow$ Large coefficients everywhere esp. when both freqs. are high

- The distribution of the 2D DFT coefficients captures variability
  - $\Rightarrow$ Most coefficients are small on background patches
  - $\Rightarrow$ Many coefficients are large on camera/tripod patches



Periodicity of complex exponentials

- We know that there are only N distinct complex exponentials
- Thus, there are only N^2 distinct 2D complex exponentials
  - $\Rightarrow$ Horizontal frequencies k and $k + N$ are equivalent
  - $\Rightarrow$ Vertical frequencies l and $l + N$ are equivalent
- Canonical sets $[0, N-1] \times [0, N-1]$ and $[-N/2, N/2] \times [-N/2, N/2]$
- 1D complex exponentials are conjugate symmetric. Thus

$$e_{(-k)(-l)N} \equiv e^{*}_{klN} \tag{18}$$

- Flipping sign of both freqs $\equiv$ Conjugation of complex exponential

Periodicity of the 2D DFT

- Consider freqs (k, l) and $(k+N, l)$. DFT at $(k+N, l)$ is

$$X(k+N, l) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m, n) e^{*}_{(k+N)lN}(m, n) \qquad (19)$$

- Complex exponentials of freqs.(k, l) and $(k+N, l)$ are equivalent

$$X(k+N, l) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m, n) e^{*}_{klN}(m, n) = X(k, l) \qquad (20)$$

- 2D DFT has period N in horizontal direction.
- Likewise, 2D DFT has period N in vertical direction

- Suffices to look at $N \times N$ adjacent frequencies
- Canonical sets $[0, N-1] \times [0, N-1]$ and $[-N/2, N/2] \times [-N/2, N/2]$

Orthogonality of complex exponentials

Theorem

Complex exponentials with nonequivalent frequencies are orthogonal

$$\langle e_{klN}, e_{\tilde{k}\tilde{l}N} \rangle = \delta(k - \tilde{k})\delta(l - \tilde{l}) \qquad (21)$$

Proof.

- From definitions of inner product and discrete complex exponential

$$\langle e_{klN}, e_{pqN} \rangle = \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} e^{j2\pi(km+ln)/N} \left(e^{j2\pi(\tilde{k}m+\tilde{l}n)/N} \right)^* \qquad (22)$$

- Separate exponents and regroup factors

$$\langle e_{klN}, e_{pqN} \rangle = \frac{1}{N} \sum_{m=0}^{N-1} e^{j2\pi km/N} \left(e^{j2\pi \tilde{k}m/N} \right)^* \frac{1}{N} \sum_{n=0}^{N-1} e^{j2\pi ln/N} \left(e^{j2\pi \tilde{l}n/N} \right)^* \qquad (23)$$

- Inner products of 1D exponentials. First is $\delta(k - \tilde{k})$, second is $\delta(l - \tilde{l})$ □

Two dimensional (2D) iDFT

Signal representation

Images

Two dimensional discrete signals

Two dimensional (2D) discrete Fourier transform (DFT)

Two dimensional (2D) inverse (i) discrete Fourier transform (DFT)

Energy conservation (Parseval's theorem)

Convolution in 2 dimensions

Applications

Discrete Cosine Transform

2D Discrete Cosine Transform

JPEG image compression

Definition of 2D iDFT

- Given a Fourier transform X, the inverse (i)DFT $x = \mathcal{F}^{-1}(X)$ is

$$x(m,n) := \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} X(k,l) e^{j2\pi(km+ln)/N} \qquad (24)$$

- Sum is over horizontal and vertical frequencies dimensions

- Recall that 2D DFT has period N in vertical and horizontal freqs.
- Any summation over $M \times N$ adjacent frequencies works as well. E.g.,

$$x(m,n) = \frac{1}{N} \sum_{k=-N/2+1}^{N/2} \sum_{l=-N/2+1}^{N/2} X(k,l) e^{j2\pi(km+ln)/N} \qquad (25)$$

iDFT is, indeed, the inverse of the DFT

Theorem

The 2D inverse DFT $\tilde{x} = \mathcal{F}^{-1}(X)$ of the 2D DFT $X = \mathcal{F}(x)$ of any given signal x is the original signal x

$$\tilde{x} \equiv \mathcal{F}^{-1}(X) \equiv \mathcal{F}^{-1}(\mathcal{F}(x)) \equiv x \tag{26}$$

- Every 2D signal can be written as a sum of 2D complex exponentials

$$x(m, n) := \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} X(k, l) e^{j2\pi(km+ln)/N} \tag{27}$$

- The coefficient for horizontal frequency k and vertical frequency f is

$$X(k, l) := \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m, n) e^{-j2\pi(km+ln)/N} \tag{28}$$

Proof of DFT inverse formula

Proof.

- To show $\tilde{x} \equiv x$ we prove $\tilde{x}(\tilde{m}, \tilde{n}) = x(\tilde{m}, \tilde{n})$ for all pairs of indices $(\tilde{m}, \tilde{n})$
- From the definition of the 2D iDFT of X we write the value $\tilde{x}(\tilde{m}, \tilde{n})$ as

$$\tilde{x}(\tilde{m}, \tilde{n}) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} X(k, l) e^{j2\pi(k\tilde{m}+l\tilde{n})/N} \tag{29}$$

- From the definition of the 2D DFT of x we write the DFT value $X(k, l)$ as

$$X(k, l) := \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m, n) e^{-j2\pi(km+ln)/N} \tag{30}$$

- Substituting expression for $X(k, l)$ into expression for $\tilde{x}(\tilde{n}, \tilde{m})$ yields

$$\tilde{x}(\tilde{m}, \tilde{n}) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \left[\frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m, n) e^{-j2\pi(km+ln)/N} \right] e^{j2\pi(k\tilde{m}+l\tilde{n})/N} \tag{31}$$

Proof of DFT inverse formula

Proof.

- Exchange summation order, pull out $x(m,n)$, and distribute $1/N$ factors

$$\tilde{x}(\tilde{m},\tilde{n}) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} x(m,n)\left[\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} \frac{1}{N} e^{-j2\pi(km+ln)/N} \frac{1}{N} e^{j2\pi(k\tilde{m}+l\tilde{n})/N}\right] \tag{29}$$

- Can pull $x(m,n)$ out because it doesn't depend neither on k nor on l

- Innermost sum is inner product between $e_{\tilde{m}\tilde{n}N}$ and e_{mnN}. Orthonormality:

$$\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} \frac{1}{N} e^{-j2\pi(km+ln)/N} \frac{1}{N} e^{j2\pi(k\tilde{m}+l\tilde{n})/N} = \langle e_{\tilde{m}\tilde{n}N}, e_{mnN}\rangle = \delta(\tilde{m}-m)\delta(\tilde{n}-n) \tag{30}$$

- Reducing to $\Rightarrow \tilde{x}(\tilde{m},\tilde{n}) = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x(m,n)\delta(\tilde{n}-n)\delta(\tilde{m}-m) = x(m,n)$

- Last equation true because only term $m=\tilde{m}, n=\tilde{n}$ is not null in the sum

□

The 2D DFT sense

- Can write image x as sum of deltas modulated by individual pixels

$$x(m,n) := \frac{1}{N}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} x(m,n)\delta(k-m, l-n) \tag{31}$$

- Also write as sum of oscillations modulated by 2D DFT coefficients

$$x(m,n) := \frac{1}{N}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} X(k,l) e^{j2\pi(km+ln)/N} \tag{32}$$

- These are mathematically analogous expressions.
- We can see (literally) pixels, but we can't see 2D DFT coefficients
- Easier to operate on the image, when written as sum of oscillations

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

- Reconstruction when using frequencies $-1 \leq k, l \leq 1$. Not too good

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

- Reconstruction when using frequencies $-2 \leq k, l \leq 2$. Not bad

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

- Using frequencies $-4 \leq k, l \leq 4$. Quite good, except for border effect

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

- Freqs. $-7 \leq k, l \leq 7$. Border effect still present. Will solve later (DCT)

Energy conservation (Parseval's theorem)

Properties of the 2D DFT

- All properties of 1D DFTs have corresponding versions for 2D DFTs
  $\Rightarrow$ Linearity, conjugate symmetry, modulation $\Leftrightarrow$ shift

- We will cover energy conservation (to study compression)

$$\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} |x(m,n)|^2 = \|x\|^2 = \|X\|^2 = \sum_{k=0}^{N-1}\sum_{l=0}^{N-1} |X(k,l)|^2 \qquad (33)$$

- Will also cover the 2D convolution theorem (to study linear filtering)

$$y = x * h \quad \Longleftrightarrow \quad Y = HX \qquad (34)$$

- Which will require defining the 2D convolution operation $x * h$

Energy conservation

Theorem (Parseval)

The energies of a signal x *and its 2D DFT* $X = \mathcal{F}(x)$ *are the same, i.e.,*

$$\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}|x(m,n)|^2 = \|x\|^2 = \|X\|^2 = \sum_{k=0}^{N-1}\sum_{l=0}^{N-1}|X(k,l)|^2 \qquad (35)$$

- Since 2D DFT is periodic, any set of adjacent freqs. would do. E.g.,

$$\|X\|^2 = \sum_{k=0}^{M-1}\sum_{l=0}^{N-1}|X(k,l)|^2 = \sum_{k=-M/2+1}^{M/2}\;\sum_{l=-N/2+1}^{N/2}|X(k,l)|^2 \qquad (36)$$

- From now on, we write $\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}(\cdot) = \sum_{m,n}(\cdot)$ and $\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}(\cdot) = \sum_{k,l}(\cdot)$
- To simplify notation. We would otherwise write up to six sums

Proof of Parseval's Theorem

Proof.

- The energy of the 2D DFT X is $\Rightarrow \|X\|^2 = \sum_{k,l} X(k,l)X^*(k,l)$

- The 2D DFT of x is $\Rightarrow X(k,l) := \frac{1}{N} \sum_{m,n} x(m,n) e^{-j2\pi(km+ln)/N}$

- Substitute expression for $X(k,l)$ into one for $\|X\|^2$ (observe conjugation)

$$\|X\|^2 = \sum_{k,l} \left[\left[\frac{1}{N} \sum_{m,n} x(m,n) e^{-j2\pi(km+ln)/N} \right] \left[\frac{1}{N} \sum_{\tilde{m},\tilde{n}} x^*(\tilde{m},\tilde{n}) e^{+j2\pi(k\tilde{m}+il\tilde{n})/N} \right] \right] \tag{37}$$

- Distribute product, exchange sum order, pull $x(m,n)$ and $x^*(\tilde{m},\tilde{n})$ out

$$\|X\|^2 = \sum_{m,n} \sum_{\tilde{m},\tilde{n}} x(m,n) x^*(\tilde{m},\tilde{n}) \left[\sum_{k,l} \frac{1}{N} e^{-j2\pi(km+ln)/N} \frac{1}{N} e^{+j2\pi(k\tilde{m}+l\tilde{n})/N} \right] \tag{38}$$

- Can pull out because $x(m,n)$ and $x^*(\tilde{m},\tilde{n})$ don't depend on (k,l)

Proof of Parseval's Theorem

Proof.

- Innermost sum is inner product between $e_{\tilde{m}\tilde{n}N}$ and e_{mnN}. Orthonormality:

$$\sum_{k,l} \frac{1}{N} e^{-j2\pi(km+ln)/N} \frac{1}{N} e^{j2\pi(k\tilde{m}+l\tilde{n})/N} = \langle e_{\tilde{m}\tilde{n}N}, e_{mnN} \rangle = \delta(\tilde{m}-m, \tilde{n}-n) \tag{37}$$

- Substitute $\delta(\tilde{m}-m, \tilde{n}-n)$ for innermost sum to simplify $\|X\|^2$ to

$$= \sum_{m,n} \sum_{\tilde{m},\tilde{n}} x(m,n)x^*(\tilde{m},\tilde{n})\delta(\tilde{m}-m, \tilde{n}-n) = \sum_{m,n} x(m,n)x^*(m,n) \tag{38}$$

- True because only terms with $m=\tilde{m}$ and $n=\tilde{n}$ are not null in the sum

- Conclude by noting that from definition of the energy of x, we have

$$\|X\|^2 = \sum_{m,n} x(m,n)x^*(m,n) = \|x\|^2 \tag{39}$$

□

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

- Energy of approximation error $\equiv$ Energy of 2D DFT coefficients dropped

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

- Energy of reconstruction error $\Rightarrow$ 32% of image's energy (4 coefficients)

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

- Energy of reconstruction error $\Rightarrow$ 9% of image's energy (16 coefficients)

Reconstruction of an image

- Separate in 16×16 patches (256 total). Compute 2D DFT of each patch
- Start with low frequencies and work up to larger frequencies

- Energy of reconstruction error $\Rightarrow$ 2% of image's energy (64 coefficients)

Convolution in 2 dimensions

2D Convolution

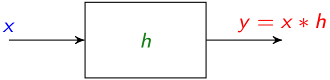
- Given 2D signal x of length $N \times N$ and filter h of length $M \times M$
- Reinterpret filter h as being null for all integers outside its range

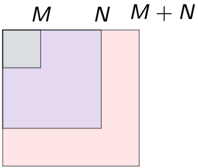
$$h(m, n) = 0, \quad \text{for all } (m, n) \notin [0, M-1] \times [0, M-1] \tag{40}$$

- Convolution of x and h is the $(N+M) \times (N+M)$ signal $y = x * h$

$$y(m, n) = \sum_{p=0}^{N} \sum_{q=0}^{N} x(p, q) h(m-p, n-q) \tag{41}$$

- Hit filter h with input x to generate output y





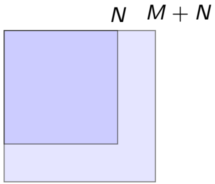
Padded signals

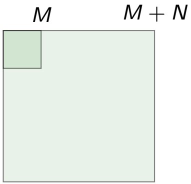
- The padded signal $\bar{x}$ is an $(N+M)\times(N+M)$ signal with

$$\bar{x}(m,n) = x(m,n), \qquad \text{for } (m,n)\in[0,N-1]\times[0,N-1]$$
$$\bar{x}(m,n) = 0, \qquad \text{else}$$

- The padded filter $\bar{h}$ is an $(N+M)\times(N+M)$ signal with

$$\bar{h}(m,n) = h(m,n), \qquad \text{for } (m,n)\in[0,M-1]\times[0,M-1]$$
$$\bar{h}(m,n) = 0, \qquad \text{else}$$





2D convolution theorem

- 2D DFTs of padded signal $\bar{X} = \mathcal{F}(\bar{x})$ and padded filter $\bar{H} = \mathcal{F}(\bar{h})$
- Regular DFT of output signal, $Y = \mathcal{F}(y)$

Theorem (2D Convolution)

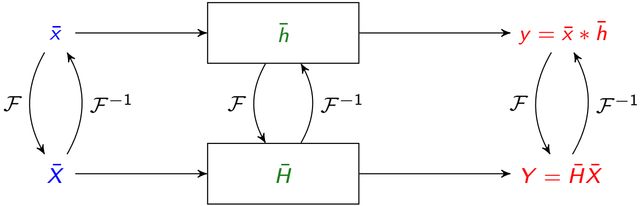
The convolution of padded signals in the space domain is equivalent to the multiplication of their 2D DFTs in the frequency domain

$$y = \bar{x} * \bar{h} \qquad \Longleftrightarrow \qquad Y = \bar{X}\bar{H} \tag{42}$$

- Transformation is obscure in space but crystal clear in frequency

Design in frequency, implement in space

- As we did in 1D, we design in frequency but implement in space



- Convolution doesn't change with padding $\Rightarrow y = \bar{x} * \bar{h} = x * h$
- 2D DFTs do change, but not by much when $M \ll N$

- Instead of padding x and h we crop y to make it $N \times N \Rightarrow \bar{y}$
- Convolution theorem becomes approximate $\Rightarrow \bar{Y} \approx HX$

  $\Rightarrow$ There are differences close to the borders of the image

Applications

Signal representation

Images

Two dimensional discrete signals

Two dimensional (2D) discrete Fourier transform (DFT)

Two dimensional (2D) inverse (i) discrete Fourier transform (DFT)

Energy conservation (Parseval's theorem)

Convolution in 2 dimensions

Applications

Discrete Cosine Transform

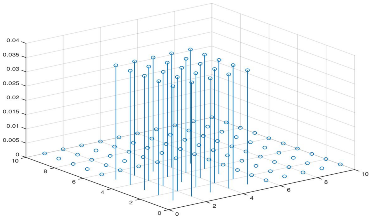
2D Discrete Cosine Transform

JPEG image compression

Averaging Filter

- An averaging filter is one with a square frequency response

$$h(m,n) = \frac{1}{M^2} \sqcap_M (m,n) \qquad (43)$$



- The convolution $y = h * x$ is an average of adjacent pixels

$$y(m,n) = \frac{1}{M^2} \sum_{p=0}^{M-1} \sum_{q=0}^{M-1} x(m-p, n-q) \qquad (44)$$

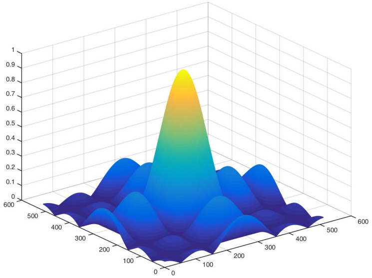
- What effect does an averaging filter have when applied to an image?

- Averaging neighboring pixels has the effect of blurring the image

- What is the counterpart of blurring in the frequency domain?

Average Filter in frequency domain

- The 2D DFT of a 2D square pulse is a 2D sinc $\Rightarrow$ low pass filter



- Blurring entails removal of high frequencies (in all directions)
  $\Rightarrow$ Smoothes edges, which makes image appear out of focus

Image Denoising

- Image is corrupted by white noise $\Rightarrow$ equal power at all frequencies

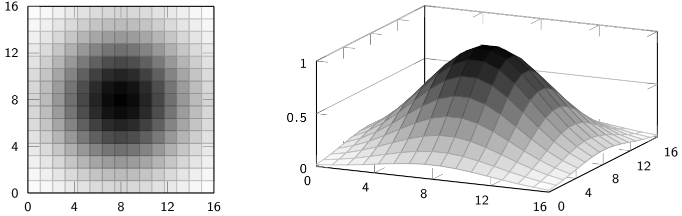
- Can remove noise with averaging filter $\Rightarrow$ Only low frequencies pass

  $\Rightarrow$ Image has low frequencies only. Noise has all frequencies

Gaussian filter

- Or, apply 2D Gaussian filter $\Rightarrow$ 2D Gaussian pulse impulse response

$$h(n,m) = g_{\mu\sigma}(n,m) = \frac{1}{2\pi\sigma^2}\exp\left[-\frac{(m-\mu)^2}{2\sigma^2} - \frac{(n-\mu)^2}{2\sigma^2}\right] \qquad (45)$$

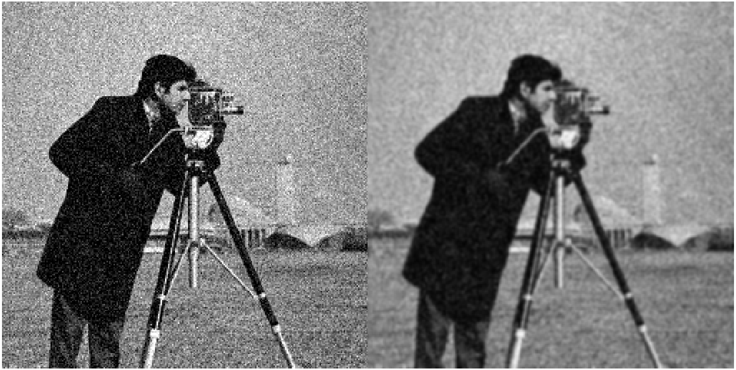


- 2D Gaussian pulse also performs averaging with nearby pixels
- Also low pass $\Rightarrow$ 2D DFT is Gaussian pulse with inverse variance
  $\Rightarrow$ Decrease σ^2 to let more frequencies pass

Gaussian filtering of a noisy image

- Remove noise with a Gaussian filter with variance $\sigma^2 = 1$

Noisy image　　　　Filtered image

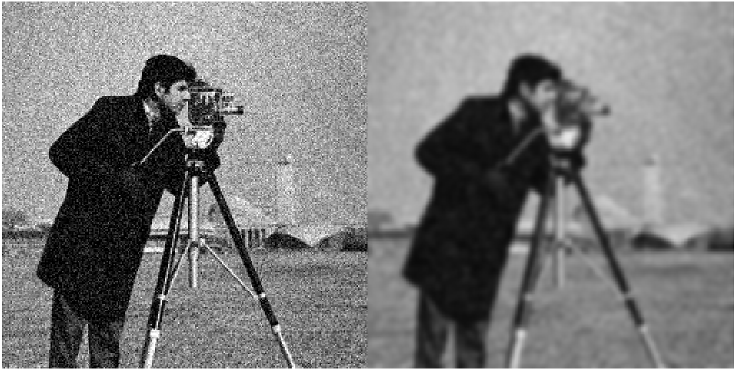


- Some noise is removed. Can remove more by increasing variance σ^2

Low pass filter of noisy image

- Remove noise with a Gaussian filter with variance $\sigma^2 = 4$

Noisy image Filtered image



- More noise removed (good), but also more blurring (not good)

Edge detection

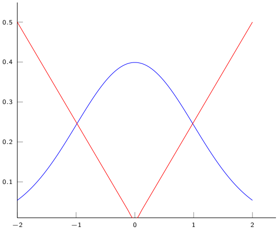
- Detect the edges of an image $\Rightarrow$ Rapid transitions
  $\Rightarrow$ A rapid transition is a high frequency $\Rightarrow$ Use a high pass filter

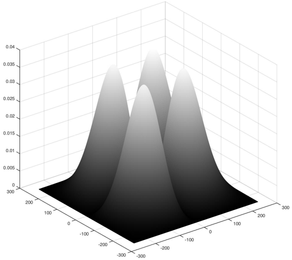
Gaussian Derivative Filter

- Multiply Gaussian filter frequency response by inverted pyramid

$$H(k,l) = G_{\mu\sigma}(k,l)|k+l| \tag{46}$$

- Derivative filter because freq. multiplication is derivation in space





- Very rapid variations are filtered out. They are regarded as noise
- Rapid, but now moderately rapid variations are considered edges

- Now applying this filter to our test image:

- After filter, only high frequencies (edges) remain in image

Image Sharpening

- We want to sharpen an image, e.g., because it's blurry, out of focus
  $\Rightarrow$ We can do that by heightening the edges
- Low frequencies are still important
  $\Rightarrow$ Want to boost high frequencies, as opposed to detecting them
- Add a constant α in frequency to let all frequencies pass

$$H(k,l) = (1-\alpha)\, G_{\mu\sigma}(k,l)|k+l| + \alpha \tag{47}$$

- In time, the constant is a delta $\Rightarrow$ we add the signal and the edges

Image Sharpening

- Increasing sharpening makes borders more defined

Discrete Cosine Transform

Signal representation

Images

Two dimensional discrete signals

Two dimensional (2D) discrete Fourier transform (DFT)

Two dimensional (2D) inverse (i) discrete Fourier transform (DFT)

Energy conservation (Parseval's theorem)

Convolution in 2 dimensions

Applications

Discrete Cosine Transform

2D Discrete Cosine Transform

JPEG image compression

Border effects in image compression

- Patches are well approximated by a subset of 2D DFT coefficients
- Except for borders. And still a problem if we retain most coefficients

- Although didn't mention, also a problem with (1D) DFTs $\Rightarrow$ Why?

The DFT and the iDFT

- Start with real signal $x : [0, N-1] \to \mathbb{R}$. The DFT of signal x is

$$X(k) := \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} \tag{48}$$

- We can recover x with the iDFT transformation defined by

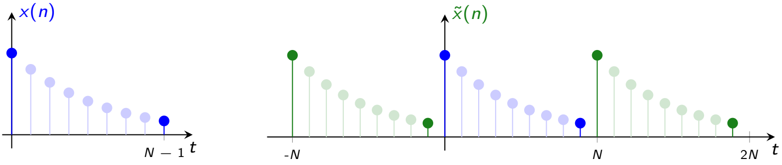
$$\tilde{x}(n) := \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N} \tag{49}$$

- We know that $\tilde{x}(n) = x(n)$ for $n \in [0, N-1]$ (inverse transform)
- But the iDFT is defined for all n
- Signal $\tilde{x}$ is periodic with period N because exponentials $e^{j2\pi kn/N}$ are

$\Rightarrow$ We say that iDFT signal $\tilde{x}$ is a periodic extension of original x

The periodic extension of the iDFT

- First sample $x(0)$ and last sample $x(N-1)$ can be very different
  $\Rightarrow$ Most likely are. Unless signal has some structure, e.g., symmetry
- This is a problem for the periodic extension
  $\Rightarrow$ The value $x(0) = \tilde{x}(N)$ appears next to $x(N-1) = \tilde{x}(N-1)$



- It's tough to approximate a jump/discontinuity $\Rightarrow$ High frequency
- Never mind. We're more than Fourier people. We're fearless transformers

Inverse discrete cosine transform

- Say that we have a transform X so that we can write signal $\tilde{x}$ as

$$\tilde{x}(n) := \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\frac{\pi k(2n+1)}{2N}\right] \tag{50}$$

- Inverse discrete cosine transform (iDCT) of X $\Rightarrow \tilde{x} = \mathcal{C}^{-1}(X)$
- No complex numbers involved. Signals and transforms assumed real

- Haven't said how to find X so that $\tilde{x}(n) = x(n)$ for $n \in [0, N-1]$
- This is done with discrete cosine transform (DCT). We'll see later

- Details are different but this is still x written as a sum of oscillations
  - $\Rightarrow$ Still expect low frequency components to be most significant
  - $\Rightarrow$ But have written cosine in a way that avoids border discontinuities

The iDCT is an even function

- Put a mirror at $N - 1/2$ and compare samples in each direction
- The sample at $n = N - 1$ can be written in terms of iDCT as

$$\tilde{x}(N-1) := \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\frac{\pi k(N-1+1/2)}{N}\right]$$

$$= \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\pi k + \frac{\pi k(-1/2)}{N}\right]$$

- The sample at $n = N$ can be written in terms of iDCT as

$$\tilde{x}(N) := \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\frac{\pi k(N+1/2)}{N}\right]$$

$$= \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\pi k + \frac{\pi k(1/2)}{N}\right]$$

- Since cosines are even, sign is irrelevant. Thus $\Rightarrow \tilde{x}(N-1) = \tilde{x}(N)$

The iDCT is an even function

- Put a mirror at $N-1/2$ and compare samples in each direction
- The sample at $n=N-2$ can be written in terms of iDCT as

$$\tilde{x}(N-2) := \frac{1}{\sqrt{N}}X(0) + \sqrt{\frac{2}{N}}\sum_{k=1}^{N-1}X(k)\cos\left[\frac{\pi k(N-2+1/2)}{N}\right]$$

$$= \frac{1}{\sqrt{N}}X(0) + \sqrt{\frac{2}{N}}\sum_{k=1}^{N-1}X(k)\cos\left[\pi k + \frac{\pi k(-3/2)}{N}\right]$$

- The sample at $n=N+1$ can be written in terms of iDCT as

$$\tilde{x}(N+1) := \frac{1}{\sqrt{N}}X(0) + \sqrt{\frac{2}{N}}\sum_{k=1}^{N-1}X(k)\cos\left[\frac{\pi k(N+1+1/2)}{N}\right]$$

$$= \frac{1}{\sqrt{N}}X(0) + \sqrt{\frac{2}{N}}\sum_{k=1}^{N-1}X(k)\cos\left[\pi k + \frac{\pi k(3/2)}{N}\right]$$

- Since cosines are even, sign is irrelevant. Thus $\Rightarrow \tilde{x}(N-2) = \tilde{x}(N+1)$

The iDCT is an even function

- Put a mirror at $N - 1/2$ and compare samples in each direction
- The sample at $n = N - 3$ can be written in terms of iDCT as

$$\tilde{x}(N-3) := \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\frac{\pi k(N-3+1/2)}{N}\right]$$

$$= \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\pi k + \frac{\pi k(-5/2)}{N}\right]$$

- The sample at $n = N + 2$ can be written in terms of iDCT as

$$\tilde{x}(N+2) := \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\frac{\pi k(N+2+1/2)}{N}\right]$$

$$= \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\pi k + \frac{\pi k(5/2)}{N}\right]$$

- Since cosines are even, sign is irrelevant. Thus $\Rightarrow \tilde{x}(N-3) = \tilde{x}(N+2)$

The iDCT is an even function

- Put a mirror at $N - 1/2$ and compare samples in each direction
- The sample at $n = N - 4$ can be written in terms of iDCT as

$$\tilde{x}(N-4) := \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\frac{\pi k (N - 4 + 1/2)}{N}\right]$$

$$= \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\pi k + \frac{\pi k(-7/2)}{N}\right]$$

- The sample at $n = N + 3$ can be written in terms of iDCT as

$$\tilde{x}(N+3) := \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\frac{\pi k (N + 3 + 1/2)}{N}\right]$$

$$= \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\pi k + \frac{\pi k(7/2)}{N}\right]$$

- Since cosines are even, sign is irrelevant. Thus $\Rightarrow \tilde{x}(N-4) = \tilde{x}(N+3)$

The even extension of the iDCT

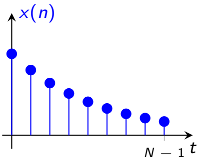
- Formalize argument to prove that the iDCT yields an even extension

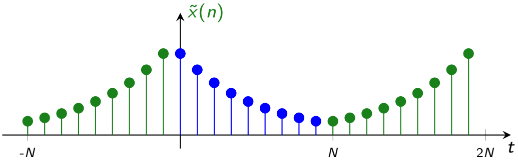
$$\tilde{x}\Big[N + (n-1)\Big] = x\Big[N - n\Big] \qquad (51)$$

- Or, to better visualize the symmetry

$$\tilde{x}\Big[(N - 1/2) + (n - 1/2)\Big] = x\Big[(N - 1/2) - (n - 1/2)\Big] \qquad (52)$$

- Signal x written as sum of oscillations without border effects





The discrete cosine transform (DCT)

- Still have to find out a way of computing the coefficients $X(k)$
- Given a real signal x, the DCT $X = \mathcal{C}(x)$ is the real signal with

$$X(0) := \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi 0(2n+1)}{2N}\right]$$

$$X(k) := \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi k(2n+1)}{2N}\right]$$

- Normalization constants are different for $k = 0$ and $k \in [1, N-1]$
- No complex numbers involved. Signals and transforms are real

DCT basis

- Define the elements of the DCT basis as the signals c_{kN} with

$$c_{0N}(n) := \frac{1}{\sqrt{N}} \qquad c_{kN}(n) := \sqrt{\frac{2}{N}} \cos\left[\frac{\pi k(2n+1)}{2N}\right]$$

- Akin to the DFT basis defined by the N complex exponentials e_{kN}

- With basis defined can write DCT of x as $\Rightarrow X(k) = \langle x, c_{kN} \rangle$

- Inner product implies the usual interpretation
  $\Rightarrow$ X(k) is how much $x(n)$ resembles oscillation of frequency k

iDCT is the inverse of the DCT

Theorem

The iDCT $\tilde{x} = \mathcal{C}^{-1}(X)$ *of the DCT* $X = \mathcal{C}(x)$ *of any given signal* x *is the original signal* x*, i.e.,*

$$\tilde{x} \equiv \mathcal{C}^{-1}(X) \equiv \mathcal{C}^{-1}(\mathcal{C}(x)) \equiv x \tag{53}$$

- Equivalence means $\tilde{x}(n) = x(n)$ for $n \in [0, N-1]$.
  - $\Rightarrow$ Otherwise, inverse transform $\tilde{x}$ is an even extension of original x
- To prove theorem, use DCT definition, iDCT definition, reverse summation order, and invoke orthogonality of the DCT basis.
- Conservation of energy (Parseval's) also holds $\Rightarrow$ orthogonality

2D Discrete Cosine Transform

Signal representation

Images

Two dimensional discrete signals

Two dimensional (2D) discrete Fourier transform (DFT)

Two dimensional (2D) inverse (i) discrete Fourier transform (DFT)

Energy conservation (Parseval's theorem)

Convolution in 2 dimensions

Applications

Discrete Cosine Transform

2D Discrete Cosine Transform

JPEG image compression

Rewriting the 1D DCT

- For 1D signal x we defined the 1D DCT $X = \mathcal{C}(x)$ as

$$X(0) := \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi 0(2n+1)}{2N}\right]$$
$$X(k) := \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi k(2n+1)}{2N}\right]$$

- Define normalization constants $\nu_0 = 1/\sqrt{2}$ and $\nu_k = \sqrt{2}$ for $k \neq 0$

$$X(k) := \frac{\nu_k}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi k(2n+1)}{2N}\right] \qquad (54)$$

- Just a definition to make notation more compact

Two dimensional discrete cosine transform

- Given a two dimensional signal x we define the 2D DCT X as

$$X(k,l) := \frac{\nu_k \nu_l}{N} \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} x(m,n) \cos\left[\frac{\pi k(2m+1)}{2N}\right] \cos\left[\frac{\pi l(2n+1)}{2N}\right] \quad (55)$$

- 2D analogous of the 1D DCT. Or DCT analogous of the 2D DFT

- Can write the double sum as a pair of nested sums

$$X(k,l) := \frac{\nu_k \nu_l}{N} \sum_{n=0}^{N-1} \left[\sum_{m=0}^{N-1} x(m,n) \cos\left[\frac{\pi k(2m+1)}{2N}\right] \right] \cos\left[\frac{\pi l(2n+1)}{2N}\right] \quad (56)$$

- The 2D DCT is the vertical DCT of the horizontals DCTs
- Equivalently, it is also the horizontal DCT of the vertical DCTs

2D DCT as an inner product

- The 2D discrete cosine of horizontal freq. k and vertical freq. l is

$$c_{klN}(n,m) := \frac{c_k}{\sqrt{N}} \cos\left[\frac{\pi k(2m+1)}{2N}\right] \frac{c_l}{\sqrt{N}} \cos\left[\frac{\pi l(2n+1)}{2N}\right] \quad (57)$$

- Use to rewrite 2D DCT as inner product $\Rightarrow X(k,l) = \langle x, c_{klN} \rangle$
- The 2D DCT element $X(k,l)$ is the inner product of x with c_{klN}
- Observe that, similar to the 2D complex exponentials, we can write

$$c_{klN}(n,m) = c_{kN} c_{lN} \quad (58)$$

- Which implies orthonormality of the c_{klN}.

Rewrite the 1D iDCT

- For given DCT X we defined the iDCT as the signal $\tilde{x}$ with values

$$\tilde{x}(n) := \frac{1}{\sqrt{N}} X(0) + \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} X(k) \cos\left[\frac{\pi k(2n+1)}{2N}\right] \tag{59}$$

- Use the same constants, $\nu_0 = 1/\sqrt{2}$ and $\nu_k = 1$ for $k \neq 0$, to write

$$\tilde{x}(n) := \sum_{k=1}^{N-1} \frac{\nu_k}{\sqrt{N}} X(k) \cos\left[\frac{\pi k(2n+1)}{2N}\right] \tag{60}$$

- Just a definition. To avoid writing four separate sums for 2D iDCT

- Given a 2D DCT X we define the 2D iDCT $\tilde{x}$ as

$$\tilde{x}(m,n) := \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} \frac{\nu_k \nu_l}{N} X(k,l) \cos\left[\frac{\pi k(2m+1)}{2N}\right] \cos\left[\frac{\pi l(2n+1)}{2N}\right] \tag{61}$$

- 2D analogous of the 1D DCT. Or DCT analogous of the 2D DFT

- The 2D iDCT is even symmetric (not periodic). In both dimensions

$$\tilde{x}\Big[(N-1/2)+(m-1/2), n\Big] = x\Big[(N-1/2)-(m-1/2), n\Big] \tag{62}$$

$$\tilde{x}\Big[m, (N-1/2)+(n-1/2)\Big] = x\Big[m, (N-1/2)-(n-1/2)\Big] \tag{63}$$

- Thus, we don't have border effects in the reconstruction. Later

iDCT is the inverse of the DCT

Theorem

The iDCT $\tilde{x} = \mathcal{C}^{-1}(X)$ of the DCT $X = \mathcal{C}(x)$ of any given signal x is the original signal x, i.e.,

$$\tilde{x} \equiv \mathcal{C}^{-1}(X) \equiv \mathcal{C}^{-1}(\mathcal{C}(x)) \equiv x \tag{64}$$

- Equivalence means $\tilde{x}(n) = x(n)$ for $n \in [0, N-1]$.
  - $\Rightarrow$ Otherwise, inverse transform $\tilde{x}$ is an even extension of original x
- To prove theorem, use DCT definition, iDCT definition, reverse summation order, and invoke orthogonality of the DCT basis.
- Conservation of energy (Parseval's) also holds $\Rightarrow$ orthogonality

Compression with the 2D DCT and 2D iDCT

- Compute 2D DCT of 16×16 patches. Reconstruct with low frequencies
- The signal is reconstructed with small error and no border effects

Compression with the 2D DCT and 2D iDCT

- Compute 2D DCT of 16×16 patches. Reconstruct with low frequencies
- The signal is reconstructed with small error and no border effects

- Reconstruction when using coefficients $0 \leq k, l \leq 4$. Not too good
- Compression factor 16 and error energy 1.59%

Compression with the 2D DCT and 2D iDCT

- Compute 2D DCT of 16×16 patches. Reconstruct with low frequencies
- The signal is reconstructed with small error and no border effects

- Reconstruction when using coefficients $0 \leq k, l \leq 6$. Quite good
- Compression factor 7.1 and error energy 0.81%

Compression with the 2D DCT and 2D iDCT

- Compute 2D DCT of 16×16 patches. Reconstruct with low frequencies
- The signal is reconstructed with small error and no border effects

- Reconstruction when using coefficients $0 \leq k, l \leq 8$. Excellent
- Compression factor 4 and error energy 0.46%

Compression with the 2D DCT and 2D iDCT

- Compute 2D DCT of 16×16 patches. Reconstruct with low frequencies
- The signal is reconstructed with small error and no border effects

- Reconstruction when using coefficients $0 \leq k, l \leq 10$. Flawless
- Compression factor 2.56 and error energy 0.26%

JPEG image compression

JPEG image compression

- Start with a color image $\Rightarrow$ three color channels x_R, x_B, x_G
  - $\Rightarrow$ Each pixel is represented by 8 bits
  - $\Rightarrow$ Values are integers in [0,255], or, equivalently [-127,128]
- Transform into luminance y and chrominance y_R and y_B
- Eye more sensitive to luminance. Sample chrominances every 2 pixels
- Work with luminance and chrominance separately.
- Separate each channel in 8×8 patches $\Rightarrow$ 64 pixels per patch
- For each patch x, compute the corresponding DCT X
  - $\Rightarrow$ Keep coefficients associated with largest frequency components
- Low frequencies more important but high frequencies not irrelevant
  - $\Rightarrow$ Introduce importance quantization

Importance quantization

- For each frequency pair k, l, define the importance coefficient $Q(k, l)$
- Encode each DCT frequent component as

$$\hat{X}(k, l) = \text{round}\left(\frac{X(k, l)}{Q(k, l)}\right) \qquad (65)$$

- If $Q(k, l) \approx 1$ there is little change $\Rightarrow \hat{X}(k, l) \approx X(k, l)$
- If $Q(k, l)$ is large we reduce the range of $\hat{X}(k, l)$
- Numbers with smaller range can be encoded with less bits

  $\Rightarrow$ Assign relatively small $Q(k, l)$ to low frequencies

  $\Rightarrow$ Assign relatively large $Q(k, l)$ to high frequencies

Importance matrix

- The importance coefficients $Q(k, l)$ form the importance matrix $\mathbf{Q}$

  $\Rightarrow$ Up to 20. Up to 50. Up to 90. More than 90.

$$\mathbf{Q} = \begin{pmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 36 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{pmatrix}$$

- Instead of top left square, we assign importance to top left triangle
- Slight asymmetry $\Rightarrow$ More importance to horizontal frequencies
- All frequency components encoded to some extent

  $\Rightarrow$ High frequency components encoded only when they are large