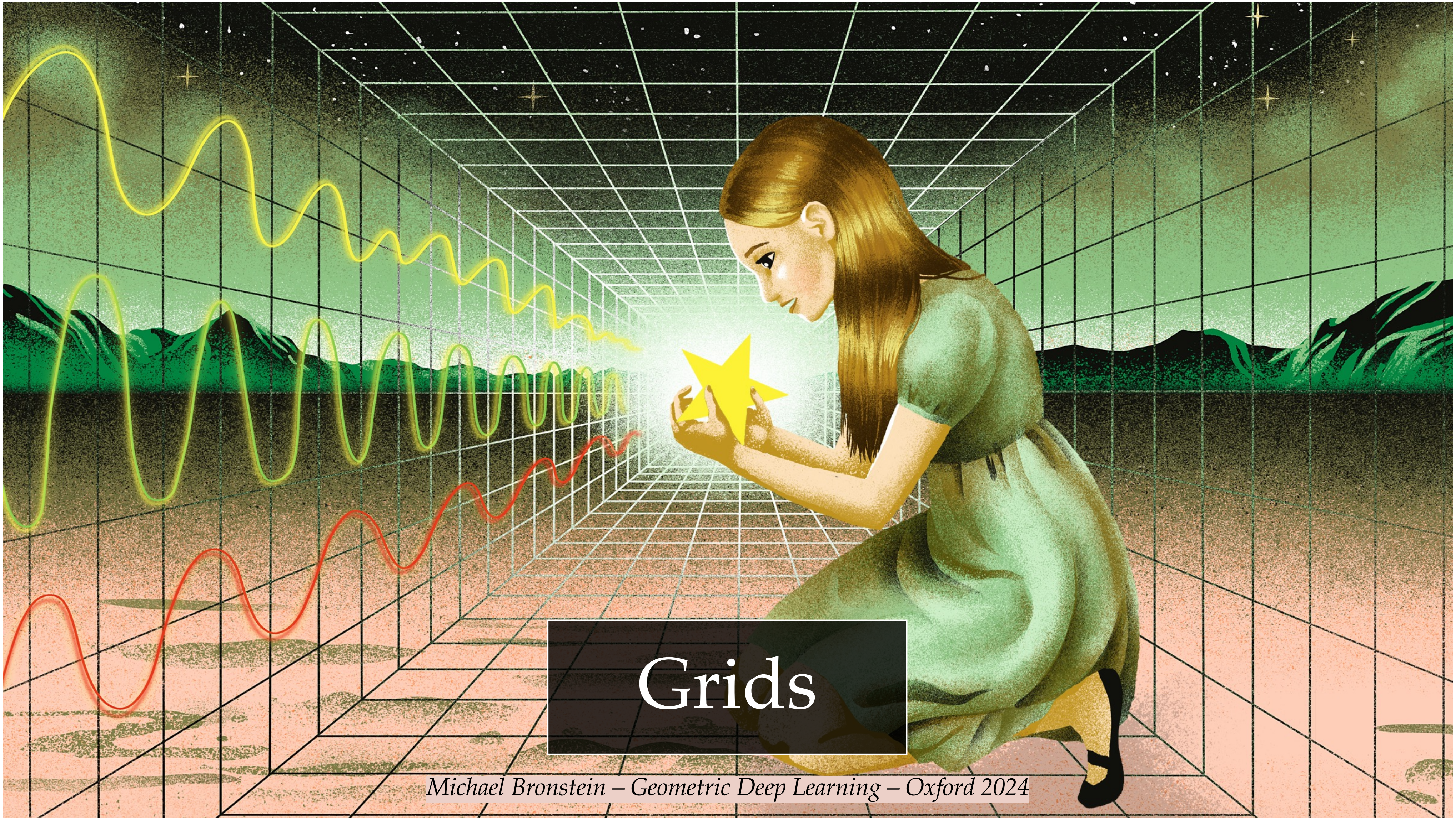
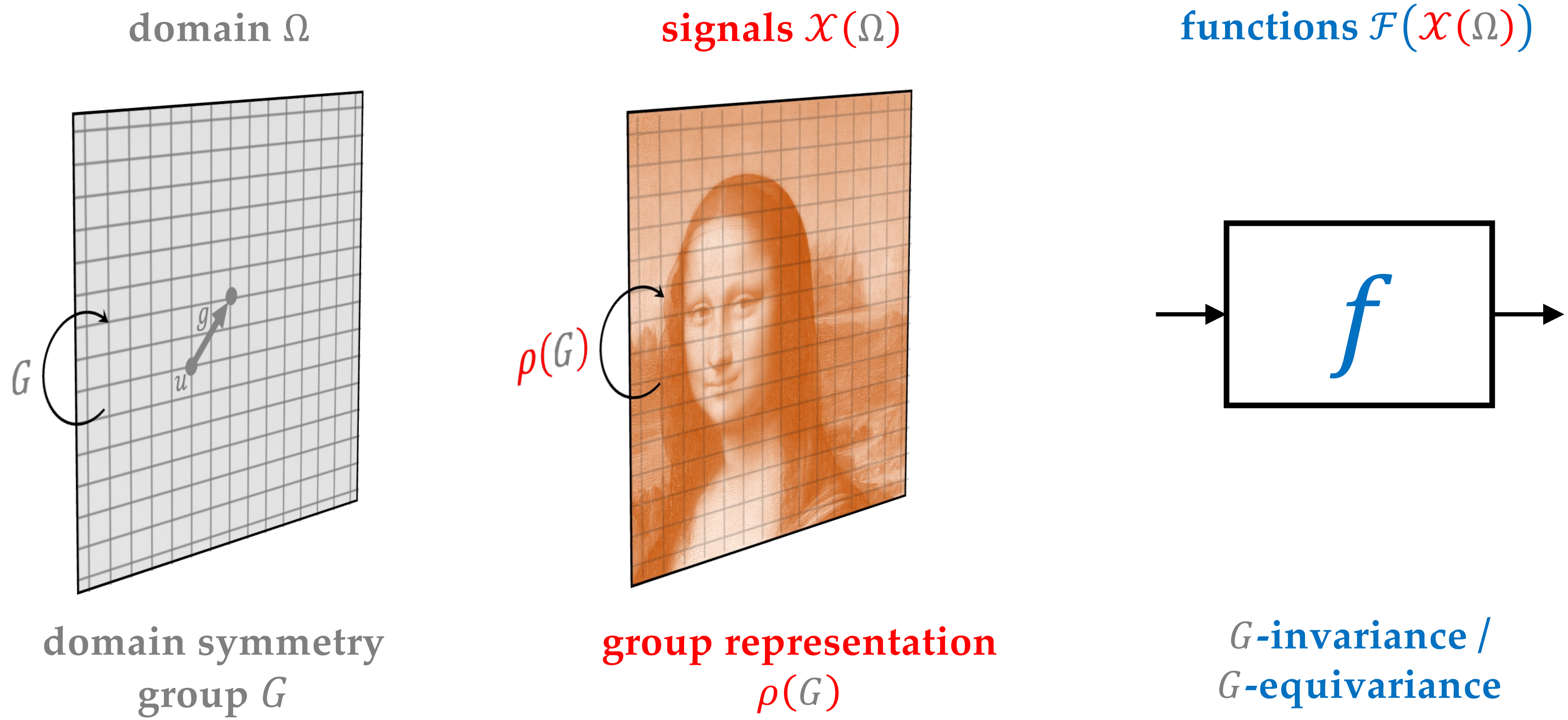

Grids
Michael Bronstein – Geometric Deep Learning – Oxford 2024

Outline

- *Grids* as special case of graphs, providing more structure
- *Translation symmetry*
- *Convolutions* are linear translation-equivariants
- *Fourier transform* diagonalises convolution operators and offers a spectral view dual to the spatial one
- *Geometric stability* and the need for multiscale representation (*wavelets*)
- *Wavelet Scattering* as a simple instance of the Geometric Deep Learning blueprint
- *Convolutional Neural Networks*

Geometric Deep Learning Blueprint



Popular architectures as instances of the Blueprint

Architecture	**Domain Ω**	**Symmetry Group G**
CNN	Grid	Translation
Spherical CNN	Sphere / SO(3)	Rotation SO(3)
Intrinsic / Mesh CNN	Manifold / Mesh	Isometry Iso(Ω) / Gauge Symmetry SO(2)
GNN	Graph	Permutation S_n
Deep Sets	Set	Permutation S_n
Transformer	Complete Graph	Permutation S_n
LSTM	1D Grid	Time warping

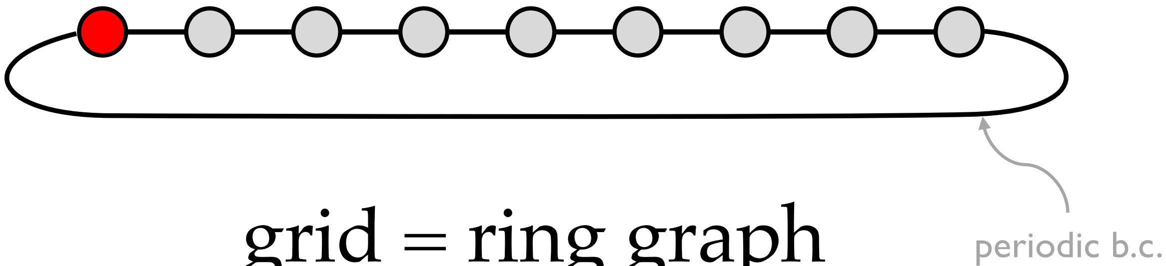
Geometric Deep Learning Building Blocks

- **Linear equivariant:** $B: \mathcal{X}(\Omega) \rightarrow \mathcal{X}'(\Omega)$ satisfying $B(gx) = gB(x)$
- **Nonlinearity:** $\sigma: \mathcal{X} \rightarrow \mathcal{X}$ applied element-wise, $(\sigma x)(u) = \sigma\big(x(u)\big)$
- **Local pooling (coarsening):** $P: \mathcal{X}(\Omega) \rightarrow \mathcal{X}\big(\widetilde{\Omega}\big)$
- **Invariant layer (global pooling):** $A: \mathcal{X} \rightarrow \mathcal{Y}$ satisfying $A(gx) = A(x)$

Geometric Deep Learning Building Blocks

- **Linear equivariant:** $B: \mathcal{X}(\Omega) \rightarrow \mathcal{X}'(\Omega)$ satisfying $B(gx) = gB(x)$
- **Nonlinearity:** $\sigma: \mathcal{X} \rightarrow \mathcal{X}$ applied element-wise, $(\sigma x)(u) = \sigma\big(x(u)\big)$
- **Local pooling (coarsening):** $P: \mathcal{X}(\Omega) \rightarrow \mathcal{X}(\widetilde{\Omega})$
- **Invariant layer (global pooling):** $A: \mathcal{X} \rightarrow \mathcal{Y}$ satisfying $A(gx) = A(x)$

Grid = Graph

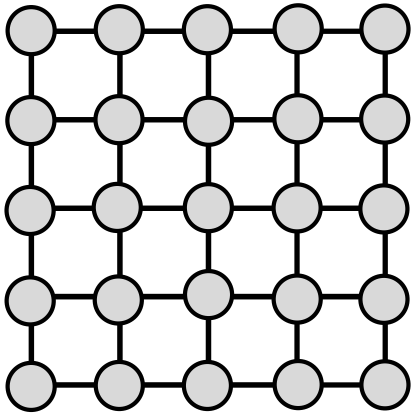


grid = ring graph

Translation on Grid with periodic b.c. = Cyclic group

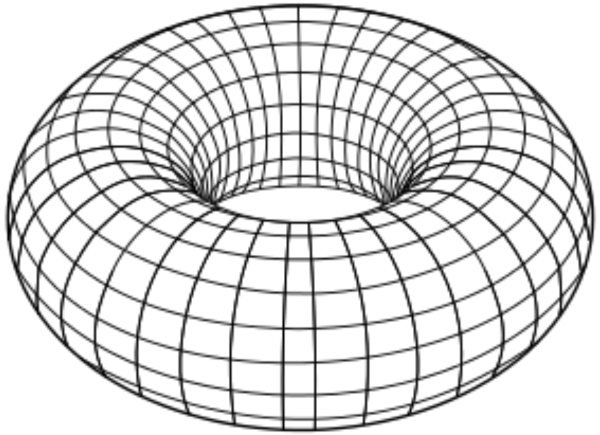
- $\Omega = \{0, \dots, n-1\} = \mathbb{Z}_n$
- 1D translations with periodic boundary conditions for the *cyclic group* $G = Z_n$
- Modulo n arithmetics:
  - $g: \Omega \to \Omega$: $u \mapsto (u+1) \bmod n$
  - $g^{-1}: \Omega \to \Omega$: $u \mapsto (u-1) \bmod n$
  - $g^k: \Omega \to \Omega$: $u \mapsto (u+k) \bmod n$
- Note that $\Omega \cong G$ (i.e., every translation group element can be identified with a point on the domain). This is *exception* rather than rule!

Multidimensional Grids

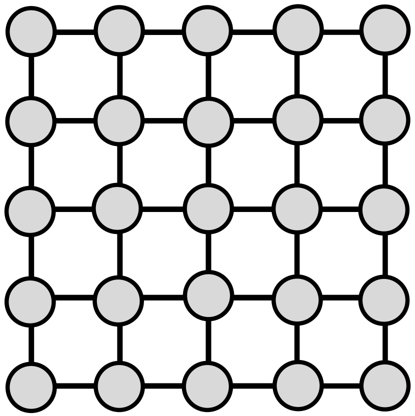


2-dimensional

$$\Omega = \{0, \dots, n-1\} \otimes \{0, \dots, n-1\}$$
$$G = Z_n \times Z_n$$

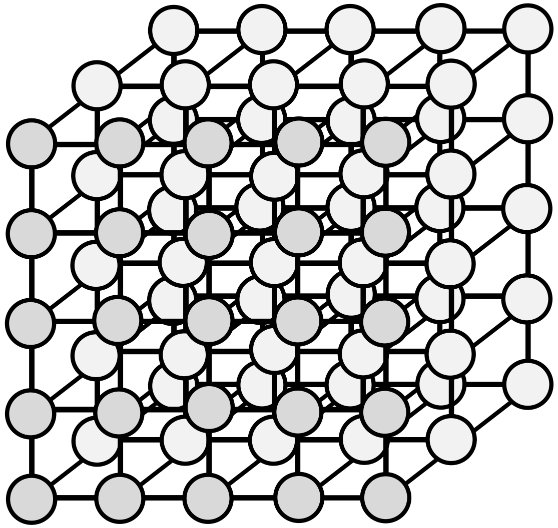


Multidimensional Grids



2-dimensional

$$\Omega = \{0, \dots, n-1\} \otimes \{0, \dots, n-1\}$$
$$G = Z_n \times Z_n$$



***d*-dimensional**

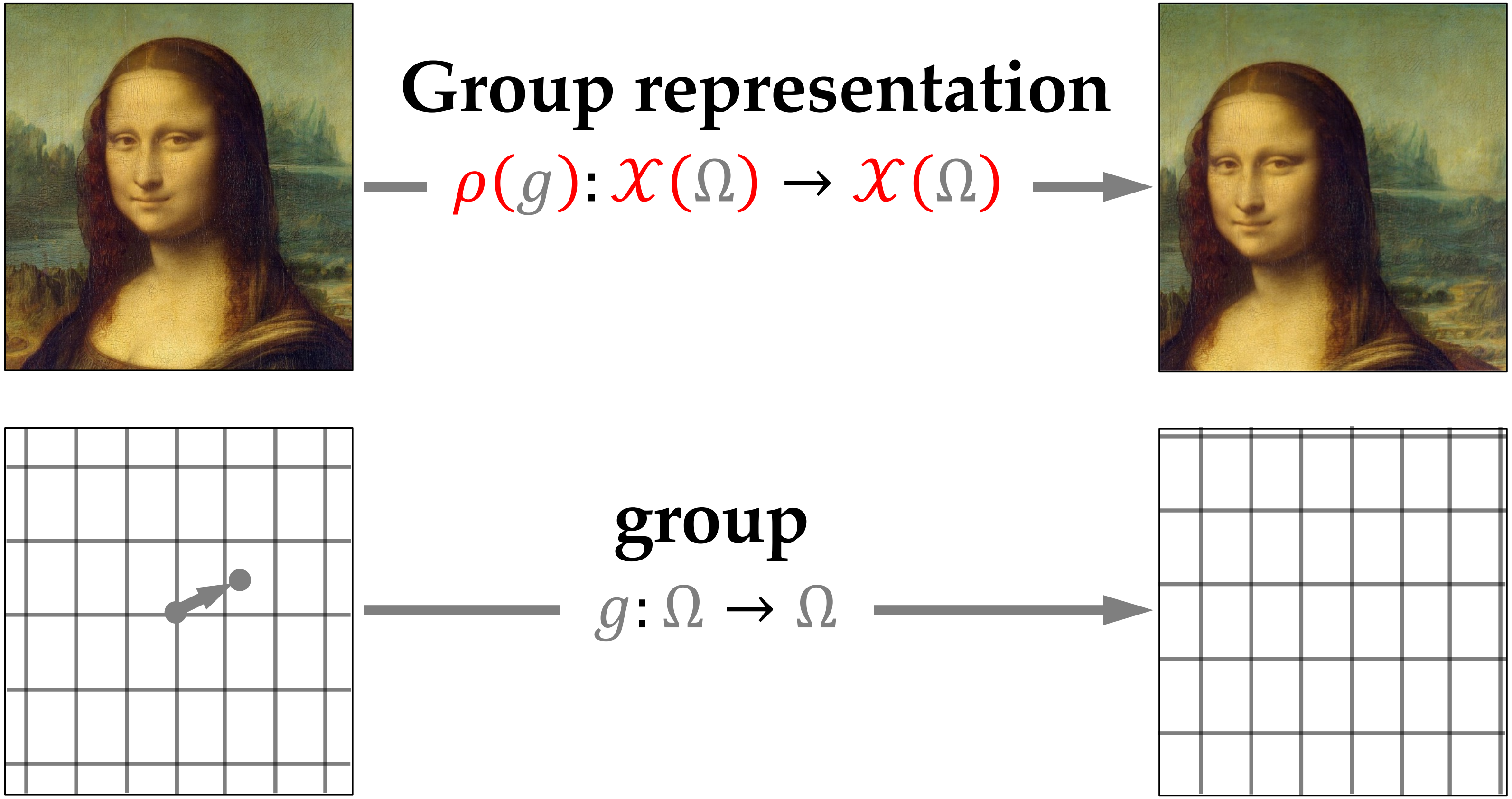
$$\Omega = \{0, \dots, n-1\}^{\otimes d}$$
$$G = Z_n \times \ \dots \ \times Z_n$$

Signals on Grids

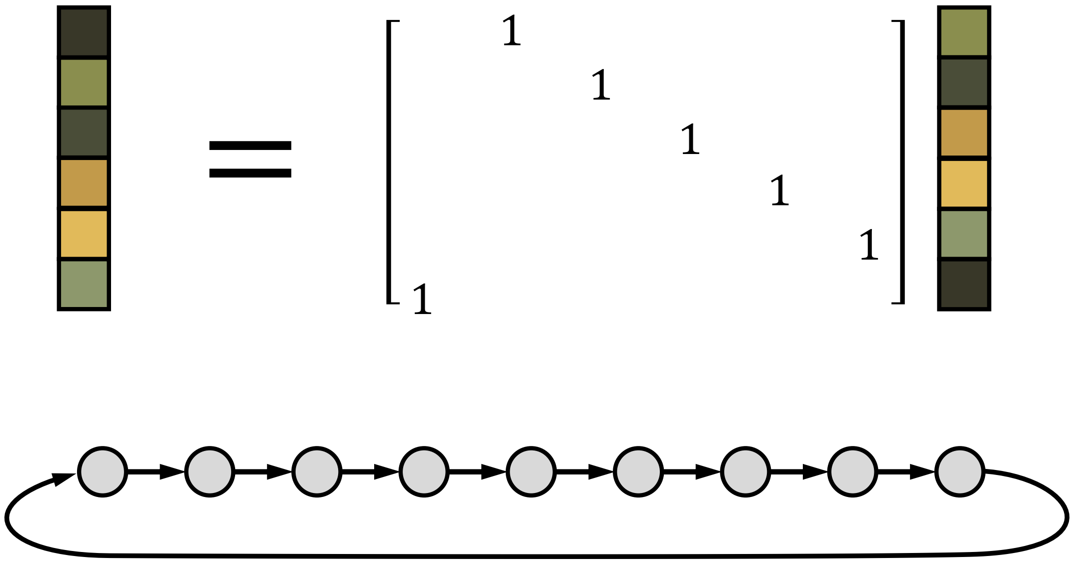
- Space of signals on 1D grid: $\mathcal{X}(\Omega) = \{x: \Omega \to \mathbb{R}\} \cong \mathbb{R}^n$
- Represented as n-dimensional vectors $\mathbf{x} = (x_0, \ldots, x_{n-1})$

Note: in this lecture, for convenience we will be using indices $0, \ldots, n-1$

“Lifting”


Group representation
$\rho(g) : \mathcal{X}(\Omega) \to \mathcal{X}(\Omega)$
group
$g : \Omega \to \Omega$

Shift operator



Note: Observe that the adjacency matrix of the directed ring graph is exactly the shift operator. We will use it when defining graph convolutions.

Shift operator

$$\mathbf{S} = \begin{bmatrix} & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \\ 1 & & & & & \end{bmatrix}$$

Shift operator

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_0 \end{bmatrix} = \begin{bmatrix} & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \\ 1 & & & & & \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

"Left shift"

Shift operator

$$\mathbf{S}^2 = \begin{bmatrix} & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \\ 1 & & & & & \\ & 1 & & & & \end{bmatrix}$$

- For every $g \in Z_n$ its representation is given by $\rho(g) = \mathbf{S}^k$ for some k

Shift operator

$$
\mathbf{S}^{\mathrm{T}} = \begin{bmatrix} & & & & & 1 \\ 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \end{bmatrix}
$$

- For every $g \in Z_n$ its representation is given by $\rho(g) = \mathbf{S}^k$ for some k
- Inverse element: $\mathbf{S}^{-1} = \mathbf{S}^{\mathrm{T}}$

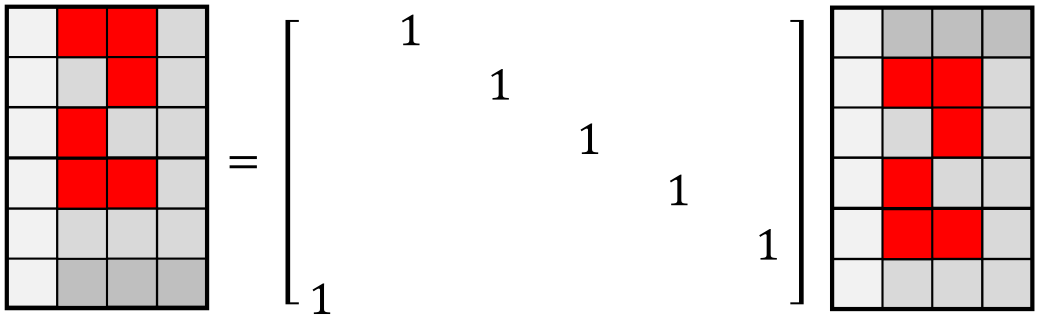
Shift operator

$$\begin{bmatrix} x_5 \\ x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} & & & & & 1 \\ 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$ "Right shift"

- For every $g \in Z_n$ its representation is given by $\rho(g) = \mathbf{S}^k$ for some k
- Inverse element: $\mathbf{S}^{-1} = \mathbf{S}^{\mathrm{T}}$
- $\mathbf{S}$ is an *orthogonal matrix* satisfying $\mathbf{S}^{\mathrm{T}}\mathbf{S} = \mathbf{S}\mathbf{S}^{\mathrm{T}} = \mathbf{I}$ (and consequently, as we will see, has orthogonal eigenvectors)

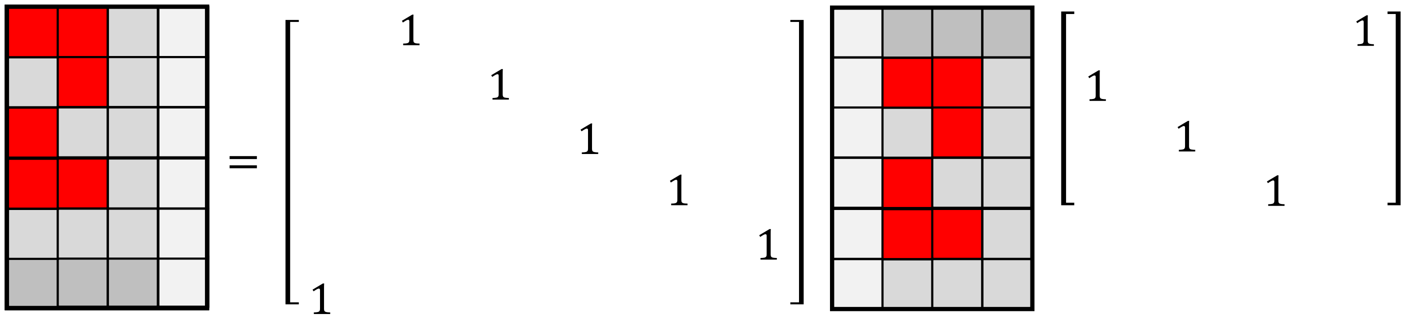
Exercise: prove that G is commutative, i.e., $gh = hg$.

2D Shift operator

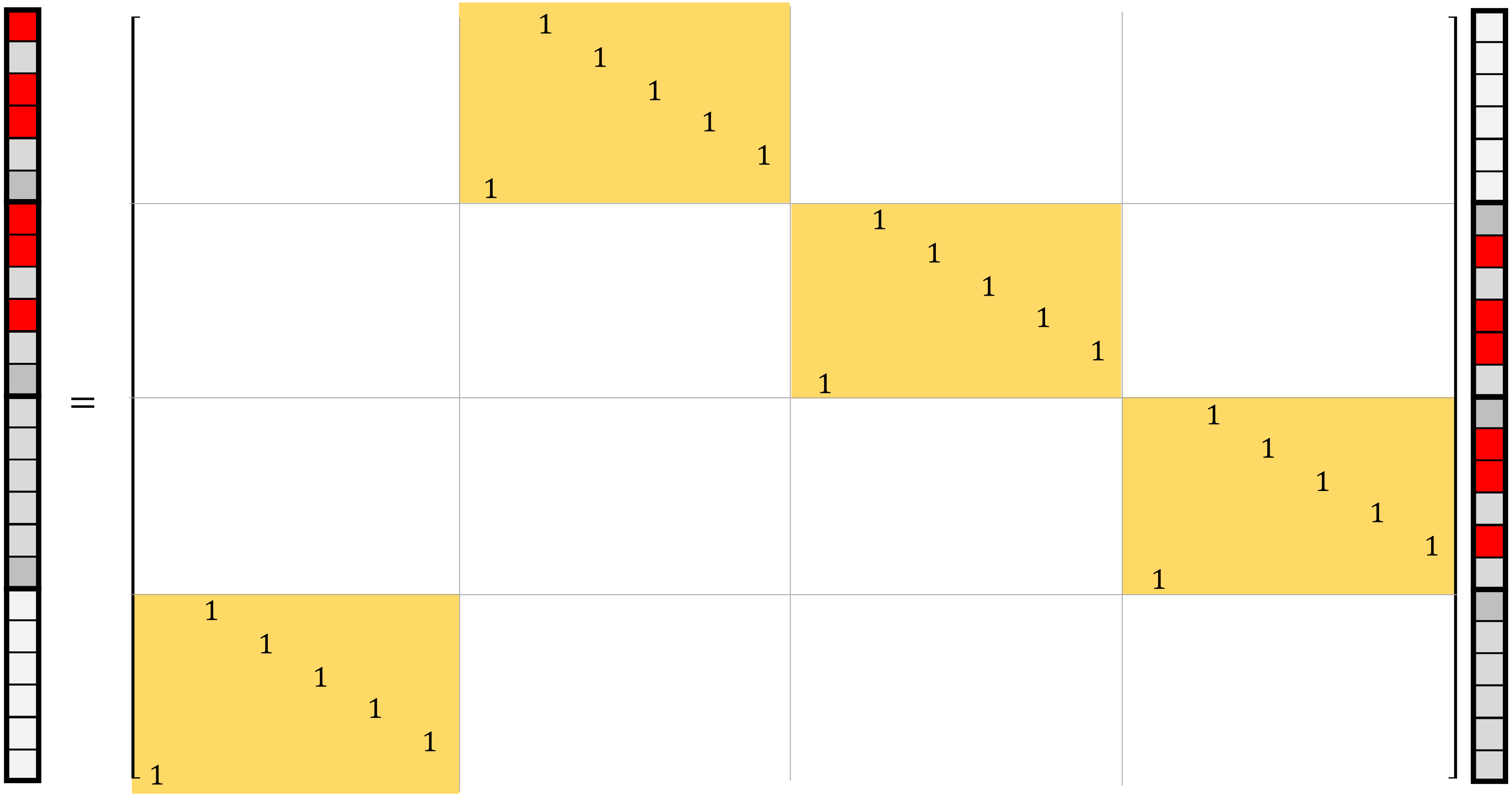


- Images $\mathcal{X}(\Omega) \cong \mathbb{R}^{n\times m}$
- Represented as $n\times m$-dimensional matrices $\mathbf{X}$
- Shifts can be applied from the *left* (row-wise or vertical)

2D Shift operator



- Images $\mathcal{X}(\Omega) \cong \mathbb{R}^{n\times m}$
- Represented as $n\times m$-dimensional matrices $\mathbf{X}$
- Shifts can be applied from the *left* (row-wise or vertical) or from the *right* (column-wise or horizontal): $\mathbf{Y} = \mathbf{S}^k\mathbf{X}\mathbf{S}^{-l}$
- Alternatively, stacking the image column-wise $\mathbf{x} = \mathrm{vec}(\mathbf{X})$ and using the *Kronecker product:* $\mathrm{vec}\left(\mathbf{S}^k\mathbf{X}\mathbf{S}^{-l}\right) = \left(\mathbf{S}^l \otimes \mathbf{S}^k\right)\mathrm{vec}(\mathbf{X})$


=
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1

Functions of Signals on Grids

- Space of **functions on grid signals** $\mathcal{F}\big(\mathcal{X}(\Omega)\big) = \{f : \mathcal{X}(\Omega) \to \mathcal{Y}\}$

Functions of Signals on Grids

- Space of **functions on grid signals** $\mathcal{F}(\mathcal{X}(\Omega)) = \{f: \mathbb{R}^n \to \mathcal{Y}\}$
- From the Geometric Deep Learning Blueprint, we know that it is sufficient to consider *local linear invariants* and *equivariants* (that can be then combined with element-wise nonlinearities and pooling)
  - *Linear invariants:* $f: \mathbb{R}^n \to \mathbb{R}$
  - *Linear equivariants:* $\mathbf{F}: \mathbb{R}^n \to \mathbb{R}^n$

Linear Invariants

- **Linear translation invariant** is a function of the form $f(\mathbf{x}) = \langle \mathbf{x}, \mathbf{v} \rangle$ for some $\mathbf{v} \in \mathbb{R}^n$, satisfying

$$\left\langle \mathbf{S}^k \mathbf{x}, \mathbf{v} \right\rangle = \langle \mathbf{x}, \mathbf{v} \rangle \text{ for all } \mathbf{x} \in \mathbb{R}^n \text{ and } k = 0, \dots, n-1$$

$\langle \mathbf{x}, \mathbf{v} \rangle$ is translation invariant iff $\mathbf{v} = \mathbf{1}$.

Exercise: prove that $\mathbf{v}$ must satisfy $\mathbf{S}\mathbf{v} = \mathbf{v}$.

Linear Equivariants

- **Linear translation equivariant** is a function of the form $\mathbf{F}(\mathbf{x}) = \mathbf{C}\mathbf{x}$ for some matrix $\mathbf{C} \in \mathbb{R}^{n \times n}$, satisfying

$$\mathbf{C}\mathbf{S}^k\mathbf{x} = \mathbf{S}^k\mathbf{C}\mathbf{x} \text{ for all } \mathbf{x} \in \mathbb{R}^n \text{ and } k = 0, \ldots, n-1$$

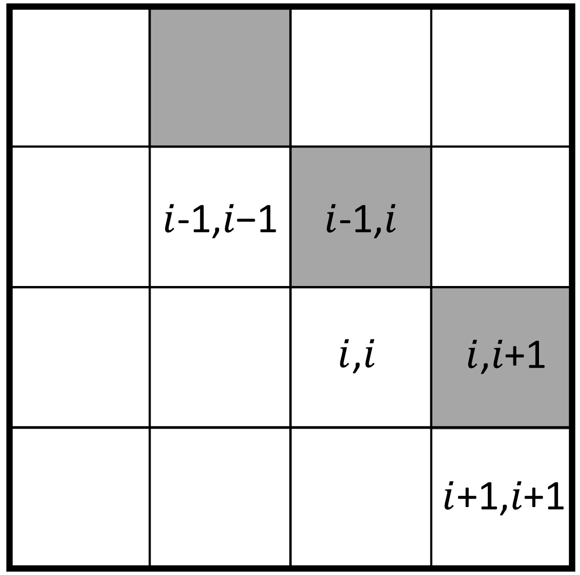
> $\mathbf{C}\mathbf{x}$ is translation-equivariant iff $\mathbf{C}$ commutes with the shift operator, $\mathbf{C}\mathbf{S} = \mathbf{S}\mathbf{C}$.

How do such matrices C look like?

CONVOLUTION

Linear Equivariants

$$(\mathbf{SC})_{ij} = \sum_{k=0}^{n-1} s_{i,k}\, c_{k,j}$$



$$s_{ik} = \begin{cases} 1 & k = i+1 \\ 0 & \text{else} \end{cases}$$

Note: all indices are mod n.

Linear Equivariants

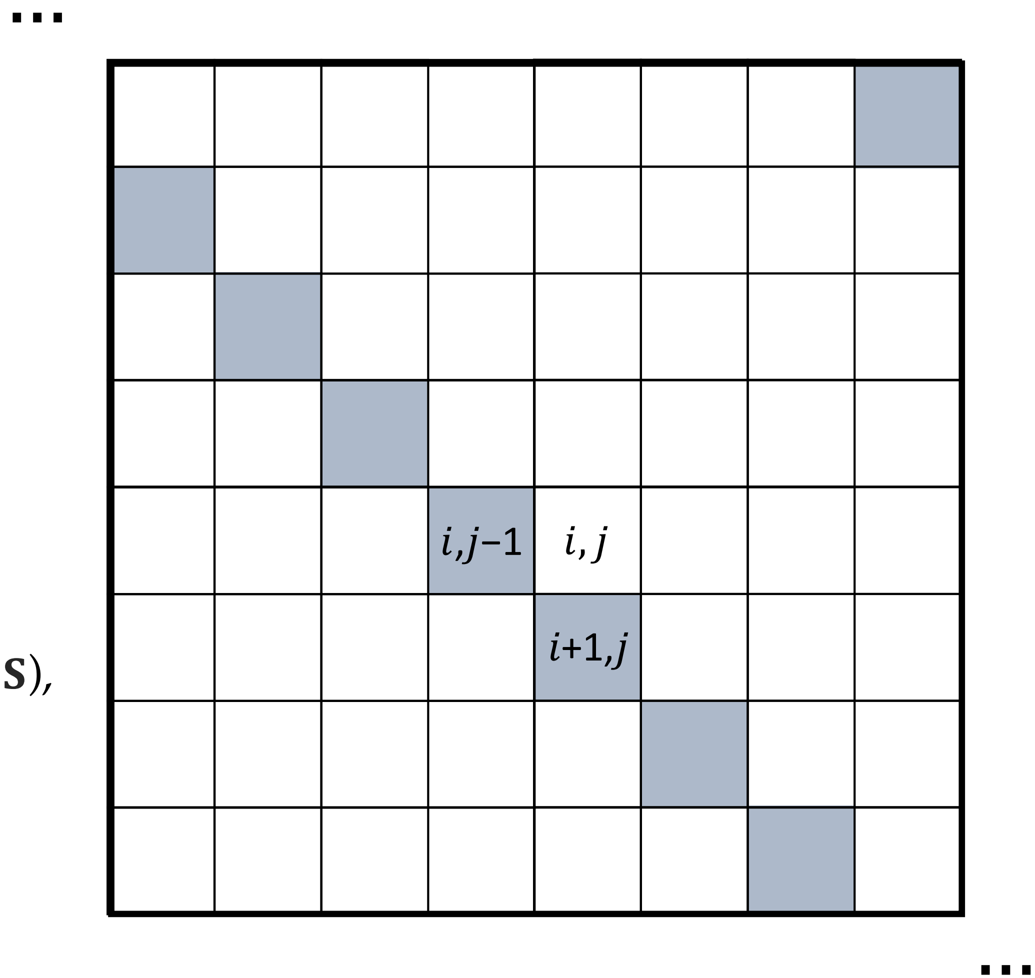
$$(\mathbf{SC})_{ij} = \sum_{k=0}^{n-1} \delta_{i-k+1}\ c_{k,j} = c_{i+1,j}$$

$$(\mathbf{CS})_{ij} = \sum_{k=0}^{n-1} c_{i,k}\ \delta_{k-j+1} = c_{i,j-1}$$

Since we assumed commutativity ($\mathbf{SC} = \mathbf{CS}$), we have

$$c_{i+1,j} = c_{i,j-1}$$

Note: all indices are mod n.



Linear Equivariants

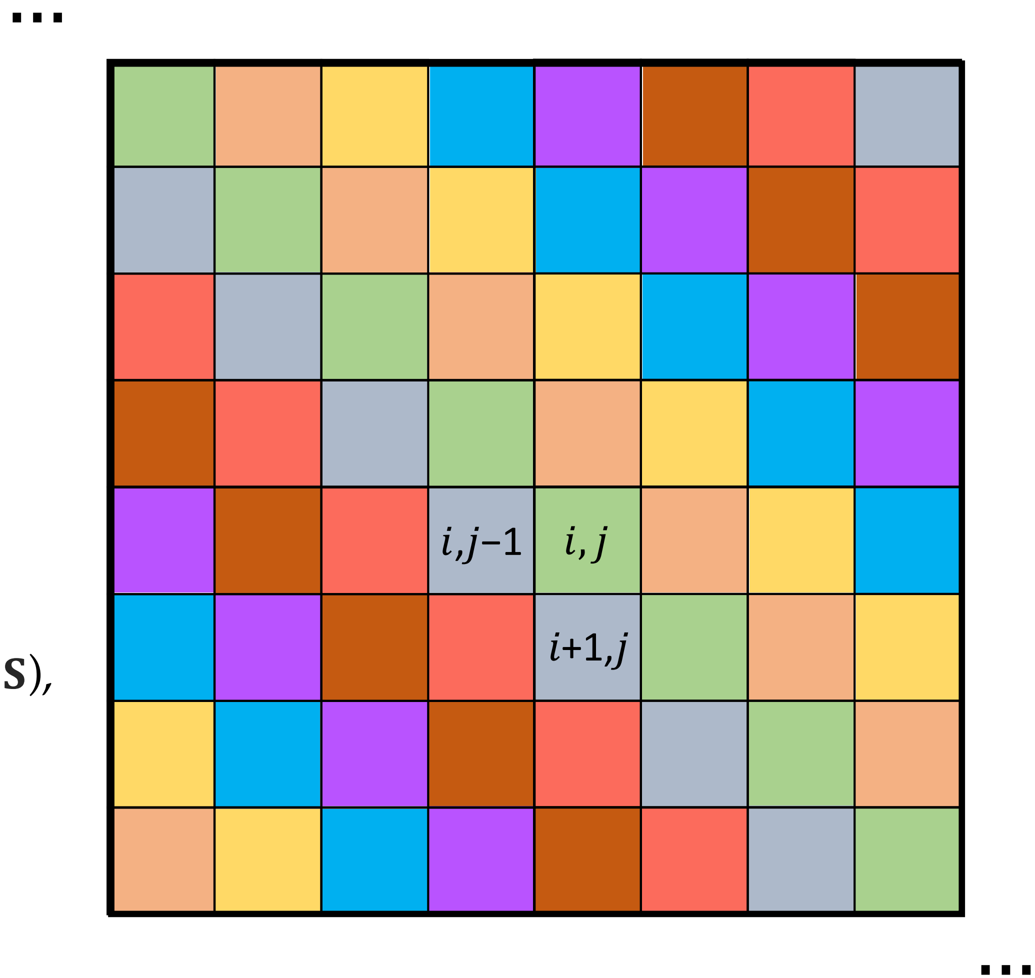
$$(\mathbf{SC})_{ij} = \sum_{k=0}^{n-1} \delta_{i-k+1}\ c_{k,j} = c_{i+1,j}$$

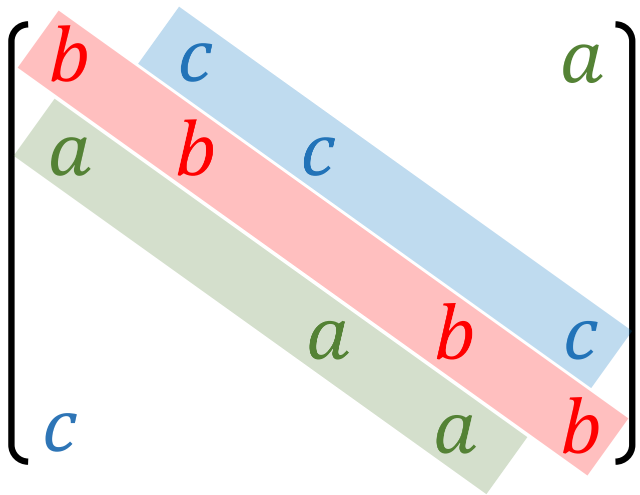
$$(\mathbf{CS})_{ij} = \sum_{k=0}^{n-1} c_{i,k}\ \delta_{k-j+1} = c_{i,j-1}$$

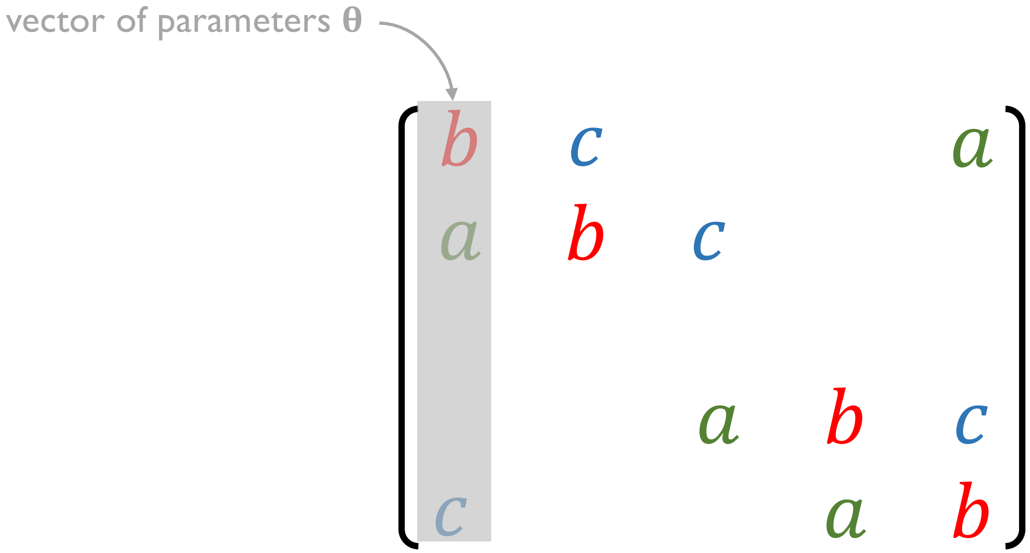
Since we assumed commutativity ($\mathbf{SC} = \mathbf{CS}$), we have

$$c_{i+1,j} = c_{i,j-1}$$

Note: all indices are mod n.




b c a
a b c
a b c
c a b



circulant matrix $\mathbf{C}(\boldsymbol{\theta})$

$$\mathbf{C}(\boldsymbol{\theta})\mathbf{x} = \begin{bmatrix} \theta_0 & \theta_{n-1} & & & \theta_1 \\ \theta_1 & \theta_0 & & & \\ & & & & \\ & & & \theta_0 & \\ \theta_{n-1} & & & & \theta_0 \end{bmatrix} \begin{bmatrix} \\ \\ \mathbf{x} \\ \\ \\ \end{bmatrix}$$

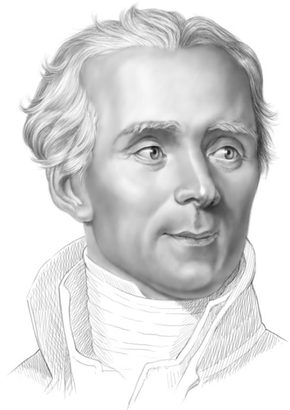
$$(\mathbf{C}(\boldsymbol{\theta})\mathbf{x})_i = \sum_{j=0}^{n-1} \theta_{i-j \bmod \mathrm{n}}\, x_j$$

Convolution

$$(\mathbf{x} \star \boldsymbol{\theta})_i = \sum_{j=0}^{n-1} \theta_{i-j \bmod n}\, x_j$$

Linear translation equivariants are convolutions

Who invented convolution?



P.-S. De Laplace

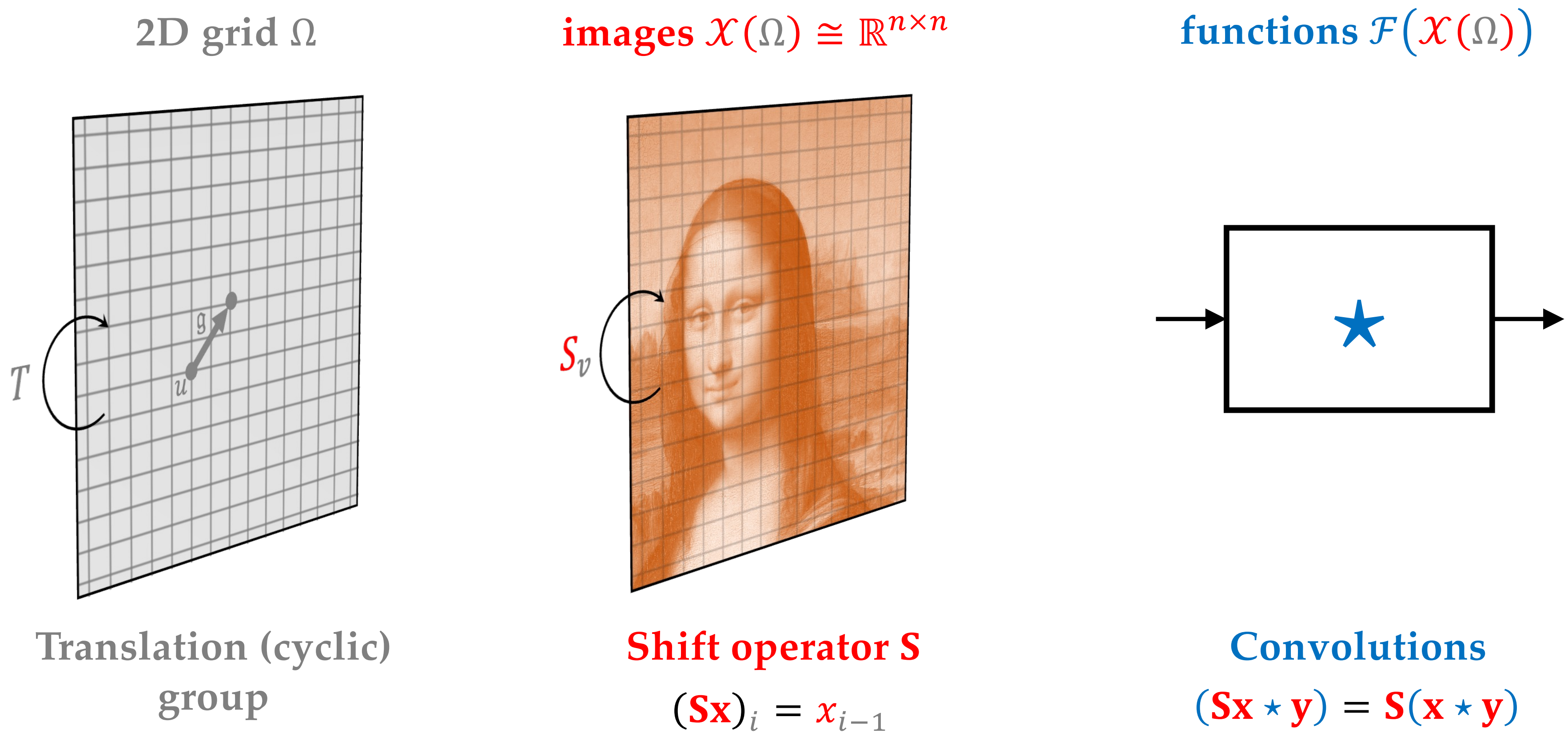
J. B. D'Alembert

résultante[1] *composizione*[2] *Faltung*[3] *convolution*[4]

свёртка

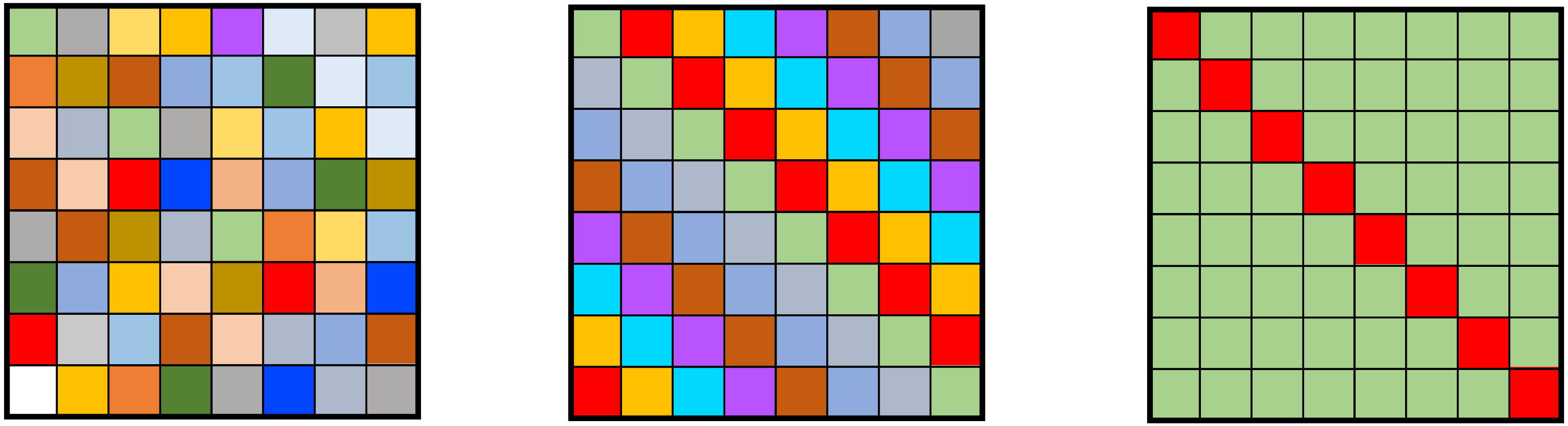
[1]Cailler 1899; [2]Volterra 1910; [3]Doetsch 1923; [4]Winter 1934; Doetsch 1937; Gardner, Barnes 1942

Geometric Deep Learning Blueprint



The story so far

- Linear translation equivariants are *convolutions* (=circulant matrices)
- Convolution *emerges* from translational symmetry
- Grids are special cases of graphs, but with more structure
- We will now study the properties of convolutions and whether linear invariants are sufficient on their own for doing ML on grids (Answer: **no!**)

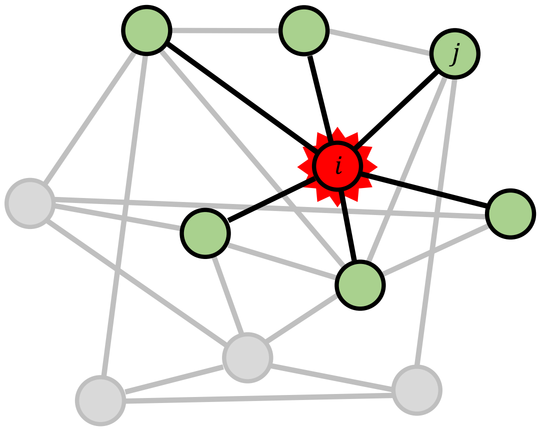


Trivial $G = \{e\}$
$\mathcal{O}(n^2)$ DOF

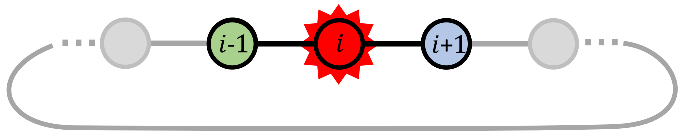
Translation
$\mathcal{O}(n)$ DOF

Permutation
$\mathcal{O}(1)$ DOF

Equivariants on Grids vs Graphs



General Graph

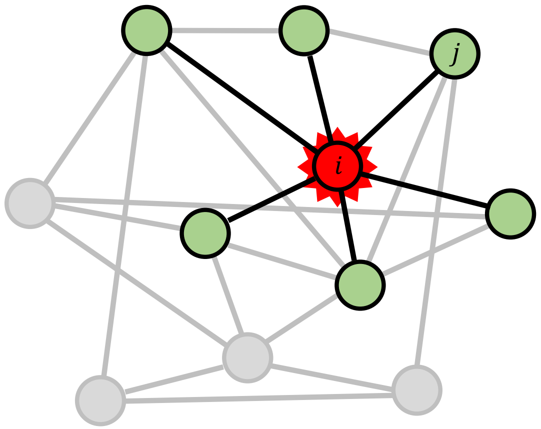


Grid

$$\phi\left(\mathbf{x}_i, \{\mathbf{x}_{j \in \mathcal{N}_i}\}\right)$$

$$\phi\left(\mathbf{x}_i, \{\mathbf{x}_{i-1}, \mathbf{x}_{i+1}\}\right)$$

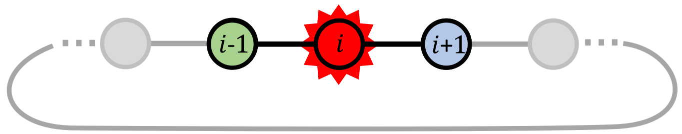
Equivariants on Grids vs Graphs



General Graph

permutation

$$\phi\left(\mathbf{x}_i, \left\{\!\left\{\mathbf{x}_{j\in\mathcal{N}_i}\right\}\!\right\}\right)$$

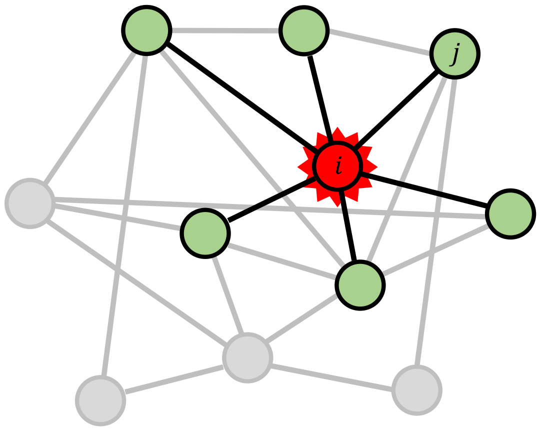


Grid

translation

$$\phi(\mathbf{x}_{i-1}, \mathbf{x}_i, \mathbf{x}_{i+1})$$

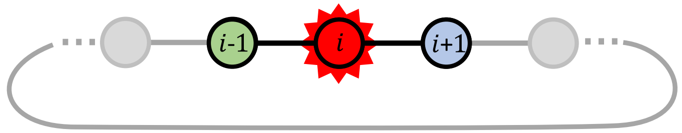
Linear Equivariants on Grids vs Graphs



General Graph

permutation

$$b\mathbf{x}_i + \sum_{j \in \mathcal{N}_i} \mathbf{x}_j$$

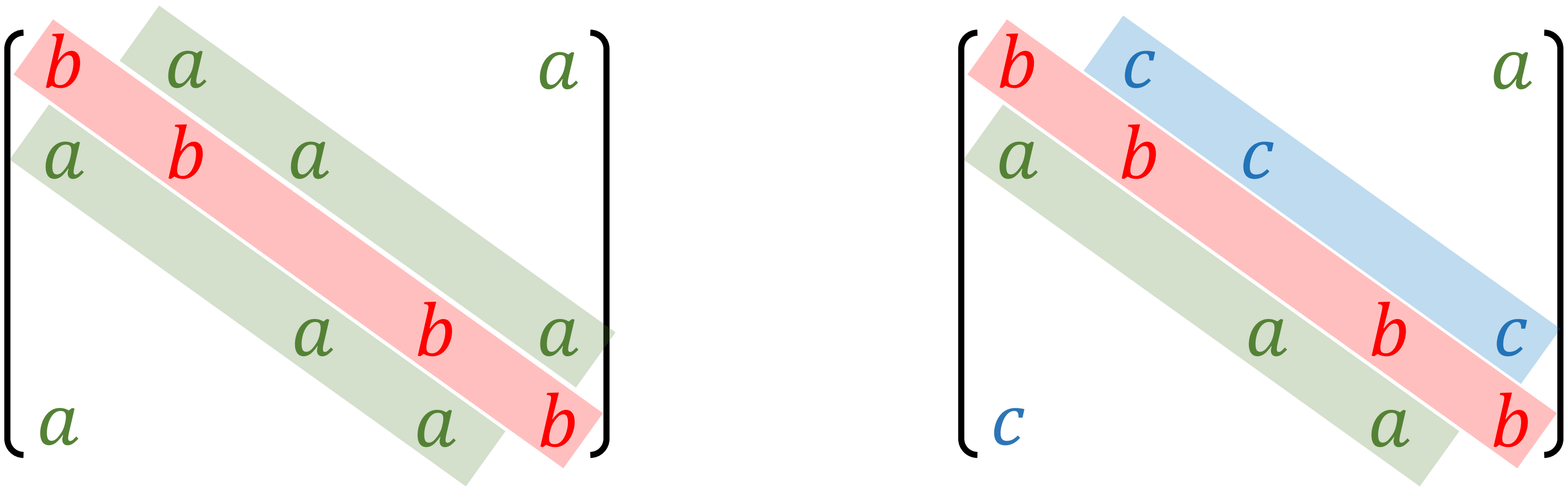


Grid

translation

$$a\mathbf{x}_{i-1} + b\mathbf{x}_i + c\mathbf{x}_{i+1}$$

Linear Equivariants on Grids vs Graphs



“Graph convolution” **Convolution**

FOURIER TRANSFORM

Commutativity & Joint Diagonalisation

Fact from Linear Algebra: Two matrices $\mathbf{A}$ and $\mathbf{B}$ commute (i.e., $\mathbf{AB} = \mathbf{BA}$) iff they are jointly diagonalizable (i.e., there exists a single invertible matrix $\mathbf{\Phi}$ such that $\mathbf{\Phi}^{-1}\mathbf{A}\mathbf{\Phi}$ and $\mathbf{\Phi}^{-1}\mathbf{B}\mathbf{\Phi}$ are diagonal).

Since our linear translation equivariants $\mathbf{C}$ commute with shift $\mathbf{S}$, we can look at the eigenvectors and eigenvalues of the shift matrix $\mathbf{S}$

Commutativity & Joint Diagonalisation

Fact from Linear Algebra: Two matrices $\mathbf{A}$ and $\mathbf{B}$ commute (i.e., $\mathbf{AB} = \mathbf{BA}$) iff they are jointly diagonalizable (i.e., there exists a single invertible matrix $\boldsymbol{\Phi}$ such that $\boldsymbol{\Phi}^{-1}\mathbf{A}\boldsymbol{\Phi}$ and $\boldsymbol{\Phi}^{-1}\mathbf{B}\boldsymbol{\Phi}$ are diagonal).

- Since $\mathbf{S}$ is orthogonal ($\mathbf{S}\mathbf{S}^{\mathrm{T}} = \mathbf{S}^{\mathrm{T}}\mathbf{S} = \mathbf{I}$), it has *orthogonal eigendecomposition* of the form

$$\mathbf{S} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^{*}, \text{ where}$$

  - $\boldsymbol{\Phi} = (\boldsymbol{\phi}_0, \dots, \boldsymbol{\phi}_{n-1})$ is the matrix of *orthogonal eigenvectors* ($\boldsymbol{\Phi}^{*}\boldsymbol{\Phi} = \mathbf{I}$)
  - $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_0, \dots, \lambda_{n-1})$ is the diagonal matrix of the corresponding eigenvalues
- Since $\mathbf{S}$ is not symmetric ($\mathbf{S} \neq \mathbf{S}^{\mathrm{T}}$), we expect complex eigenvalues/eigenvectors
- Commuting matrices have same eigenvectors but different eigenvalues

Eigenvalues & Eigenvectors of the Shift matrix

- Let $\mathbf{x} \neq \mathbf{0}$. Then, for all $i \bmod n$:

$$\mathbf{Sx} = \lambda \mathbf{x} \qquad \Longleftrightarrow \qquad x_{i+1} = \lambda x_i$$

Note: all indices are $\bmod\ n$.

Eigenvalues & Eigenvectors of the Shift matrix

- Let $\mathbf{x} \neq \mathbf{0}$. Then, for all i mod n:

$$\mathbf{S}^n\mathbf{x} = \lambda^n\mathbf{x} \qquad \Leftrightarrow \qquad x_{i+n} = x_i = \lambda^n x_i$$

- Since at least one $x_k \neq 0$, we have $\lambda^n = 1$, from which it follows that the **eigenvalues** of $\mathbf{S}$ are the complex *roots of unity*

$$\lambda_k = \mathrm{e}^{2\pi \mathrm{i} k/n} \text{ for } k = 0, \dots, n-1$$

- We can now use $x_{i+j} = \lambda^j x_i$ to express the entries of the kth **eigenvector** of $\mathbf{S}$:

$$(\boldsymbol{\phi}_k)_{j+0} = \lambda_k^j (\boldsymbol{\phi}_k)_0$$

Note: all indices are $\mathrm{mod}\ n$.

Eigenvalues & Eigenvectors of the Shift matrix

- Let $\mathbf{x} \neq \mathbf{0}$. Then, for all i mod n:

$$\mathbf{S}^n\mathbf{x} = \lambda^n\mathbf{x} \qquad \Leftrightarrow \qquad x_{i+n} = x_i = \lambda^n x_i$$

- Since at least one $x_k \neq 0$, we have $\lambda^n = 1$, from which it follows that the **eigenvalues** of $\mathbf{S}$ are the complex *roots of unity*

$$\lambda_k = \mathrm{e}^{2\pi \mathrm{i} k/n} \text{ for } k = 0, \dots, n-1$$

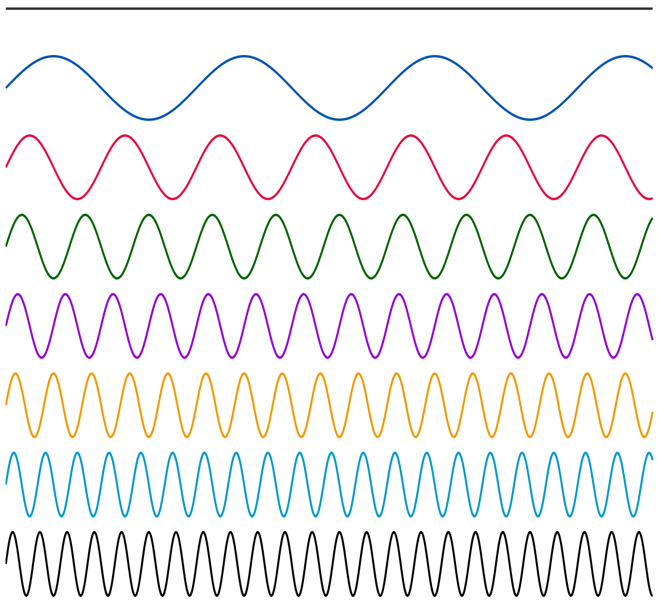
- We can now use $x_{i+j} = \lambda^j x_i$ to express the entries of the kth **eigenvector** of $\mathbf{S}$:

$$\boldsymbol{\phi}_k = c\left(1, \mathrm{e}^{2\pi \mathrm{i} k/n}, \mathrm{e}^{2\pi \mathrm{i} 2k/n}, \dots, \mathrm{e}^{2\pi \mathrm{i}(n-1)k/n}\right)$$

where $c = 1/\sqrt{n}$ such that $\|\boldsymbol{\phi}_k\| = 1$.

Note: all indices are mod n.

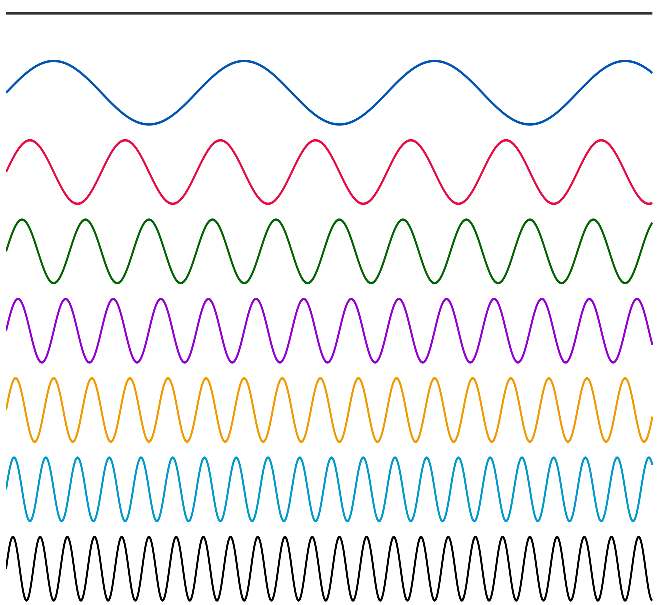
Discrete Fourier Transform

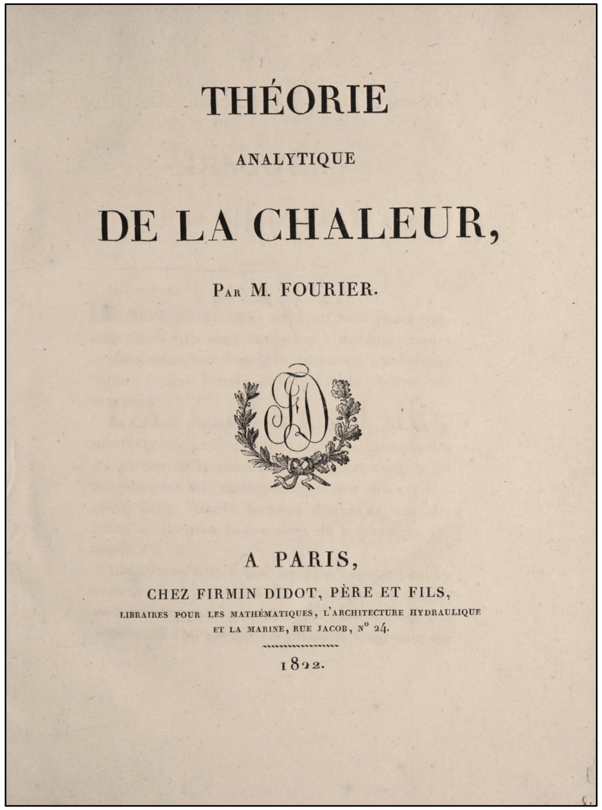


$$\boldsymbol{\Phi} = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 & & 1 \\ 1 & e^{i\frac{2\pi}{n}} & & e^{i\frac{2\pi}{n}(n-1)} \\ \vdots & \vdots & \ddots & \\ 1 & e^{i\frac{2\pi}{n}(n-1)} & & e^{i\frac{2\pi}{n}(n-1)^2} \end{bmatrix}$$

- Columns of $\boldsymbol{\Phi}$ form an *orthogonal basis*, referred to as the **Fourier basis**
- Multiplication of a vector $\hat{\mathbf{x}} = \boldsymbol{\Phi}^* \mathbf{x}$ is called the **Discrete Fourier Transform** (DFT) of $\mathbf{x}$

Fourier Transform





THÉORIE

ANALYTIQUE

DE LA CHALEUR,

Par M. FOURIER.

A PARIS,

CHEZ FIRMIN DIDOT, PÈRE ET FILS,

LIBRAIRES POUR LES MATHÉMATIQUES, L'ARCHITECTURE HYDRAULIQUE

ET LA MARINE, RUE JACOB, N° 24.

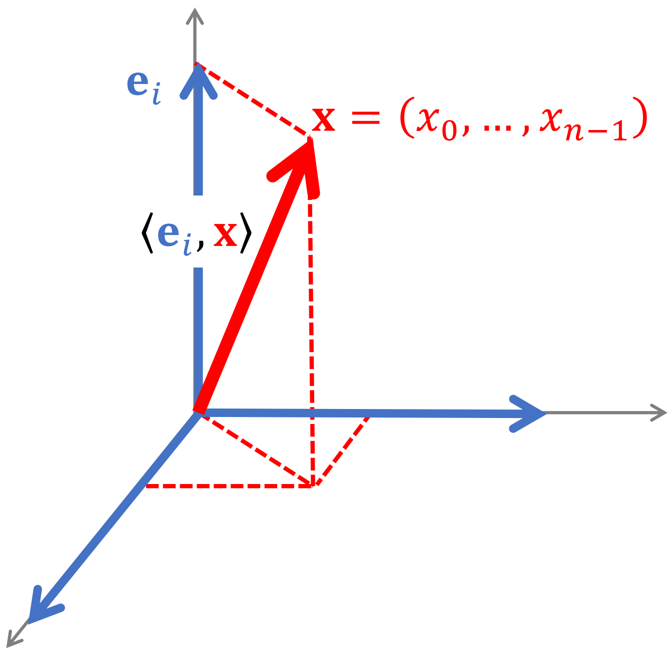
1822.

J. Fourier

1822

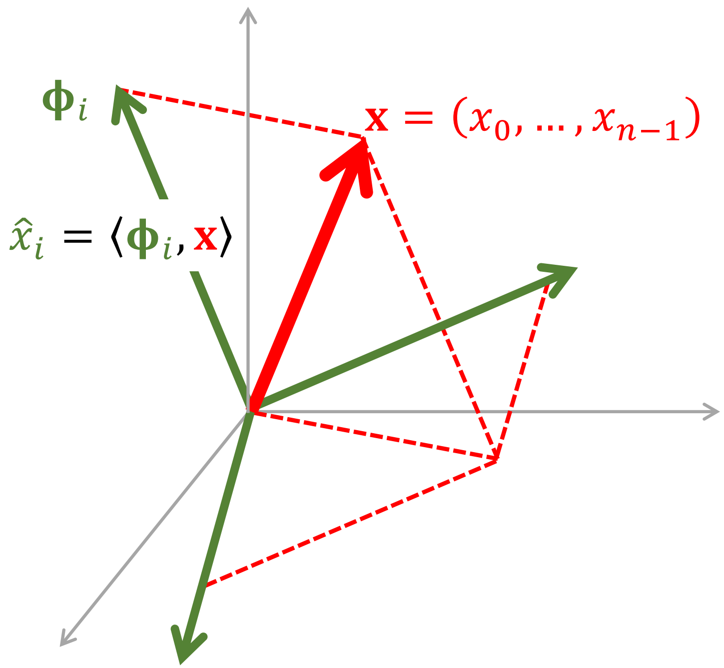
Fourier 1822

Geometric intuition



$$\mathbf{x} = \sum_{i=0}^{n-1} \langle \mathbf{e}_i, \mathbf{x} \rangle \mathbf{e}_i \quad (n\text{D Pythagoras})$$

Geometric intuition



$$\mathbf{x} = \sum_{i=0}^{n-1} \langle \boldsymbol{\phi}_i, \mathbf{x} \rangle \boldsymbol{\phi}_i = \boldsymbol{\Phi}\boldsymbol{\Phi}^* \mathbf{x}$$

- Parceval theorem:

$$\langle \hat{\mathbf{x}}, \hat{\mathbf{y}} \rangle = (\boldsymbol{\Phi}^* \mathbf{x})^* \boldsymbol{\Phi}^* \mathbf{y} = \mathbf{x}^* \boldsymbol{\Phi}\boldsymbol{\Phi}^* \mathbf{y} = \langle \mathbf{x}, \mathbf{y} \rangle$$

- Plancherel theorem (special case):

$$\|\hat{\mathbf{x}}\| = \|\mathbf{x}\|$$

i.e., Fourier transform is an *isometry*

Convolution in the Fourier domain

- DFT diagonalises the shift operator $\boldsymbol{S}$ and all matrices commuting with it (linear translation-equivariants we called *circulant matrices* or *convolutions*)

$$\mathbf{C} = \mathbf{\Phi \Lambda \Phi}^*$$

- Express convolution in the Fourier basis

$$\mathbf{Cx} = \mathbf{C\Phi\Phi}^*\mathbf{x} = (\mathbf{\Phi\Lambda\Phi}^*)\mathbf{\Phi}(\mathbf{\Phi}^*\mathbf{x})$$

Convolution in the Fourier domain

- DFT diagonalises the shift operator $\boldsymbol{S}$ and all matrices commuting with it (linear translation-equivariants we called *circulant matrices* or *convolutions*)

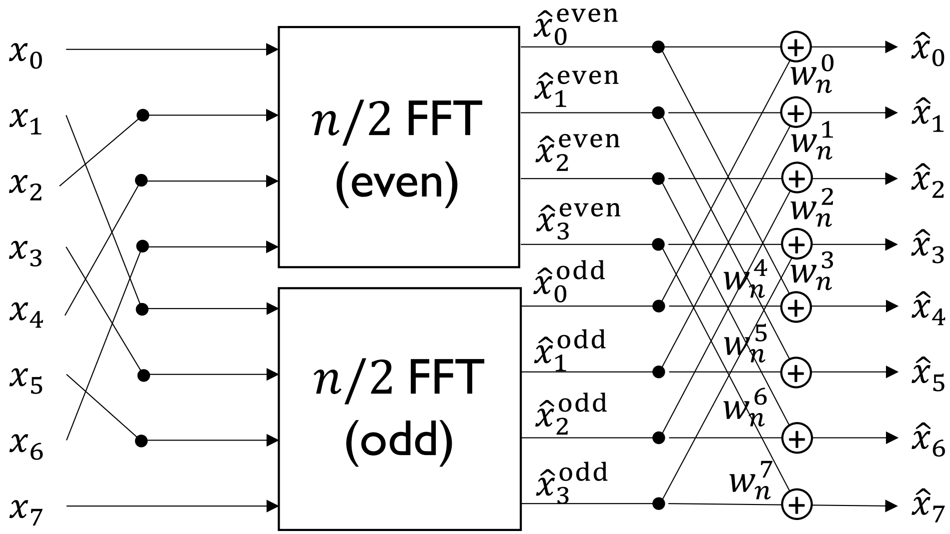
$$\mathbf{C} = \mathbf{\Phi}\mathbf{\Lambda}\mathbf{\Phi}^*$$

- Express convolution in the Fourier basis

$$\mathbf{C}\mathbf{x} = \mathbf{C}\mathbf{\Phi}\mathbf{\Phi}^*\mathbf{x} = \mathbf{\Phi}(\mathbf{\Lambda}\hat{\mathbf{x}})$$

  - Convolution becomes *element-wise product* in Fourier basis! (*"DFT diagonalises convolution"*)
  - The Fourier basis matrices $\mathbf{\Phi}, \mathbf{\Phi}^*$ have a highly-redundant structure, allowing to compute $\hat{\mathbf{x}} = \mathbf{\Phi}^*\mathbf{x}$ (forward DFT) and $\mathbf{x} = \mathbf{\Phi}\hat{\mathbf{x}}$ (inverse DFT) in complexity $\mathcal{O}(n \log n)$ rather than $\mathcal{O}(n^2)$.
  - Such methods are known as Fast Fourier Transforms (FFT) and have been the bread-and-butter of digital signal processing since the late 1960s

Fast Fourier Transform



$$w_n^k = e^{-\mathrm{i}\frac{2\pi}{n}k}$$

(Gauss 1805); Cooley, Tukey 1965

DFT of Convolution

- Last remaining bit: how do the eigenvalues of $\mathbf{C} = \mathbf{\Phi\Lambda\Phi}^*$ look like?
- Recall that $\mathbf{C} = \mathbf{C}(\boldsymbol{\theta})$ constructed from circularly-shifted column vectors $\boldsymbol{\theta}$
- Let us express $\mathbf{C}\boldsymbol{\phi}_k$=$\lambda_k\boldsymbol{\phi}_k$:

$$\begin{bmatrix} \theta_0 & \theta_{n-1} & \cdots & \theta_1 \\ \theta_1 & \theta_0 & & \theta_2 \\ \vdots & & \ddots & \vdots \\ \theta_{n-1} & \theta_{n-2} & \cdots & \theta_0 \end{bmatrix} \begin{bmatrix} 1 \\ e^{\mathrm{i}\frac{2\pi}{n}k} \\ \vdots \\ e^{\mathrm{i}\frac{2\pi}{n}k(n-1)} \end{bmatrix} = \lambda_k \begin{bmatrix} 1 \\ e^{\mathrm{i}\frac{2\pi}{n}k} \\ \vdots \\ e^{\mathrm{i}\frac{2\pi}{n}k(n-1)} \end{bmatrix}$$

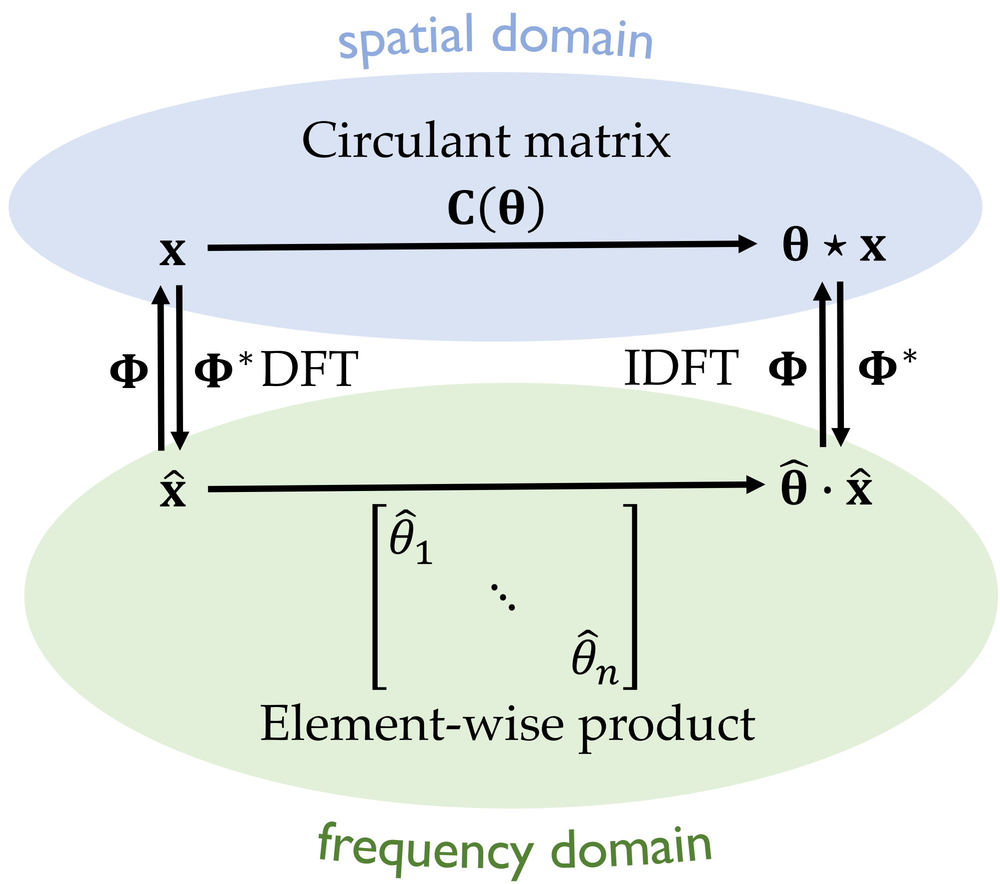
$$\lambda_k = \theta_0 + \theta_{n-1}e^{\mathrm{i}\frac{2\pi}{n}k} + \cdots + \theta_1 e^{\mathrm{i}\frac{2\pi}{n}k(n-1)}$$

$$= \theta_0 + \theta_1 e^{-\mathrm{i}\frac{2\pi}{n}k} + \cdots + \theta_{n-1}e^{-\iota\frac{2\pi}{n}k(n-1)} = \sum_{l=0}^{n-1} \theta_l e^{-\mathrm{i}\frac{2\pi}{n}kl} = \hat{\theta}_k$$

DFT of Convolution

- Last remaining bit: how do the eigenvalues of $\mathbf{C} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^*$ look like?
- Recall that $\mathbf{C} = \mathbf{C}(\boldsymbol{\theta})$ constructed from circularly-shifted column vectors $\boldsymbol{\theta}$

> A circulant matrix can be expressed in the Fourier domain as $\mathbf{C}(\boldsymbol{\theta}) = \boldsymbol{\Phi} \, \mathrm{diag}\big(\hat{\boldsymbol{\theta}}\big) \, \boldsymbol{\Phi}^*$.


spatial domain
Circulant matrix
$\mathbf{C}(\boldsymbol{\theta})$
$\mathbf{x}$
$\boldsymbol{\theta} \star \mathbf{x}$
$\boldsymbol{\Phi}$
$\boldsymbol{\Phi}^*$ DFT
IDFT $\boldsymbol{\Phi}$
$\boldsymbol{\Phi}^*$
$\hat{\mathbf{x}}$
$\hat{\boldsymbol{\theta}} \cdot \hat{\mathbf{x}}$
$\begin{bmatrix} \hat{\theta}_1 & & \\ & \ddots & \\ & & \hat{\theta}_n \end{bmatrix}$
Element-wise product
frequency domain

CONTINUOUS CASE

Continuous Fourier Transform

- Let $x, y \in L_2(\mathbb{R})$
- Continuous **convolution**

$$(x \star y)(u) = \int_{-\infty}^{+\infty} x(v)y(u - v)\mathrm{d}v$$

- Continuous **Fourier Transform**

$$\hat{x}(\xi) = \int_{-\infty}^{+\infty} x(u)e^{-2\pi \mathrm{i}\xi u}\mathrm{d}u \qquad x(u) = \int_{-\infty}^{+\infty} \hat{x}(\xi)e^{+2\pi \mathrm{i}\xi u}\mathrm{d}\omega$$

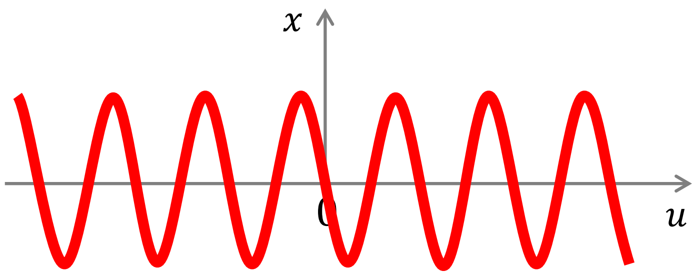
  - $\omega = 2\pi\xi$ is often referred to as **frequency**
  - Continuous Fourier basis: orthogonal functions $\phi_\xi(u) = e^{+2\pi \mathrm{i}\xi u}$ s.t. $\langle \phi_\xi, \phi_\zeta \rangle = \delta(\xi - \zeta)$
  - Fourier transform: orthogonal projection on the basis $\hat{x}(\xi) = \langle x, \phi_\xi \rangle$

Continuous Fourier Transform: Properties

• Shift	$x(u-v)$	Phase	$e^{-2\pi i \xi v}\hat{x}(\xi)$
• Derivative	$\frac{\mathrm{d}}{\mathrm{d}u}x(u)$	Moment	$i\omega\hat{x}(\xi)$
• Convolution	$(x \star y)(u)$	Product	$\hat{x}(\xi)\cdot\hat{y}(\xi)$
• Product	$(x \cdot y)(u)$	Convolution	$\hat{x}(\xi)\star\hat{y}(\xi)$
• Scaling	$x(\alpha u), \alpha > 0$	Inv. Scaling	$\frac{1}{\alpha}\hat{x}\left(\frac{\xi}{\alpha}\right)$

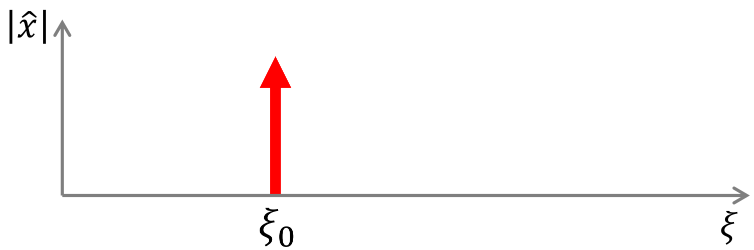
- Parseval theorem: $\langle x, y\rangle = \langle \hat{x}, \hat{y}\rangle$
- Plancherel theorem: $\|x\| = \|\hat{x}\|$ (isometry)

Harmonics and Deltas



Space

$$x(u) = e^{2\pi i \xi_0 u}$$

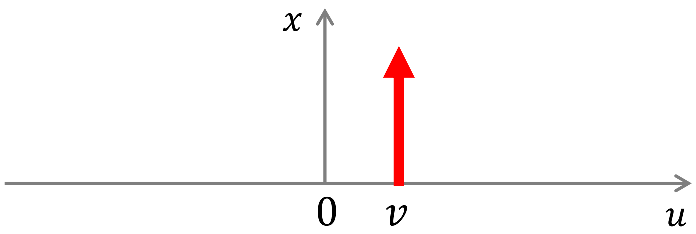


Frequency

$$\hat{x}(\xi) = \delta(\xi - \xi_0)$$

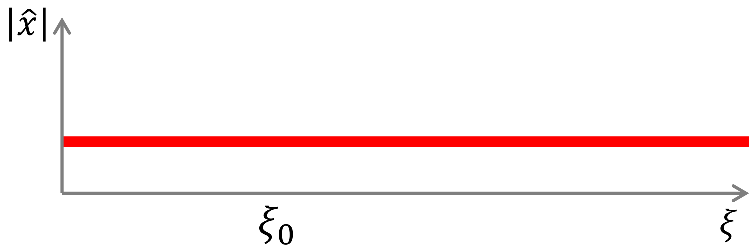
Note: Dirac delta is a "generalised function." It can be regarded as a linear functional on Hilbert space.

Harmonics and Deltas



Space

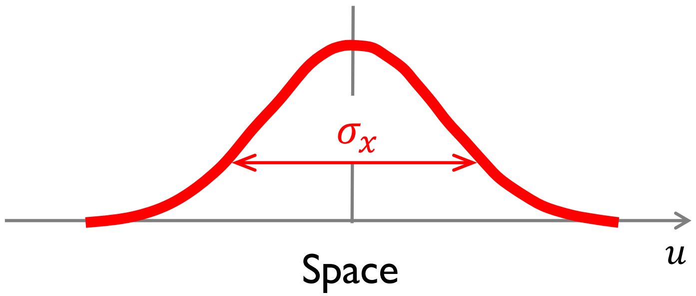
$$x(u) = \delta(u - v)$$

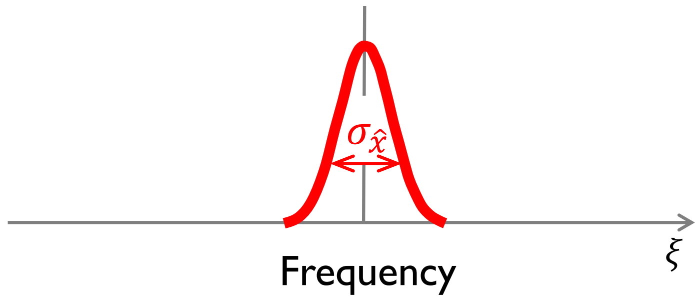


Frequency

$$\hat{x}(\xi) = \int_{-\infty}^{+\infty} \delta(u - v) e^{-2\pi \mathrm{i} \xi u} \mathrm{d}u = e^{-2\pi \mathrm{i} \xi v}$$

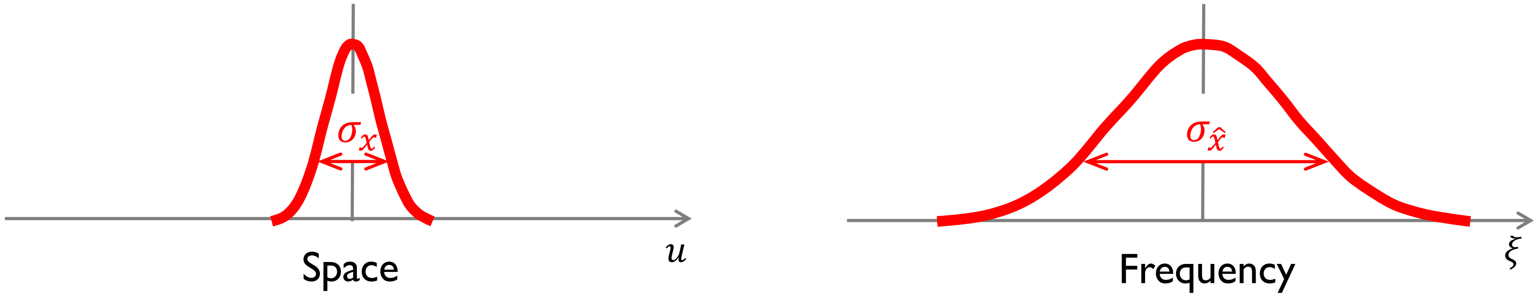
Localisation in Space and Frequency





$$x(\alpha u) \qquad\qquad \frac{1}{\alpha}\hat{x}\left(\frac{\xi}{\alpha}\right)$$

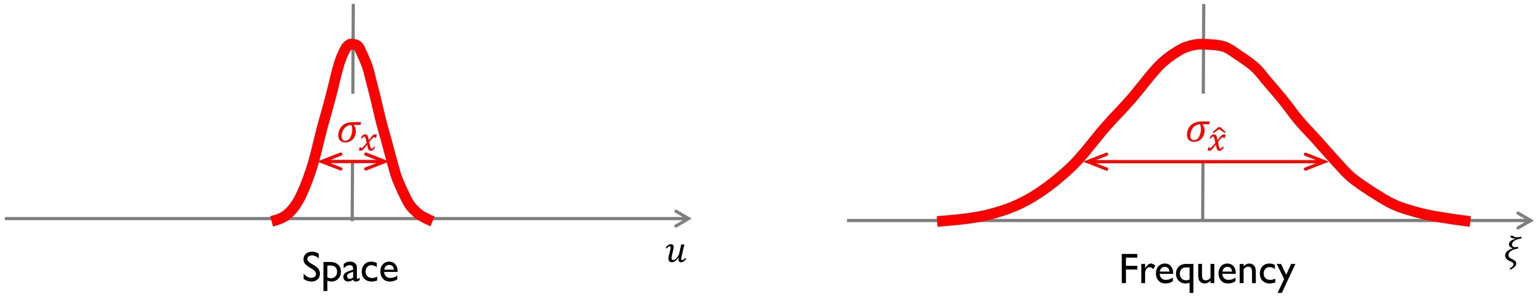
Localisation in Space and Frequency



$x(\alpha u)$

$\frac{1}{\alpha}\hat{x}\left(\frac{\xi}{\alpha}\right)$

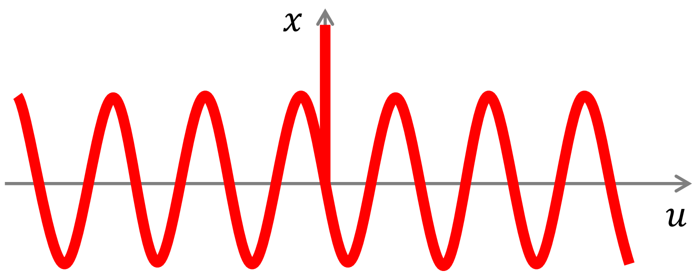
Localisation in Space and Frequency



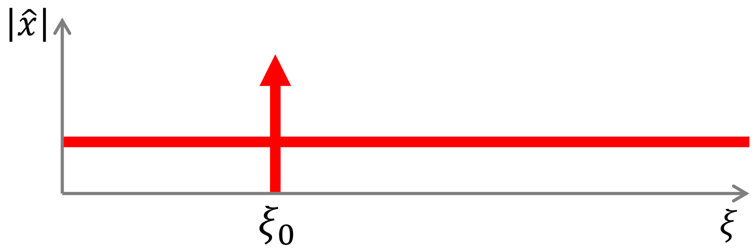
$$\sigma_x^2 \, \sigma_{\hat{x}}^2 \geq \frac{1}{16\pi^2}$$

Exercise: prove.

Spectrogram, or Windowed Fourier Transform

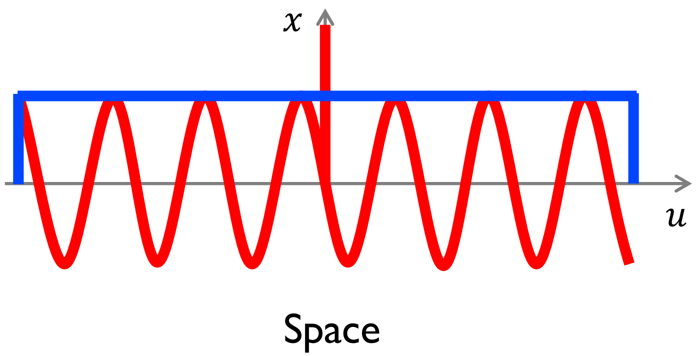


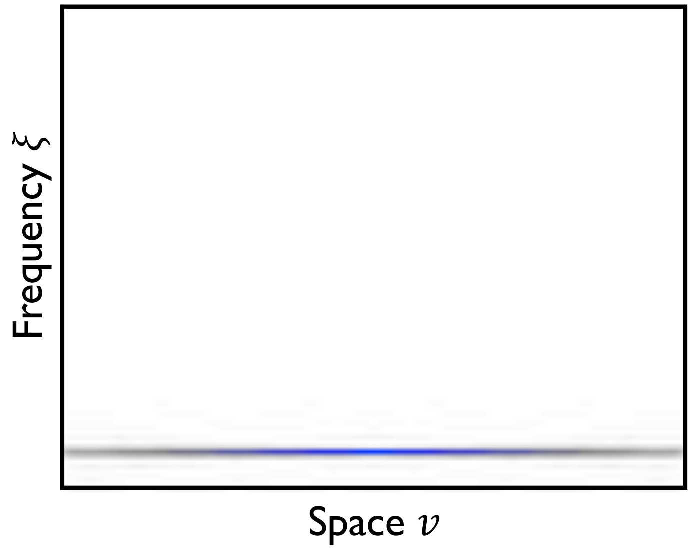
Space



Frequency

Spectrogram, or Windowed Fourier Transform

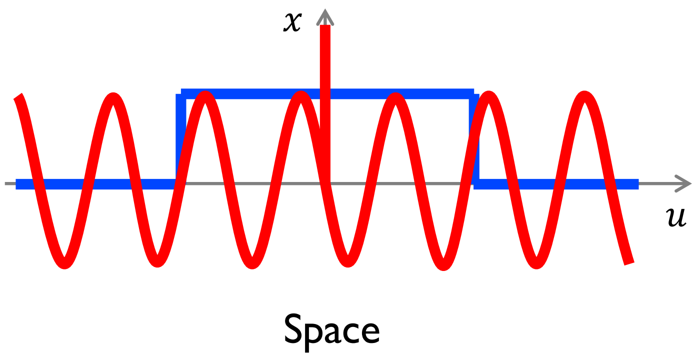


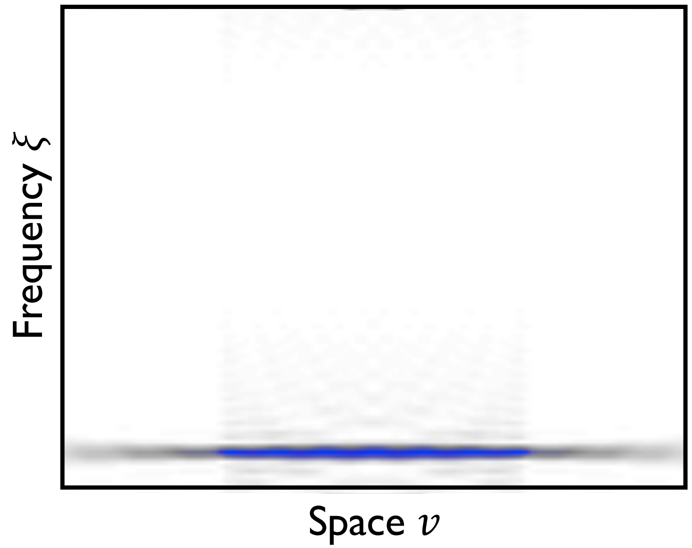


- **Windowed Fourier Transform:** FT applied to a signal multiplied by a local window h, shifted to different positions v

$$\hat{x}_h(v, \xi) = \int_{-\infty}^{+\infty} h(u - v)x(u)e^{-2\pi i \xi u} du$$

Spectrogram, or Windowed Fourier Transform

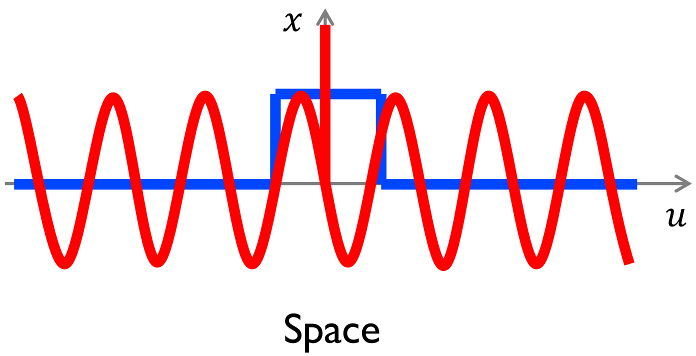


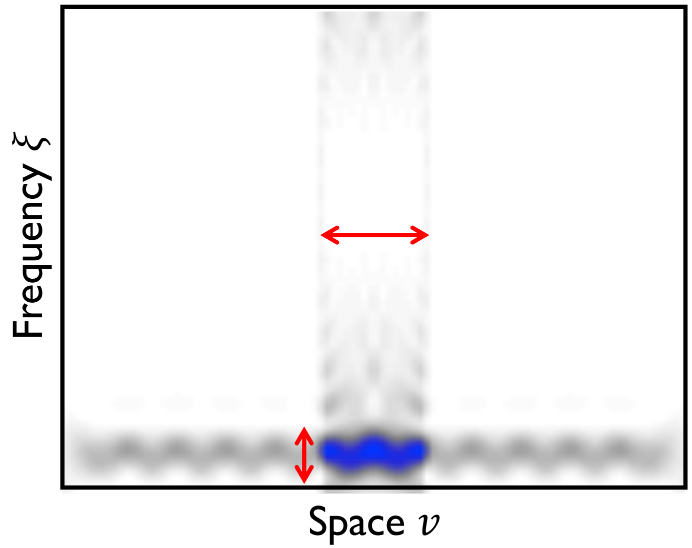


- **Windowed Fourier Transform:** FT applied to a signal multiplied by a local window h, shifted to different positions v

$$\hat{x}_h(v,\xi) = \int_{-\infty}^{+\infty} h(u-v)x(u)e^{-2\pi i \xi u}\,du$$

Spectrogram, or Windowed Fourier Transform

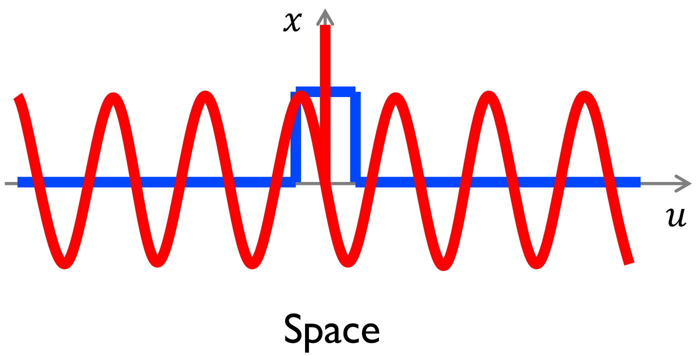


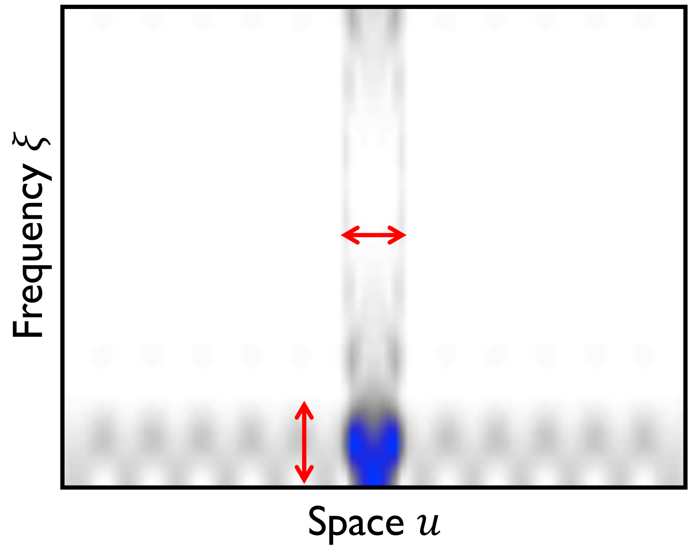


- **Windowed Fourier Transform:** FT applied to a signal multiplied by a local window h, shifted to different positions v

$$\hat{x}_h(v,\xi) = \int_{-\infty}^{+\infty} h(u-v)x(u)e^{-2\pi i \xi u}\mathrm{d}u$$

Spectrogram, or Windowed Fourier Transform





- **Windowed Fourier Transform:** FT applied to a signal multiplied by a local window h, shifted to different positions v

$$\hat{x}_h(v,\xi) = \int_{-\infty}^{+\infty} h(u-v)x(u)e^{-2\pi i \xi u}\mathrm{d}u$$

GEOMETRIC STABILITY

Fourier invariants

- Geometric Deep Learning Blueprint: compose linear equivariants with non-linear element-wise functions to obtain *non-linear invariants*
- Fourier Transform modulus:

$$f(x) = |\hat{x}|$$

is a *non-linear translation invariant*

Proof: Use the property of the FT transforming shift $x(u-v)$ into complex phase $e^{-2\pi i \xi v}\hat{x}(\xi)$. Taking the absolute value then removes the complex phase.

Fourier invariants

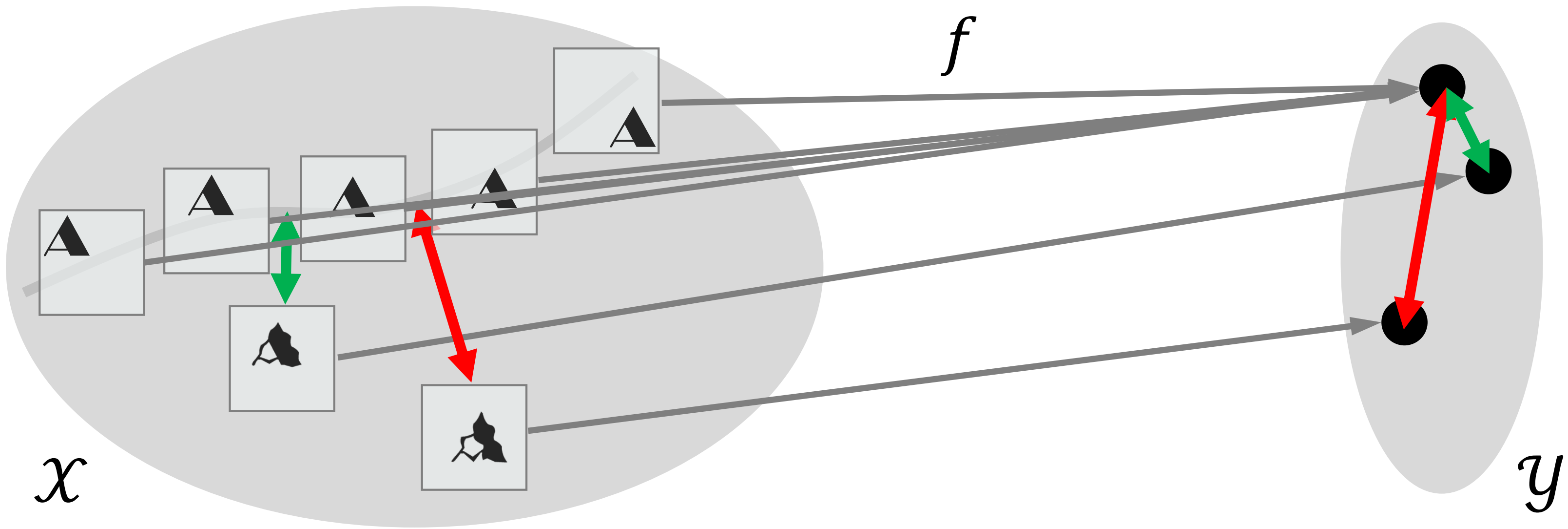
- Geometric Deep Learning Blueprint: compose linear equivariants with non-linear element-wise functions to obtain *non-linear invariants*
- Fourier Transform modulus:

$$f(x) = |\hat{x}|$$

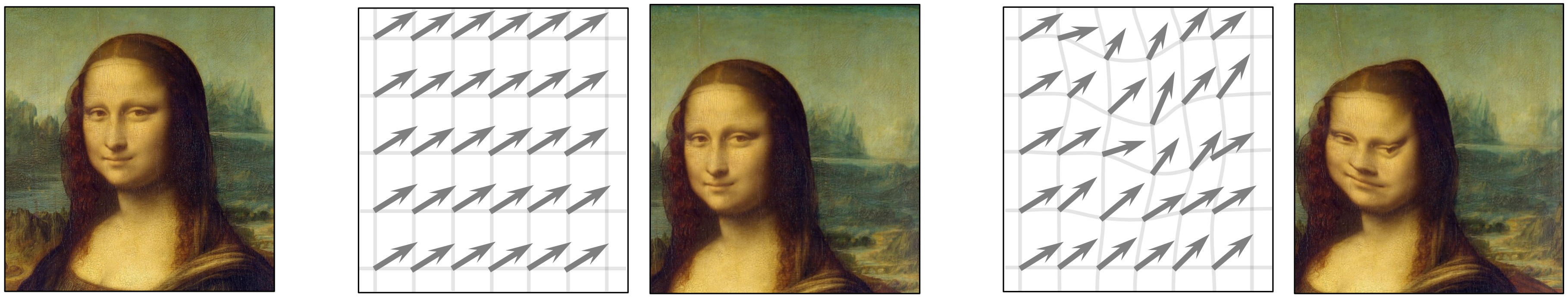
is a *non-linear translation invariant*

Is this invariant stable?

Reminder: Approximate Group Invariance



Reminder: Approximate Group Invariance

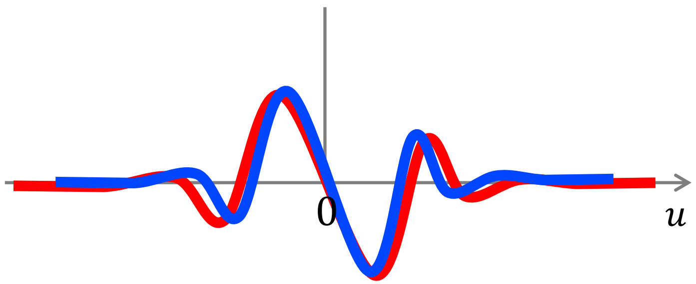


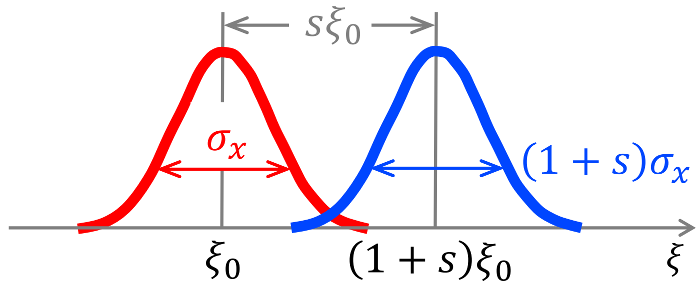
Translation **Warping**

- The translation group is a *subgroup* of a much bigger group of *diffeomorphisms* ("warpings") $\tau: \Omega \to \Omega$ acting in the plane
- The smoothness of τ (**Dirichlet energy**) measures how "far" it is from translation

$$\|\nabla\tau\|^2 = \int_{\mathbb{R}^2} \|\nabla\tau(u)\|^2 \mathrm{d}u = 0$$

Stability of Fourier modulus: 1D example





- Let $x(u) = h(u)e^{2\pi i \xi_0 u}$ be a signal concentrated around frequency ξ_0 with bandwidth σ_x
- Warping $(\tau x)(u) = x\big((1+s)u\big)$, with $s \ll 1$
- If $(1+s)\xi_0 - \xi_0 = s\xi_0 \gg (2+s)\sigma_x$, then $\| |\hat{x}| - |\widehat{\tau x}| \| = \mathcal{O}(\|x\|)$

  i.e., $\| |\hat{x}| - |\widehat{\tau x}| \| / \|x\| = \mathcal{O}(1)$ rather than $\mathcal{O}(s)$, indicating that this **invariant is unstable!**

Exercise: prove.

What do we learn from this example?

- *Low frequencies* are stable
- *High frequencies* are unstable (dilation affects them more)
- Removing high frequencies looses information
- Key challenge: how to "stabilise" high frequencies without loosing information?

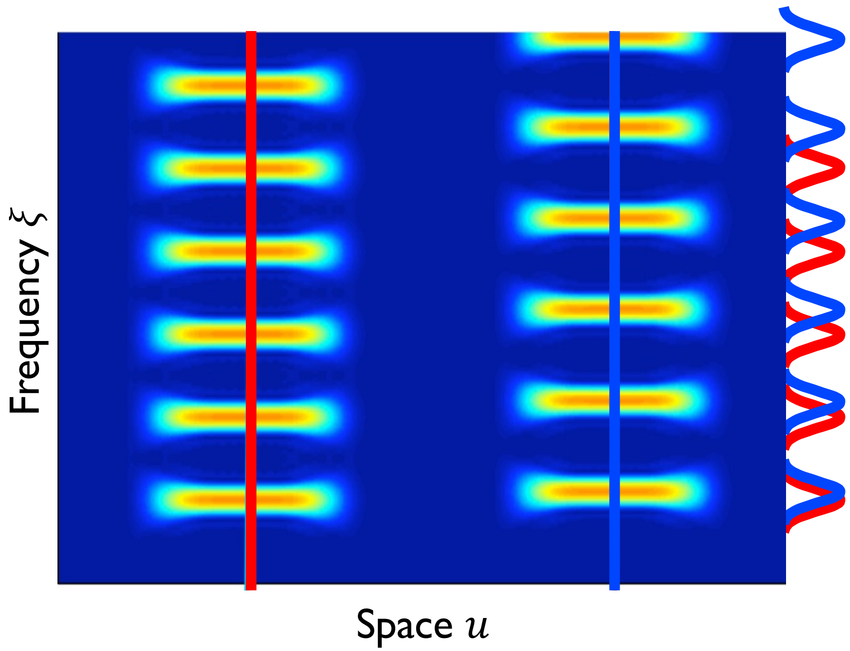
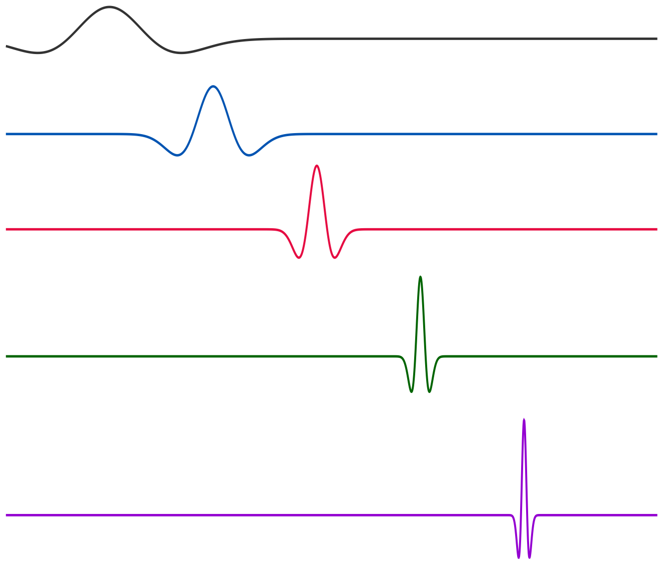


Image: Mallat (IPAM lectures)

WAVELETS & MULTISCALE ANALYSIS

Wavelets

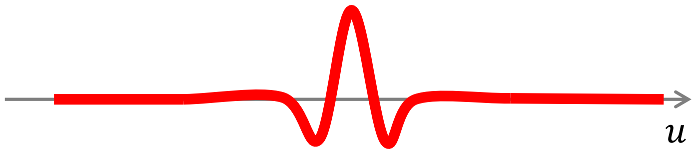


I. Daubechies

S. Mallat

Haar 1909; Gabor 1946; Grossman, Morlet 1984 (invented the term), Meyer 1990 (2017 Abel prize for wavelets); Daubechies 1988; Mallat 1989; Coifman 1991

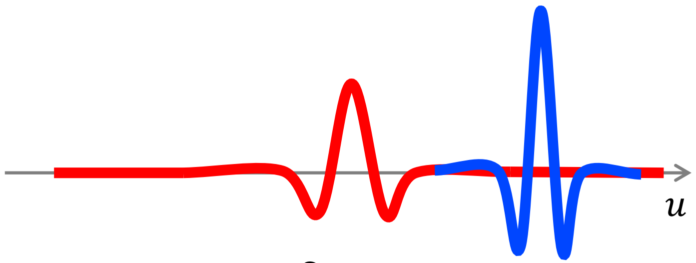
Wavelet Transform



Mother wavelet

$\psi(u)$

Wavelet Transform

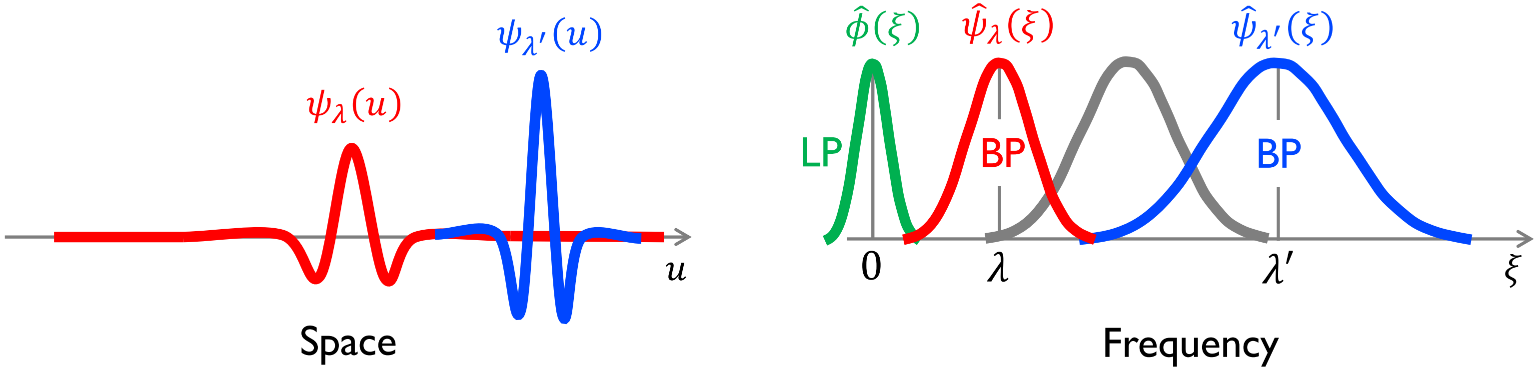


Dilated and **translated wavelet**

$$\psi_\lambda(u) = \lambda^{-1/2}\psi\left(\frac{u-a}{\lambda}\right)$$

Wavelet Transform: $(Wx)(u,\lambda) = \lambda^{-1/2}\int_{-\infty}^{+\infty}\psi\left(\frac{v-u}{\lambda}\right)x(v)\mathrm{d}v$

Wavelet Transform

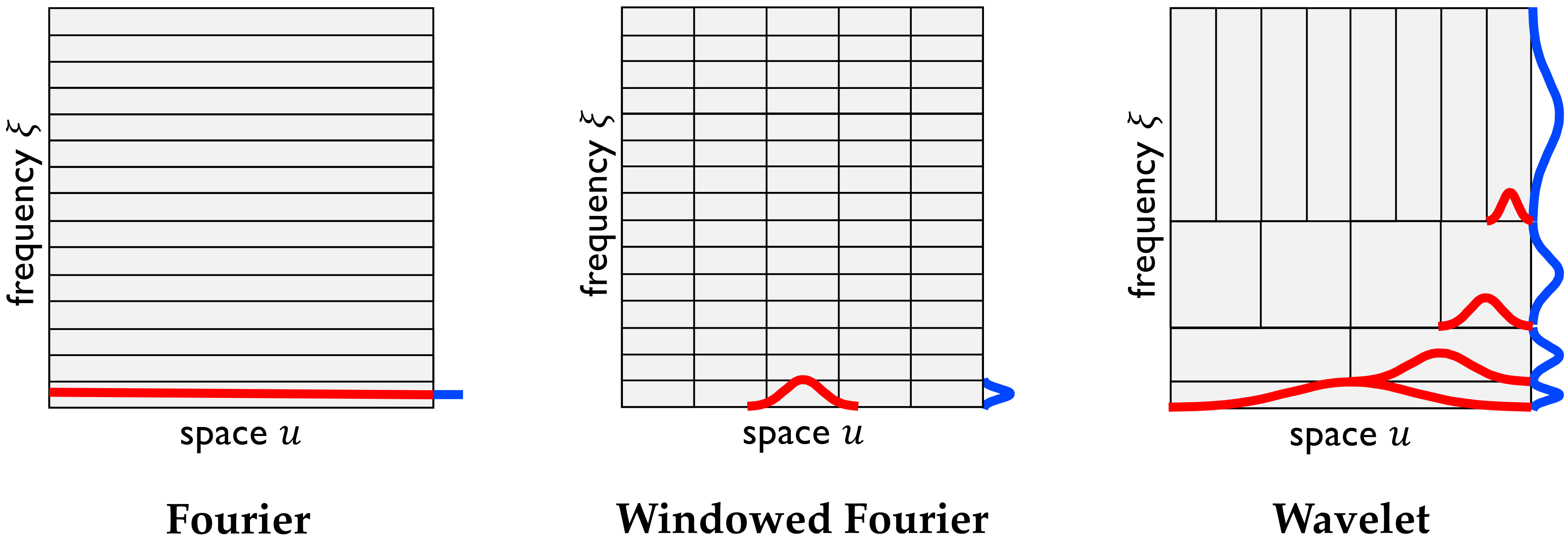


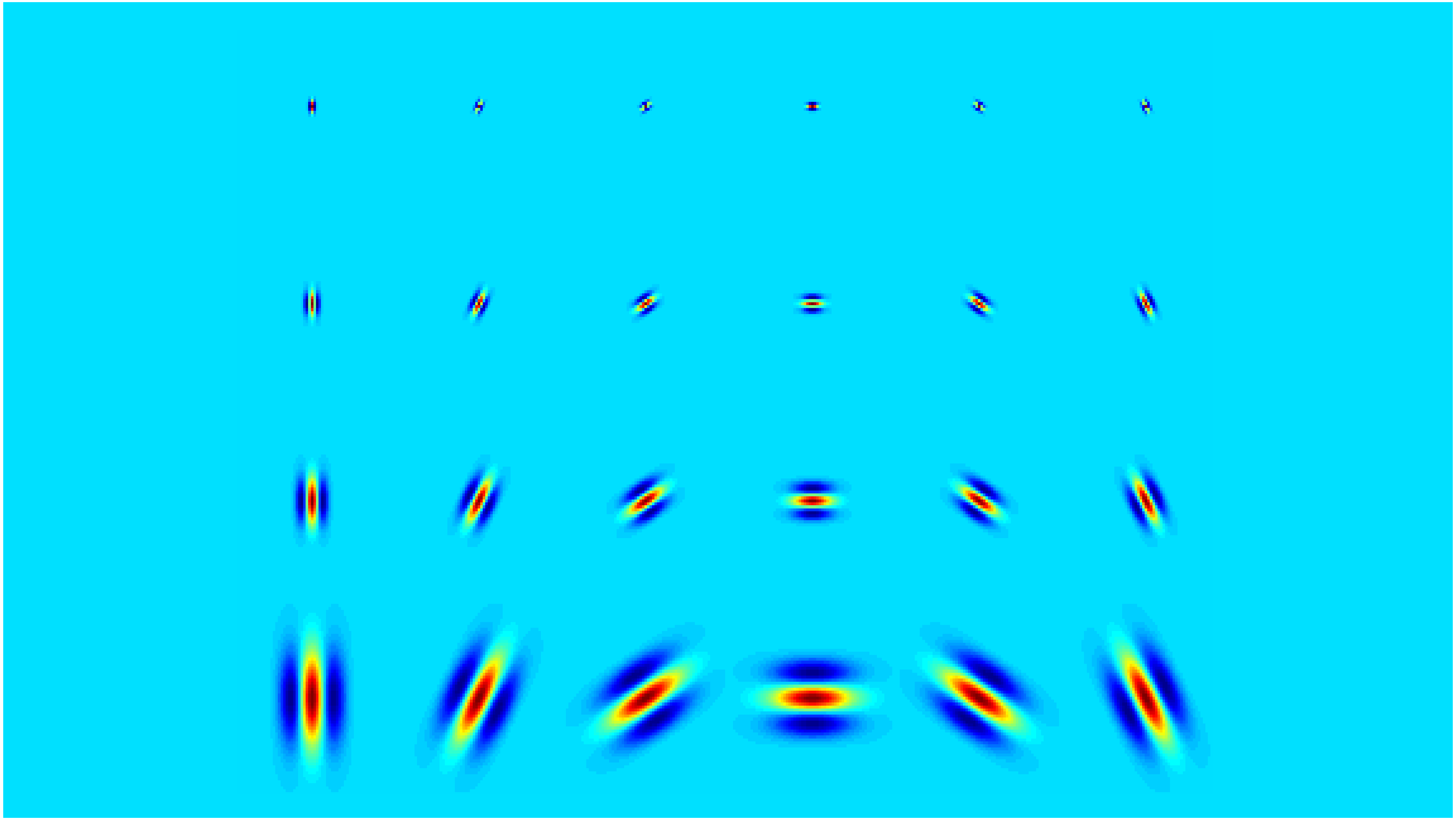
Dyadically **dilated** and **translated wavelet**

$$\psi_\lambda(u) = \lambda^{-1/2}\psi\left(\frac{u-a}{\lambda}\right) \qquad \lambda = 2^{-j},\, a = 2^{-j}k$$

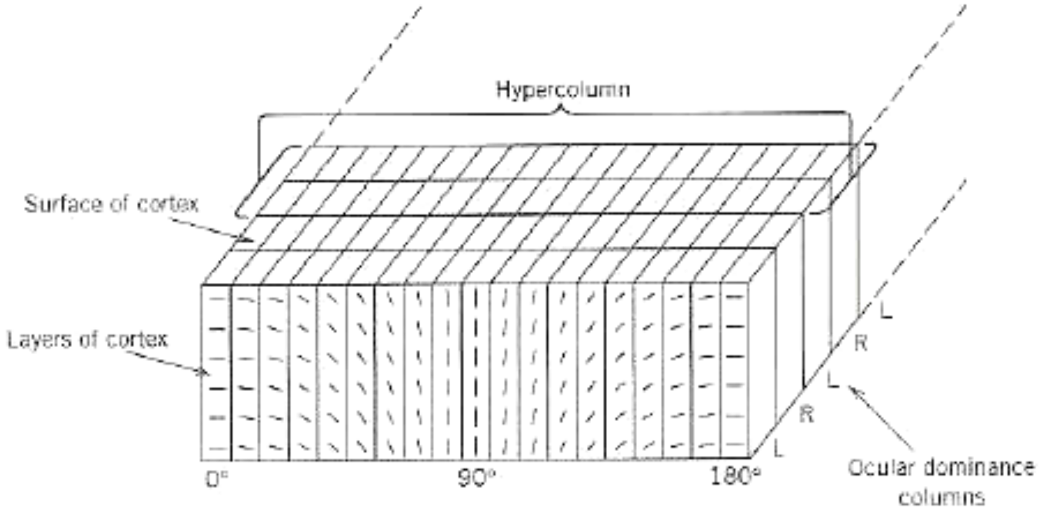
Discrete Wavelet Transform: $c_{jk} = (Wx)\left(2^{-j}k, 2^{-j}k\right)$

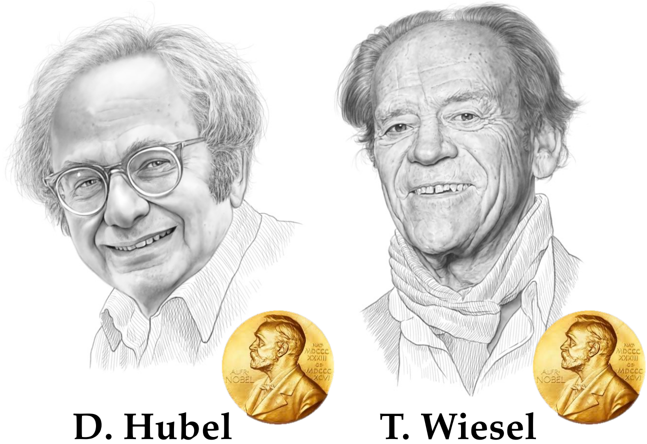
Wavelets vs Fourier vs Windowed Fourier





Organisation of the brain





1959

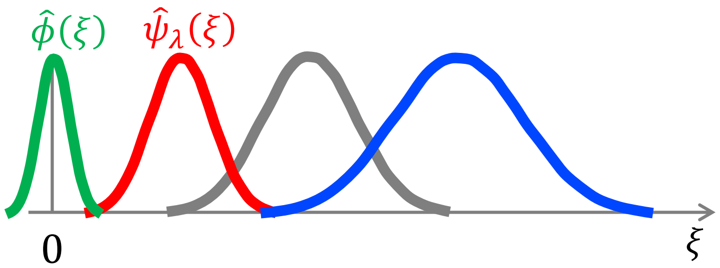
Hubel, Wiesel 1959; 1962

Wavelet Transform

$$(Wx)(u,\lambda) = \lambda^{-1/2} \int_{-\infty}^{+\infty} \psi\left(\frac{v-u}{\lambda}\right) x(v) \mathrm{d}v\,, \qquad \lambda = 2^{-j}, u = 2^{-j}k$$

- Convolution with a **filter bank** $Wx = \{x \star \phi,\ x \star \psi_\lambda\}_\lambda$ where ϕ is LP filter ("averaging")
- Appropriate design choice $|\hat{\phi}(\xi)|^2 + \sum_\lambda |\hat{\psi}_\lambda(\xi)|^2 = 1$ guarantees **unitarity**

$$\|Wx\| = \|x\|.$$



Question why do we care about unitarity (energy preservation)?

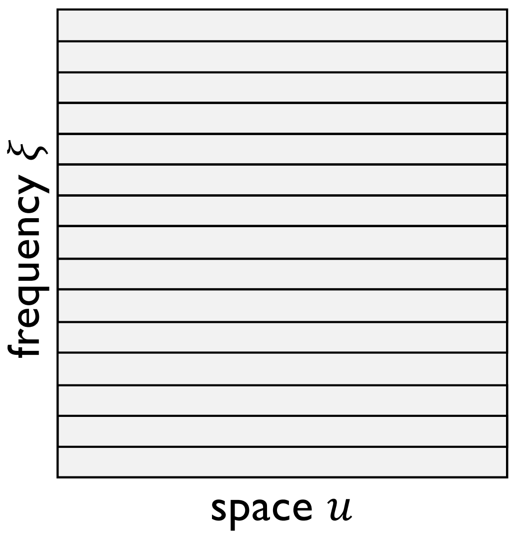
Wavelet Transform

$$(Wx)(u,\lambda) = \lambda^{-1/2} \int_{-\infty}^{+\infty} \psi\left(\frac{v-u}{\lambda}\right) x(v)\mathrm{d}v\,, \qquad \lambda = 2^{-j}, u = 2^{-j}k$$

- Convolution with a **filter bank** $Wx = \{x \star \phi,\ x \star \psi_\lambda\}_\lambda$ where ϕ is LP filter ("averaging")
- Appropriate design choice guarantees **unitarity** $\|Wx\| = \|x\|$.
- $x \star \psi_\lambda$ is **translation-equivariant**
- $\|x \star \psi_\lambda\|_1 = \int_{-\infty}^{+\infty} |(x \star \psi_\lambda)(u)|\mathrm{d}u$ is a **stable** and **translation-invariant**… however, it looses a lot of information!

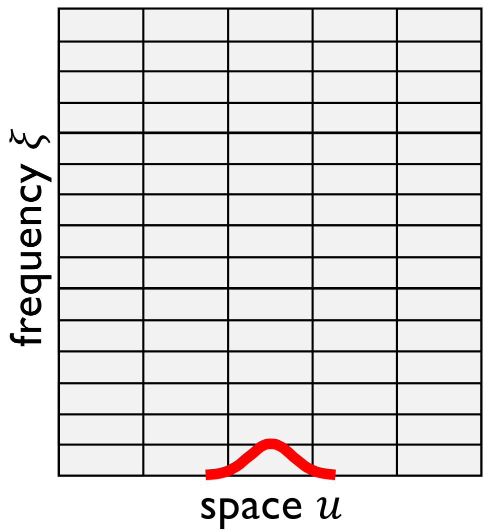
Note: Absolute value is special, and can be derived from deformation stability + unitarity (see Mallat's lecture)

Properties of Wavelets

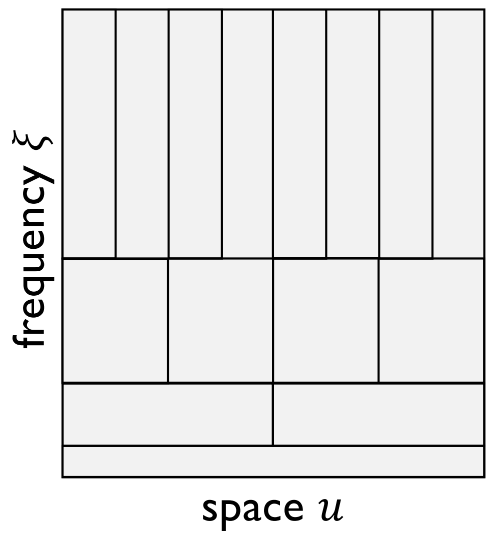


Fourier

$|\hat{\phi}(\xi)|$
translation invariant



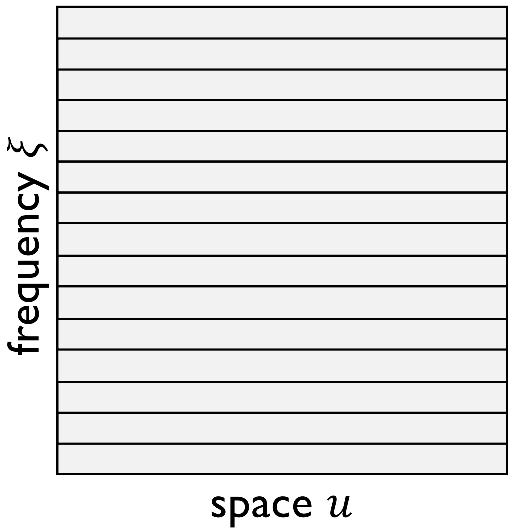
Windowed Fourier



Wavelet

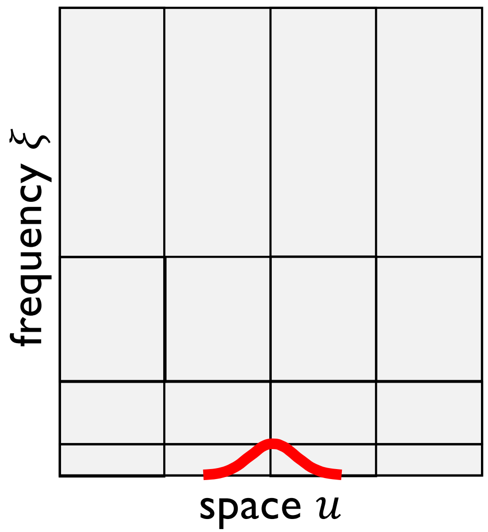
$\{|x \star \psi_\lambda|\}_\lambda$
translation equivariant

Properties of Wavelets



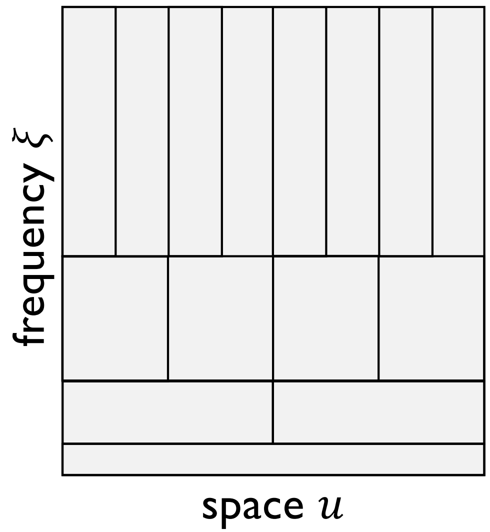
Fourier

$|\hat{\phi}(\xi)|$
translation invariant



Wavelet+averaging

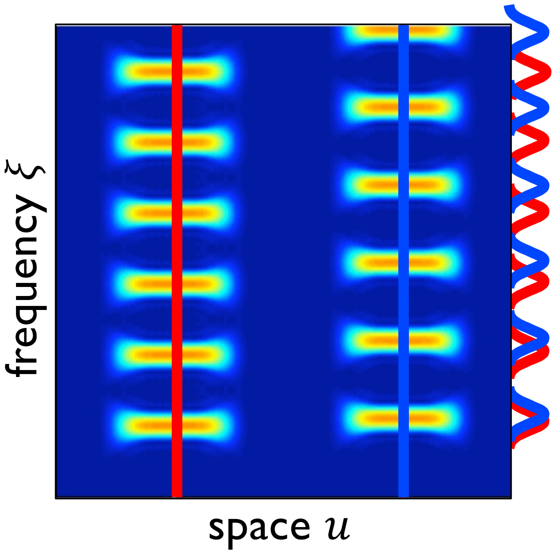
$\{|x \star \psi_\lambda| \star \phi\}_\lambda$
locally translation-invariant
deformation stable



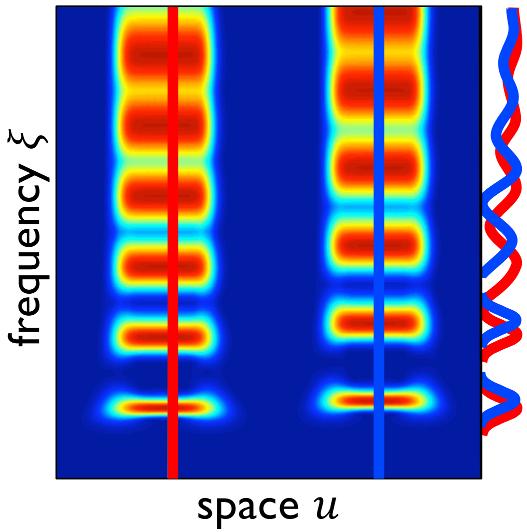
Wavelet

$\{|x \star \psi_\lambda|\}_\lambda$
translation equivariant

Effect of Averaging

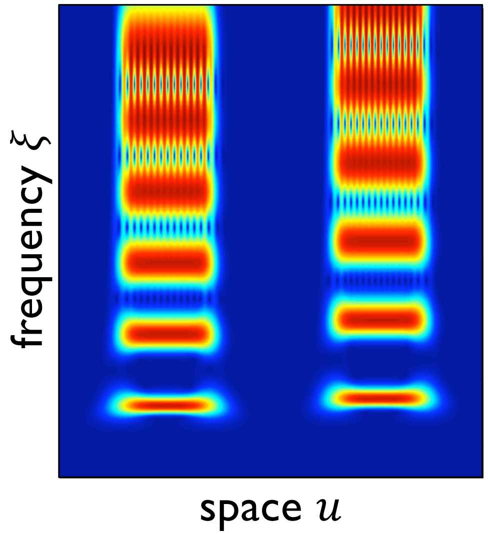


Windowed Fourier



Wavelet+averaging

$\{|x \star \psi_\lambda| \star \phi\}_\lambda$

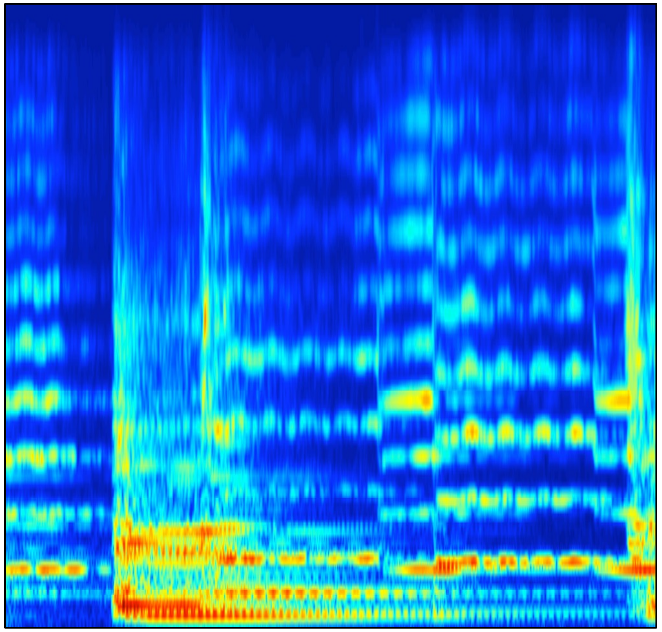


Wavelet

$\{|x \star \psi_\lambda|\}_\lambda$

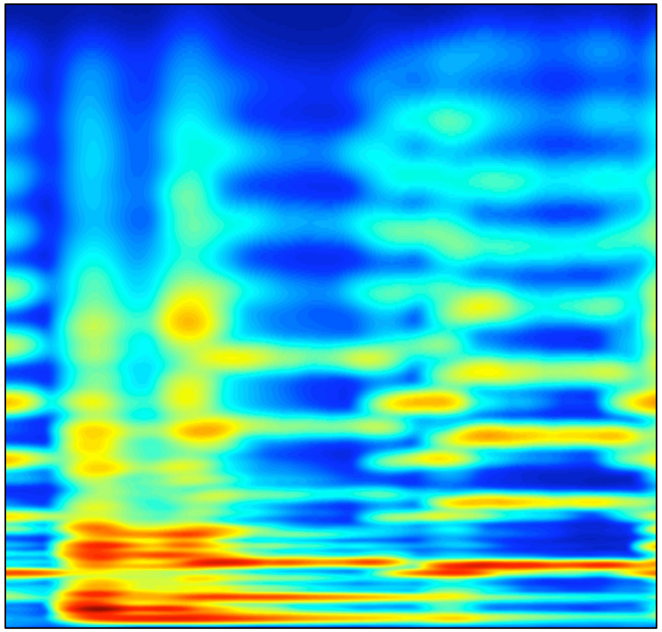
Image: Mallat (IPAM lectures)

Effect of Averaging



Wavelet

$\{|x \star \psi_\lambda|\}_\lambda$

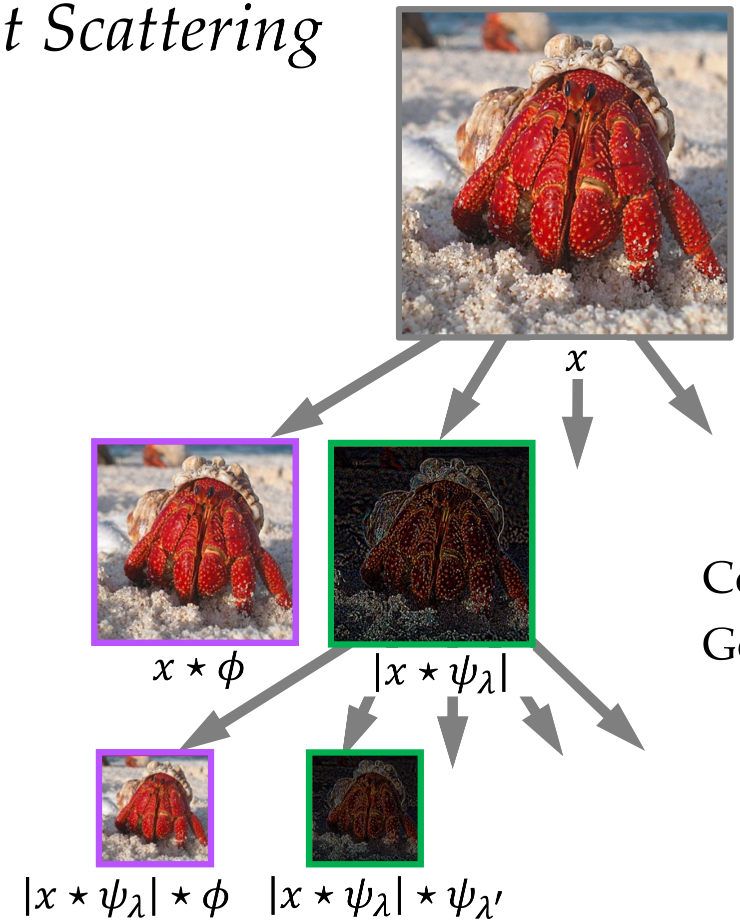


Wavelet+averaging

$\{|x \star \psi_\lambda| \star \phi\}_\lambda$

Image: Mallat (IPAM lectures)

Wavelet Scattering



Scattering transform: iteration of

$$Ux = \{x \star \phi, |x \star \psi_\lambda|\}_\lambda$$

Combine the three main building blocks of Geometric Deep Learning Blueprint:

- Linear invariant: lowpass filter $x \star \phi$
- Linear equivariant: wavelet filter bank $\{x \star \psi_\lambda\}_\lambda$
- Element-wise nonlinearity: $\rho = |\quad|$

Mallat 2010; Bruna, Mallat 2012

Wavelet Scattering Transform

$$Ux = \left|\left|\dots\left|\left|x \star \psi_{\lambda_1}\right| \star \psi_{\lambda_2}\right|\dots\right| \star \psi_{\lambda_2}\right| \star \phi$$

- "Geometric deep learning without learning"
- Progressively pushes high-frequency information to low-frequency

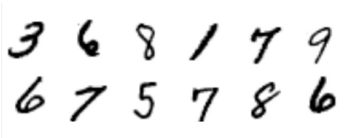
Theorem: With appropriate selection of wavelets, the Scattering Transform U is:

- Contracting $\|Ux - Uy\| \leq \|x - y\|$
- Energy-preserving $\|Ux\| = \|x\|$
- Stable to deformations $\|Ux - U(\tau x)\| \leq c\|\nabla\tau\|\|x\|$

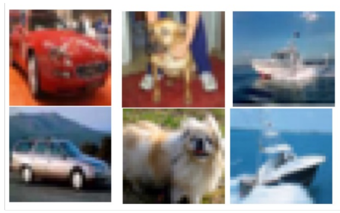
Bruna, Mallat 2012

Wavelet Scattering Transform

MNIST



CIFAR-10



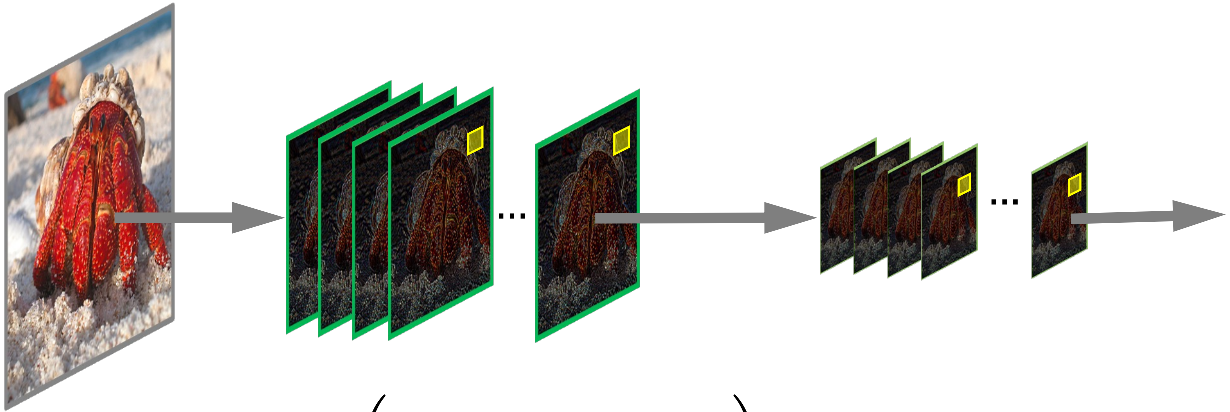
ImageNet

	MNIST	CIFAR-10	ImageNet
Deep net	0.5%	8%	11%
Scattering	0.5%	23%	52%

What is missing?

Bruna, Mallat 2012

Hybrid Scattering & Convolutional Neural Networks

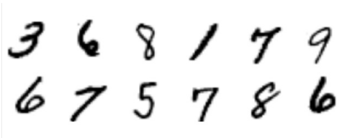


$$x^{(k+1)} = \rho\left(\sum_{\lambda} \boldsymbol{\theta}_{\lambda}\left(x^{(k)} \star \psi_{\lambda}\right)\right), \qquad \theta \in \mathbb{R}^{C_m \times C_{m+1}}$$

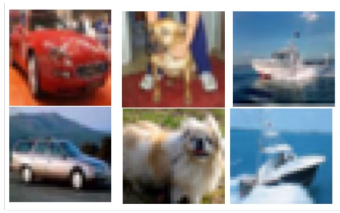
Oyallon et al. 2019; Zarka et al. 2020

Hybrid Scattering

MNIST



CIFAR-10



ImageNet

	MNIST	CIFAR-10	ImageNet
Deep net	0.5%	8%	11%
Scattering	0.5%	23%	52%
Hybrid scattering	0.5%	8%	11%

Oyallon et al. 2019; Zarka et al. 2020

Takeaways

- Linear translation-equivariants are *convolutions*
- Convolutions are diagonalised in the *Fourier basis* (=eigenvectors of the shift)
- Fourier invariants are *unstable*
- *Wavelet scattering* is a simple instance of Geometric Deep Learning without training, highlighting the importance of the Scale Separation prior
- Convolutional Neural Networks: culmination of many years of fine-tuning, strikes the right balance between geometric priors and expressive power
- Next lectures: extensions of convolutions *&* related concepts to other objects (homogeneous space, manifolds, meshes, and graphs)

Key Concepts

- Convolution
- Fourier transform
- Uncertainty principles
- Wavelets & Wavelet scattering
- Convolutional Neural Networks

Main References

- M. Bronstein et al., Geometric deep learning, *arXiv:2104.13478, 2021*. Section 4.1 "Graphs and sets" and Section 5.3 "Graph neural networks"
- B. Bamieh, Discovering Transforms: A Tutorial on circulant matrices, circular convolution, and the Discrete Fourier Transform, *arXiv:1805.05533*, 2018. Derivation of convolutions and Fourier transforms. For a shorter version, see the blog post M. Bronstein, Deriving convolution from first principles, *Towards Data Science* 2020.
- J. Bruna, S. Mallat, Invariant scattering convolutional networks, *PAMI* 35(8):1872–1885, 2013. Scattering transforms and geometric stability