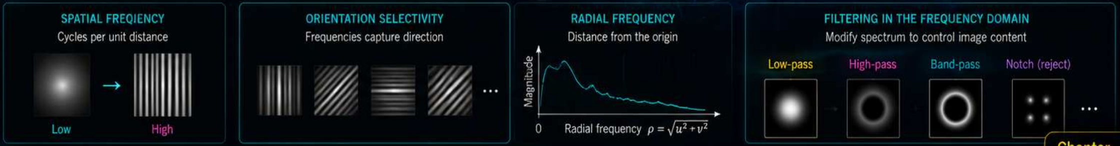
The 2-Dimensional Discrete Fourier Transform and Applications

Frequency-domain representations, spectra of images, and filtering in two dimensions

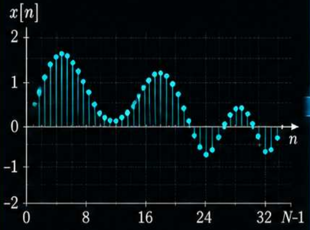


Chapter

Review of the 1-D Discrete Fourier Transform

From a sampled signal to its frequency-domain representation

1-D DISCRETE-TIME SIGNAL $x[n]$



Example: sum of two sinusoids ($N = 32$ samples)

THE 1-D DISCRETE FOURIER TRANSFORM (DFT)

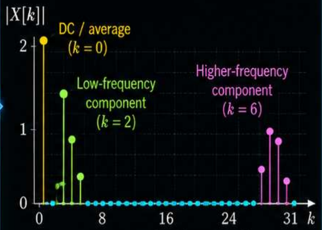
Forward DFT (analysis)

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi k n}{N}}, \qquad k = 0, 1, \ldots, N-1$$

Inverse DFT (synthesis) — *preview*

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k]\, e^{j\frac{2\pi k n}{N}}, \qquad n = 0, 1, \ldots, N-1$$

MAGNITUDE SPECTRUM $|X[k]|$



Frequency bin k $(0 \ldots N-1)$

NOTATION

- $x[n]$ is the N-point discrete-time signal
- $X[k]$ is the complex DFT coefficient at frequency bin k
- N is the number of samples
- $e^{\pm j\frac{2\pi kn/N}{N}}$ are orthogonal complex exponential basis functions

INTUITION: THE SIGNAL IS A WEIGHTED SUM OF COMPLEX EXPONENTIALS

$$x[n] = X[0] \cdot e^{j\frac{2\pi\cdot 0\cdot n}{N}} + X[2] \cdot e^{j\frac{2\pi\cdot 2\cdot n}{N}} + \cdots + X[6]\, e^{j\frac{2\pi\cdot 6\cdot n}{N}} + \cdots$$

X[0] (DC / average)

constant

X[2] (low frequency)

slow oscillation

X[6] (higher frequency)

faster oscillation

Each term has a magnitude $|X[k]|$ and a phase $\angle X[k]$ that sets its weight.

FREQUENCY BINS AND NORMALIZED FREQUENCY

Each index k corresponds to a normalized frequency:

$$f_k = \frac{k}{N} \text{ cycles/sample}, \qquad k = 0, 1, \ldots, N-1$$

- $f = 0$ $(k = 0)$ → DC / average
- $0 < f < \frac{1}{2}$ → positive frequencies
- $f = \frac{1}{2}$ $(k = \frac{N}{2}, N \text{ even})$ → Nyquist
- $\frac{1}{2} < f < 1$ → negative frequencies (implicit for real $x[n]$)

Takeaway: The 1-D DFT converts a sampled signal into complex frequency coefficients $X[k]$ by projecting $x[n]$ onto an orthogonal set of complex exponentials; the inverse DFT combines those coefficients to reconstruct the signal exactly.

Frequency Bins, Magnitude, and Phase

How to interpret the DFT coefficients $X[k]$

DFT coefficient in polar (magnitude–phase) form

$$X[k] = |X[k]|\, e^{j\phi[k]}$$

$|X[k]|$ tells how much of frequency bin k is present.

$\phi[k]$ tells the phase / shift of that component.

DFT (forward) for context

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi kn}{N}}$$

$$k = 0, 1, \cdots, N-1$$

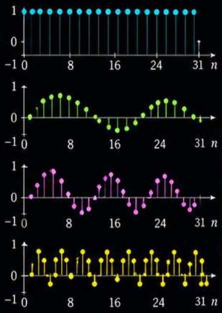
Basis signals (examples of frequency bins)

$k = 0$ DC (constant sequence)

$k = 1$ very low frequency

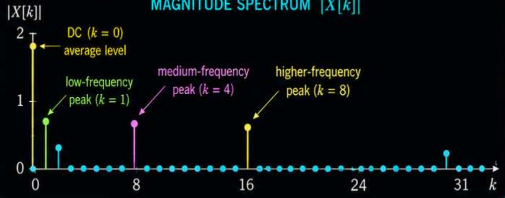
$k = 4$ medium frequency

$k = 8$ higher frequency

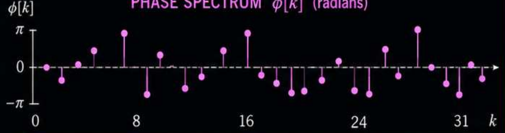


As k increases, the oscillation rate increases.

MAGNITUDE SPECTRUM $|X[k]|$



PHASE SPECTRUM $\phi[k]$ (radians)



Phase indicates the alignment (time shift) of each frequency component.

WHAT FREQUENCY BINS TELL US

$k = 0$	Average / DC term (overall level of the signal)
Small k	Slow variations / smooth trends
Larger k	Rapid changes / fine detail
For real $x[n]$	Bins k and $N-k$ form conjugate pairs: $X[N-k] = X^*[k]$

NORMALIZED FREQUENCY

Frequency bin k corresponds to

$$\text{normalized frequency} = \frac{k}{N} \text{ cycles / sample}$$

0 0.5 (Nyquist) 1 cycles / sample

Takeaway: $|X[k]|$ tells how strong each frequency bin k is in the signal; $\phi[k]$ tells its phase (shift).
Together across all bins, they fully describe the signal in the frequency domain.

The Inverse DFT and Perfect Reconstruction

How the frequency-domain coefficients rebuild the signal exactly

1) THE INVERSE DFT

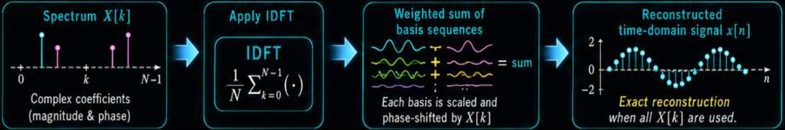
$$\underbrace{x[n]}_{\text{time-domain sample}} = \underbrace{\frac{1}{N}}_{\text{normalization factor}} \sum_{k=0}^{N-1} \underbrace{X[k]}_{\text{DFT coefficient (magnitude \& phase)}} \underbrace{e^{j\frac{2\pi k n}{N}}}_{k\text{-th complex exponential basis sequence}}, \qquad \underbrace{n = 0, 1, \ldots, N-1}_{\text{reconstructed samples}}$$

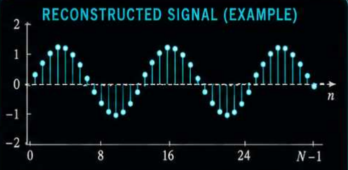
2) SYNTHESIS INTERPRETATION

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} \underbrace{|X[k]|}_{\text{magnitude = weight}} \underbrace{e^{j\phi[k]}}_{\text{phase = shift}} \underbrace{e^{j\frac{2\pi k n}{N}}}_{k\text{-th complex exponential basis sequence}}$$

WHY THE INVERSE DFT WORKS

- N The inverse DFT adds all N orthogonal basis sequences to synthesize the signal.
- Each coefficient $X[k]$ scales (by $|X[k]|$) and phase-shifts (by $\phi[k]$) its basis sequence $e^{j2\pi kn/N}$.
- Keeping both magnitude and phase for every k yields **perfect reconstruction:** the IDFT recovers $x[n]$ exactly.



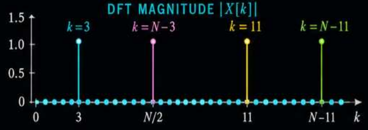


IDFT reproduces the original samples exactly.

MINI EXAMPLE: TWO SINUSOIDS

Signal composed of two sinusoids:

$$x[n] = \cos\left(\frac{2\pi \cdot 3n}{N}\right) + 0.7\cos\left(\frac{2\pi \cdot 11n}{N} + \frac{\pi}{6}\right)$$



Only four bins are (nearly) nonzero:

- $k = 3$ and $k = N-3$ (cosine pair)
- $k = 11$ and $k = N-11$ (cosine pair)

All other bins ≈ 0.

IDFT recombines these components exactly.

Perfect reconstruction of $x[n]$.

Why the 1-D DFT Matters

Intuition, interpretation, and practical uses

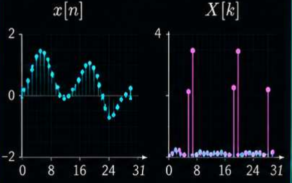
$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi kn}{N}} \qquad x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]\, e^{j\frac{2\pi kn}{N}}$$

- $x[n]$: N-point discrete-time signal
- $X[k]$: DFT coefficient at bin k
- $k, n = 0, 1, \ldots, N-1$
- N: number of samples / bins
- Basis functions: $e^{-j2\pi kn/N}$ (orthogonal)

1 INTERPRETATION

The DFT rewrites a sampled signal as orthogonal discrete frequencies.

Samples (time domain) → DFT coefficients (frequency bins)



2 WHAT STUDENTS SHOULD REMEMBER

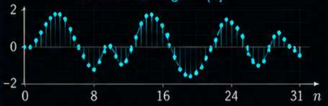
- Magnitude $|X[k]|$ tells how much (strength of each component).
- Phase $\phi[k] = \angle X[k]$ tells where / alignment (shift of each component).
- k indexes normalized frequency (k/N cycles per sample).
- Real-valued signals have conjugate symmetry: $X[N-k] = X^*[k]$

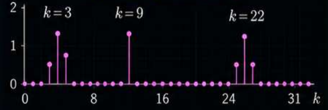
3 APPLICATIONS

- **Spectral analysis** identify dominant frequencies
- **Filtering** keep or remove frequency bins
- **Compression** keep the most important bins
- **Image & audio processing** transform, enhance, denoise
- **Convolution via multiplication** in the frequency domain

4 MINI WORKED EXAMPLE

Signal: sum of three sinusoids ($N = 32$)





Peaks at $k = 3, 9, 22$ match the three sinusoids.

Takeaway: The 1-D DFT is a change of basis from samples to frequency bins; the inverse DFT recombines those bins to reconstruct the signal exactly.

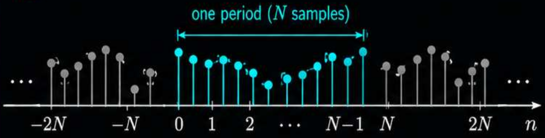
The 1-D DFT as a Periodic Representation

An N-point DFT treats the finite record as one period of a periodic sequence

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi}{N}kn}, \quad k = 0, 1, \ldots, N-1$$

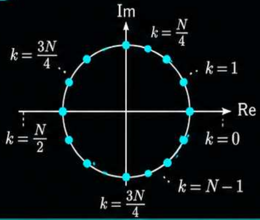
The DFT analyzes one period of a length-N sequence.

1 TIME-DOMAIN PERIODIC EXTENSION



$$x_p[n] = x[(n \bmod N)] \qquad x_p[n+N] = x_p[n]$$

2 FREQUENCY BINS AS EQUALLY SPACED SAMPLES



$$\omega_k = \frac{2\pi k}{N}$$

$$X[k] = X(e^{j\omega})\Big|_{\omega=\omega_k}$$

1 ONE PERIOD IN TIME

the N samples are interpreted as a repeating sequence.

2 DISCRETE FREQUENCY SAMPLES

the spectrum is sampled at N equally spaced frequencies.

3 WRAP-AROUND INTUITION

modulo-N indexing is built into the DFT.

Takeaway: The N-point DFT represents one period of a periodic discrete-time signal and samples its spectrum at N equally spaced frequencies.

Frequency-Domain Periodicity and Circular Convolution

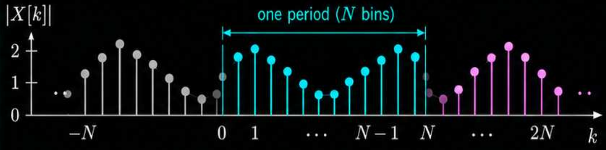
The DFT spectrum repeats every N bins, and this periodicity leads naturally to circular convolution

$$X[k+N] = X[k]$$

The spectrum of an N-point DFT is periodic in k with period N.

1 SPECTRUM REPEATS EVERY N BINS

The DFT spectrum is periodic in the frequency index k with period N.



Why $X[k+N] = X[k]$?

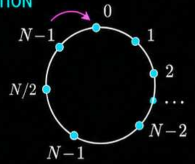
From the DFT:

$$e^{-j\frac{2\pi}{N}(k+N)n} = e^{-j\frac{2\pi}{N}kn}\, e^{-j2\pi n} = e^{-j\frac{2\pi}{N}kn}$$

since $e^{-j2\pi n} = 1$ for all integer n.

2 WRAP-AROUND INTUITION

Frequency-bin indices wrap modulo N.



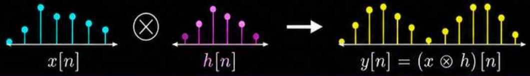
bin indices wrap modulo N

3 WHY IT MATTERS: CIRCULAR CONVOLUTION

Time-domain convolution implied by DFT periodicity is circular.

$$y[n] = (x \otimes h)[n] = \sum_{m=0}^{N-1} x[m]\, h[(n-m) \bmod N]$$

$$Y[k] = X[k]\, H[k], \qquad k = 0, 1, \ldots, N-1$$



Takeaway: In the DFT, frequency bins repeat every N, and the implied wrap-around structure leads to circular convolution.

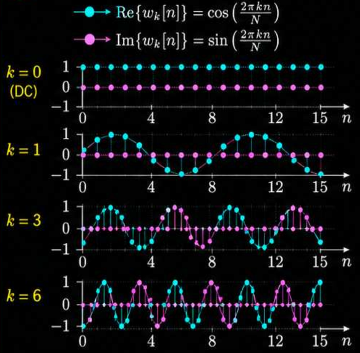
Orthogonality of the DFT Kernels

The complex exponential kernels form an orthogonal basis for length-N signals

$$w_k[n] = e^{j\frac{2\pi}{N}kn}, \qquad k = 0, 1, \ldots, N-1$$
$$w_k^*[n] = e^{-j\frac{2\pi}{N}kn}$$

Analysis uses the conjugate kernel; synthesis uses the forward kernel.

① EXAMPLES OF KERNELS ($N = 16$)



② ORTHOGONALITY PROOF

Inner product of two length-N sequences:

$$\langle a, b \rangle = \sum_{n=0}^{N-1} a[n]\, b^*[n]$$

Inner product of DFT kernels:

$$\langle w_k, w_\ell \rangle = \sum_{n=0}^{N-1} e^{j\frac{2\pi}{N}(k-\ell)n}$$
$$= \begin{cases} N, & \text{if } k = \ell \\ 0, & \text{if } k \neq \ell \end{cases}$$

- If $k = \ell$: each term is $e^{j0} = 1$, so the sum is N.
- If $k \neq \ell$: it is a geometric series whose terms lie on the unit circle and sum to 0.

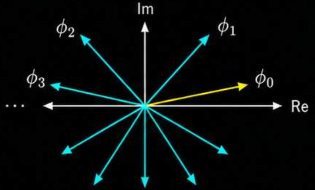
③ NORMALIZED ORTHONORMAL BASIS

Normalize the kernels:

$$\phi_k[n] = \frac{1}{\sqrt{N}}\, e^{j\frac{2\pi}{N}kn}$$

They are orthonormal:

$$\langle \phi_k, \phi_\ell \rangle = \delta[k-\ell]$$



Takeaway: The DFT kernels are orthogonal; after normalization they form an orthonormal basis for the length-N signal space.

Projection onto DFT Kernels and Reconstruction

Each DFT coefficient is an inner-product projection, and the inverse DFT recombines those coordinates

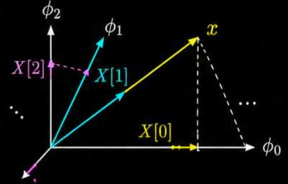
$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi}{N}kn} = \langle x, w_k \rangle$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k]\; w_k[n]$$

Analysis = projection; synthesis = weighted recombination.

1 PROJECTION INTERPRETATION

The DFT kernels $\{\phi_k\}$ form an orthonormal basis.



DFT coefficient = inner product (projection).

$$\langle x, \phi_k \rangle = \frac{X[k]}{\sqrt{N}}$$

$\{\phi_k\}_{k=0}^{N-1}$ orthonormal: $\langle \phi_k, \phi_\ell \rangle = \delta[k-\ell]$

Each $X[k]$ is the coordinate of x along ϕ_k.

2 WHAT EACH COEFFICIENT MEANS

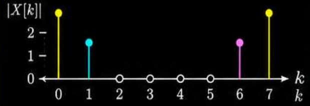
Each $X[k]$ is complex.

$$X[k] = |X[k]|\, e^{j\phi[k]}$$

Magnitude $|X[k]|$ = length / weight

Phase $\phi[k]$ = shift / rotation

Example spectrum ($N = 8$):

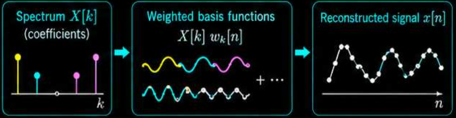


Nonzero bins at $k = 0, 1, 6, 7$.

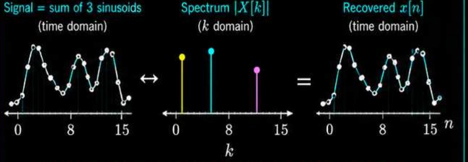
Others are zero.

3 INVERSE DFT RECONSTRUCTION

Inverse DFT adds scaled kernels to rebuild $x[n]$.



Mini-example ($N = 16$):



Keeping all coefficients gives exact reconstruction.

Takeaway: The DFT computes the coordinates of the signal in an orthogonal kernel basis; the inverse DFT adds those weighted basis functions to recover the signal exactly.

DFT Property: Circular Time Shift

Shifting a discrete signal in time leaves the magnitude unchanged and adds a linear phase term

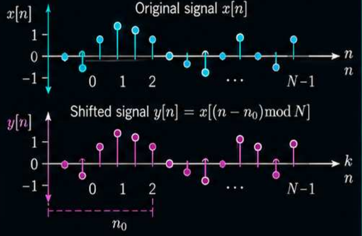
Let
$$y[n] = x[(n - n_0) \mod N]$$

Then the N-point DFT satisfies
$$Y[k] = X[k]\, e^{-j\frac{2\pi k n_0}{N}}, \qquad k = 0, 1, \ldots, N-1$$

$$|Y[k]| = |X[k]| \qquad\qquad \angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$$

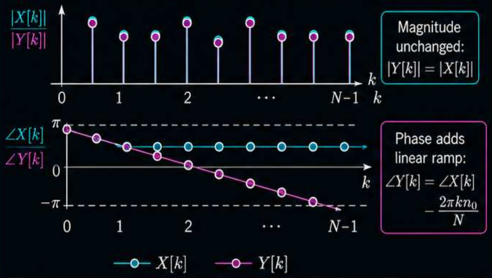
A circular shift in time does not change the DFT magnitude; it only adds a linear phase ramp.

TIME DOMAIN: CIRCULAR SHIFT



Time domain: samples move by n_0 positions (circularly).

FREQUENCY DOMAIN: DFT RELATIONSHIP



WHY THIS MATTERS

Delay / alignment analysis
Time shifts appear as linear phase ramps.

Image registration and cyclic shifts
Cyclic shifts in images preserve magnitude.

Phase carries position information
Magnitude shows what, phase shows where.

Takeaway: Circular time shift $\leftrightarrow$ multiplication by $e^{-j\frac{2\pi k n_0}{N}}$ in frequency; magnitude stays the same, phase changes linearly.

DFT Property: Time Dilation / Expansion

Expanding a discrete signal by inserting zeros compresses the spectrum and creates repeated spectral images

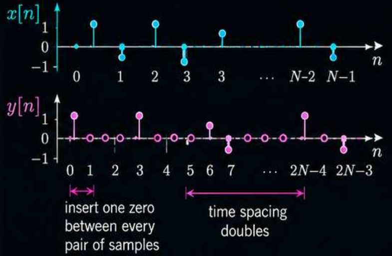
Let $y[n] = x[n/L]$ for $n = 0, \pm L, \pm 2L, \ldots$ and $y[n] = 0$ otherwise.

$$Y[k] = \sum_{n=0}^{LN-1} y[n]\, e^{-j\frac{2\pi kn}{LN}} = \sum_{m=0}^{N-1} x[m]\, e^{-j\frac{2\pi km}{N}} = X[k \bmod N] \quad \text{for } k = 0, 1, \ldots, LN-1.$$

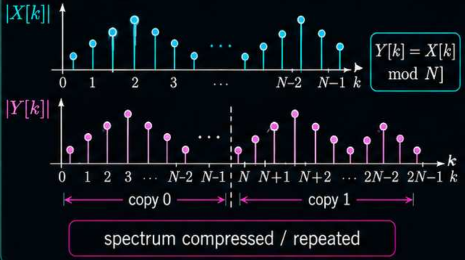
Intuitively (DTFT relation): $Y(e^{j\omega}) = X(e^{j\omega L})$

Time expansion by L inserts zeros between samples, stretches the signal in time, and compresses the spectrum while repeating it L times over 0 to 2π.

TIME DOMAIN: EXPANSION BY $L = 2$



FREQUENCY DOMAIN: SPECTRAL IMAGES



WHAT TO REMEMBER

- Expansion in time → compression in frequency
- Inserted zeros do NOT add new information
- In the discrete periodic spectrum, copies repeat across the frequency axis

Takeaway: Dilation (time expansion) by L spreads the samples apart in time and causes the DFT spectrum to repeat L times across the frequency axis.

DFT Property: Zero Padding

Appending zeros in time does not add information; it samples the same spectrum more densely

Define zero padding of length $M > N$:

$$x_p[n] = x[n], \quad 0 \le n \le N-1; \qquad x_p[n] = 0, \quad N \le n \le M-1, \quad M > N$$

The M-point DFT of the padded signal:

$$X_p[k] = \sum_{n=0}^{M-1} x_p[n]\, e^{-j\frac{2\pi k n}{M}} = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi k n}{M}}, \qquad k = 0, 1, \ldots, M-1$$

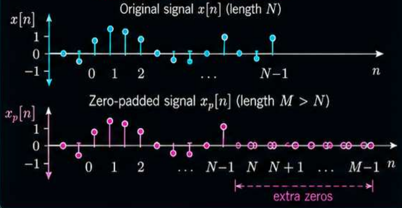
Interpretation (samples of the same DTFT):

$$X_p[k] = X(e^{j\omega})\Big|_{\omega = \frac{2\pi k}{M}}$$

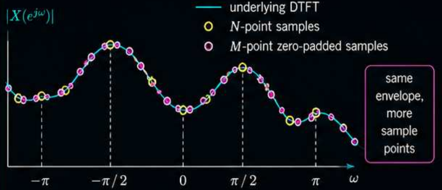
Zero padding keeps the underlying DTFT the same, but evaluates it on a denser frequency grid.

Important: zero padding improves visual spectral resolution / interpolation, but it does not improve true frequency resolution or create new information.

TIME DOMAIN: APPEND ZEROS



FREQUENCY DOMAIN: DENSER SAMPLES OF THE SAME DTFT



WHY ZERO PADDING IS USEFUL

- Makes spectral peaks easier to visualize
- Helps estimate peak location / interpolate between bins
- Does not change the physical bandwidth or add information

Takeaway: Zero padding in time gives a denser-looking DFT in frequency because it samples the same DTFT on a finer grid—useful for display and peak estimation, but not new information.

DFT Property: Circular Time Shift

Shifting a discrete signal in time leaves the magnitude unchanged and adds a linear phase term

Let $$y[n] = x[(n - n_0) \mod N]$$

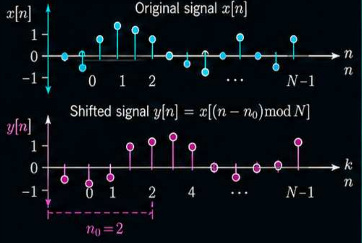
Then the N-point DFT satisfies

$$Y[k] = X[k]\, e^{-j\frac{2\pi k n_0}{N}}, \qquad k = 0, 1, \ldots, N-1$$

$$|Y[k]| = |X[k]| \qquad\qquad \angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$$

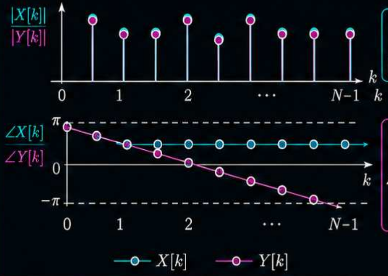
A circular shift in time does not change the DFT magnitude; it only adds a linear phase ramp.

TIME DOMAIN: CIRCULAR SHIFT



Time domain: samples move by n_0 positions (circularly).

FREQUENCY DOMAIN: DFT RELATIONSHIP



Magnitude unchanged: $|Y[k]| = |X[k]|$

Phase adds linear ramp: $\angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$

WHY THIS MATTERS

Delay / alignment analysis
Time shifts appear as linear phase ramps.

Image registration and cyclic shifts
Cyclic shifts in images preserve magnitude.

Phase carries position information
Magnitude shows what, phase shows where.

Takeaway: Circular time shift $\leftrightarrow$ multiplication by $e^{-j\frac{2\pi k n_0}{N}}$ in frequency; magnitude stays the same, phase changes linearly.

1-D DFT as a Matrix–Vector Operation

Forward transform: represent the discrete signal as a column vector and compute its spectrum by matrix multiplication

$$\mathbf{X} = \mathbf{F}_N\,\mathbf{x}$$

$$\mathbf{x} = [x[0] \quad x[1] \quad x[2] \quad \cdots \quad x[N-1]]^T$$

$$\mathbf{X} = [X[0] \quad X[1] \quad X[2] \quad \cdots \quad X[N-1]]^T$$

SIGNAL VECTOR

time-domain samples

$$\mathbf{x} = \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ \vdots \\ x[N-1] \end{bmatrix} \in \mathbb{C}^{N\times 1}$$

DFT MATRIX (ANALYSIS MATRIX)

$$W_N = e^{-j\,2\pi/N}$$

$$\mathbf{F}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N & W_N^2 & \cdots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix}$$

$$[\mathbf{F}_N]_{k,n} = e^{-j\,2\pi\,kn/N}, \qquad k,n = 0,1,\ldots,N-1$$

CONCEPTUAL FLOW

Signal vector | Multiply by DFT matrix | Spectrum vector

$$\mathbf{x} = \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ \vdots \\ x[N-1] \end{bmatrix} \in \mathbb{C}^{N\times 1} \;\rightarrow\; \boxed{\mathbf{F}_N \; (N\times N)} \;\rightarrow\; \mathbf{X} = \begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ \vdots \\ X[N-1] \end{bmatrix} \in \mathbb{C}^{N\times 1}$$

COMPONENTWISE FORWARD DFT

$$X[k] = \sum_{n=0}^{N-1} x[n]\,e^{-j\,2\pi\,kn/N}, \qquad k = 0,1,\ldots,N-1$$

Each row of $\mathbf{F}_N$ projects the signal onto one complex exponential kernel.

KEY INTERPRETATION

- Forward DFT = analysis
- Maps time samples to frequency coefficients
- Each $X[k]$ is a complex coefficient for the k-th frequency bin
- The columns/rows are built from powers of the primitive root W_N

Takeaway: The forward 1-D DFT is a linear transformation that maps the sample vector x into the spectrum vector $\mathbf{X}$ through multiplication by the DFT matrix $\mathbf{F}_N$.

Inverse 1-D DFT in Matrix–Vector Form

The synthesis matrix reconstructs the time-domain signal from the spectrum vector

$$\mathbf{x} = \mathbf{F}_N^{-1}\mathbf{X} = \frac{1}{N}\mathbf{F}_N^H\mathbf{X}$$

$\mathbf{F}_N^H$ = conjugate transpose of $\mathbf{F}_N$

$\mathbf{x} = [\,x[0] \quad x[1] \quad \dots \quad x[N-1]\,]^T$

$\mathbf{X} = [\,X[0] \quad X[1] \quad \dots \quad X[N-1]\,]^T$

INVERSE DFT MATRIX

$$\mathbf{F}_N^{-1} = \frac{1}{N}\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \cdots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \cdots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \cdots & W_N^{-(N-1)(N-1)} \end{bmatrix}$$

$$[\mathbf{F}_N^{-1}]_{n,k} = \frac{1}{N}\,e^{+j\frac{2\pi n k}{N}}, \quad n,k = 0,1,\dots,N-1$$

$$W_N = e^{-j\frac{2\pi}{N}}$$

ORTHOGONALITY / UNITARY RELATION

$$\mathbf{F}_N^H\mathbf{F}_N = \mathbf{F}_N\mathbf{F}_N^H = N\mathbf{I}$$

therefore

$$\mathbf{F}_N^{-1} = \frac{1}{N}\mathbf{F}_N^H$$

Rows/columns of $\mathbf{F}_N$ are orthogonal complex exponential basis vectors.

$$\boxed{\mathbf{x}} \xrightarrow{\mathbf{F}_N} \boxed{\mathbf{X}} \xrightarrow{\frac{1}{N}\mathbf{F}_N^H} \boxed{\mathbf{x}}$$

exact reconstruction

COMPONENTWISE INVERSE DFT

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]\,e^{+j\frac{2\pi k n}{N}}, \quad n = 0,1,\dots,N-1$$

- inverse DFT = synthesis
- each coefficient $X[k]$ scales one basis sequence
- all scaled basis sequences are added to rebuild $x[n]$

WHAT RECONSTRUCTION DOES

Spectrum vector $\mathbf{X}$

$$\mathbf{X} = \begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ \vdots \\ X[N-1] \end{bmatrix}$$

Weighted basis functions $X[k]\,w_k[n]$

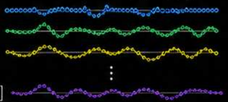
$X[0]\,w_0[n]$ +

$X[1]\,w_1[n]$ +

$X[2]\,w_2[n]$ +

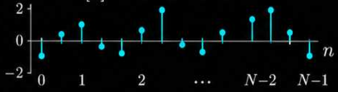
$\vdots$

$X[N-1]\,w_{N-1}[n]$



Reconstructed signal $x[n]$

$x[n]$ = sum of all terms



Takeaway: The inverse 1-D DFT is a linear synthesis operation: multiplying the spectrum vector by $\frac{1}{N}\mathbf{F}_N^H$ exactly reconstructs the time-domain signal.

Why the Inverse Comes from the Hermitian Transpose

Standard DFT versus unitary DFT

Standard DFT (matrix F_N):

$$\mathbf{X} = \mathbf{F}_N\, x \quad \Big| \quad x = \mathbf{F}_N^{-1}\mathbf{X} = \frac{1}{N} F_N^H \mathbf{X} \quad \Big| \quad \mathbf{F}_N^H \mathbf{F}_N = N I$$

Unitary DFT (matrix U_N):

$$U_N = \frac{1}{\sqrt{N}} \mathbf{F}_N \quad \Big| \quad \mathbf{X}_u = U_N\, x \quad \Big| \quad x = U_N^H \mathbf{X}_u = U_N^H U_N\, x \quad \Big| \quad U_N^H U_N = I$$

☆ **Important:** the expression $x = \mathbf{F}^H \mathbf{X} = \mathbf{F}^H \mathbf{F} x$ is exactly true for the **unitary** DFT matrix U.
For the standard DFT matrix F, the correct inverse is $x = \frac{1}{N} F^H \mathbf{X}$.

- **Orthogonal** complex exponential columns: $\mathbf{F}^H F = N I$.
- Hermitian transpose gives **matched synthesis:** $\mathbf{F}^H$ pairs each basis with its conjugate counterpart.
- **Normalization** determines the inverse: unitary $\Rightarrow \mathbf{F}^{-1} = \mathbf{F}^H$; standard $\Rightarrow \mathbf{F}^{-1} = \frac{1}{N} F^H$.

Signal (time/space)
$x \in \mathbb{C}^N$

Spectrum (frequency)
$\mathbf{X} \in \mathbb{C}^N$

DFT

$$x \longrightarrow \boxed{\mathbf{F}_N} \longrightarrow \mathbf{X}$$

$$x \longleftarrow \boxed{\mathbf{F}_N^{-1} = \frac{1}{N} F_N^H} \longleftarrow \mathbf{X}$$

Inverse DFT

Length-4 Example: Explicit DFT Matrices

Write out F_4, its Hermitian transpose, and verify orthogonality

1) Explicit DFT matrix ($N = 4$)

$$W_4 = e^{-j2\pi/4} = -j$$

$$F_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

2) Hermitian transpose (F_4^H)

$$F_4^H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix}$$

3) Orthogonality proof

$$F_4^H F_4 = 4 I_4$$

$$= \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}$$

4) Unitary matrix from F_4

$$U_4 = \frac{1}{2} F_4 \quad \rightarrow \quad U_4^H U_4 = I_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

☆ The columns of F_4 are orthogonal complex exponentials. After dividing by $\sqrt{4} = 2$, they become *orthonormal*.

Length-4 Signal Example and Exact Reconstruction

Compute $X = \mathbf{F}_4 x$ and recover x from the Hermitian transpose

1) Example signal

$$x = \begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix}$$

2) Forward DFT $(N = 4)$

$$X = \mathbf{F}_4 x = \begin{bmatrix} 2 \\ 1-3j \\ 0 \\ 1+3j \end{bmatrix}$$

3) Transform flow (forward and inverse)

$$\overset{x}{\begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix}} \xrightarrow{\mathbf{F}_4} \overset{X}{\begin{bmatrix} 2 \\ 1-3j \\ 0 \\ 1+3j \end{bmatrix}} \xrightarrow{\text{Inverse}} \overset{x_{\text{recovered}}}{\begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix}}$$

4) Exact reconstruction: standard DFT inverse

$$x = \frac{1}{4}\mathbf{F}_4^H X = \frac{1}{4}\mathbf{F}_4^H \mathbf{F}_4 x = \frac{1}{4}(4I_4)\,x = x$$

Recovered x: $\begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix} = x$

5) Exact reconstruction: unitary DFT inverse

Define unitary spectrum:

$$X_u = U_4 x = \frac{1}{2} X = \begin{bmatrix} 1 \\ \frac{1-3j}{2} \\ 0 \\ \frac{1+3j}{2} \end{bmatrix}$$

Then recover:

$$x = U_4^H X_u = U_4^H U_4 x = I_4 x = x$$

Recovered x: $\begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix} = x$

Unitary spectrum values: $X_u = \left[\overset{1}{\ } 1-3j\right]_2 \;\; 0 \;\; 1+3j\,]_2$

Takeaway: For the standard DFT, $\mathbf{F}^{-1} = \frac{1}{N}\mathbf{F}^H$; for the unitary DFT, $U^{-1} = U^H$.

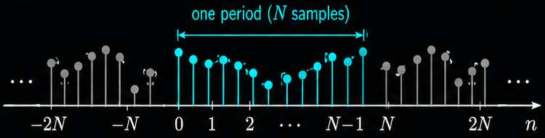
The 1-D DFT as a Periodic Representation

An N-point DFT treats the finite record as one period of a periodic sequence

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi}{N}kn}, \quad k = 0, 1, \ldots, N-1$$

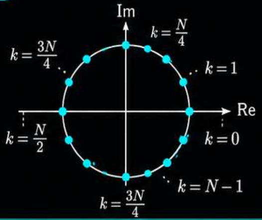
The DFT analyzes one period of a length-N sequence.

1 TIME-DOMAIN PERIODIC EXTENSION



$$x_p[n] = x[(n \bmod N)] \qquad x_p[n+N] = x_p[n]$$

2 FREQUENCY BINS AS EQUALLY SPACED SAMPLES



$$\omega_k = \frac{2\pi k}{N}$$

$$X[k] = X(e^{j\omega})\Big|_{\omega=\omega_k}$$

1 ONE PERIOD IN TIME
the N samples are interpreted as a repeating sequence.

2 DISCRETE FREQUENCY SAMPLES
the spectrum is sampled at N equally spaced frequencies.

3 WRAP-AROUND INTUITION
modulo-N indexing is built into the DFT.

Takeaway: The N-point DFT represents one period of a periodic discrete-time signal and samples its spectrum at N equally spaced frequencies.

Frequency-Domain Periodicity and Circular Convolution

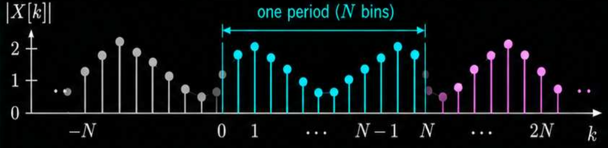
The DFT spectrum repeats every N bins, and this periodicity leads naturally to circular convolution

$$X[k+N] = X[k]$$

The spectrum of an N-point DFT is periodic in k with period N.

1 SPECTRUM REPEATS EVERY N BINS

The DFT spectrum is periodic in the frequency index k with period N.



Why $X[k+N] = X[k]$?

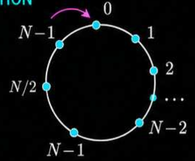
From the DFT:

$$e^{-j\frac{2\pi}{N}(k+N)n} = e^{-j\frac{2\pi}{N}kn}\, e^{-j2\pi n} = e^{-j\frac{2\pi}{N}kn}$$

since $e^{-j2\pi n} = 1$ for all integer n.

2 WRAP-AROUND INTUITION

Frequency-bin indices wrap modulo N.



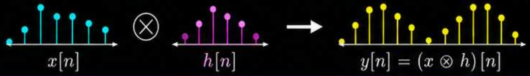
bin indices wrap modulo N

3 WHY IT MATTERS: CIRCULAR CONVOLUTION

Time-domain convolution implied by DFT periodicity is circular.

$$y[n] = (x \otimes h)[n] = \sum_{m=0}^{N-1} x[m]\, h[(n-m) \bmod N]$$

$$Y[k] = X[k]\, H[k], \qquad k = 0, 1, \ldots, N-1$$



Takeaway: In the DFT, frequency bins repeat every N, and the implied wrap-around structure leads to circular convolution.

Orthogonality of the DFT Kernels

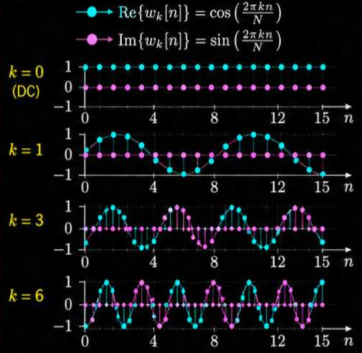
The complex exponential kernels form an orthogonal basis for length-N signals

$$w_k[n] = e^{j\frac{2\pi}{N}kn}, \qquad k = 0, 1, \ldots, N-1$$

$$w_k^*[n] = e^{-j\frac{2\pi}{N}kn}$$

Analysis uses the conjugate kernel; synthesis uses the forward kernel.

① EXAMPLES OF KERNELS ($N = 16$)



② ORTHOGONALITY PROOF

Inner product of two length-N sequences:

$$\langle a, b\rangle = \sum_{n=0}^{N-1} a[n]\, b^*[n]$$

Inner product of DFT kernels:

$$\langle w_k, w_\ell\rangle = \sum_{n=0}^{N-1} e^{j\frac{2\pi}{N}(k-\ell)n}$$

$$= \begin{cases} N, & \text{if } k = \ell \\ 0, & \text{if } k \neq \ell \end{cases}$$

- If $k = \ell$: each term is $e^{j0} = 1$, so the sum is N.
- If $k \neq \ell$: it is a geometric series whose terms lie on the unit circle and sum to 0.

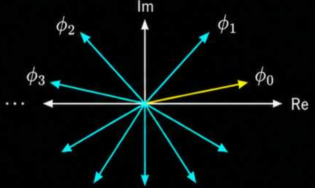
③ NORMALIZED ORTHONORMAL BASIS

Normalize the kernels:

$$\phi_k[n] = \frac{1}{\sqrt{N}}\, e^{j\frac{2\pi}{N}kn}$$

They are orthonormal:

$$\langle \phi_k, \phi_\ell\rangle = \delta[k-\ell]$$



Takeaway: The DFT kernels are orthogonal; after normalization they form an orthonormal basis for the length-N signal space.

Projection onto DFT Kernels and Reconstruction

Each DFT coefficient is an inner-product projection, and the inverse DFT recombines those coordinates

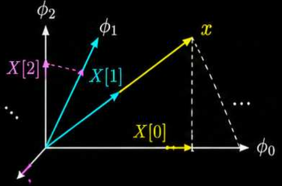
$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi}{N}kn} = \langle x, w_k \rangle$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k]\; w_k[n]$$

Analysis = projection; synthesis = weighted recombination.

1 PROJECTION INTERPRETATION

The DFT kernels $\{\phi_k\}$ form an orthonormal basis.



DFT coefficient = inner product (projection).

$$\langle x, \phi_k \rangle = \frac{X[k]}{\sqrt{N}}$$

$\{\phi_k\}_{k=0}^{N-1}$ orthonormal: $\langle \phi_k, \phi_\ell \rangle = \delta[k-\ell]$

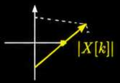
Each $X[k]$ is the coordinate of x along ϕ_k.

2 WHAT EACH COEFFICIENT MEANS

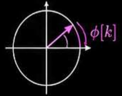
Each $X[k]$ is complex.

$$X[k] = |X[k]|\, e^{j\phi[k]}$$

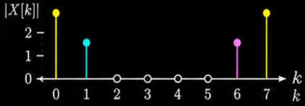
Magnitude $|X[k]|$ = length / weight



Phase $\phi[k]$ = shift / rotation



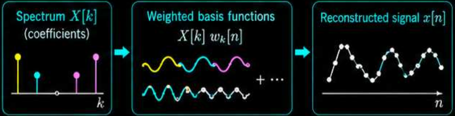
Example spectrum ($N = 8$):



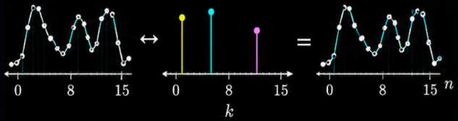
Nonzero bins at $k = 0, 1, 6, 7$.
Others are zero.

3 INVERSE DFT RECONSTRUCTION

Inverse DFT adds scaled kernels to rebuild $x[n]$.



Mini-example ($N = 16$):



Keeping all coefficients gives exact reconstruction.

Takeaway: The DFT computes the coordinates of the signal in an orthogonal kernel basis; the inverse DFT adds those weighted basis functions to recover the signal exactly.

DFT Property: Circular Time Shift

Shifting a discrete signal in time leaves the magnitude unchanged and adds a linear phase term

Let

$$y[n] = x[(n - n_0) \mod N]$$

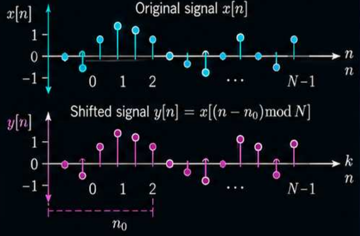
Then the N-point DFT satisfies

$$Y[k] = X[k]\, e^{-j\frac{2\pi k n_0}{N}}, \qquad k = 0, 1, \ldots, N-1$$

$$|Y[k]| = |X[k]| \qquad\qquad \angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$$

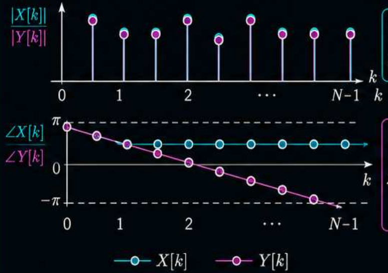
A circular shift in time does not change the DFT magnitude; it only adds a linear phase ramp.

TIME DOMAIN: CIRCULAR SHIFT



Time domain: samples move by n_0 positions (circularly).

FREQUENCY DOMAIN: DFT RELATIONSHIP



Magnitude unchanged: $|Y[k]| = |X[k]|$

Phase adds linear ramp: $\angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$

WHY THIS MATTERS

Delay / alignment analysis
Time shifts appear as linear phase ramps.

Image registration and cyclic shifts
Cyclic shifts in images preserve magnitude.

Phase carries position information
Magnitude shows what, phase shows where.

Takeaway: Circular time shift ↔ multiplication by $e^{-j\frac{2\pi k n_0}{N}}$ in frequency; magnitude stays the same, phase changes linearly.

DFT Property: Time Dilation / Expansion

Expanding a discrete signal by inserting zeros compresses the spectrum and creates repeated spectral images

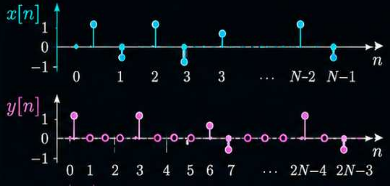
Let $y[n] = x[n/L]$ for $n = 0, \pm L, \pm 2L, \ldots$ and $y[n] = 0$ otherwise.

$$Y[k] = \sum_{n=0}^{LN-1} y[n]\, e^{-j\frac{2\pi kn}{LN}} = \sum_{m=0}^{N-1} x[m]\, e^{-j\frac{2\pi km}{N}} = X[k \mod N] \quad \text{for } k = 0, 1, \ldots, LN-1.$$

Intuitively (DTFT relation): $Y(e^{j\omega}) = X(e^{j\omega L})$

Time expansion by L inserts zeros between samples, stretches the signal in time, and compresses the spectrum while repeating it L times over 0 to 2π.

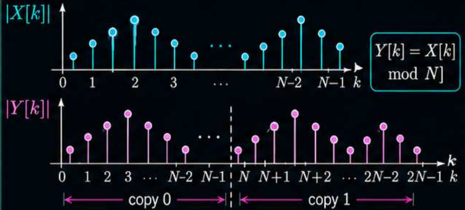
TIME DOMAIN: EXPANSION BY $L = 2$



insert one zero between every pair of samples

time spacing doubles

FREQUENCY DOMAIN: SPECTRAL IMAGES



$Y[k] = X[k \mod N]$

spectrum compressed / repeated

WHAT TO REMEMBER

- Expansion in time → compression in frequency
- Inserted zeros do NOT add new information
- In the discrete periodic spectrum, copies repeat across the frequency axis

Takeaway: Dilation (time expansion) by L spreads the samples apart in time and causes the DFT spectrum to repeat L times across the frequency axis.

DFT Property: Zero Padding

Appending zeros in time does not add information; it samples the same spectrum more densely

Define zero padding of length $M > N$:

$$x_p[n] = x[n], \quad 0 \le n \le N-1; \qquad x_p[n] = 0, \quad N \le n \le M-1, \quad M > N$$

The M-point DFT of the padded signal:

$$X_p[k] = \sum_{n=0}^{M-1} x_p[n]\, e^{-j\frac{2\pi kn}{M}} = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi kn}{M}}, \qquad k = 0, 1, \ldots, M-1$$

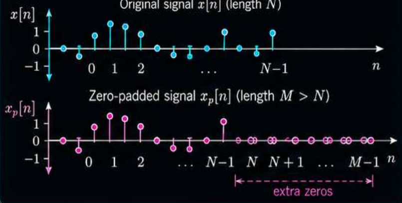
Interpretation (samples of the same DTFT):

$$X_p[k] = X(e^{j\omega})\Big|_{\omega = \frac{2\pi k}{M}}$$

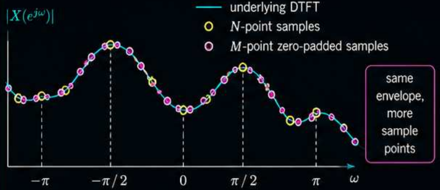
Zero padding keeps the underlying DTFT the same, but evaluates it on a denser frequency grid.

Important: zero padding improves visual spectral resolution / interpolation, but it does not improve true frequency resolution or create new information.

TIME DOMAIN: APPEND ZEROS



FREQUENCY DOMAIN: DENSER SAMPLES OF THE SAME DTFT



WHY ZERO PADDING IS USEFUL

- Makes spectral peaks easier to visualize
- Helps estimate peak location / interpolate between bins
- Does not change the physical bandwidth or add information

Takeaway: Zero padding in time gives a denser-looking DFT in frequency because it samples the same DTFT on a finer grid—useful for display and peak estimation, but not new information.

DFT Property: Circular Time Shift

Shifting a discrete signal in time leaves the magnitude unchanged and adds a linear phase term

Let $$y[n] = x[(n-n_0) \mod N]$$

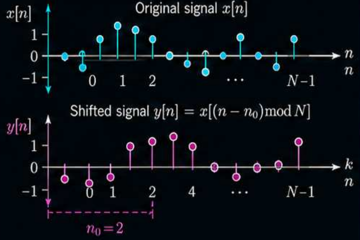
Then the N-point DFT satisfies

$$Y[k] = X[k]\, e^{-j\frac{2\pi k n_0}{N}}, \qquad k = 0, 1, \dots, N-1$$

$$|Y[k]| = |X[k]| \qquad\qquad \angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$$

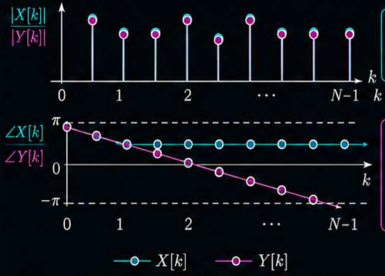
A circular shift in time does not change the DFT magnitude; it only adds a linear phase ramp.

TIME DOMAIN: CIRCULAR SHIFT



Time domain: samples move by n_0 positions (circularly).

FREQUENCY DOMAIN: DFT RELATIONSHIP



Magnitude unchanged: $|Y[k]| = |X[k]|$

Phase adds linear ramp: $\angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$

WHY THIS MATTERS

Delay / alignment analysis
Time shifts appear as linear phase ramps.

Image registration and cyclic shifts
Cyclic shifts in images preserve magnitude.

Phase carries position information
Magnitude shows what, phase shows where.

Takeaway: Circular time shift $\leftrightarrow$ multiplication by $e^{-j\frac{2\pi k n_0}{N}}$ in frequency; magnitude stays the same, phase changes linearly.

DFT Property: Circular Time Shift

Shifting a discrete signal in time leaves the magnitude unchanged and adds a linear phase term

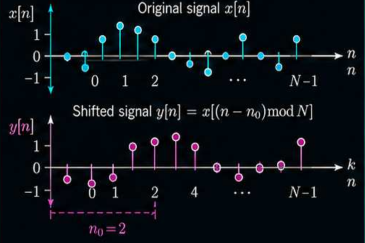
Let
$$y[n] = x[(n - n_0) \mod N]$$

Then the N-point DFT satisfies
$$Y[k] = X[k]\, e^{-j\frac{2\pi k n_0}{N}}, \qquad k = 0, 1, \ldots, N-1$$

$$|Y[k]| = |X[k]| \qquad\qquad \angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$$

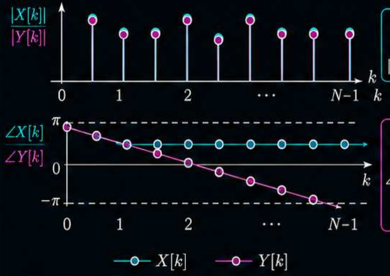
A circular shift in time does not change the DFT magnitude; it only adds a linear phase ramp.

TIME DOMAIN: CIRCULAR SHIFT



Time domain: samples move by n_0 positions (circularly).

FREQUENCY DOMAIN: DFT RELATIONSHIP



Magnitude unchanged: $|Y[k]| = |X[k]|$

Phase adds linear ramp: $\angle Y[k] = \angle X[k] - \frac{2\pi k n_0}{N}$

WHY THIS MATTERS

Delay / alignment analysis
Time shifts appear as linear phase ramps.

Image registration and cyclic shifts
Cyclic shifts in images preserve magnitude.

Phase carries position information
Magnitude shows what, phase shows where.

Takeaway: Circular time shift $\leftrightarrow$ multiplication by $e^{-j\frac{2\pi k n_0}{N}}$ in frequency; magnitude stays the same, phase changes linearly.

1-D DFT as a Matrix–Vector Operation

Forward transform: represent the discrete signal as a column vector and compute its spectrum by matrix multiplication

$$\mathbf{X} = \mathbf{F}_N\,\mathbf{x}$$

$$\mathbf{x} = [x[0] \;\; x[1] \;\; x[2] \;\; \cdots \;\; x[N-1]]^T$$
$$\mathbf{X} = [X[0] \;\; X[1] \;\; X[2] \;\; \cdots \;\; X[N-1]]^T$$

SIGNAL VECTOR

time-domain samples

$$\mathbf{x} = \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ \vdots \\ x[N-1] \end{bmatrix} \in \mathbb{C}^{N\times 1}$$

DFT MATRIX (ANALYSIS MATRIX)

$$W_N = e^{-j\,2\pi/N}$$

$$\mathbf{F}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N & W_N^2 & \cdots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix}$$

$$[\mathbf{F}_N]_{k,n} = e^{-j\,2\pi\,kn/N}, \qquad k,n = 0,1,\ldots,N-1$$

CONCEPTUAL FLOW

Signal vector → Multiply by DFT matrix → Spectrum vector

$$\mathbf{x} = \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ \vdots \\ x[N-1] \end{bmatrix} \in \mathbb{C}^{N\times 1} \;\rightarrow\; \boxed{\begin{matrix}\mathbf{F}_N \\ (N\times N)\end{matrix}} \;\rightarrow\; \mathbf{X} = \begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ \vdots \\ X[N-1] \end{bmatrix} \in \mathbb{C}^{N\times 1}$$

COMPONENTWISE FORWARD DFT

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\,2\pi\,kn/N}, \qquad k = 0,1,\ldots,N-1$$

Each row of $\mathbf{F}_N$ projects the signal onto one complex exponential kernel.

KEY INTERPRETATION

- Forward DFT = analysis
- Maps time samples to frequency coefficients
- Each $X[k]$ is a complex coefficient for the k-th frequency bin
- The columns/rows are built from powers of the primitive root W_N

Takeaway: The forward 1-D DFT is a linear transformation that maps the sample vector x into the spectrum vector $\mathbf{X}$ through multiplication by the DFT matrix $\mathbf{F}_N$.

Inverse 1-D DFT in Matrix–Vector Form

The synthesis matrix reconstructs the time-domain signal from the spectrum vector

$$\mathbf{x} = \mathbf{F}_N^{-1}\mathbf{X} = \frac{1}{N}\mathbf{F}_N^{H}\mathbf{X}$$

$\mathbf{F}_N^H$ = conjugate transpose of $\mathbf{F}_N$

$\mathbf{x} = [\,x[0] \;\; x[1] \;\; \dots \;\; x[N-1]\,]^T$

$\mathbf{X} = [\,X[0] \;\; X[1] \;\; \dots \;\; X[N-1]\,]^T$

INVERSE DFT MATRIX

$$\mathbf{F}_N^{-1} = \frac{1}{N}\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \cdots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \cdots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \cdots & W_N^{-(N-1)(N-1)} \end{bmatrix}$$

$$[\mathbf{F}_N^{-1}]_{n,k} = \frac{1}{N}\, e^{+j\frac{2\pi n k}{N}}, \quad n,k = 0,1,\dots,N-1$$

$$W_N = e^{-j\frac{2\pi}{N}}$$

ORTHOGONALITY / UNITARY RELATION

$$\mathbf{F}_N^H\mathbf{F}_N = \mathbf{F}_N\mathbf{F}_N^H = N\mathbf{I}$$

therefore

$$\mathbf{F}_N^{-1} = \frac{1}{N}\mathbf{F}_N^H$$

Rows/columns of $\mathbf{F}_N$ are orthogonal complex exponential basis vectors.

$$\boxed{\mathbf{x}} \xrightarrow{\mathbf{F}_N} \boxed{\mathbf{X}} \xrightarrow{\frac{1}{N}\mathbf{F}_N^H} \boxed{\mathbf{x}}$$

exact reconstruction

COMPONENTWISE INVERSE DFT

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]\, e^{+j\frac{2\pi k n}{N}}, \quad n = 0,1,\dots,N-1$$

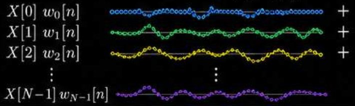
- inverse DFT = synthesis
- each coefficient $X[k]$ scales one basis sequence
- all scaled basis sequences are added to rebuild $x[n]$

WHAT RECONSTRUCTION DOES

Spectrum vector $\mathbf{X}$

$$\mathbf{X} = \begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ \vdots \\ X[N-1] \end{bmatrix}$$

Weighted basis functions $X[k]\,w_k[n]$



Reconstructed signal $x[n]$

$x[n]$ = sum of all terms

Takeaway: The inverse 1-D DFT is a linear synthesis operation: multiplying the spectrum vector by $\frac{1}{N}\mathbf{F}_N^H$ exactly reconstructs the time-domain signal.

Why the Inverse Comes from the Hermitian Transpose

Standard DFT versus unitary DFT

Standard DFT (matrix F_N):

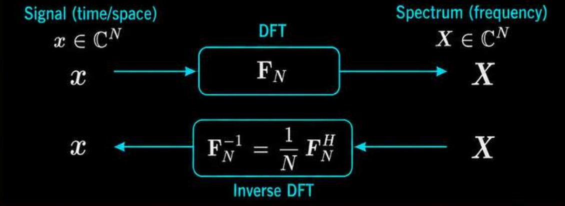
$$\mathbf{X} = \mathbf{F}_N\, x \quad \Big| \quad x = \mathbf{F}_N^{-1}\mathbf{X} = \frac{1}{N} F_N^H \mathbf{X} \quad \Big| \quad \mathbf{F}_N^H \mathbf{F}_N = N I$$

Unitary DFT (matrix U_N):

$$U_N = \frac{1}{\sqrt{N}} \mathbf{F}_N \quad \Big| \quad \mathbf{X}_u = U_N\, x \quad \Big| \quad x = U_N^H \mathbf{X}_u = U_N^H U_N\, x \quad \Big| \quad U_N^H U_N = I$$

Important: the expression $x = \mathbf{F}^H \mathbf{X} = \mathbf{F}^H \mathbf{F} x$ is exactly true for the unitary DFT matrix U.
For the standard DFT matrix F, the correct inverse is $x = \frac{1}{N} F^H \mathbf{X}$.

- Orthogonal complex exponential columns: $\mathbf{F}^H F = NI$.
- Hermitian transpose gives matched synthesis: $\mathbf{F}^H$ pairs each basis with its conjugate counterpart.
- Normalization determines the inverse: unitary $\Rightarrow \mathbf{F}^{-1} = \mathbf{F}^H$; standard $\Rightarrow \mathbf{F}^{-1} = \frac{1}{N} F^H$.



Length-4 Example: Explicit DFT Matrices

Write out F_4, its Hermitian transpose, and verify orthogonality

1) Explicit DFT matrix ($N = 4$)

$$W_4 = e^{-j2\pi/4} = -j$$

$$\boldsymbol{F}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

2) Hermitian transpose ($\boldsymbol{F}_4^H$)

$$\boldsymbol{F}_4^H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix}$$

3) Orthogonality proof

$$\boldsymbol{F}_4^H \boldsymbol{F}_4 = 4\boldsymbol{I}_4$$

$$= \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}$$

4) Unitary matrix from $\boldsymbol{F}_4$

$$U_4 = \frac{1}{2} F_4 \quad \rightarrow \quad U_4^H U_4 = \boldsymbol{I}_4$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

☆ The columns of F_4 are orthogonal complex exponentials. After dividing by $\sqrt{4} = 2$, they become *orthonormal*.

Length-4 Signal Example and Exact Reconstruction

Compute $X = \mathbf{F}_4 x$ and recover x from the Hermitian transpose

1) Example signal

$$x = \begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix}$$

2) Forward DFT $(N = 4)$

$$X = \mathbf{F}_4 x = \begin{bmatrix} 2 \\ 1-3j \\ 0 \\ 1+3j \end{bmatrix}$$

3) Transform flow (forward and inverse)

$$\underset{x}{\begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix}} \xrightarrow{\ \mathbf{F}_4\ } \underset{X}{\begin{bmatrix} 2 \\ 1-3j \\ 0 \\ 1+3j \end{bmatrix}} \xrightarrow{\ \text{Inverse}\ } \underset{x_{\text{recovered}}}{\begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix}}$$

4) Exact reconstruction: standard DFT inverse

$$x = \frac{1}{4}\mathbf{F}_4^H X = \frac{1}{4}\mathbf{F}_4^H \mathbf{F}_4 x = \frac{1}{4}(4I_4)\,x = x$$

Recovered x: $\begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix} = x$

5) Exact reconstruction: unitary DFT inverse

Define unitary spectrum:

$$X_u = U_4 x = \frac{1}{2} X = \begin{bmatrix} 1 \\ \frac{1-3j}{2} \\ 0 \\ \frac{1+3j}{2} \end{bmatrix}$$

Then recover:

$$x = U_4^H X_u = U_4^H U_4 x = I_4 x = x$$

Recovered x: $\begin{bmatrix} 1 \\ 2 \\ 0 \\ -1 \end{bmatrix} = x$

Unitary spectrum values: $X_u = \Big[\, ^{1}\ 1-3j\Big]_2 \quad 0 \quad 1+3j\Big]_2$

Takeaway: For the standard DFT, $\mathbf{F}^{-1} = \frac{1}{N}\mathbf{F}^H$; for the unitary DFT, $U^{-1} = U^H$.

Overview of the 2-D Discrete Fourier Transform

From a sampled image to its 2-D frequency-domain representation

SPATIAL DOMAIN IMAGE $f[m,n]$

Size: $M \times N$ pixels

THE 2-D DISCRETE FOURIER TRANSFORM (DFT)

Forward 2-D DFT (analysis)

$$F[u,v] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[m,n]\, e^{-j\,2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)},$$

$$u = 0, \dots, M-1, \quad v = 0, \dots, N-1$$

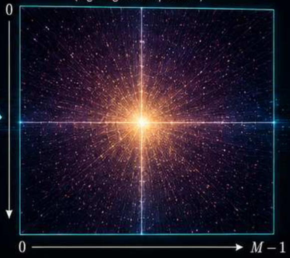
Inverse 2-D DFT (synthesis)

$$f[m,n] = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F[u,v]\, e^{+j\,2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)},$$

$$m = 0, \dots, M-1, \quad n = 0, \dots, N-1$$

FREQUENCY DOMAIN $|F[u,v]|$

(log-magnitude spectrum)

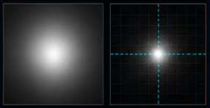


Zero frequency (DC) at center

NOTATION

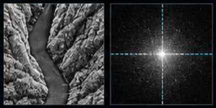
- $f[m,n]$ = image intensity (real-valued)
- $F[u,v]$ = complex spectrum (frequency coefficients)
- (m,n) = spatial coordinates (row, column)
- (u,v) = frequency coordinates (horizontal, vertical)
- $M \times N$ = image size (rows M, columns N)
- $j = \sqrt{-1}$ = imaginary unit

1 LOW FREQUENCIES



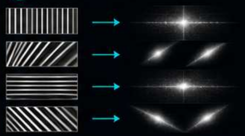
Overall illumination and coarse shapes

2 HIGH FREQUENCIES



Detail, sharp transitions, and texture

3 ORIENTATION



Orientation in the image becomes direction in the spectrum.

TAKEAWAY: The 2-D DFT rewrites an image as a weighted sum of 2-D complex exponentials; magnitude shows what frequencies are present, and phase preserves spatial structure.

2-D DFT as Matrix Multiplication

Represent the image as a matrix and compute its spectrum with left and right DFT matrices

$$\mathbf{F} = \mathbf{W}_M \, f \, \mathbf{W}_N^T \qquad \Big| \qquad f = \frac{1}{MN} \mathbf{W}_M^H \, \mathbf{F} \, \mathbf{W}_N^H$$

$f = [f[m,n]] \in \mathbb{C}^{M \times N}$
image matrix

$\mathbf{F} = [F[u,v]] \in \mathbb{C}^{M \times N}$
spectrum matrix

DFT MATRICES

$$[\mathbf{W}_M]_{u,m} = e^{-j2\pi um/M}, \quad u, m = 0, \ldots, M-1$$

$$\mathbf{W}_M = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_M & W_M^2 & \cdots & W_M^{M-1} \\ 1 & W_M^2 & W_M^4 & \cdots & W_M^{2(M-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_M^{M-1} & W_M^{2(M-1)} & \cdots & W_M^{(M-1)(M-1)} \end{bmatrix}$$

where $W_M = e^{-j2\pi/M}$

$$[\mathbf{W}_N]_{v,n} = e^{-j2\pi vn/N}, \quad v, n = 0, \ldots, N-1$$

$$\mathbf{W}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N & W_N^2 & \cdots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix}$$

where $W_N = e^{-j2\pi/N}$

IMAGE MATRIX f

52	55	61	66
63	59	55	90
62	59	68	113
63	65	84	100

$M \times N$

Multiply on the right by $\mathbf{W}_N^T$

DFT along rows (across m)

INTERMEDIATE MATRIX $\mathbf{G} = f \, \mathbf{W}_N^T$

$\cdots$	$\cdots$	$\cdots$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

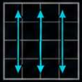
Multiply on the left by $\mathbf{W}_M$

DFT along columns (down n)

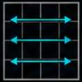
SPECTRUM MATRIX $\mathbf{F}$

$M \times N$

INTERPRETATION



Left multiplication by $\mathbf{W}_M$ mixes rows and performs DFT along the columns (transforms vertical coordinate $n \to v$).



Right multiplication by $\mathbf{W}_N^T$ mixes columns and performs DFT along the rows (transforms horizontal coordinate $m \to u$).

Each element $F[u,v]$ is one 2-D frequency coefficient.

TAKEAWAY: The 2-D DFT is a linear transformation on a matrix: rows and columns are transformed by DFT matrices to produce the 2-D spectrum.

Separable Computation of the 2-D DFT

A 2-D DFT is two cascaded 1-D DFTs: first across one dimension, then across the other

$M \times N$ image

$\mathbf{f}$

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

$$\mathbf{F} = \mathbf{W}_M \left(\mathbf{f}\, \mathbf{W}_N^T\right) = \left(\mathbf{W}_M\, \mathbf{f}\right) \mathbf{W}_N^T$$

$$F[u,v] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[m,n]\, e^{-j2\pi\left(\frac{um}{M} + \frac{vn}{N}\right)} \qquad u = 0,\ldots,M-1, \quad v = 0,\ldots,N-1$$

1 ROW TRANSFORMS

Apply a 1-D DFT to every row

$\mathbf{G} = \mathbf{f}\mathbf{W}_N^T$

$\mathbf{f}$

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

DFT $\rightarrow$

$\mathbf{G}$ (row-transformed)

10	$-2+2j$	-2	$-2-2j$
26	$-2+2j$	-2	$-2-2j$
42	$-2+2j$	-2	$-2-2j$
58	$-2+2j$	-2	$-2-2j$

2 COLUMN TRANSFORMS

Apply a 1-D DFT to every column

$\mathbf{F} = \mathbf{W}_M\mathbf{G}$

$\mathbf{G}$ (from step 1)

10	$-2+2j$	-2	$-2-2j$
26	$-2+2j$	-2	$-2-2j$
42	$-2+2j$	-2	$-2-2j$
58	$-2+2j$	-2	$-2-2j$

DFT $\downarrow$

$\mathbf{F}$ (final 2-D DFT)

120	-8	0	0
$-32+32j$	0	0	0
0	0	0	0
$-32-32j$	0	0	0

3 EQUIVALENT ORDER

Column-first then row-first gives the same result $\mathbf{F}$

$\mathbf{f}$

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Rows then Columns $\rightarrow$ $\boxed{\mathbf{F}}$

$=$

Columns then Rows $\rightarrow$ $\boxed{\mathbf{F}}$

VECTORIZATION / KRONECKER FORM

vec(**f**)

$$\mathrm{vec}(\mathbf{F}) = (\mathbf{W}_N \otimes \mathbf{W}_M)\, \mathrm{vec}(\mathbf{f})$$

$\otimes$: Kronecker product

PRACTICAL NOTE

This separability is why the 2-D FFT is efficient: apply fast 1-D FFTs to rows and columns.

Cost: $O(MN(\log_2 M + \log_2 N))$ versus $O(M^2N^2)$ for direct 2-D DFT.

TAKEAWAY: A 2-D DFT is separable: it is computed by two 1-D transforms, making implementation and intuition much easier.

2-D DFT Property: Circular Spatial Shift

Shifting an image in space leaves the magnitude unchanged and adds a linear phase plane

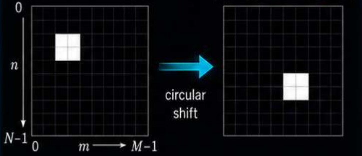
$$g[m,n] = f[(m-m_0) \bmod M,\ (n-n_0) \bmod N] \qquad G[u,v] = F[u,v]\, e^{-j2\pi\left(\frac{um_0}{M}+\frac{vn_0}{N}\right)}$$

$$|G[u,v]| = |F[u,v]| \qquad \angle G[u,v] = \angle F[u,v] - 2\pi\left(\frac{um_0}{M}+\frac{vn_0}{N}\right)$$

1) SPATIAL DOMAIN: CIRCULAR SHIFT

Original image $f[m,n]$

Shifted image $g[m,n]$ (shift by $m_0 = 3$, $n_0 = 2$)

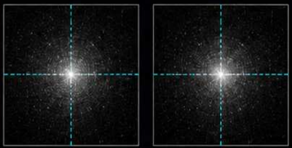


Shift: $m_0 = 3$ (right), $n_0 = 2$ (down)

2) MAGNITUDE SPECTRUM: UNCHANGED

$|F[u,v]|$ (original)

$|G[u,v]|$ (shifted)



Magnitude is identical:

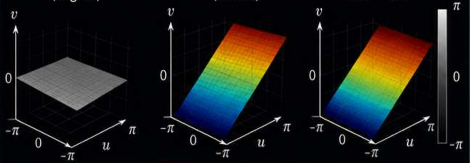
$|G[u,v]| = |F[u,v]|$

3) PHASE SPECTRUM: LINEAR PHASE RAMP ADDED

$\angle F[u,v]$ (original)

$\angle G[u,v]$ (shifted)

Phase difference $\angle G - \angle F$



Linear phase ramp:

$\angle G[u,v] = \angle F[u,v] - 2\pi\left(\frac{um_0}{M}+\frac{vn_0}{N}\right)$

INTERPRETATION

- A circular translation in the spatial domain corresponds to a linear phase ramp in frequency.
- The magnitude spectrum is unchanged, so texture and energy distribution are preserved.

APPLICATIONS

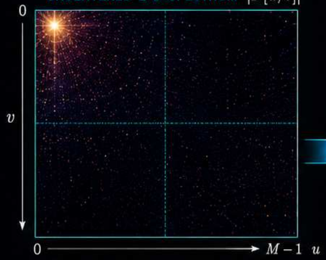
- Image registration and alignment.
- Phase correlation (uses phase differences to detect translation).

TAKEAWAY: A circular shift in the image domain changes only phase, not magnitude; the phase becomes a linear ramp in frequency.

Spectrum Centering by Modulation

Multiplying by $(-1)^{m+n}$ shifts the 2-D spectrum so the DC term moves from-re corner to the center

UNCENTERED 2-D SPECTRUM $|F[u,v]|$



DC at upper-left corner (low frequencies packed in the corner → hard to see detail)

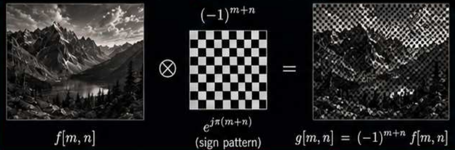
$$g[m,n] = (-1)^{m+n} f[m,n] = e^{j\pi(m+n)} f[m,n]$$

For even M, N,

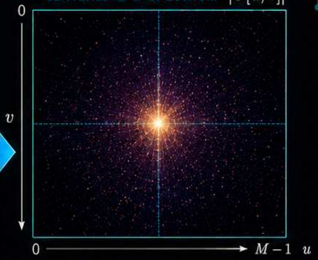
$$G[u,v] = F\left[\left(u - \frac{M}{2}\right) \bmod M,\ \left(v - \frac{N}{2}\right) \bmod N\right]$$

This is a special case of modulation / frequency shift.

MODULATION (MULTIPLY IN SPACE)



CENTERED 2-D SPECTRUM $|G[u,v]|$



DC moved to center (low frequencies now centered → easy to see and process)

WHY IT WORKS (DERIVATION SKETCH)

$$g[m,n] = e^{j\pi m}\, e^{j\pi n}\, f[m,n]$$

$$\Rightarrow G[u,v] = F[(u-u_0) \bmod M, (v-v_0) \bmod N]$$

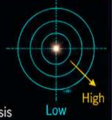
where $u_0 = \frac{M}{2}$, $v_0 = \frac{N}{2}$ (since M, N even)

Multiplying by $e^{j\pi m}\ e^{j\pi n}$ shifts the spectrum by $\frac{M}{2}$ columns and $\frac{N}{2}$ rows.

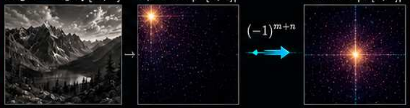
WHY CENTERED SPECTRA ARE USED

- Low frequencies (most energy) are at the center → easier to visualize and interpret
- Radially symmetric filters (e.g., low-pass, band-pass, Gaussian) are easy to design and apply
- Frequency-domain operations and masks become more intuitive (distance from center = frequency)
- Essential for filtering, enhancement, and feature analysis

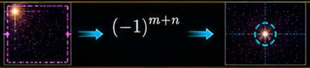
Distance from center = frequency



EXAMPLE (REAL IMAGE)



TAKEAWAY: The factor $(-1)^{m+n}$ is a simple modulation that re-centers the spectrum and makes low frequencies visually intuitive.



2-D DFT Property: Zero Padding

Padding an image with zeros does not add information; it samples the same 2-D DTFT more densely

1) DEFINE ZERO-PADDED IMAGE

Original image size: $M \times N$

Padded image size: $M_p \times N_p$

with $M_p \geq M,\ N_p \geq N$

$$f_p[m,n] = \begin{cases} f[m,n], & 0 \leq m \leq M-1,\ 0 \leq n \leq N-1 \\ 0, & otherwise \end{cases}$$

(the original image placed in the top-left corner; zeros elsewhere)

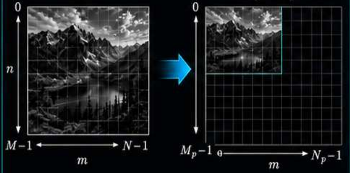
2) DFT OF THE ZERO-PADDED IMAGE

$$F_p[u,v] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[m,n]\ e^{-j2\pi\left(\frac{um}{M_p} + \frac{vn}{N_p}\right)}, \quad u = 0,\ldots,M_p-1,\ \ v = 0,\ldots,N_p-1$$

3) INTERPRETATION

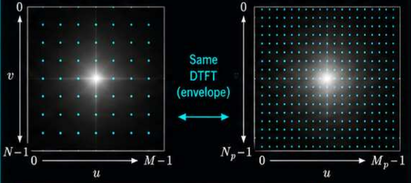
Zero padding does not change the underlying continuous-space DTFT of $f[m,n]$.
It evaluates the same DTFT on a **denser frequency grid** in both dimensions.

1) IMAGE: ORIGINAL vs ZERO-PADDED



$M_p \geq M,\ N_p \geq N$

② SPECTRUM: SAME DTFT, DENSER SAMPLES

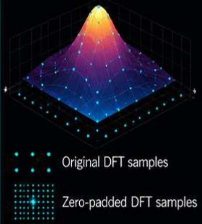


• Sample locations in the 2-D frequency domain

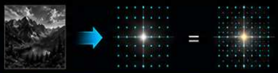
③ KEY POINTS

- ✓ Better spectral display (smoother appearance)
- ✓ Easier peak localization (more sample points)
- ✓ No new information added
- ✓ No true increase in resolution (bandwidth unchanged)

Same spectrum envelope, more samples



TAKEAWAY: Zero padding improves spectral visualization and interpolation in frequency, but it **does not** create new image content.



Worked 4x4 Example of the 2-D DFT

Explicit matrices for a small image and its spectrum

4×4 IMAGE f

$$\mathbf{f} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

4×4 DFT MATRIX $\mathbf{W}_4$

$$W_4 = e^{-j2\pi/4} = -j$$

$$\mathbf{W}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

FORWARD 2-D DFT (MATRIX FORM)

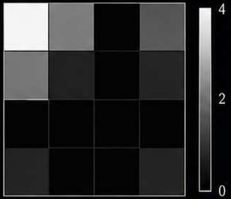
$$\boxed{\mathbf{F} = \mathbf{W}_4\, \mathbf{f}\, \mathbf{W}_4^T}$$

SPECTRUM $\mathbf{F}$ (explicit result)

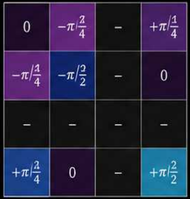
$$\mathbf{F} = \begin{bmatrix} 4 & 2-2j & 0 & 2+2j \\ 2-2j & -2j & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 2+2j & 2 & 0 & 2j \end{bmatrix}$$

All values simplified. $j^2 = -1$.

4×4 MAGNITUDE $|\mathbf{F}|$



PHASE $\angle\mathbf{F}$ (radians)



Complex value $a + jb$ $\Rightarrow$ magnitude $|a + jb| = \sqrt{a^2 + b^2}$ phase $\angle(a + jb) = \tan^{-1}(b/a)$

NOTE: Most energy is in the low frequencies (top-left). The 2×2 block is a coarse, smooth structure, so its spectrum is concentrated near DC (0,0).

INVERSE 2-D DFT (REMINDER)

$$\mathbf{f} = \frac{1}{16}\, \mathbf{W}_4^H\, \mathbf{F}\, \mathbf{W}_4^H \qquad (\text{since } N_x N_y = 4 \cdot 4 = 16)$$

$\mathbf{F}$ spectrum $\xrightarrow{\text{inverse DFT}}$ $\mathbf{f}$ image $\xrightarrow{\text{forward DFT}}$ $\mathbf{F}$ spectrum

TAKEAWAY: A tiny 4x4 image already shows the full 2-D DFT story: matrix multiplication, complex coefficients, and dominant low-frequency energy.

Large 1024x1024 Image Example and Applications

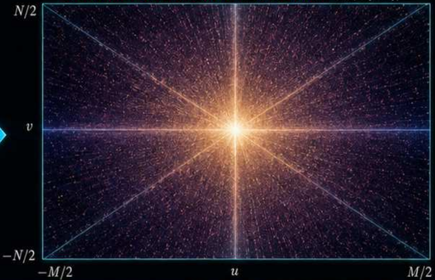
How the 2-D DFT reveals structure, scale, and orientation in a real image

$$\mathbf{F} = \mathrm{DFT2}\{f\} \qquad f = \mathrm{IDFT2}\{\mathbf{F}\}$$

$$F[u,v] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f[m,n]\, e^{-j2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)} \qquad f[m,n] = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F[u,v]\, e^{+j2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)}$$

1024x1024 IMAGE

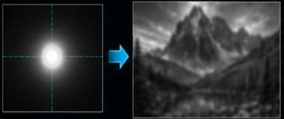
CENTERED LOG-MAGNITUDE SPECTRUM $\log|F[u,v]|$



USES & APPLICATIONS

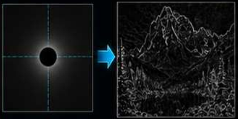
- **Image Enhancement** — *boost contrast, sharpen, suppress noise*
- **Compression** — *retain low frequencies, discard high*
- **Denoising** — *remove noise in the frequency domain*
- **Restoration** — *deblur, dehaze, improve quality*
- **Deconvolution** — *undo blur using known PSF*
- **Pattern Analysis** — *detect textures, orientations, periodicity*

① LOW-PASS FILTERING



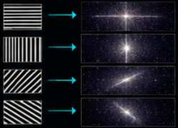
Keeps low frequencies → *smooth structure, blurry details.*

② HIGH-PASS FILTERING



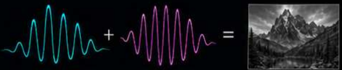
Keeps high frequencies → *sharp transitions, fine detail.*

③ ORIENTATION INTERPRETATION



- Horizontal patterns → horizontal spectral ridge
- Vertical patterns → vertical spectral ridge
- Diagonal (/) patterns → diagonal ridge
- Diagonal (\) patterns → diagonal ridge

TAKEAWAY: For large images, the 2-D DFT is a powerful lens: low frequencies describe coarse content, high frequencies *describe* detail, and spectral geometry reveals structure and direction.



Color Image and RGB Spectra (1024×1024)

A 2-D DFT can be applied separately to the red, green, and blue channels of a color image

Color image $f[m,n]$ (1024×1024)

$$F_R[u,v] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} R[m,n]\, e^{-j2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)}$$

$$F_G[u,v] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} G[m,n]\, e^{-j2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)}$$

$$F_B[u,v] = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} B[m,n]\, e^{-j2\pi\left(\frac{um}{M}+\frac{vn}{N}\right)}$$

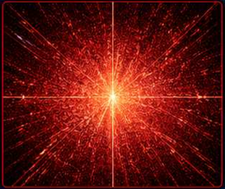
Display usually uses centered log-magnitude spectra:

$$\log\left(1+|F_c[u,v]|\right), \quad c \in \{R,G,B\}$$

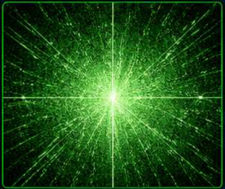
Interpretation

- Each color channel has its own spatial-frequency content:
- Low frequencies near the center describe smooth color variations.
- High frequencies away from the center describe edges, texture, and fine detail.
- Differences among R, G, and B spectra reveal color-dependent structure.

Red channel $|F_R[u,v]|$



Green channel $|F_G[u,v]|$



Blue channel $|F_B[u,v]|$

Takeaway: A color image is analyzed by applying the 2-D DFT separately to R, G, and B; the resulting spectra show how spatial detail and structure are distributed in each color channel.

Ideal Filters in the 2-D Frequency Domain

Binary masks in the spectrum keep some frequencies and reject the rest

Centered radial frequency distance

$$D(u,v) = \sqrt{\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2}$$

Ideal low-pass filter

$$H_{LP}(u,v) = \begin{cases} 1, & D(u,v) \le D_0 \\ 0, & \text{otherwise} \end{cases}$$

Ideal high-pass filter

$$H_{HP}(u,v) = 1 - H_{LP}(u,v)$$

Frequency-domain filtering

$$G(u,v) = H(u,v)\, F(u,v)$$
$$g[m,n] = \mathrm{IDFT}_2\{G(u,v)\}$$

1) Input image and centered spectrum

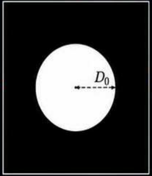
Input image $f[m,n]$ (1024×1024)

Log-magnitude spectrum $\log(1 + |F_c(u,v)|)$

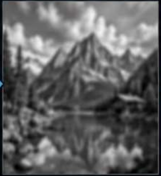
Low frequencies are concentrated near the center; high frequencies lie farther from the center.

2) Ideal low-pass filter

Mask $H_{LP}(u,v)$



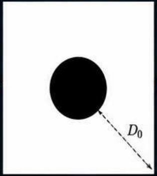
Filtered image $g_{LP}[m,n]$



- Keeps coarse (low-frequency) structure.
- Removes fine detail and texture.
- Sharp cutoff in frequency (ringing may occur).

3) Ideal high-pass filter

Mask $H_{HP}(u,v)$



Filtered image $g_{HP}[m,n]$

- Suppresses DC and low frequencies.
- Emphasizes edges, fine detail, and texture.
- Can reduce smooth background variations.

Takeaway: Ideal filters are simple binary frequency masks: low-pass keeps a centered disk of low frequencies, while high-pass removes it and preserves detail-rich high frequencies.

Ideal Band-Stop and Band-Pass Filters

Ring-shaped binary masks reject or preserve a selected band of spatial frequencies

Centered radial frequency distance

$$D(u,v) = \sqrt{\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2}$$

Ideal band-reject (band-stop) filter

$$H_{BR}(u,v) = \begin{cases} 0, & D_1 \le D(u,v) \propto D_2 \\ 1, & \text{otherwise} \end{cases}$$

Ideal band-pass filter

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

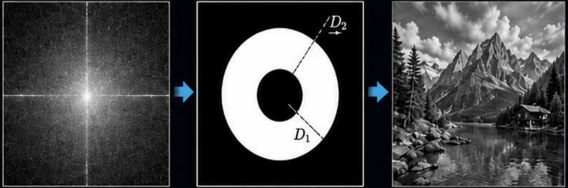
Frequency-domain filtering

$$G(u,v) = H(u,v)\, F(u,v)$$

$$g[m,n] = \mathrm{IDFT}_2\{G(u,v)\}$$

1) Ideal band-reject (band-stop) filter

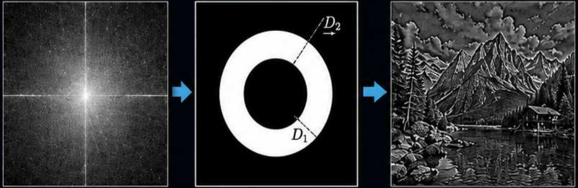
Input spectrum $F(u,v)$ | Ring-shaped binary mask $H_{BR}(u,v)$ | Filtered image $g_{BR}[m,n]$



- Rejects a band of mid frequencies between D_1 and D_2.
- Preserves very low frequencies (illumination, coarse structure) and very high frequencies (fine detail and edges).
- Useful for removing repetitive textures or specific frequency bands (e.g., moiré).

2) Ideal band-pass filter

Input spectrum $F(u,v)$ | Ring-shaped binary mask $H_{BP}(u,v)$ | Filtered image $g_{BP}[m,n]$



- Passes only the band of frequencies between D_1 and D_2.
- Suppresses very low and very high frequencies.
- Emphasizes details and textures within a selected range of scales.

Note on ringing: Because the mask has abrupt discontinuities, the spatial impulse response oscillates and ringing artifacts may appear near edges.

Sharp cutoff in frequency → oscillatory (sinc-like) response in space

Butterworth Filters in the 2-D Frequency Domain

Smooth frequency-domain filters replace the hard cutoff of ideal masks

Distance from the frequency center

$$D(u,v) = \sqrt{\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2}$$

Butterworth low-pass filter

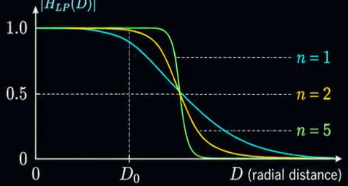
$$H_{LP}(u,v) = \frac{1}{1 + \left(\frac{D(u,v)}{D_0}\right)^{2n}}$$

Butterworth high-pass filter

$$H_{HP}(u,v) = 1 - H_{LP}(u,v) = \frac{1}{1 + \left(\frac{D_0}{D(u,v)}\right)^{2n}}$$

D_0 : cutoff radius
n : filter order (positive integer)

1) Concept: radial transfer magnitude

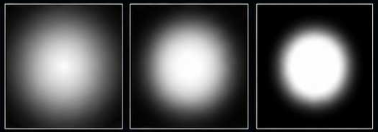


Higher order n gives a sharper transition around the cutoff radius D_0.

2) Butterworth low-pass masks (varying order n)

$H_{LP}(u,v)$ magnitude

$n = 1$ | $n = 2$ | $n = 5$



Smoother (gentle roll-off) → Sharper (steeper roll-off)

Same cutoff radius D_0 (brighter region passes)

3) Butterworth high-pass masks (varying order n)

$H_{HP}(u,v)$ magnitude

$n = 1$ | $n = 2$ | $n = 5$

Smoother (gentle roll-off) → Sharper (steeper roll-off)

Same cutoff radius D_0 (darker region passes)

4) Interpretation: why Butterworth filters are useful

- Butterworth filters provide a smooth transition between passband and stopband in the frequency domain.
- The order n controls the steepness of the transition: higher n approximates the ideal (but never truly binary).
- They generally produce less ringing in the spatial domain compared to ideal filters with the same cutoff.
- The transition is smooth rather than abrupt, reducing sensitivity to small shifts in the cutoff frequency.

Takeaway: Butterworth filters offer a tunable, smooth alternative to ideal filters—choose the order n to balance sharpness of transition and spatial-domain ringing.

Butterworth Band-Reject and Band-Pass Filters

Smooth ring-shaped transfer functions attenuate or preserve a selected frequency band

Butterworth band-reject filter

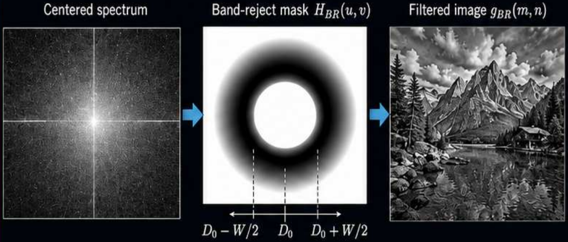
$$H_{BR}(u,v) = \frac{1}{1 + \left(\frac{D(u,v)\,W}{D(u,v)^2 - D_0^2}\right)^{2n}}$$

Butterworth band-pass filter

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

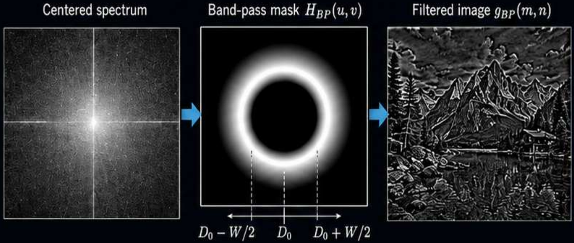
D_0 : center radius of the rejected band
W : bandwidth (thickness of the band)
n : order (controls transition steepness)

Butterworth band-reject filter



- Smoothly suppresses frequencies in an annular band.
- Effective for removing periodic patterns, ripples, and repetitive texture/noise.
- Preserves both low and high frequencies.

Butterworth band-pass filter



- Smoothly passes frequencies in an annular band.
- Keeps structures and details within a selected frequency range.
- Useful for isolating texture scales or specific pattern frequencies.

Note: Because the transition is gradual, Butterworth band filters usually produce less ringing (fewer oscillations in the spatial domain) than ideal binary ring filters.

Ideal versus Butterworth Filtering

Hard cutoffs are simple but ring: smooth transitions reduce artifacts

Low-pass transfer functions

Ideal: $H_{\text{ideal}}(u,v) = \begin{cases} 1, & D(u,v) \le D_0 \\ 0, & \text{otherwise} \end{cases}$

Butterworth (order n): $H_{\text{BW}}^{\text{LP}}(u,v) = \dfrac{1}{1+\left(\frac{D(u,v)}{D_0}\right)^{2n}}$

High-pass transfer functions

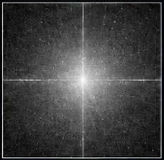
Ideal: $H_{\text{ideal}}^{\text{HP}}(u,v) = \begin{cases} 0, & D(u,v) \le D_0 \\ 1, & \text{otherwise} \end{cases}$

Butterworth (order n): $H_{\text{BW}}^{\text{HP}}(u,v) = \dfrac{1}{1+\left(\frac{D_0}{D(u,v)}\right)^{2n}}$

Common input

Input image $f[m,n]$ (1024×1024)

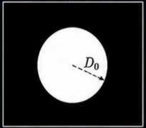
Centered log-magnitude spectrum $\log(1+|F(u,v)|)$



LOW-PASS COMPARISON

Ideal Low-Pass (cutoff D_0)

Mask $H_{LP}(u,v)$

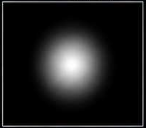


Filtered image $g[m,n]$

- Hard cutoff → ringing (Gibbs phenomenon).
- Preserves sharp edges but creates halos.
- More visible artifacts near high-contrast edges.

Butterworth Low-Pass (order $n=4$, D_0)

Mask $H_{LP}(u,v)$



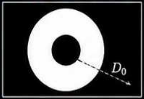
Filtered image $g[m,n]$

- Smooth transition reduces ringing.
- Edges remain clear but without halos.
- Order n controls sharpness of transition.

HIGH-PASS COMPARISON

Ideal High-Pass (cutoff D_0)

Mask $H_{HP}(u,v)$

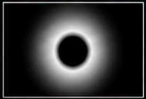


Filtered image $g[m,n]$

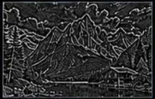
- Hard cutoff → ringing around edges.
- Emphasizes edges strongly (but with halos).
- Sensitive to noise (edges look noisy).

Butterworth High-Pass (order $n=4$, D_0)

Mask $H_{HP}(u,v)$



Filtered image $g[m,n]$



- Smooth transition reduces ringing and noise.
- Edges enhanced without harsh halos.
- Cleaner, more natural high-frequency emphasis.

Pros & Cons

Ideal Filters

- Simple, exact binary support.
- Maximally sharp cutoff.

- Strong ringing in spatial domain.
- Artifacts around edges.
- Not robust to noise near cutoff.

Butterworth Filters

- Tunable order n for trade-off.
- Smooth transition band.
- Less ringing, cleaner results.

- Slightly less selective near cutoff. (transition band not zero)

Takeaway: The choice of transfer function $H(u,v)$ controls both the frequencies you keep and the artifacts you see in space. Butterworth filters are usually preferred in practice for smoother, more natural results with less ringing.

Ideal Filters in the 2-D Frequency Domain

Binary masks in the spectrum keep some frequencies and reject the rest

Centered radial frequency distance

$$D(u,v) = \sqrt{\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2}$$

Ideal low-pass filter

$$H_{LP}(u,v) = \begin{cases} 1, & D(u,v) \le D_0 \\ 0, & \text{otherwise} \end{cases}$$

Ideal high-pass filter

$$H_{HP}(u,v) = 1 - H_{LP}(u,v)$$

Frequency-domain filtering

$$G(u,v) = H(u,v)\,F(u,v)$$

$$g[m,n] = \mathrm{IDFT}_2\{G(u,v)\}$$

1) Input image and centered spectrum

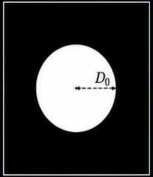
Input image $f[m,n]$ (1024×1024)

Log-magnitude spectrum $\log(1 + |F_c(u,v)|)$

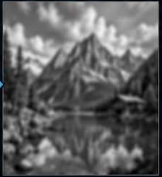
Low frequencies are concentrated near the center; high frequencies lie farther from the center.

2) Ideal low-pass filter

Mask $H_{LP}(u,v)$



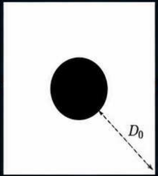
Filtered image $g_{LP}[m,n]$



- Keeps coarse (low-frequency) structure.
- Removes fine detail and texture.
- Sharp cutoff in frequency (ringing may occur).

3) Ideal high-pass filter

Mask $H_{HP}(u,v)$



Filtered image $g_{HP}[m,n]$

- Suppresses DC and low frequencies.
- Emphasizes edges, fine detail, and texture.
- Can reduce smooth background variations.

Takeaway: Ideal filters are simple binary frequency masks: low-pass keeps a centered disk of low frequencies, while high-pass removes it and preserves detail-rich high frequencies.

Ideal Band-Stop and Band-Pass Filters

Ring-shaped binary masks reject or preserve a selected band of spatial frequencies

Centered radial frequency distance

$$D(u,v) = \sqrt{\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2}$$

Ideal band-reject (band-stop) filter

$$H_{BR}(u,v) = \begin{cases} 0, & D_1 \le D(u,v) < D_2 \\ 1, & \text{otherwise} \end{cases}$$

Ideal band-pass filter

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

Frequency-domain filtering

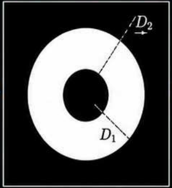
$$G(u,v) = H(u,v)\, F(u,v)$$

$$g[m,n] = \mathrm{IDFT}_2\{G(u,v)\}$$

1) Ideal band-reject (band-stop) filter

Input spectrum $F(u,v)$

Ring-shaped binary mask $H_{BR}(u,v)$

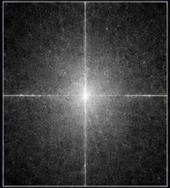


Filtered image $g_{BR}[m,n]$

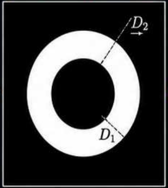
- Rejects a band of mid frequencies between D_1 and D_2.
- Preserves very low frequencies (illumination, coarse structure) and very high frequencies (fine detail and edges).
- Useful for removing repetitive textures or specific frequency bands (e.g., moiré).

2) Ideal band-pass filter

Input spectrum $F(u,v)$



Ring-shaped binary mask $H_{BP}(u,v)$



Filtered image $g_{BP}[m,n]$

- Passes only the band of frequencies between D_1 and D_2.
- Suppresses very low and very high frequencies.
- Emphasizes details and textures within a selected range of scales.

Note on ringing: Because the mask has abrupt discontinuities, the spatial impulse response oscillates and ringing artifacts may appear near edges.

Sharp cutoff in frequency → oscillatory (sinc-like) response in space

Butterworth Filters in the 2-D Frequency Domain

Smooth frequency-domain filters replace the hard cutoff of ideal masks

Distance from the frequency center

$$D(u,v) = \sqrt{\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2}$$

Butterworth low-pass filter

$$H_{LP}(u,v) = \frac{1}{1 + \left(\frac{D(u,v)}{D_0}\right)^{2n}}$$

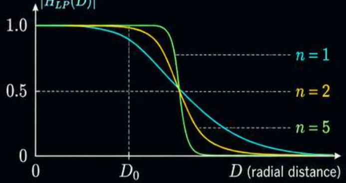
Butterworth high-pass filter

$$H_{HP}(u,v) = 1 - H_{LP}(u,v) = \frac{1}{1 + \left(\frac{D_0}{D(u,v)}\right)^{2n}}$$

D_0 : cutoff radius

n : filter order (positive integer)

1) Concept: radial transfer magnitude

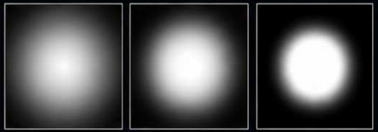


Higher order n gives a sharper transition around the cutoff radius D_0.

2) Butterworth low-pass masks (varying order n)

$H_{LP}(u,v)$ magnitude

$n = 1$ $n = 2$ $n = 5$



Smoother (gentle roll-off) → Sharper (steeper roll-off)

Same cutoff radius D_0 (brighter region passes)

3) Butterworth high-pass masks (varying order n)

$H_{HP}(u,v)$ magnitude

$n = 1$ $n = 2$ $n = 5$

Smoother (gentle roll-off) → Sharper (steeper roll-off)

Same cutoff radius D_0 (darker region passes)

4) Interpretation: why Butterworth filters are useful

- Butterworth filters provide a smooth transition between passband and stopband in the frequency domain.
- The order n controls the steepness of the transition: higher n approximates the ideal (but never truly binary).
- They generally produce less ringing in the spatial domain compared to ideal filters with the same cutoff.
- The transition is smooth rather than abrupt, reducing sensitivity to small shifts in the cutoff frequency.

Takeaway: Butterworth filters offer a tunable, smooth alternative to ideal filters—choose the order n to balance sharpness of transition and spatial-domain ringing.

Butterworth Band-Reject and Band-Pass Filters

Smooth ring-shaped transfer functions attenuate or preserve a selected frequency band

Butterworth band-reject filter

$$H_{BR}(u,v) = \frac{1}{1 + \left(\dfrac{D(u,v)\, W}{D(u,v)^2 - D_0^2} \right)^{2n}}$$

Butterworth band-pass filter

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

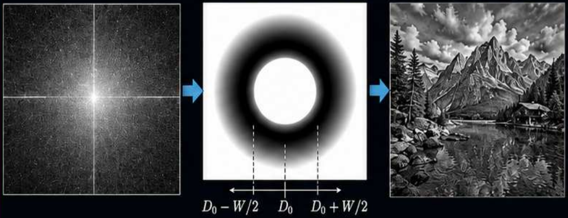
D_0 : center radius of the rejected band

W : bandwidth (thickness of the band)

n : order (controls transition steepness)

Butterworth band-reject filter

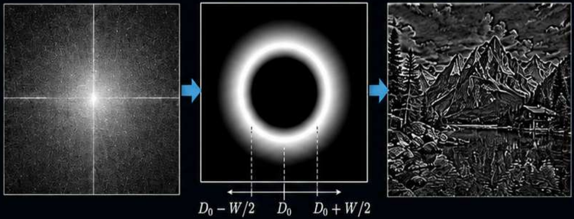
Centered spectrum | Band-reject mask $H_{BR}(u,v)$ | Filtered image $g_{BR}(m,n)$



- Smoothly suppresses frequencies in an annular band.
- Effective for removing periodic patterns, ripples, and repetitive texture/noise.
- Preserves both low and high frequencies.

Butterworth band-pass filter

Centered spectrum | Band-pass mask $H_{BP}(u,v)$ | Filtered image $g_{BP}(m,n)$



- Smoothly passes frequencies in an annular band.
- Keeps structures and details within a selected frequency range.
- Useful for isolating texture scales or specific pattern frequencies.

Note: Because the transition is gradual, Butterworth band filters usually produce less ringing (fewer oscillations in the spatial domain) than ideal binary ring filters.

Ideal versus Butterworth Filtering

Hard cutoffs are simple but ring: smooth transitions reduce artifacts

Low-pass transfer functions

Ideal: $H_{\text{ideal}}(u,v) = \begin{cases} 1, & D(u,v) \le D_0 \\ 0, & \text{otherwise} \end{cases}$

Butterworth (order n): $H_{\text{BW}}^{\text{LP}}(u,v) = \dfrac{1}{1+\left(\frac{D(u,v)}{D_0}\right)^{2n}}$

High-pass transfer functions

Ideal: $H_{\text{ideal}}^{\text{HP}}(u,v) = \begin{cases} 0, & D(u,v) \le D_0 \\ 1, & \text{otherwise} \end{cases}$

Butterworth (order n): $H_{\text{BW}}^{\text{HP}}(u,v) = \dfrac{1}{1+\left(\frac{D_0}{D(u,v)}\right)^{2n}}$

Common input

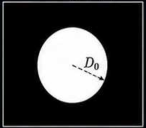
Input image $f[m,n]$ (1024×1024)

Centered log-magnitude spectrum $\log(1+|F(u,v)|)$

LOW-PASS COMPARISON

Ideal Low-Pass (cutoff D_0)

Mask $H_{LP}(u,v)$

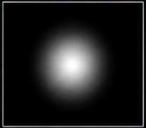


Filtered image $g[m,n]$

- Hard cutoff → ringing (Gibbs phenomenon).
- Preserves sharp edges but creates halos.
- More visible artifacts near high-contrast edges.

Butterworth Low-Pass (order $n=4$, D_0)

Mask $H_{LP}(u,v)$



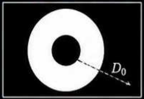
Filtered image $g[m,n]$

- Smooth transition reduces ringing.
- Edges remain clear but without halos.
- Order n controls sharpness of transition.

HIGH-PASS COMPARISON

Ideal High-Pass (cutoff D_0)

Mask $H_{HP}(u,v)$

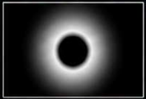


Filtered image $g[m,n]$

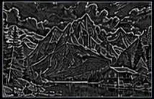
- Hard cutoff → ringing around edges.
- Emphasizes edges strongly (but with halos).
- Sensitive to noise (edges look noisy).

Butterworth High-Pass (order $n=4$, D_0)

Mask $H_{HP}(u,v)$



Filtered image $g[m,n]$



- Smooth transition reduces ringing and noise.
- Edges enhanced without harsh halos.
- Cleaner, more natural high-frequency emphasis.

Pros & Cons

Ideal Filters

- Simple, exact binary support.
- Maximally sharp cutoff.

- Strong ringing in spatial domain.
- Artifacts around edges.
- Not robust to noise near cutoff.

Butterworth Filters

- Tunable order n for trade-off.
- Smooth transition band.
- Less ringing, cleaner results.

- Slightly less selective near cutoff. (transition band not zero)

Takeaway: The choice of transfer function $H(u,v)$ controls both the frequencies you keep and the artifacts you see in space. Butterworth filters are usually preferred in practice for smoother, more natural results with less ringing.