# Scattering Bricks to Build Invariants

*Joan Bruna, Joakim Anden, Stéphane Mallat*
*Laurent Sifre, Irène Waldspurger*

**École Normale Supérieure**

# High Dimensional Classification

*CalTech 101*

Anchor

Joshua Tree

Beaver

Lotus

Water Lily

- Considerable variability in each class: **not low-dimensional**
- Euclidean distances are meaningless on **raw data**
- Need to find Informative Invariants.

# Curse of Dimensionality

- Analysis in high dimension: $x \in \mathbb{R}^d$ with $d \geq 10^6$.

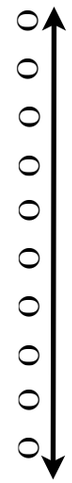
- Points are far away in high dimensions $d$:

# Curse of Dimensionality

- Analysis in high dimension: $x \in \mathbb{R}^d$ with $d \geq 10^6$.

- Points are far away in high dimensions $d$:

  - 10 points cover $[0, 1]$ at a distance $10^{-1}$



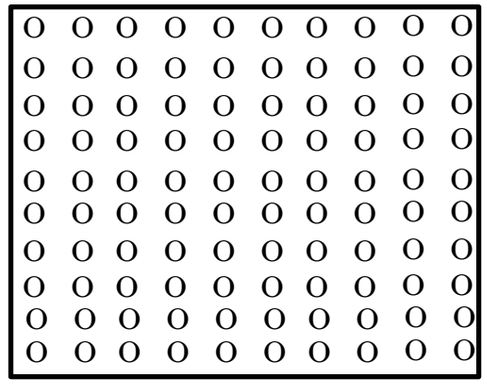
# Curse of Dimensionality

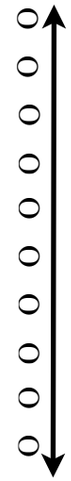
- Analysis in high dimension: $x \in \mathbb{R}^d$ with $d \geq 10^6$.

- Points are far away in high dimensions $d$:

  - 10 points cover $[0,1]$ at a distance $10^{-1}$

  - 100 points for $[0,1]^2$





# Curse of Dimensionality

- Analysis in high dimension: $x \in \mathbb{R}^d$ with $d \geq 10^6$.

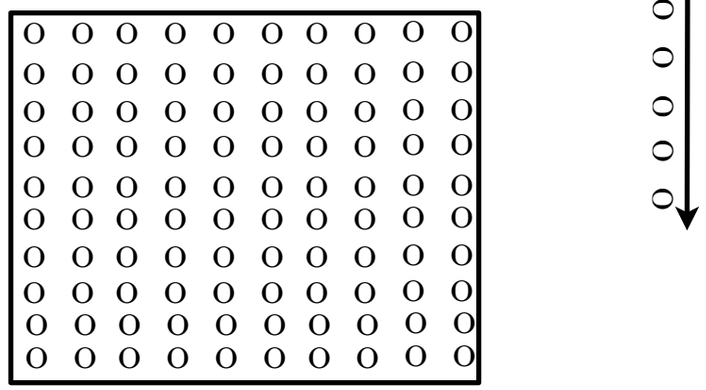
- Points are far away in high dimensions $d$:

  - 10 points cover $[0,1]$ at a distance $10^{-1}$

  - 100 points for $[0,1]^2$

  - need $10^d$ points over $[0,1]^d$

    impossible if $d \geq 20$



# Curse of Dimensionality

- Analysis in high dimension: $x \in \mathbb{R}^d$ with $d \geq 10^6$.

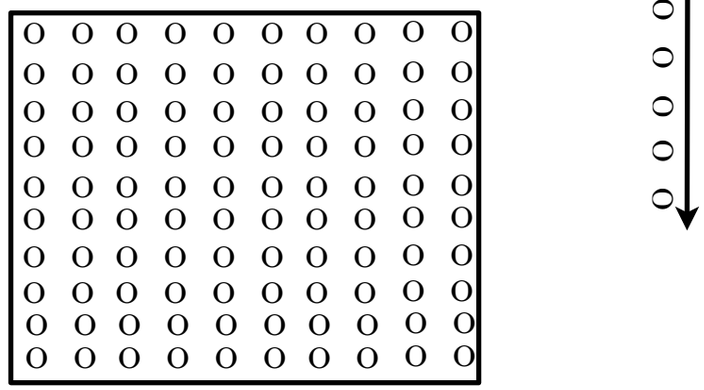
- Points are far away in high dimensions $d$:

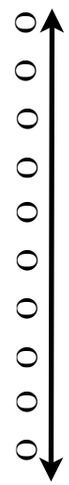
  - 10 points cover $[0,1]$ at a distance $10^{-1}$

  - 100 points for $[0,1]^2$

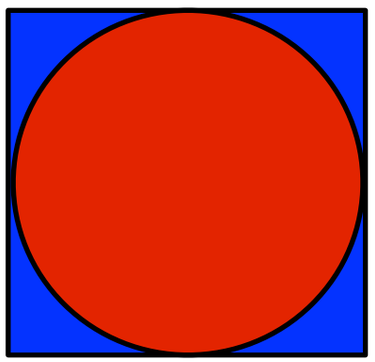
  - need $10^d$ points over $[0,1]^d$

    impossible if $d \geq 20$





$$\lim_{d \to \infty} \frac{\text{volume sphere of radius r}}{\text{volume } [0,r]^d} = 0$$



points are concentrated in $2^d$ corners!

# Curse of Dimensionality

- Analysis in high dimension: $x \in \mathbb{R}^d$ with $d \geq 10^6$.

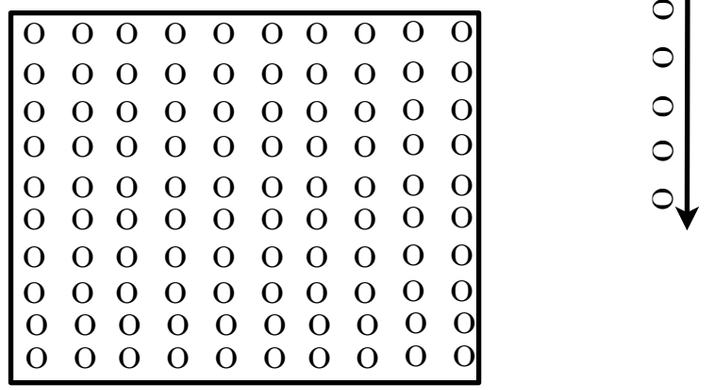
- Points are far away in high dimensions $d$:

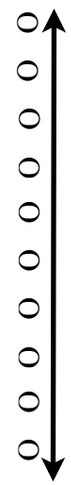
  - 10 points cover $[0,1]$ at a distance $10^{-1}$

  - 100 points for $[0,1]^2$

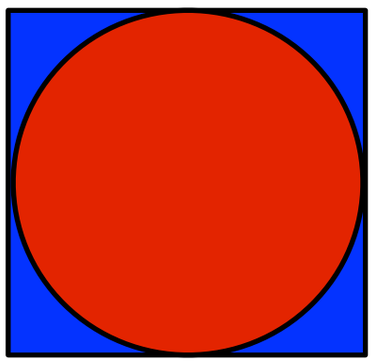
  - need $10^d$ points over $[0,1]^d$

    impossible if $d \geq 20$





$$\lim_{d \to \infty} \frac{\text{volume sphere of radius r}}{\text{volume } [0,r]^d} = 0$$



points are concentrated in $2^d$ corners!

$\Rightarrow$ Euclidean metrics are not appropriate on **raw data**.

# Kernel Learning

Representation

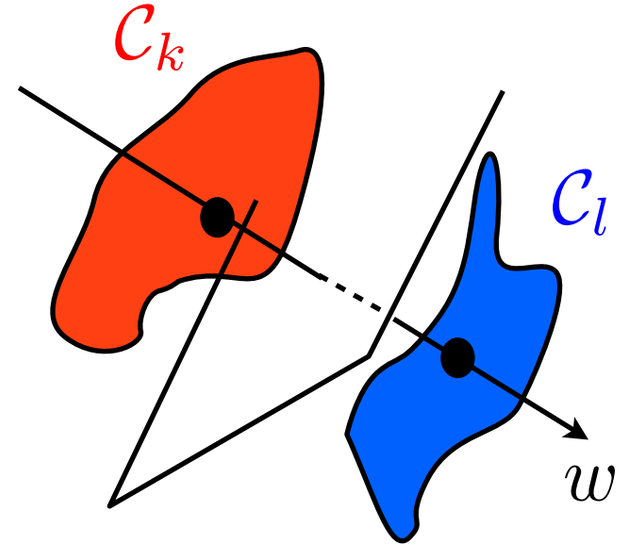
Supervised
Linear classification

$$x \longrightarrow \boxed{\Phi} \longrightarrow \Phi x = \{\Phi_n x\}_n \longrightarrow \boxed{\langle \Phi x, w \rangle \overset{?}{\geq} T} \longrightarrow \text{class}$$

Euclidean metric

For any two classes $\mathcal{C}_k$ and $\mathcal{C}_l$ finds $w$ so that

$$\langle \Phi x, w \rangle = \sum_n w_n \, \Phi_n x$$




is nearly invariant and different in any $\mathcal{C}_k$ and $\mathcal{C}_l$.

# Kernel Learning

Representation

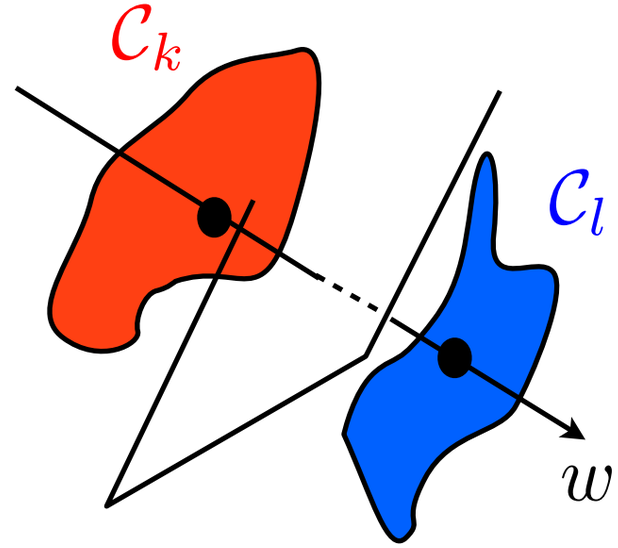
Supervised
Linear classification

$$x \longrightarrow \boxed{\Phi} \longrightarrow \Phi x = \{\Phi_n x\}_n \longrightarrow \boxed{\langle \Phi x, w \rangle \overset{?}{\geq} T} \longrightarrow \text{class}$$

Euclidean metric

For any two classes $\mathcal{C}_k$ and $\mathcal{C}_l$ finds $w$ so that

$$\langle \Phi x, w \rangle = \sum_n w_n \, \Phi_n x$$



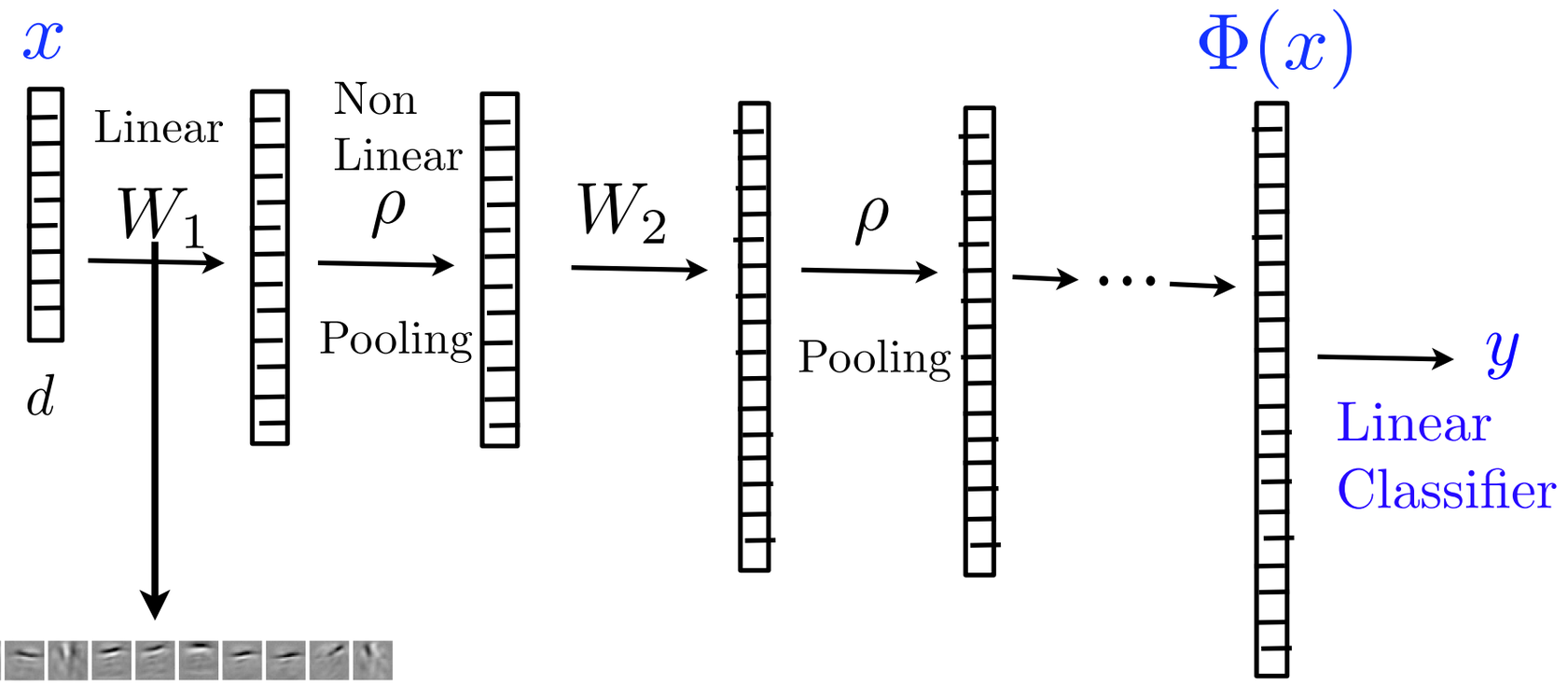

is nearly invariant and different in any $\mathcal{C}_k$ and $\mathcal{C}_l$.

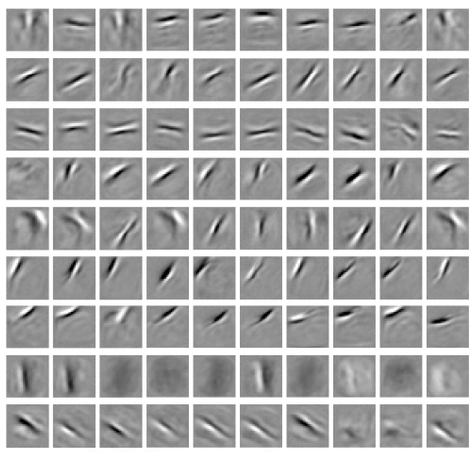
- How to construct $\Phi$ ?

# Deep Neural Networks

*Hinton, LeCun*

Hierarchical invariance



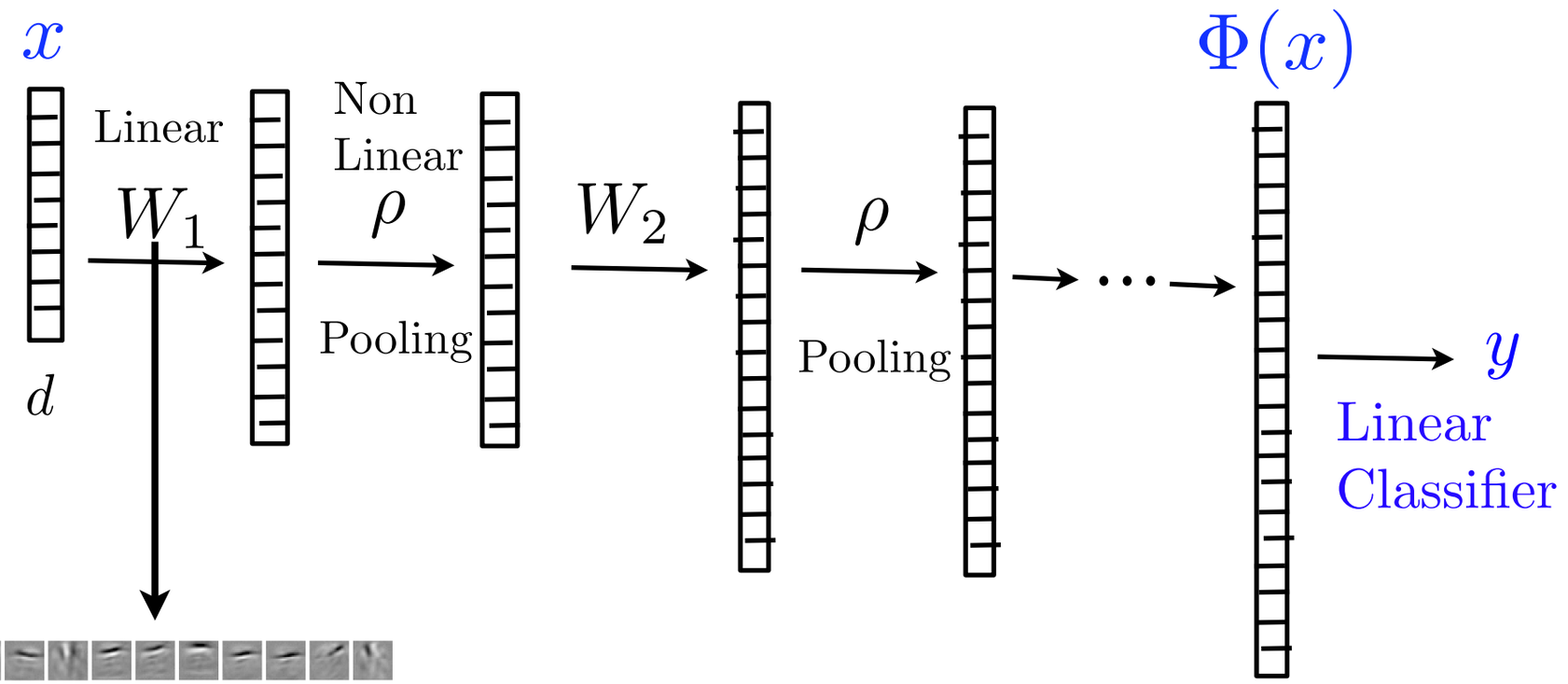



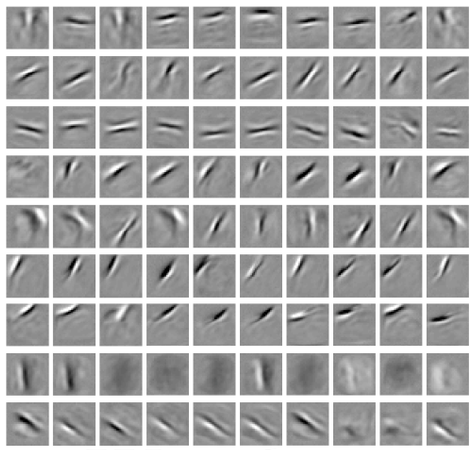
Wavelets

# Deep Neural Networks

*Hinton, LeCun*

Hierarchical invariance



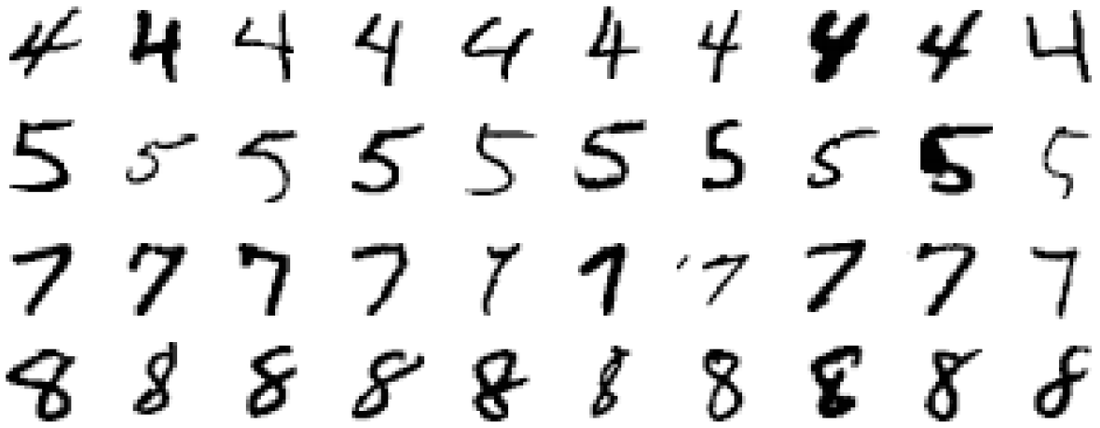



Wavelets

- Why wavelets ?
- What non-linearities ?
- Why cascading ?
- Role of reconstruction ?
- Role of sparsity ?
- How to do Unsupervised Learning ?

# Translations and Deformations

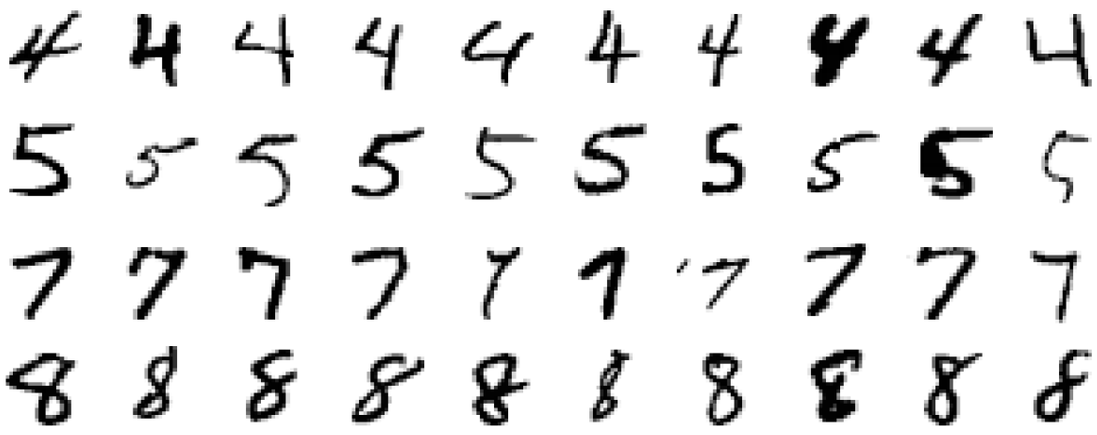
- Patterns are translated and deformed



Invariance to Translations
Two dimensional group: $\mathbb{R}^2$

# Translations and Deformations

- Patterns are translated and deformed



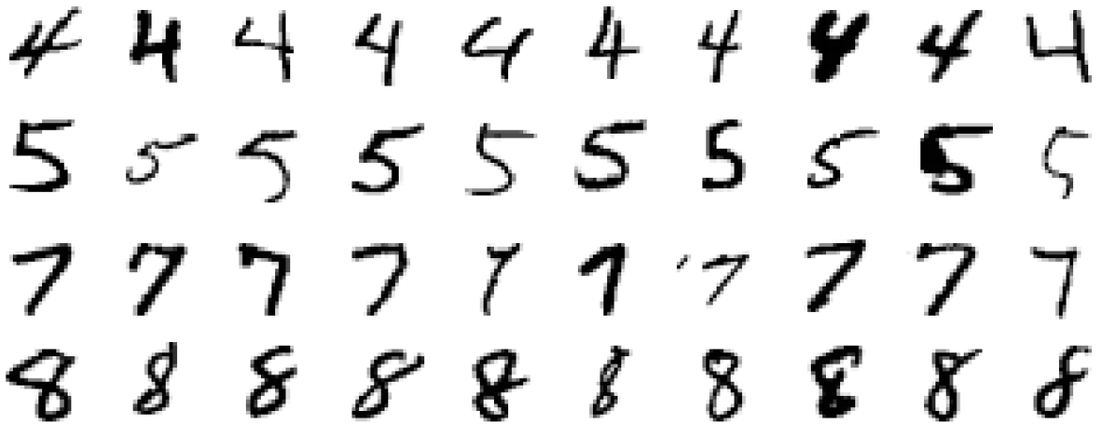
Invariance to Translations
Two dimensional group: $\mathbb{R}^2$

Deformations are actions of diffeomorphisms: infinite group.

Each digit is invariant to a specific set of small deformations

# Translations and Deformations

- Patterns are translated and deformed

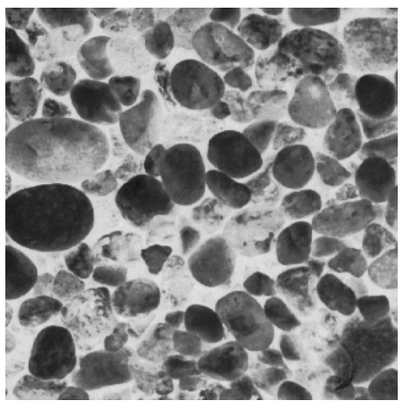
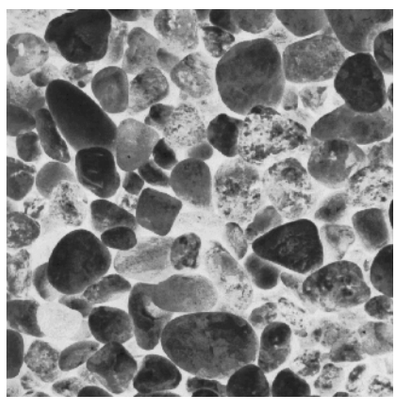
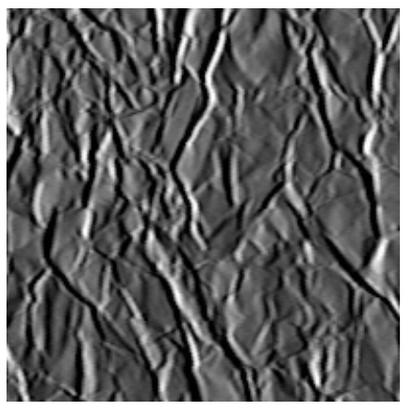


Invariance to Translations
Two dimensional group: $\mathbb{R}^2$

Deformations are actions of diffeomorphisms: infinite group.

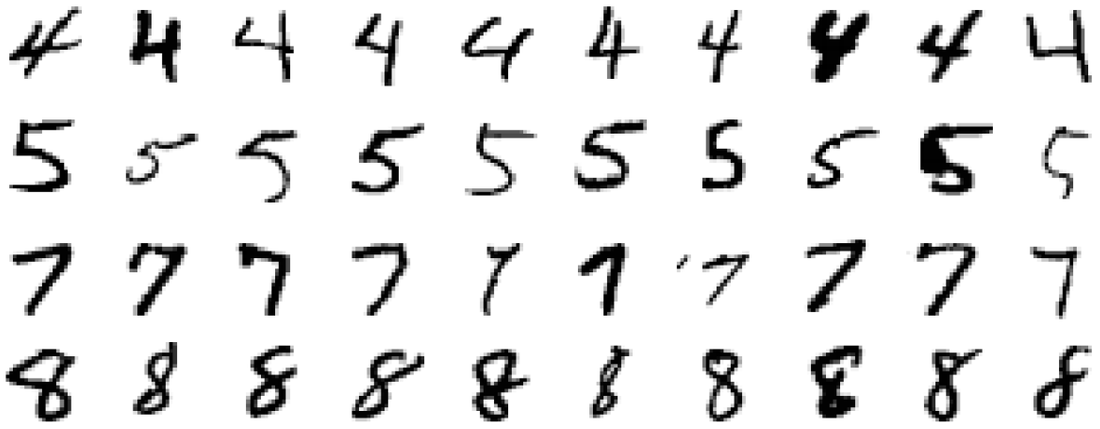
Each digit is invariant to a specific set of small deformations

- Textures are stationary (translation invariant) processes


# Translations and Deformations

- Patterns are translated and deformed



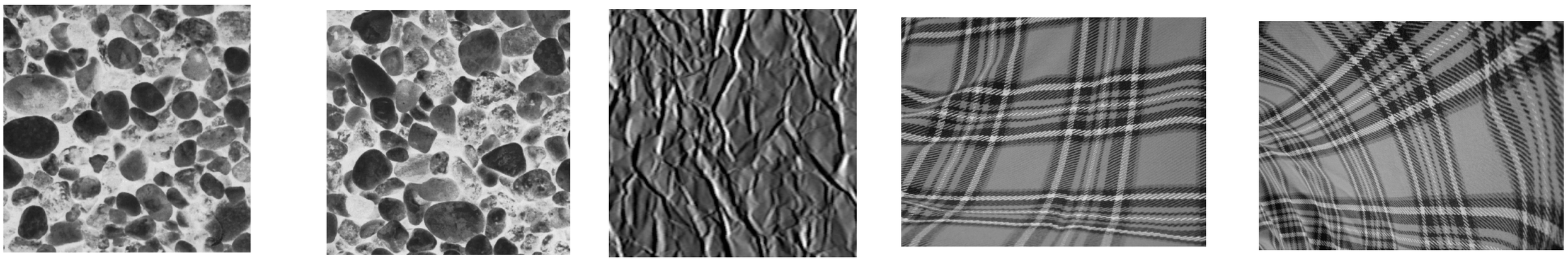
Invariance to Translations
Two dimensional group: $\mathbb{R}^2$

Deformations are actions of diffeomorphisms: infinite group.

Each digit is invariant to a specific set of small deformations

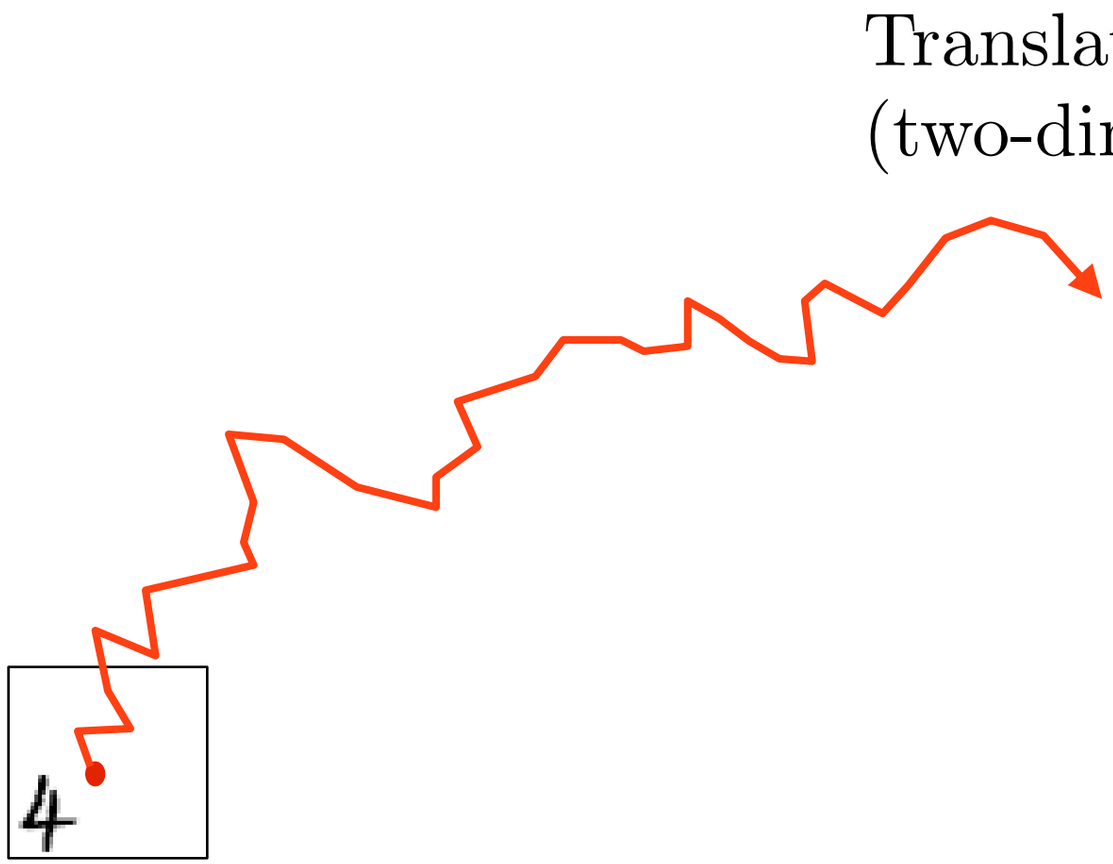
- Textures are stationary (translation invariant) processes

with deformations



# Translation Invariance

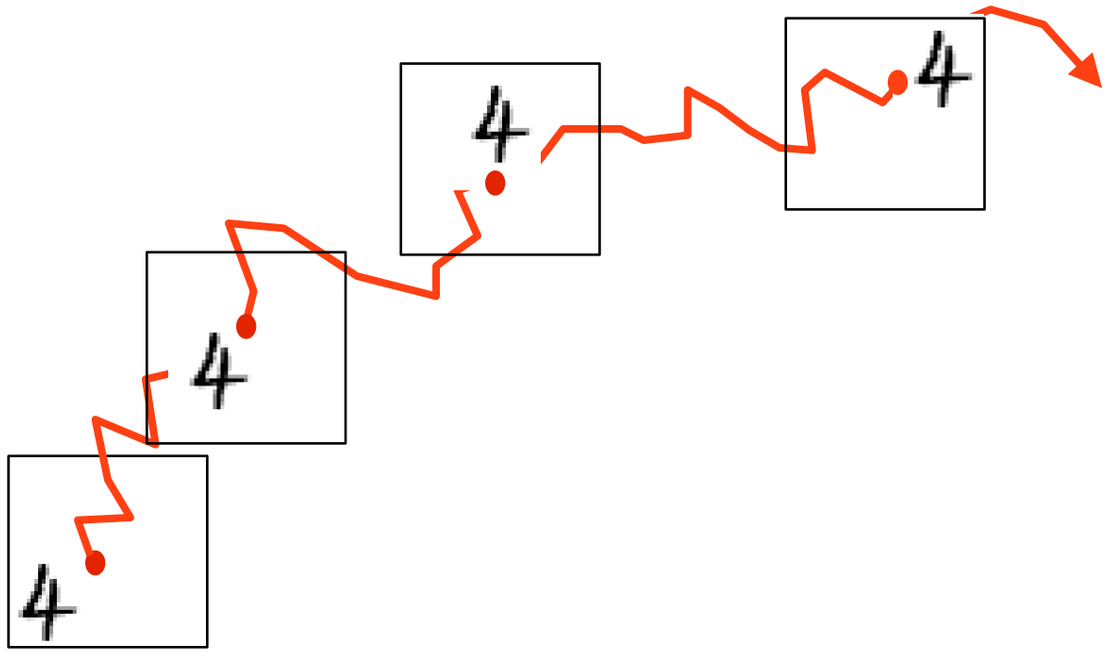
- Specific deformation invariance must be learned.



# Translation Invariance

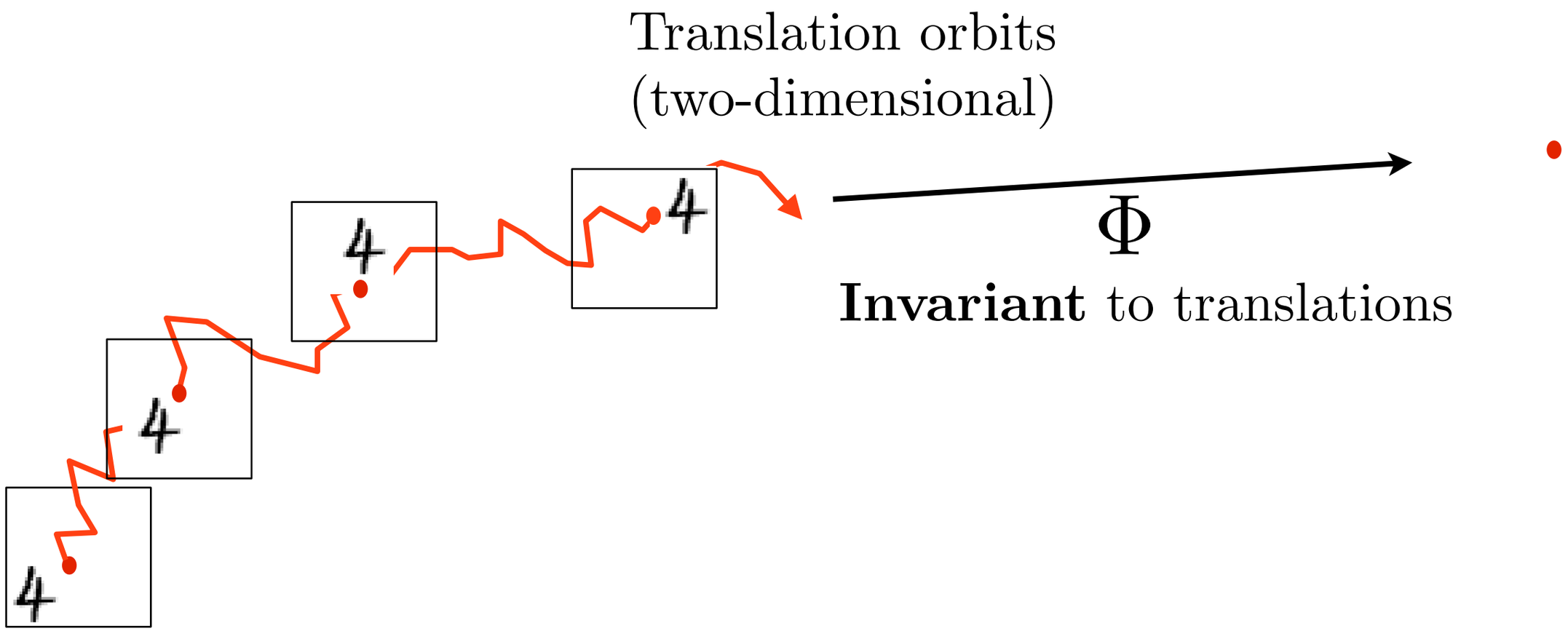
- Specific deformation invariance must be learned.

Translation orbits
(two-dimensional)



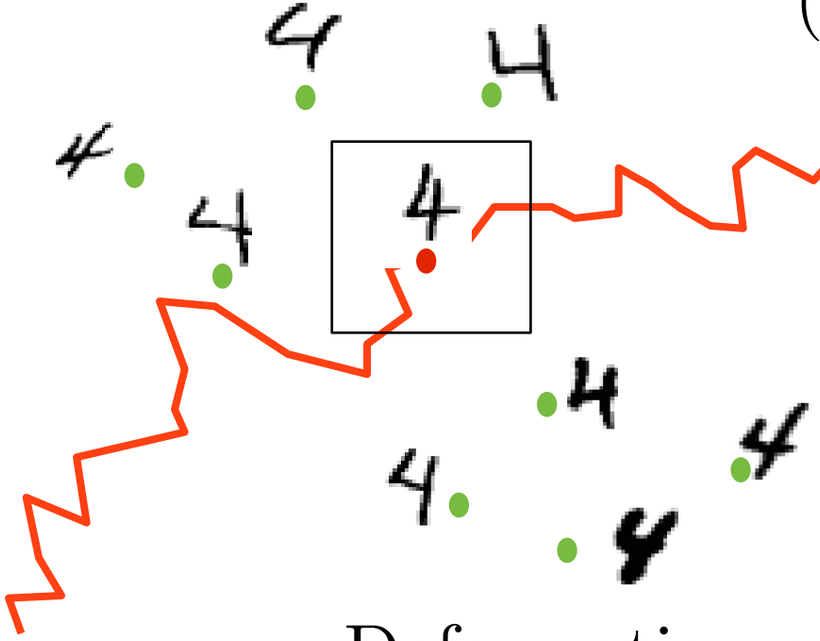
# Translation Invariance

- Specific deformation invariance must be learned.



# Translation Invariance

- Specific deformation invariance must be learned.



Translation orbits
(two-dimensional)

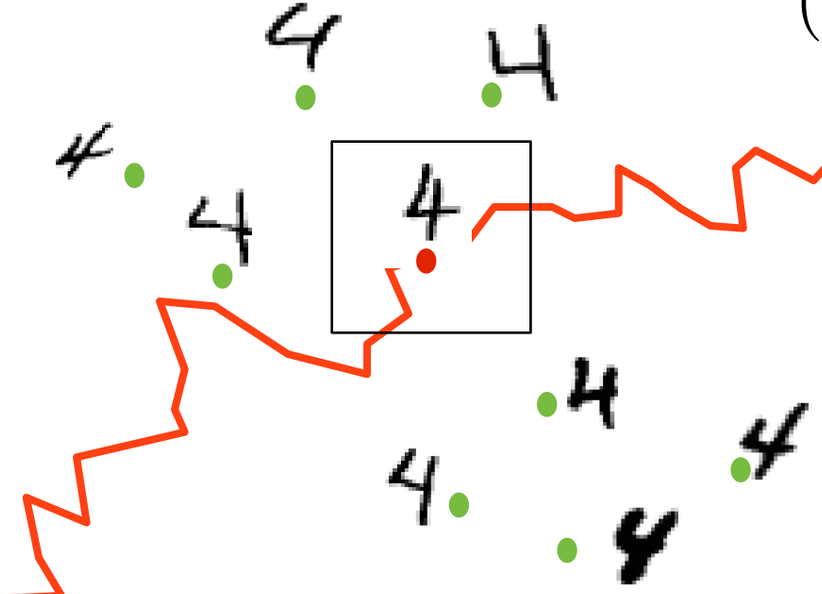
Deformation orbits
(high dimensional)

$\Phi$

**Invariant** to translations

# Translation Invariance

- Specific deformation invariance must be learned.



Translation orbits
(two-dimensional)

Deformation orbits
(high dimensional)

$\Phi$

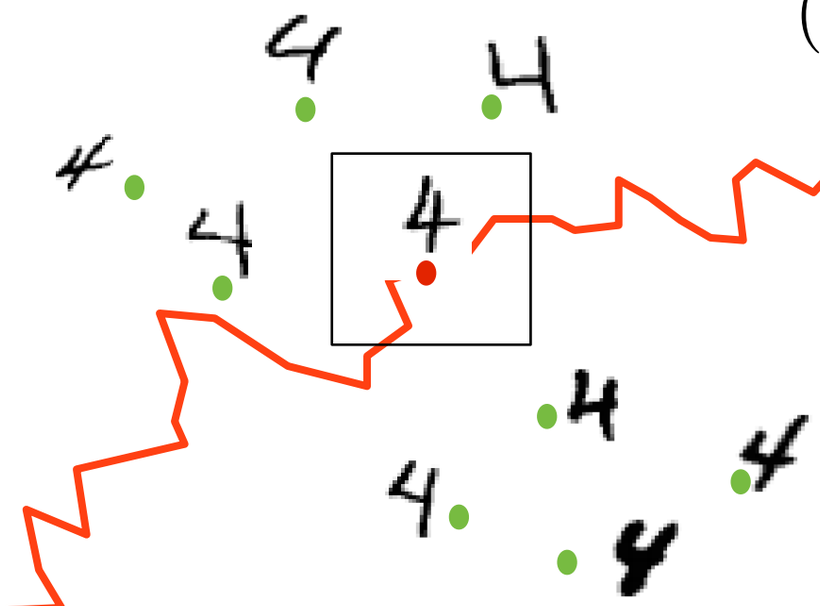
**Invariant** to translations
"**Linearizes**" deformations

# Translation Invariance

- Specific deformation invariance must be learned.

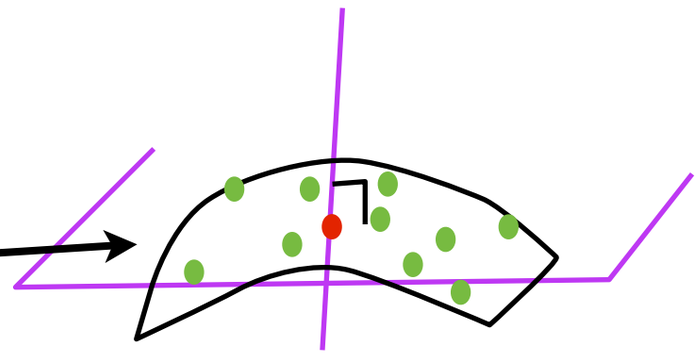
Supervised learning:

Translation orbits
(two-dimensional)



Deformation orbits
(high dimensional)

$\Phi$



**Invariant** to translations

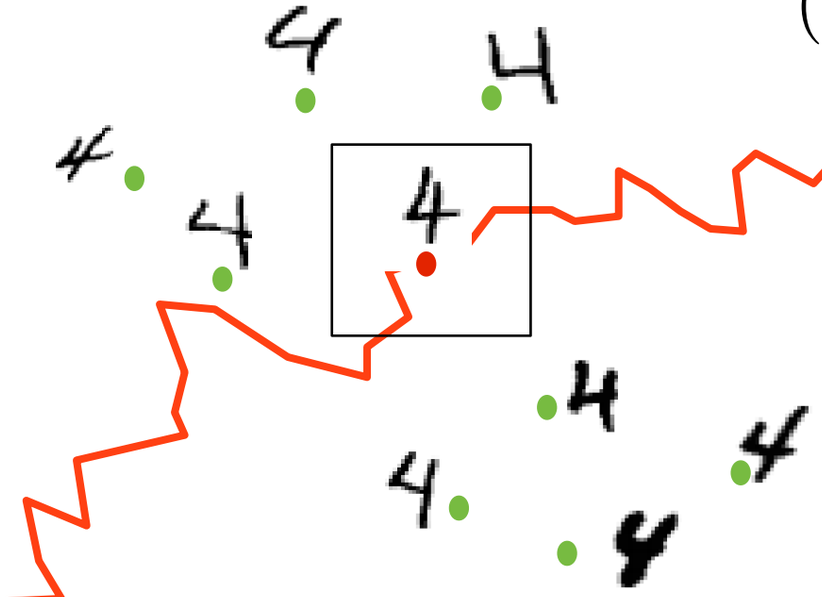
"**Linearizes**" deformations

# Translation Invariance

- Specific deformation invariance must be learned.

Supervised learning:

Translation orbits
(two-dimensional)

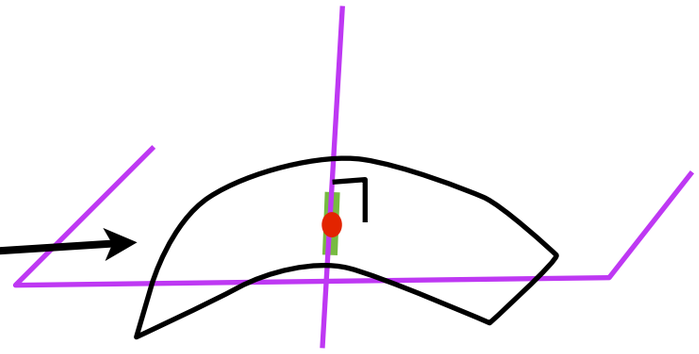


Deformation orbits
(high dimensional)

$\Phi$

**Invariant** to translations

"**Linearizes**" deformations



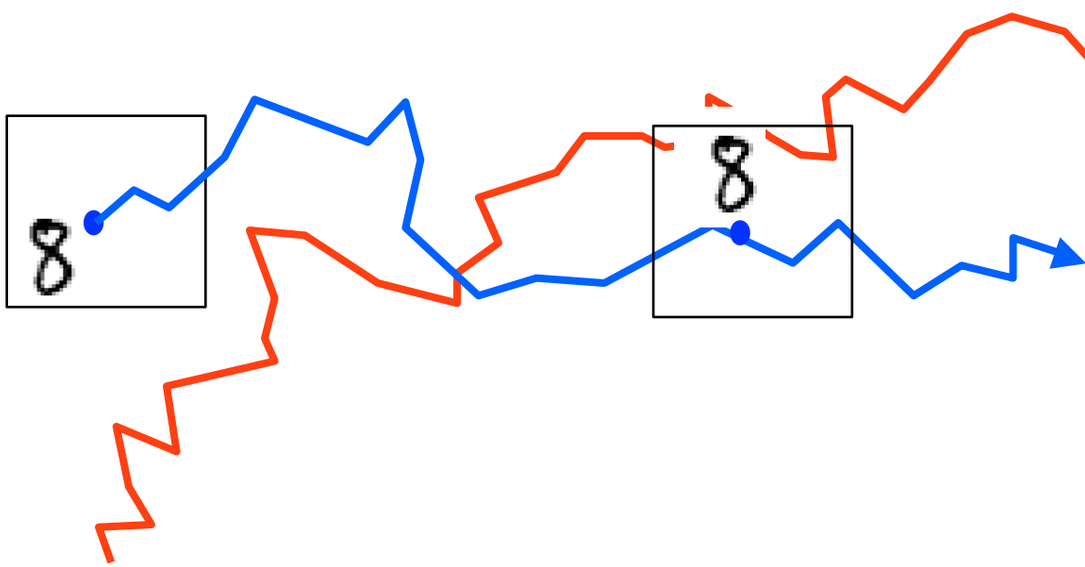
$\mathbf{P}_{\mathbf{V}_k^\perp}$
nearly invariant
to deformations

# Translation Invariance

- Specific deformation invariance must be learned.

Supervised learning:

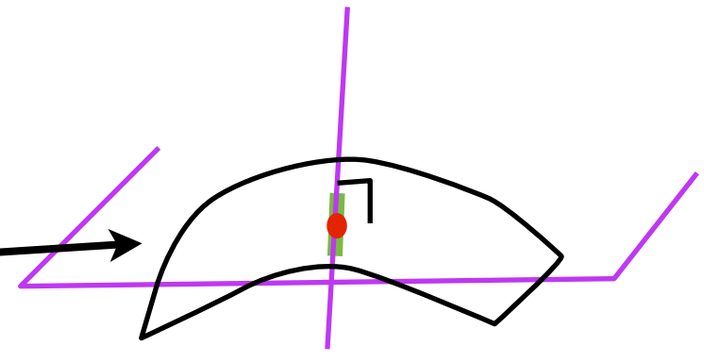
Translation orbits
(two-dimensional)



$\Phi$

**Invariant** to translations

**"Linearizes"** deformations



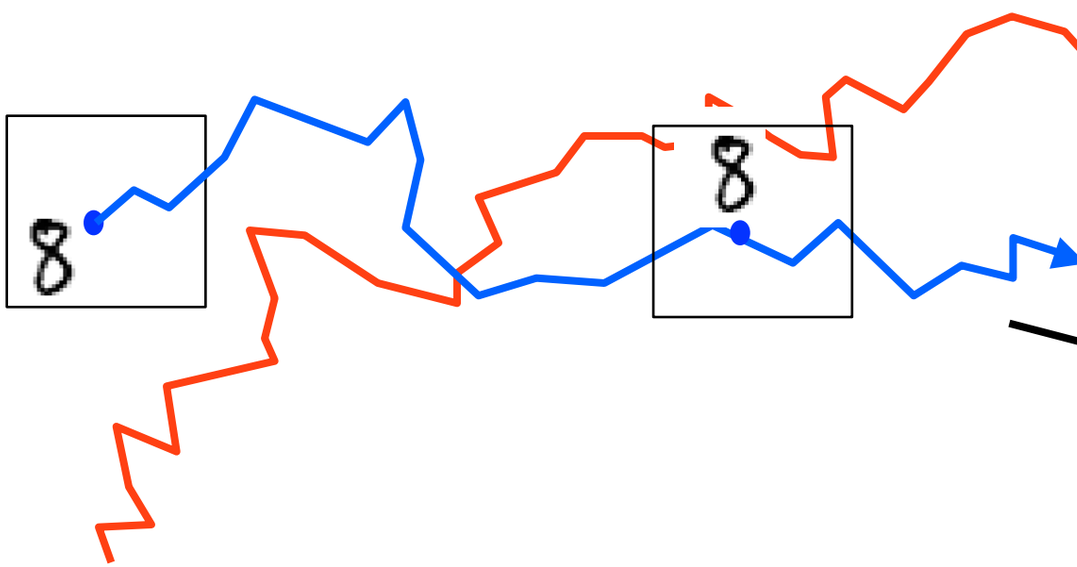
$\mathbf{P}_{\mathbf{V}_k^\perp}$
nearly invariant
to deformations

# Translation Invariance

- Specific deformation invariance must be learned.

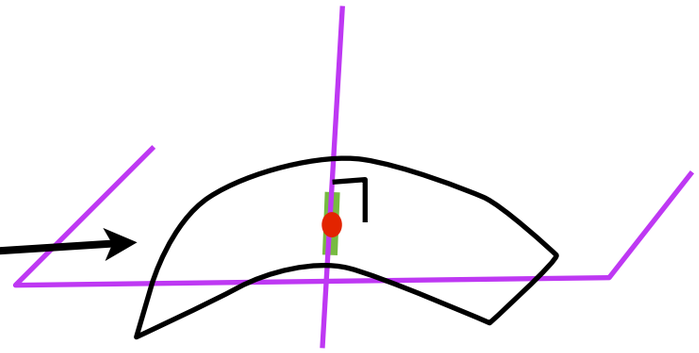
Supervised learning:

Translation orbits
(two-dimensional)



$\Phi$

**Invariant** to translations
**"Linearizes"** deformations
**Discriminant**

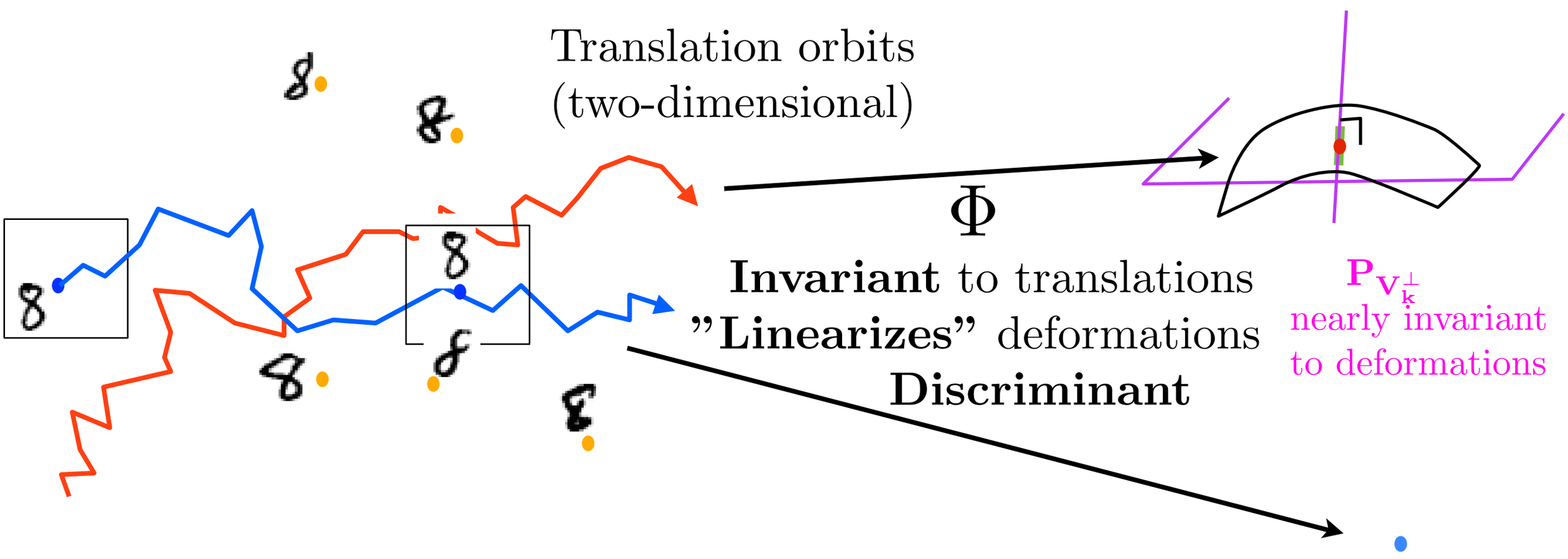


$\mathbf{P}_{\mathbf{V}_{\mathbf{k}}^{\perp}}$
nearly invariant
to deformations

# Translation Invariance

- Specific deformation invariance must be learned.

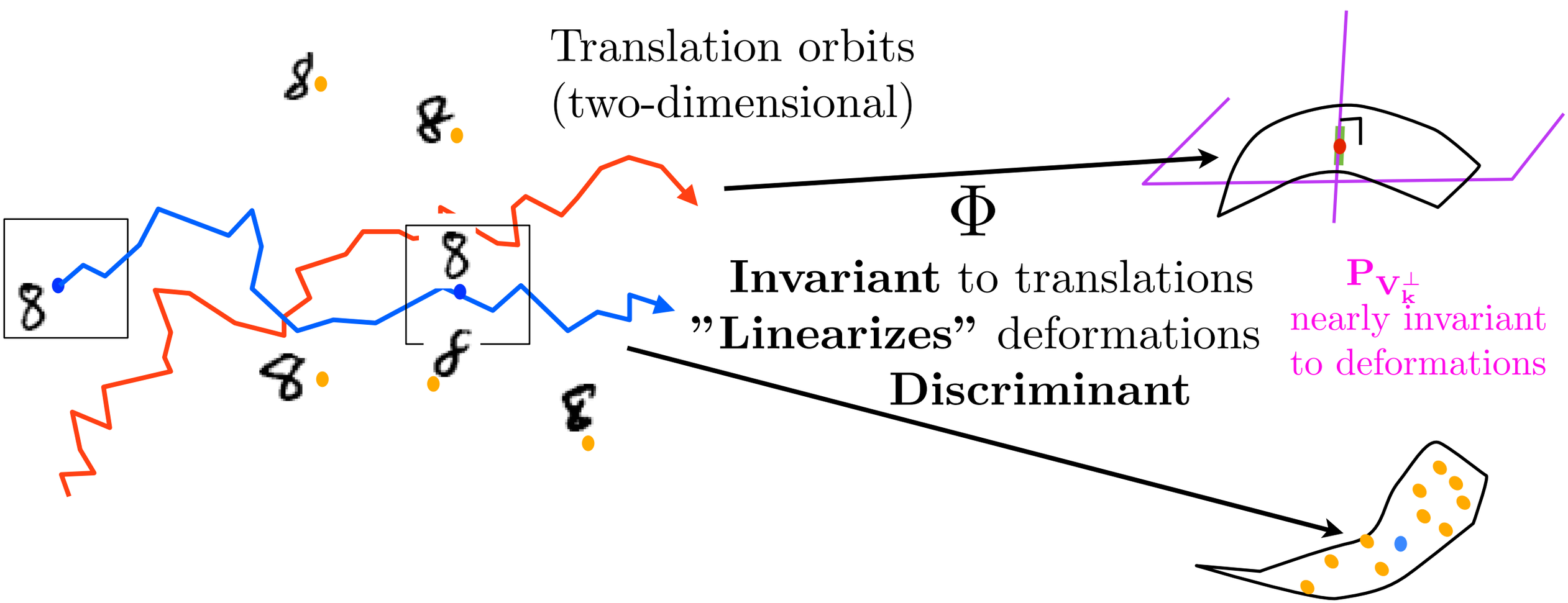
Supervised learning:



# Translation Invariance

- Specific deformation invariance must be learned.

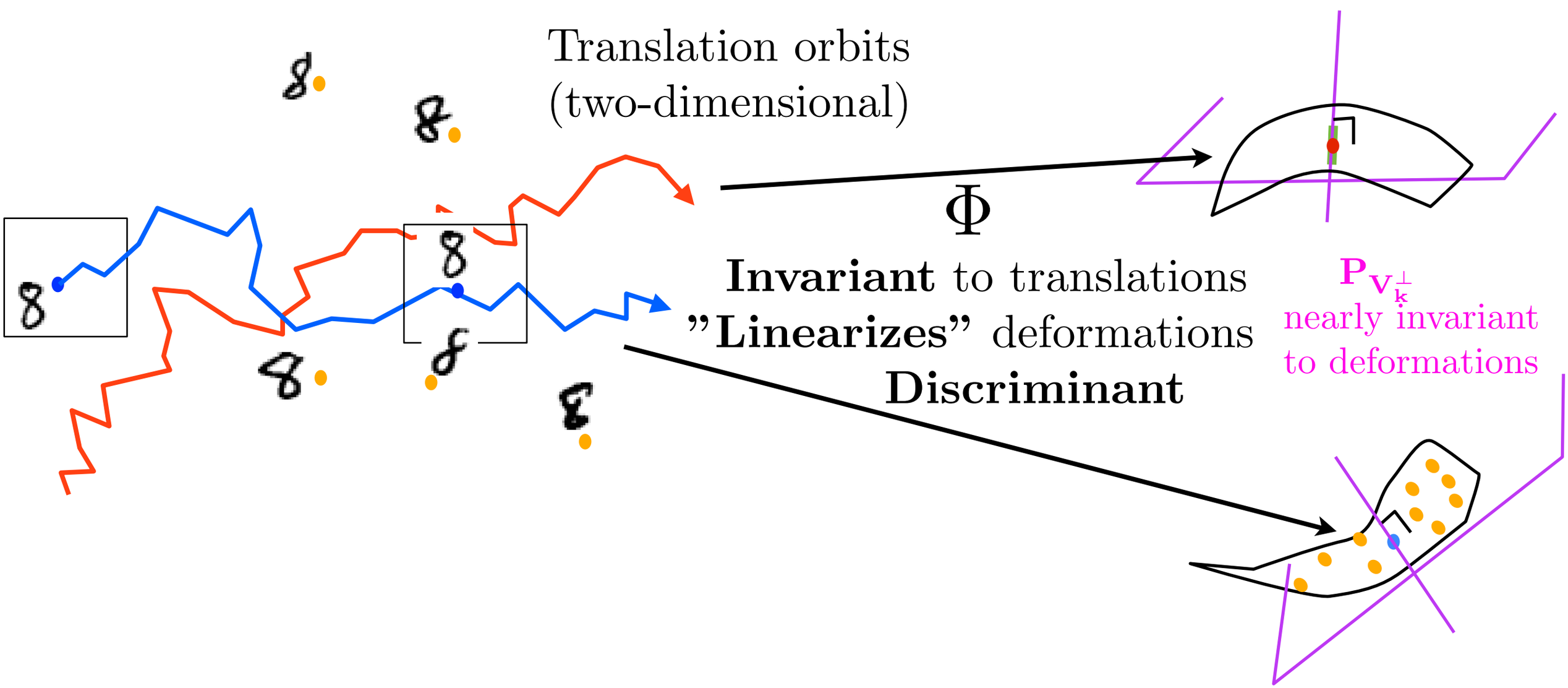
Supervised learning:



# Translation Invariance

- Specific deformation invariance must be learned.

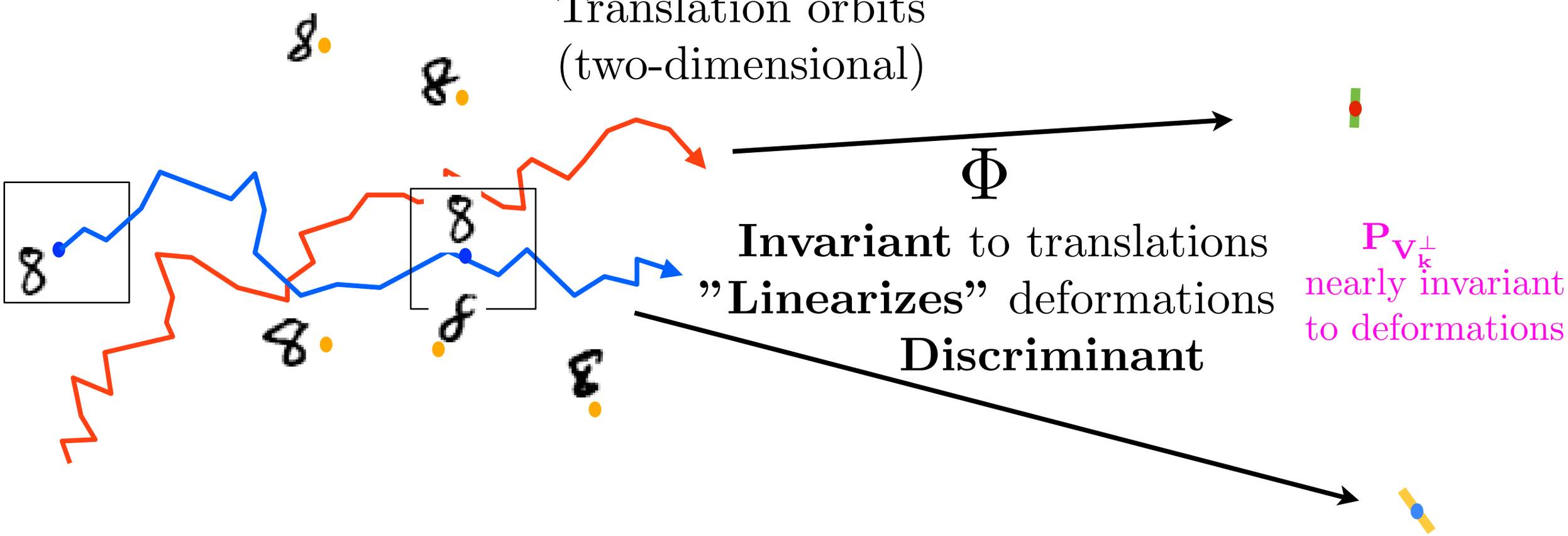
Supervised learning:



# Translation Invariance

- Specific deformation invariance must be learned.

Supervised learning:



# Stable Translation Invariants

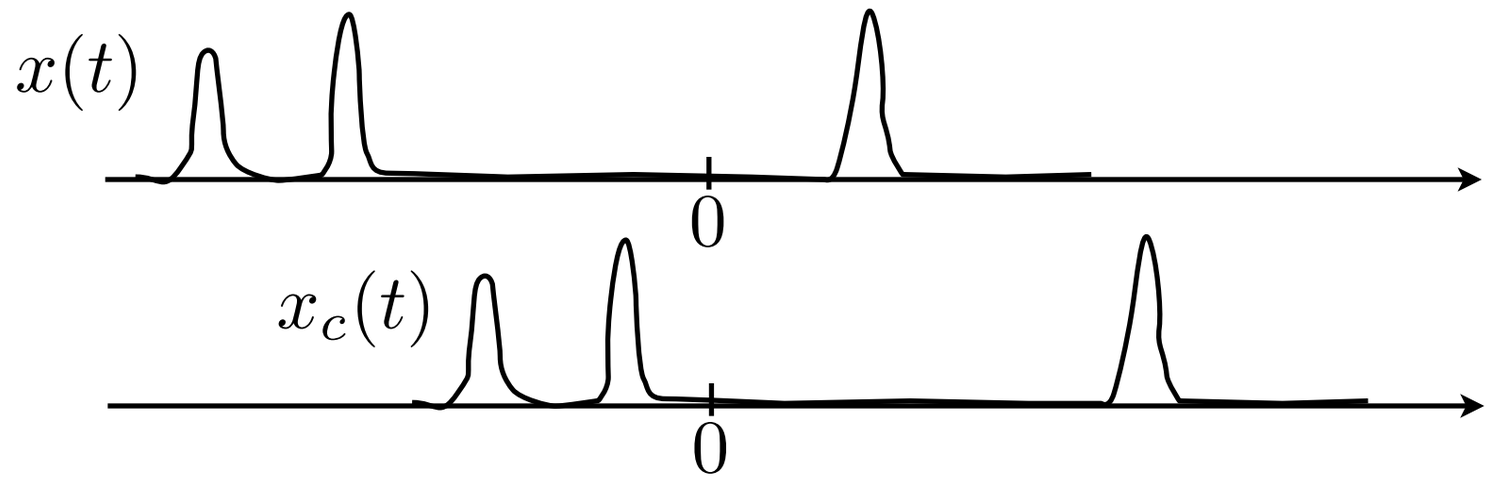
- **Invariance** to translations $x_c(t) = x(t - c)$

$$\forall c \in \mathbf{R} \quad , \quad \Phi(x_c) = \Phi(x) \ .$$

# Stable Translation Invariants

- **Invariance** to translations $x_c(t) = x(t - c)$

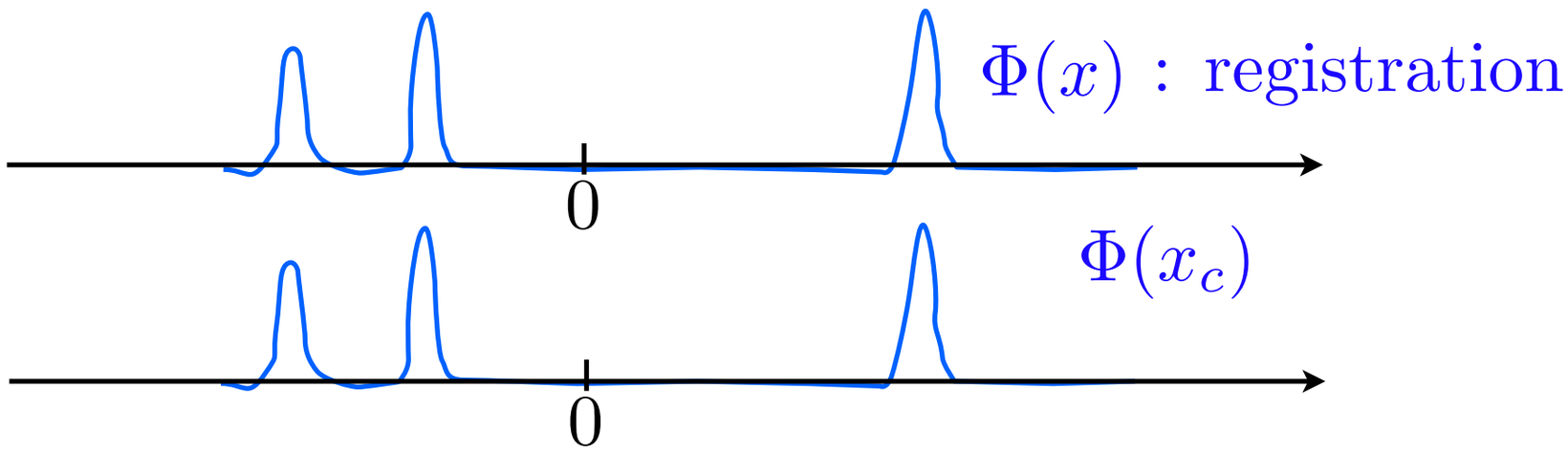
$$\forall c \in \mathbf{R} \quad , \quad \Phi(x_c) = \Phi(x) \ .$$



# Stable Translation Invariants

- **Invariance** to translations $x_c(t) = x(t - c)$

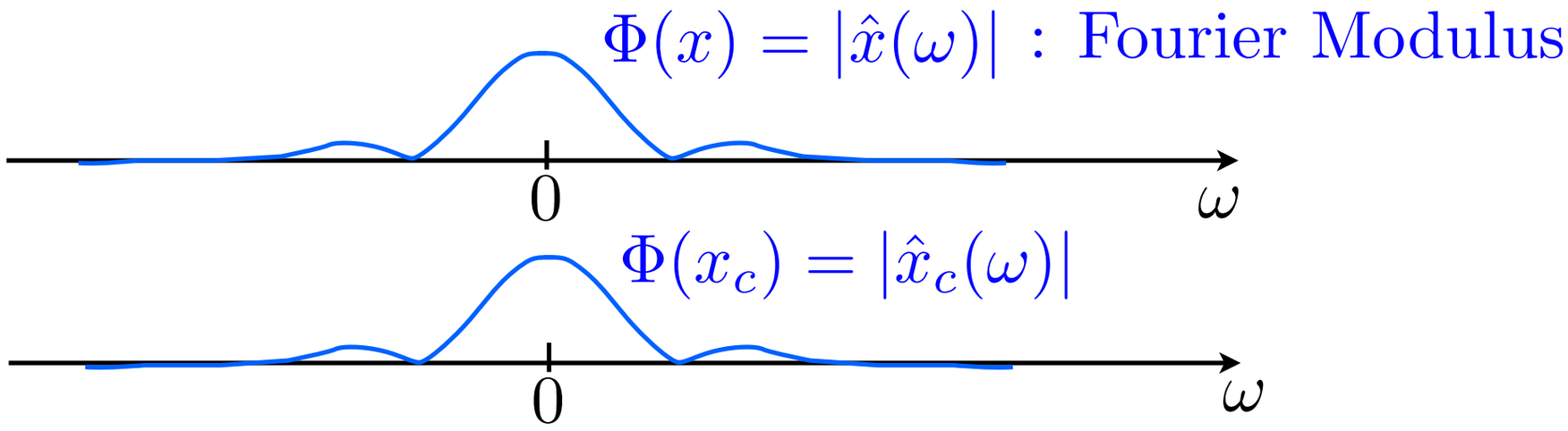
$$\forall c \in \mathbf{R} \quad , \quad \Phi(x_c) = \Phi(x) \ .$$



# Stable Translation Invariants

- **Invariance** to translations $x_c(t) = x(t - c)$

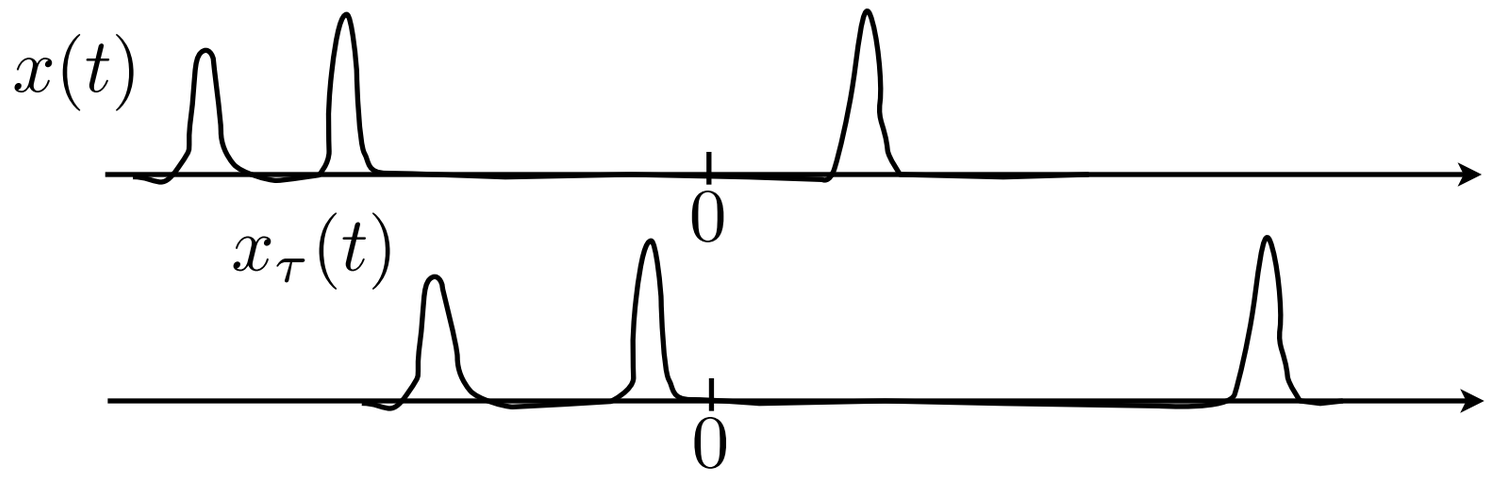
$$\forall c \in \mathbf{R} \quad , \quad \Phi(x_c) = \Phi(x) \ .$$



# Stable Translation Invariants

- **Invariance** to translations $x_c(t) = x(t - c)$

$$\forall c \in \mathbf{R} \quad , \quad \Phi(x_c) = \Phi(x) \ .$$

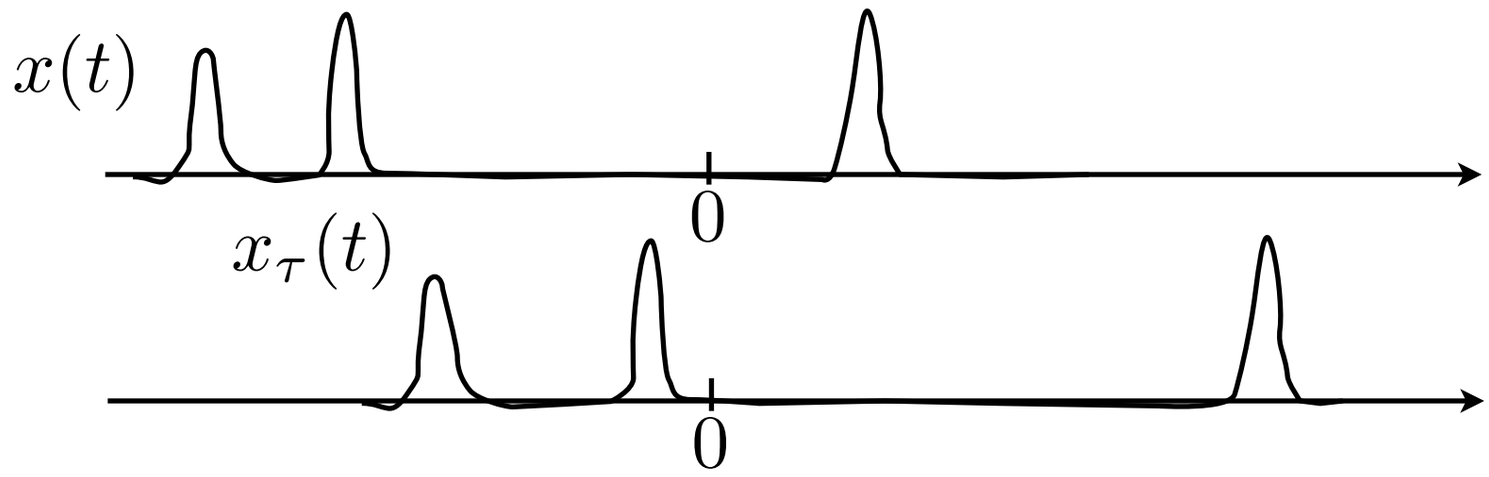



- **Lipschitz stable** to deformations $x_\tau(t) = x(t - \tau(t))$

# Stable Translation Invariants

- **Invariance** to translations $x_c(t) = x(t - c)$

$$\forall c \in \mathbf{R} \quad , \quad \Phi(x_c) = \Phi(x) \ .$$




- **Lipschitz stable** to deformations $x_\tau(t) = x(t - \tau(t))$

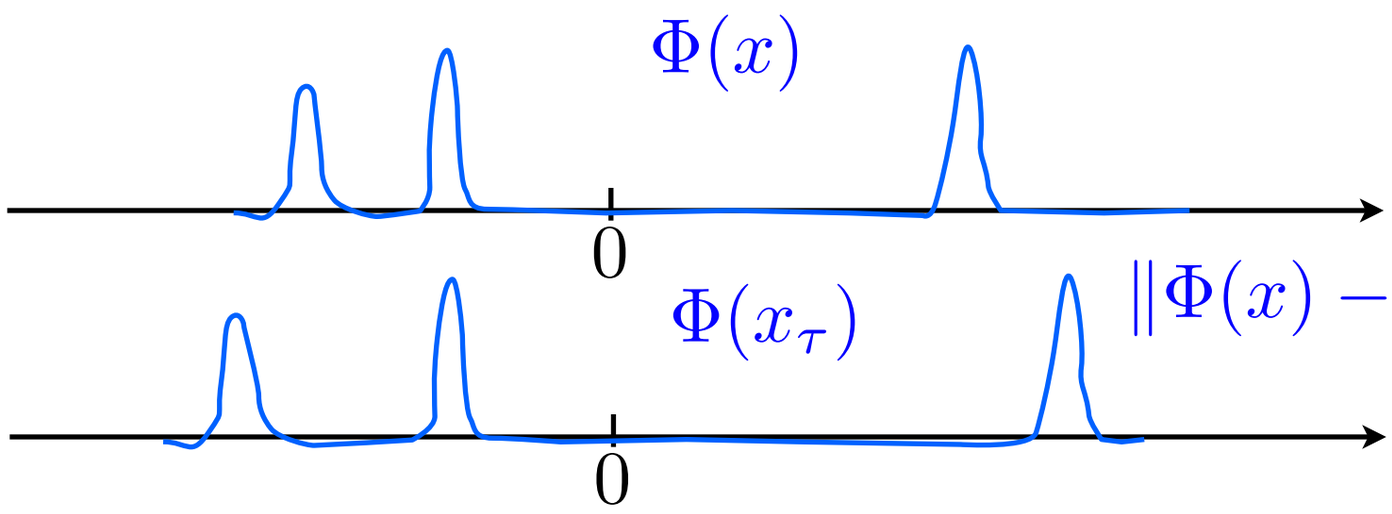
small deformations of $x$ $\implies$ small modifications of $\Phi(x)$

$$\forall \tau \quad , \quad \|\Phi(x_\tau) - \Phi(x)\| \leq C \underbrace{\sup_t |\nabla \tau(t)|}_{\text{deformation size}} \|x\| \ .$$

# Stable Translation Invariants

- **Invariance** to translations $x_c(t) = x(t - c)$

$$\forall c \in \mathbf{R} \quad , \quad \Phi(x_c) = \Phi(x) \ .$$




Not stable

$$\|\Phi(x) - \Phi(x_\tau)\| \gg \sup_t |\tau'(t)| \, \|x\|$$

- **Lipschitz stable** to deformations $x_\tau(t) = x(t - \tau(t))$

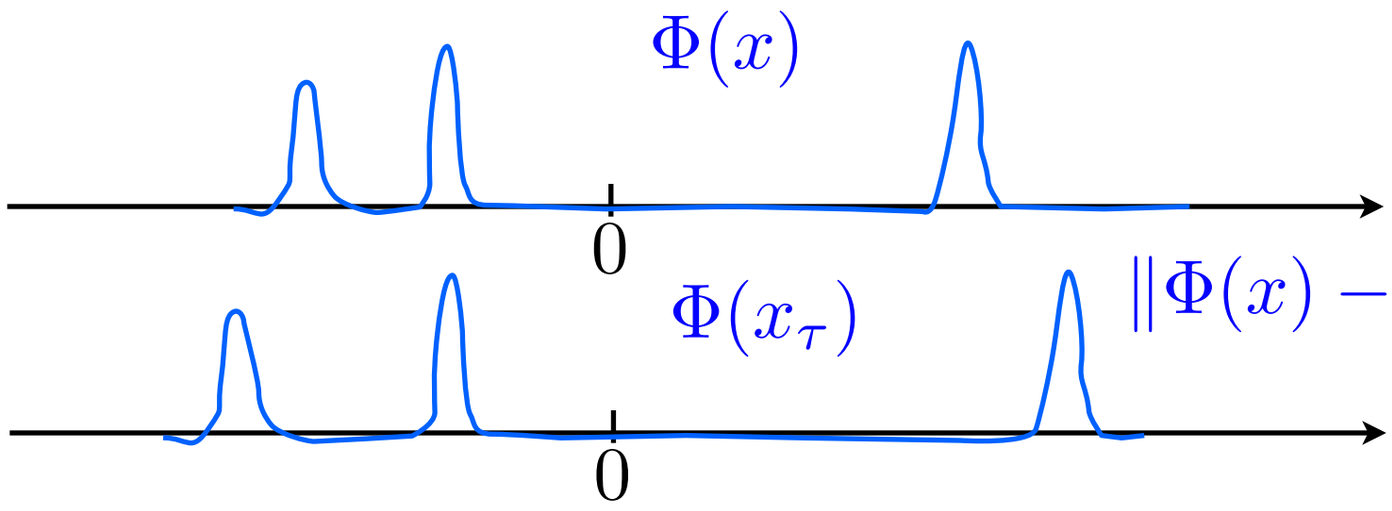
small deformations of $x \implies$ small modifications of $\Phi(x)$

$$\forall \tau \quad , \quad \|\Phi(x_\tau) - \Phi(x)\| \leq C \underbrace{\sup_t |\nabla \tau(t)|}_{\text{deformation size}} \|x\| \ .$$

# Stable Translation Invariants

- **Invariance** to translations $x_c(t) = x(t-c)$

$$\forall c \in \mathbf{R} \quad , \quad \Phi(x_c) = \Phi(x) \ .$$




Not stable

$$\|\Phi(x) - \Phi(x_\tau)\| \gg \sup_t |\tau'(t)| \, \|x\|$$

Fourier invariants are not stable either.

- **Lipschitz stable** to deformations $x_\tau(t) = x(t - \tau(t))$

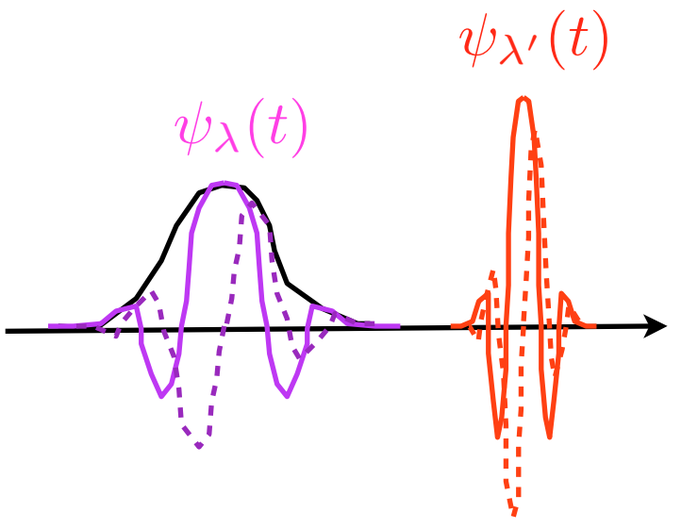
small deformations of $x \implies$ small modifications of $\Phi(x)$

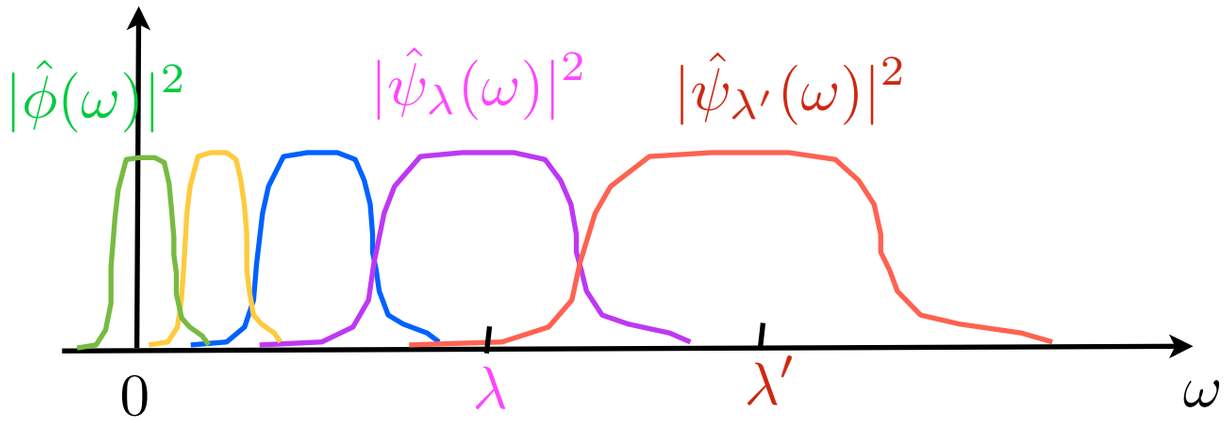
$$\forall \tau \quad , \quad \|\Phi(x_\tau) - \Phi(x)\| \leq C \sup_t |\nabla \tau(t)| \, \|x\| \ .$$

deformation size

# Wavelet Transform

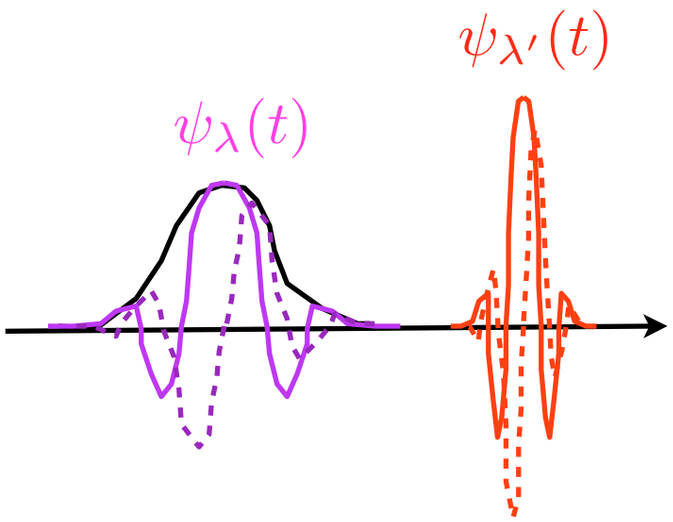
- Complex wavelet: $\psi(t) = \psi^a(t) + i\,\psi^b(t)$
- Dilated: $\psi_\lambda(t) = 2^{-j}\,\psi(2^{-j}t)$ with $\lambda = 2^{-j}$ .

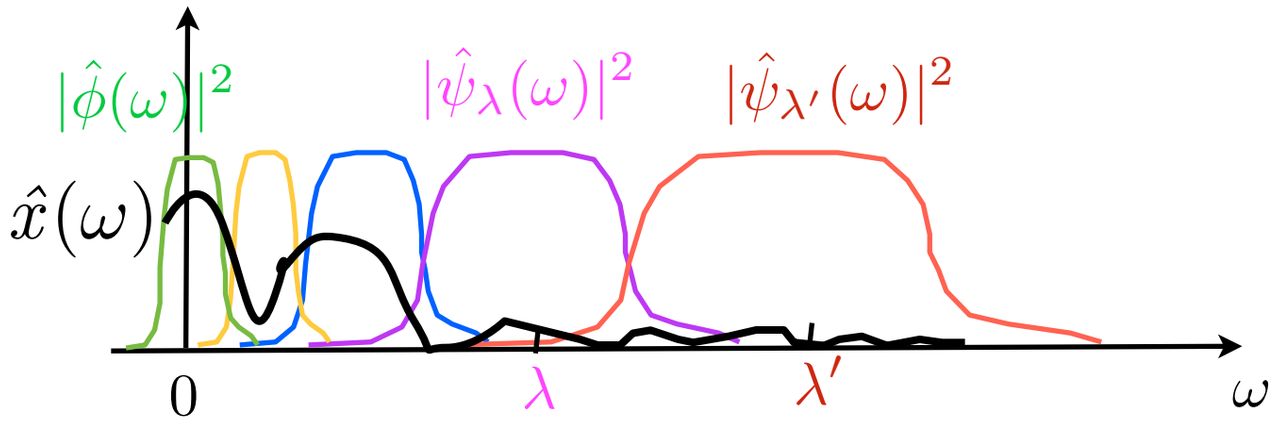


# Wavelet Transform

- Complex wavelet: $\psi(t) = \psi^a(t) + i\,\psi^b(t)$
- Dilated: $\psi_\lambda(t) = 2^{-j}\,\psi(2^{-j}t)$ with $\lambda = 2^{-j}$ .



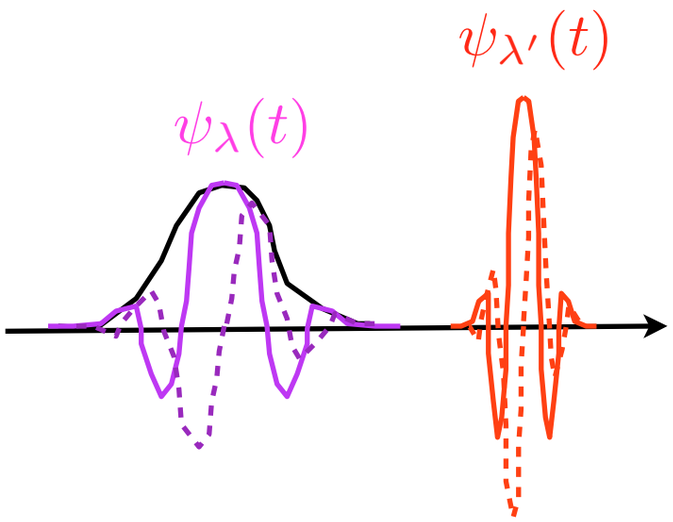



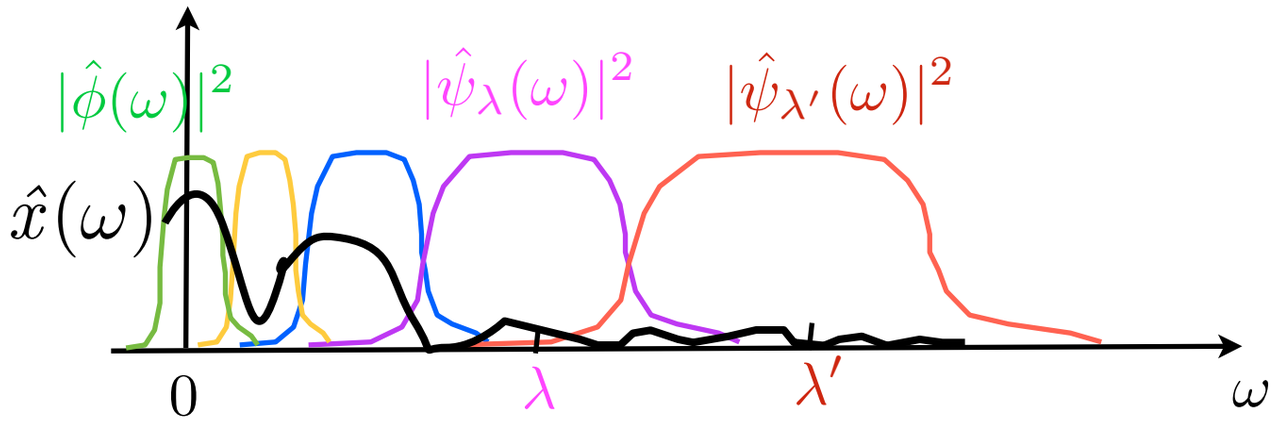

- Wavelet transform: $x \star \psi_\lambda(t) = \int x(u)\,\psi_\lambda(t-u)\,du$

$$Wx = \begin{pmatrix} x \star \phi(t) \\ x \star \psi_\lambda(t) \end{pmatrix}_{t,\lambda}$$

# Wavelet Transform

- Complex wavelet: $\psi(t) = \psi^a(t) + i\,\psi^b(t)$
- Dilated: $\psi_\lambda(t) = 2^{-j}\,\psi(2^{-j}t)$ with $\lambda = 2^{-j}$ .




- Wavelet transform: $x \star \psi_\lambda(t) = \int x(u)\,\psi_\lambda(t-u)\,du$

$$Wx = \begin{pmatrix} x \star \phi(t) \\ x \star \psi_\lambda(t) \end{pmatrix}_{t,\lambda}$$

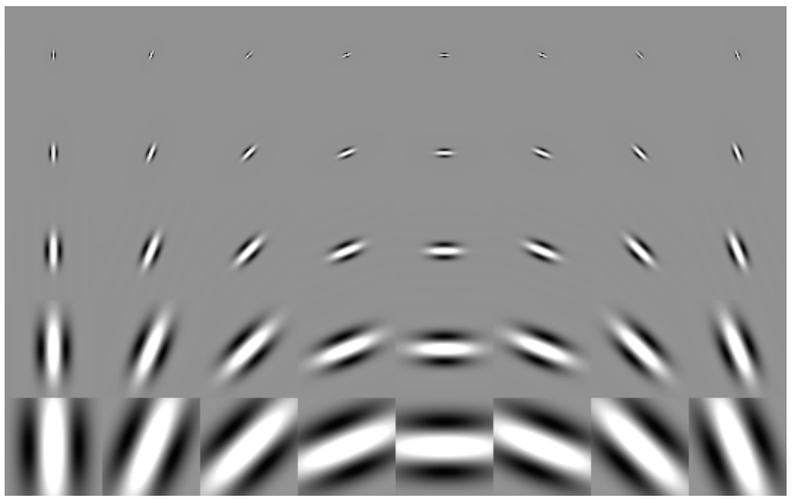
Unitary: $\|Wx\|^2 = \|x\|^2$ .

# Image Wavelet Transform

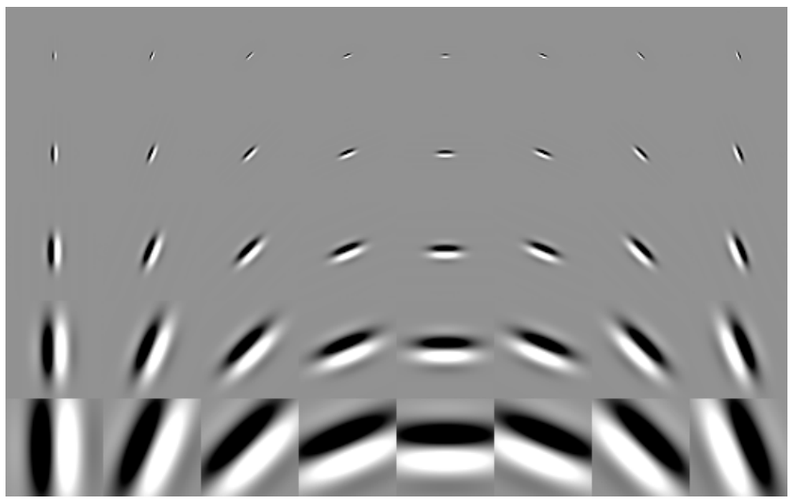
- Complex wavelet: $\psi(t) = \psi^a(t) + i\,\psi^b(t) \quad , \quad t = (t_1, t_2)$

  rotated and dilated: $\psi_\lambda(t) = 2^{-j}\,\psi(2^{-j} r t) \quad \text{with} \quad \lambda = (2^j, r)$

real parts



imaginary parts

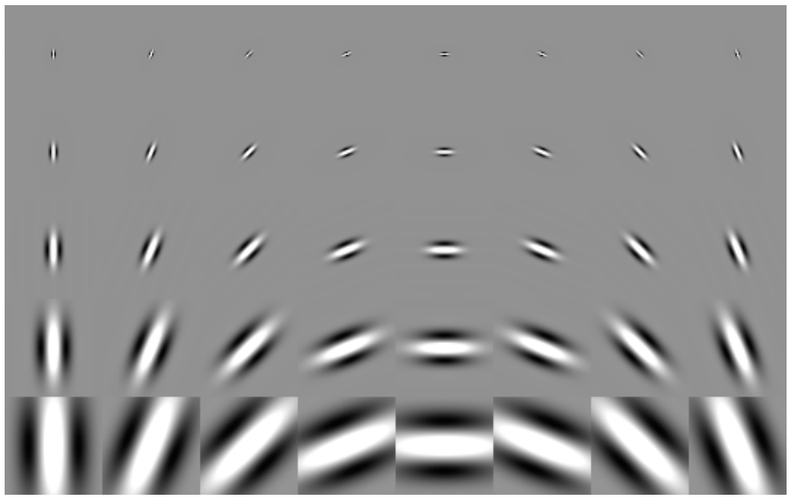


# Image Wavelet Transform

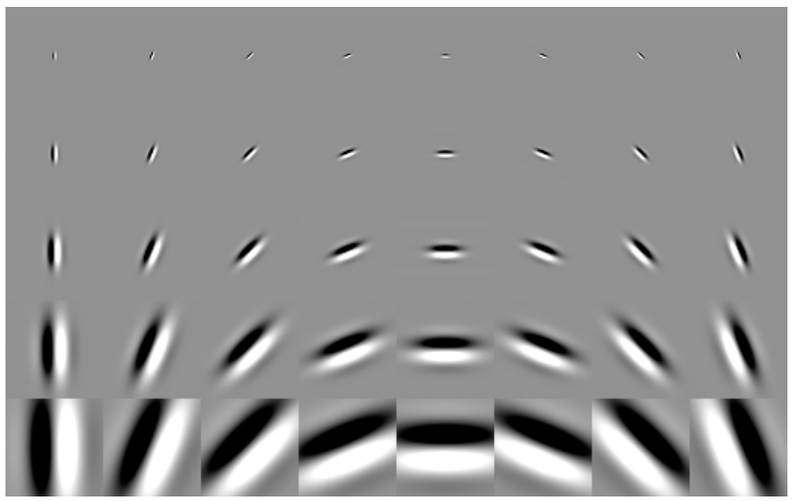
- Complex wavelet: $\psi(t) = \psi^a(t) + i\,\psi^b(t) \quad , \quad t = (t_1, t_2)$

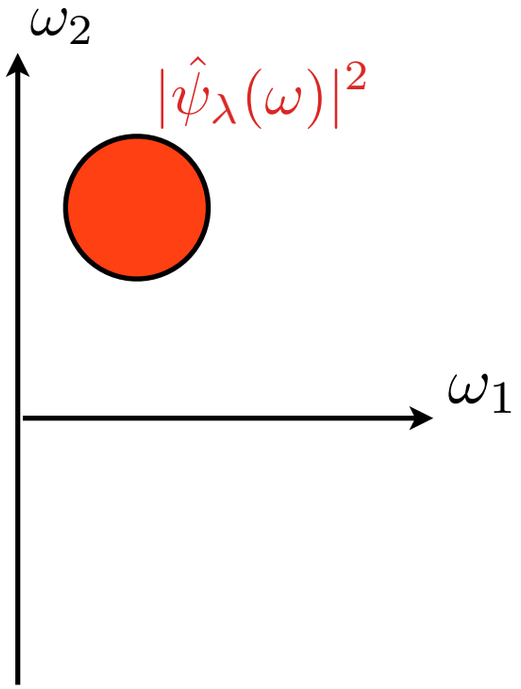
  rotated and dilated: $\psi_\lambda(t) = 2^{-j}\,\psi(2^{-j} r t)$ with $\lambda = (2^j, r)$

real parts



imaginary parts



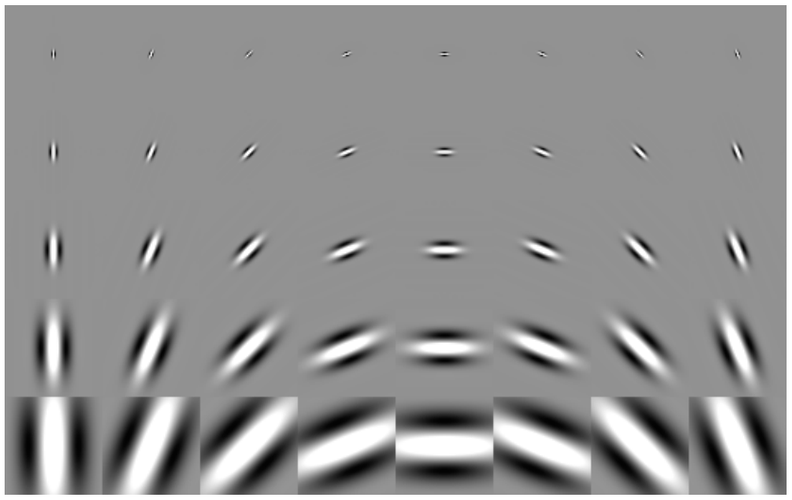


# Image Wavelet Transform

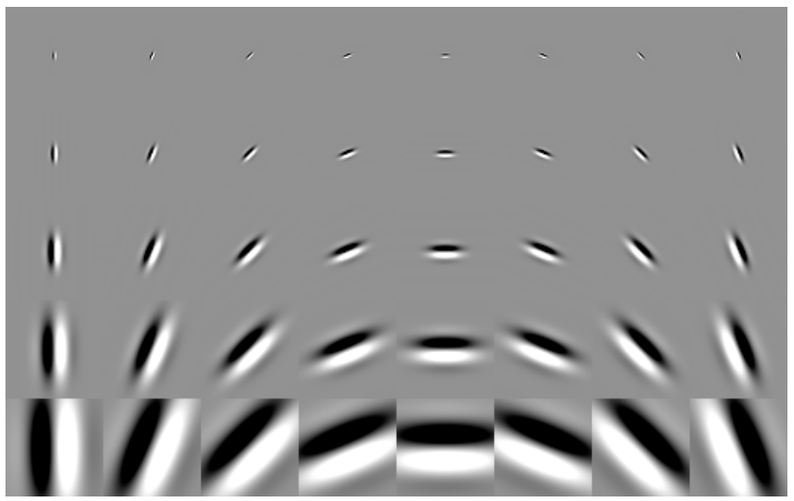
- Complex wavelet: $\psi(t) = \psi^a(t) + i\,\psi^b(t) \quad , \quad t = (t_1, t_2)$

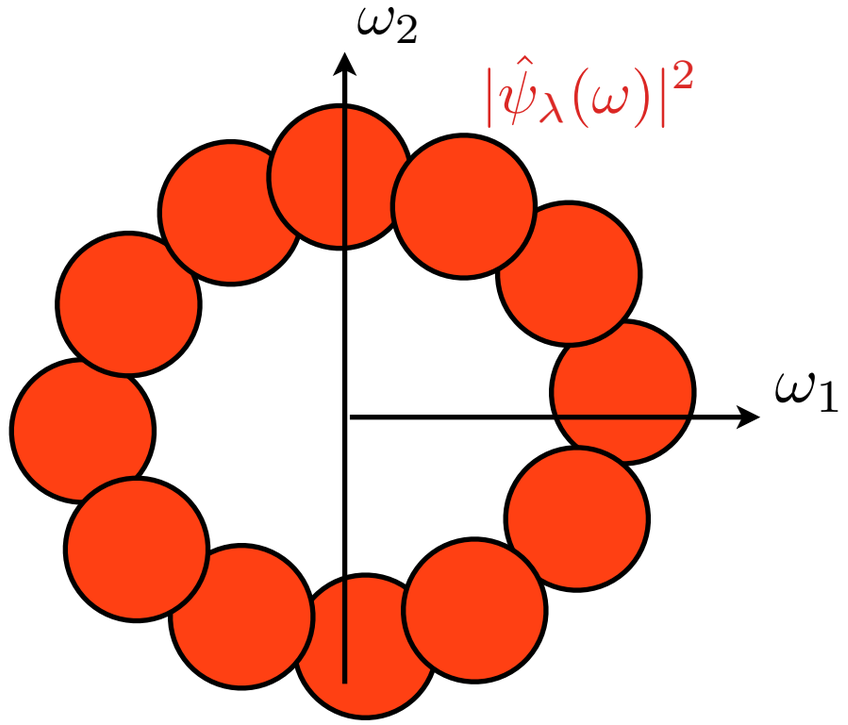
  rotated and dilated: $\psi_\lambda(t) = 2^{-j}\,\psi(2^{-j} r t)$ with $\lambda = (2^j, r)$

real parts



imaginary parts



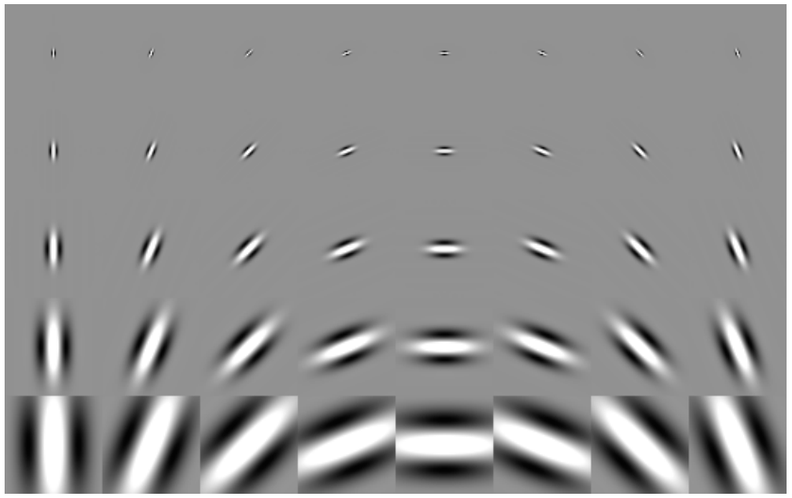


# Image Wavelet Transform

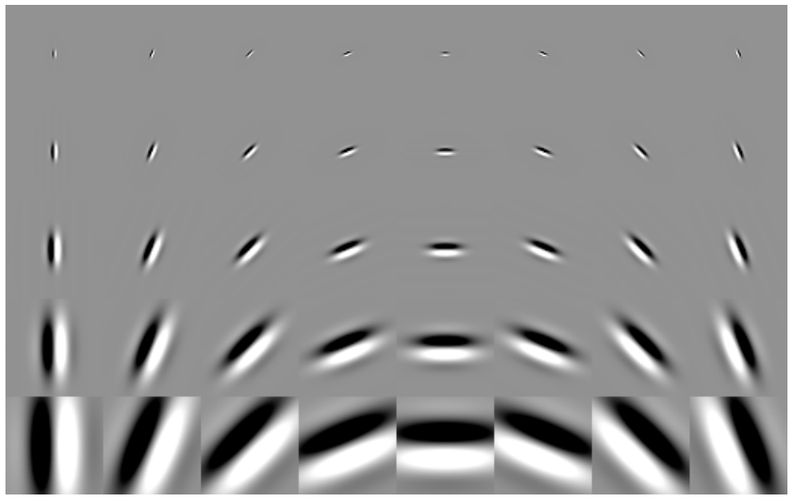
- Complex wavelet: $\psi(t) = \psi^a(t) + i\,\psi^b(t) \quad , \quad t = (t_1, t_2)$

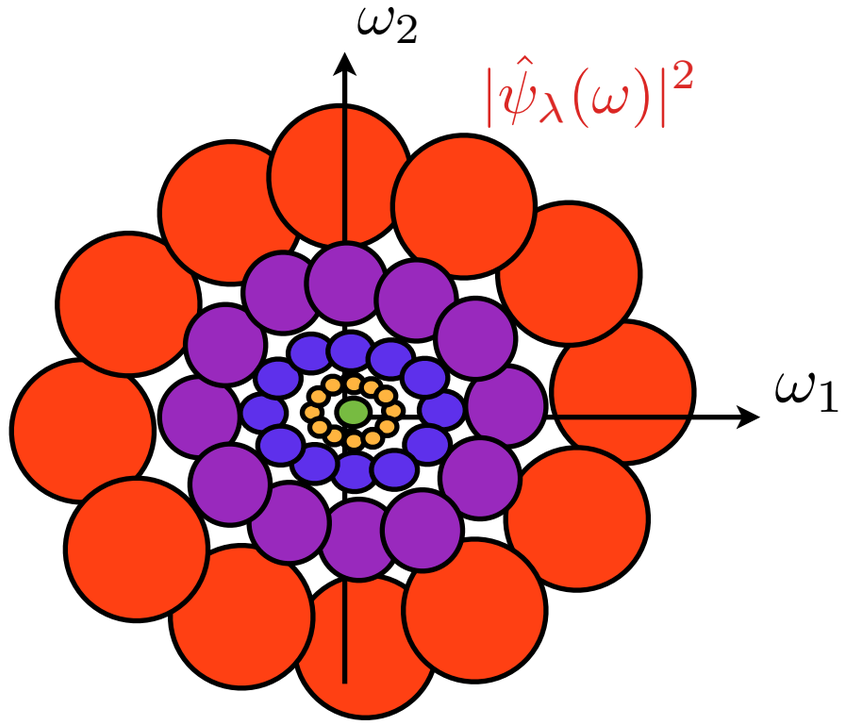
  rotated and dilated: $\psi_\lambda(t) = 2^{-j}\,\psi(2^{-j} r t)$ with $\lambda = (2^j, r)$

real parts



imaginary parts



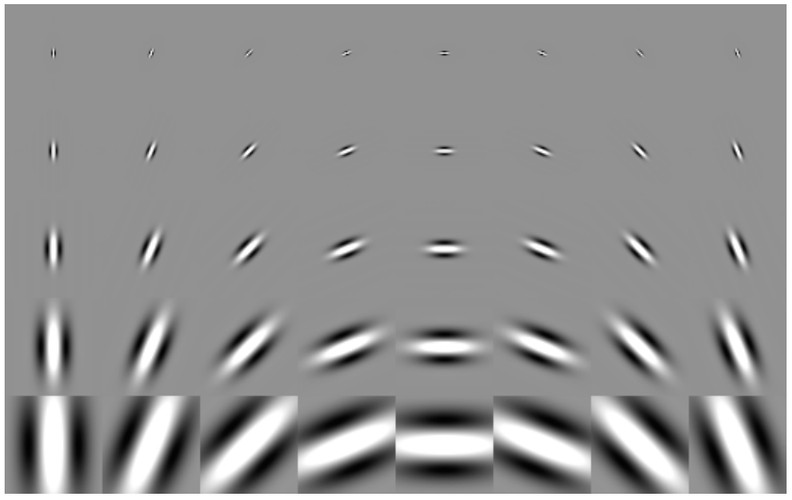


# Image Wavelet Transform

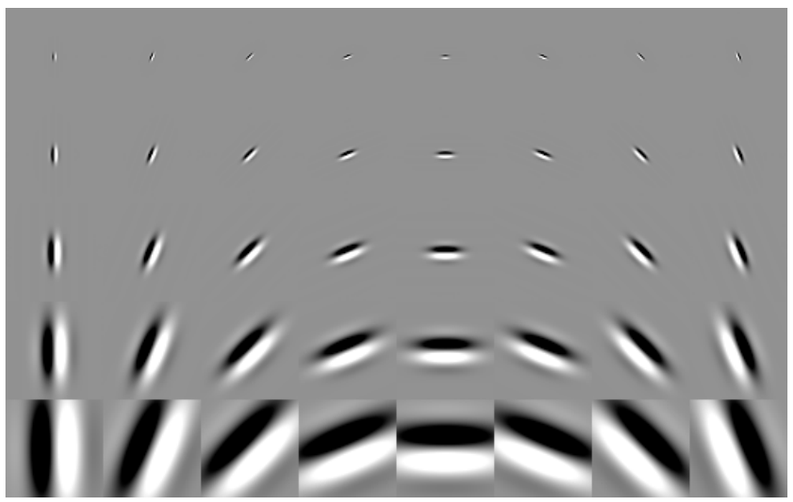
- Complex wavelet: $\psi(t) = \psi^a(t) + i\,\psi^b(t) \quad , \quad t = (t_1, t_2)$

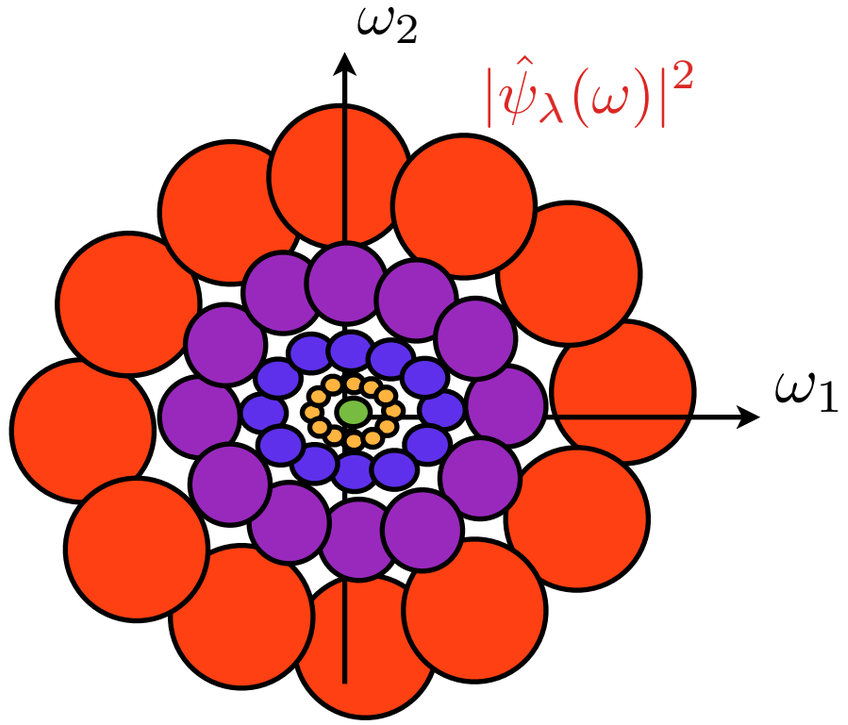
  rotated and dilated: $\psi_\lambda(t) = 2^{-j}\,\psi(2^{-j} r t)$ with $\lambda = (2^j, r)$

real parts



imaginary parts



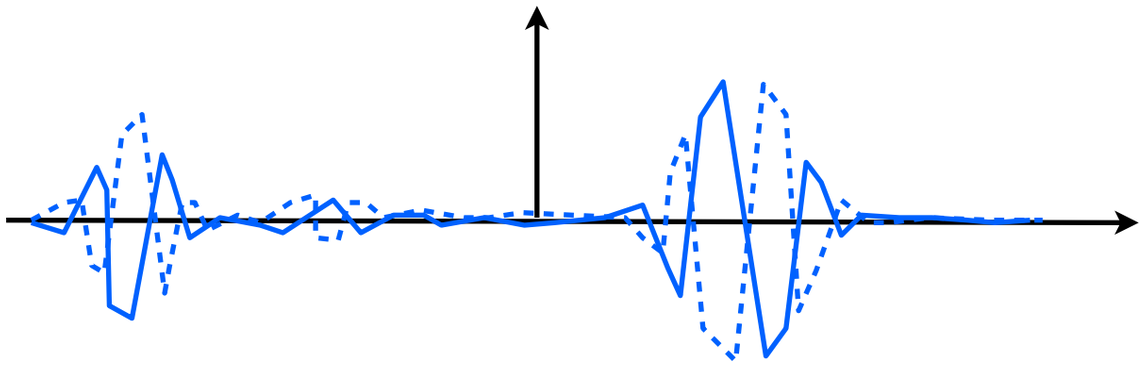



- Wavelet transform: $Wx = \begin{pmatrix} x \star \phi(t) \\ x \star \psi_\lambda(t) \end{pmatrix}_{t,\lambda}$

Unitary: $\|Wx\|^2 = \|x\|^2$ .

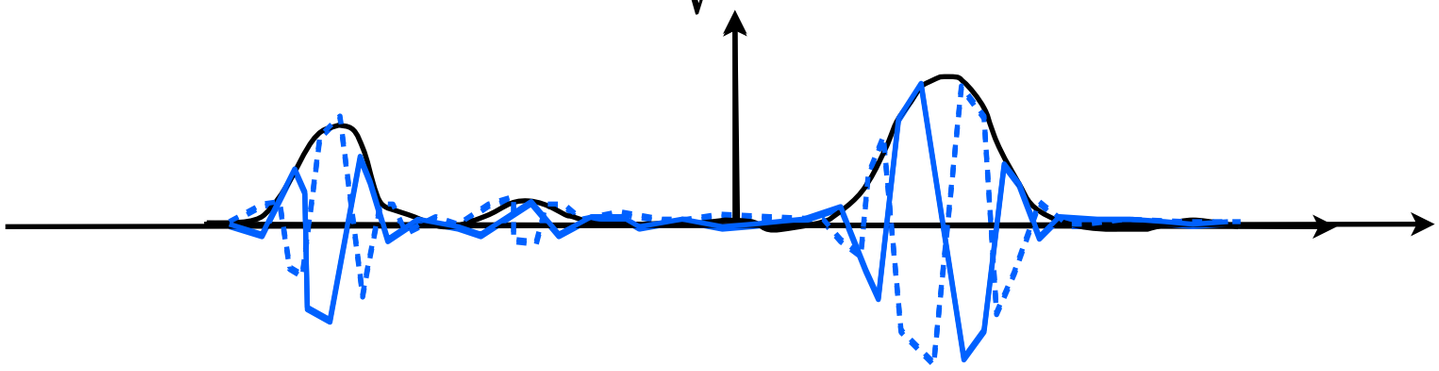
# Wavelet Translation Invariance

$$x \star \psi_{\lambda_1}(t) = x \star \psi^a_{\lambda_1}(t) + i\, x \star \psi^b_{\lambda_1}(t)$$



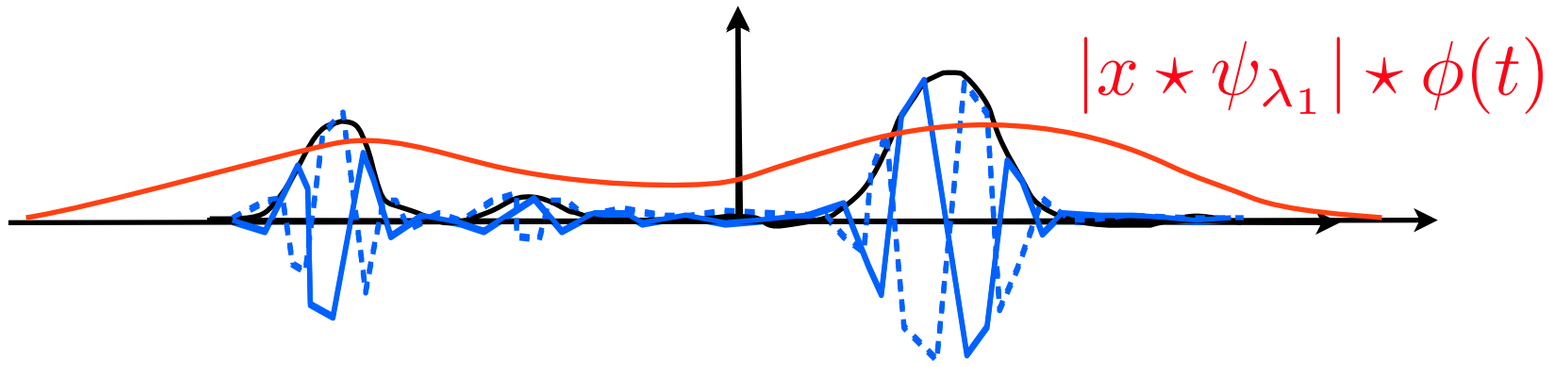
# Wavelet Translation Invariance

$$|x \star \psi_{\lambda_1}(t)| = \sqrt{|x \star \psi^a_{\lambda_1}(t)|^2 + |x \star \psi^b_{\lambda_1}(t)|^2} \quad \text{pooling}$$



- The modulus $|x \star \psi_{\lambda_1}|$ is a regular envelop

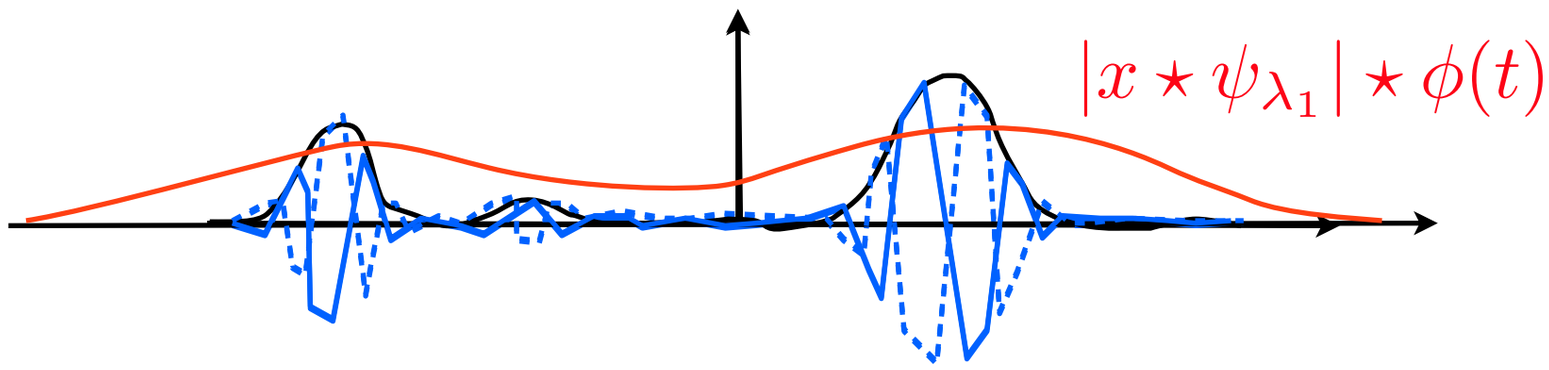
# Wavelet Translation Invariance




- The modulus $|x \star \psi_{\lambda_1}|$ is a regular envelop

- The average $|x \star \psi_{\lambda_1}| \star \phi(t)$ is invariant to small translations relatively to the support of $\phi$.

# Wavelet Translation Invariance




- The modulus $|x \star \psi_{\lambda_1}|$ is a regular envelop

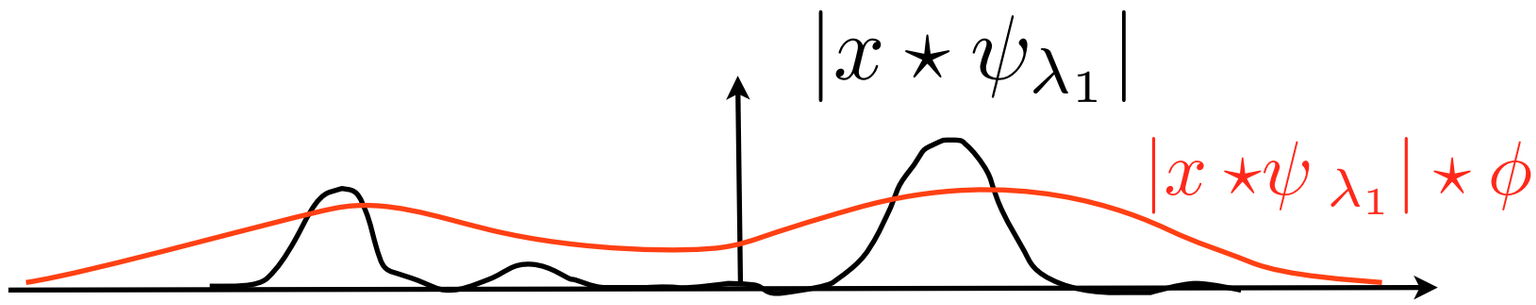
- The average $|x \star \psi_{\lambda_1}| \star \phi(t)$ is invariant to small translations relatively to the support of $\phi$.

- Full translation invariance at the limit:

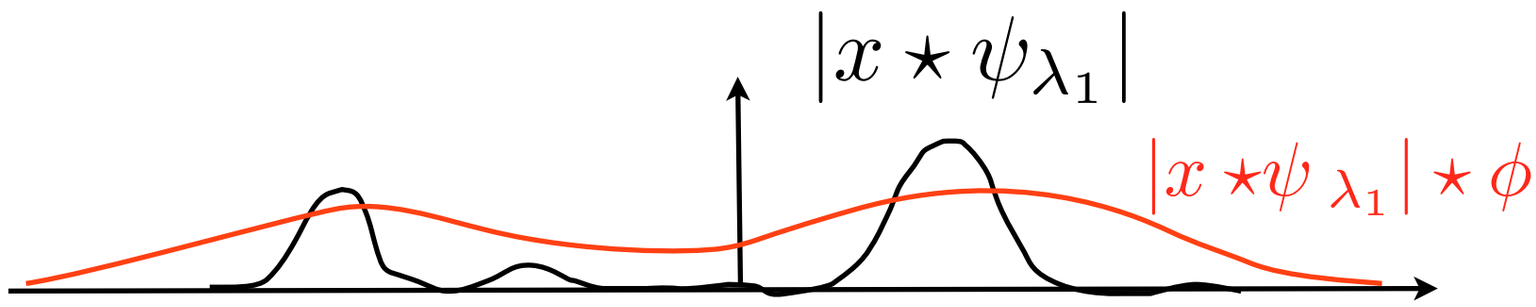
$$\lim_{\phi \to 1} |x \star \psi_{\lambda_1}| \star \phi(t) = \int |x \star \psi_{\lambda_1}(u)| \, du = \|x \star \psi_{\lambda_1}\|_1$$

but few invariants.

# Recovering Lost Information



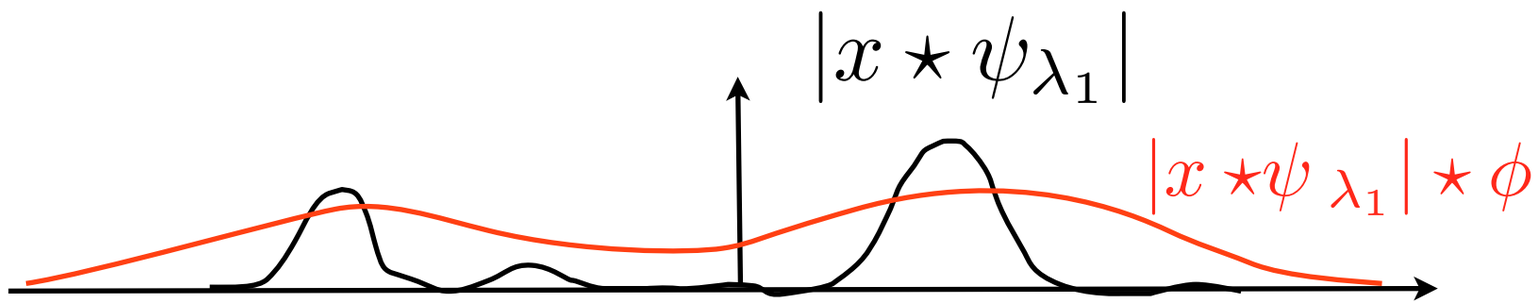
# Recovering Lost Information




- The high frequencies of $|x \star \psi_{\lambda_1}|$ are in wavelet coefficients:

$$W|x \star \psi_{\lambda_1}| = \begin{pmatrix} |x \star \psi_{\lambda_1}| \star \phi(t) \\ |x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}(t) \end{pmatrix}_{t,\lambda_2}$$

# Recovering Lost Information




- The high frequencies of $|x \star \psi_{\lambda_1}|$ are in wavelet coefficients:

$$W|x \star \psi_{\lambda_1}| = \begin{pmatrix} |x \star \psi_{\lambda_1}| \star \phi(t) \\ |x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}(t) \end{pmatrix}_{t,\lambda_2}$$

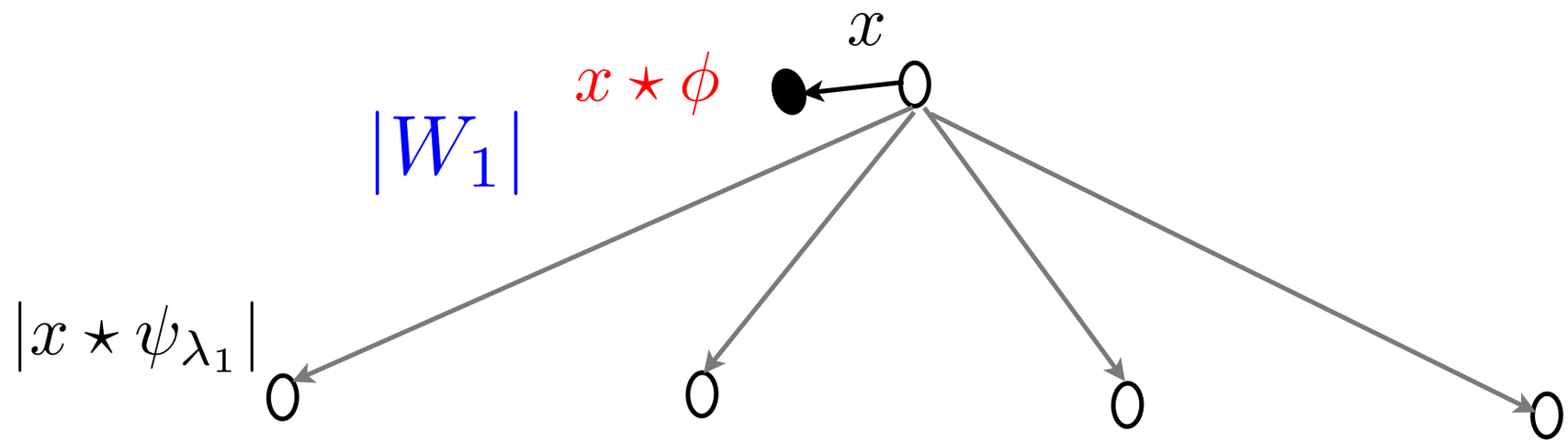
- Translation invariance by time averaging the amplitude:

$$\forall \lambda_1 , \lambda_2 , \quad ||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(t)$$

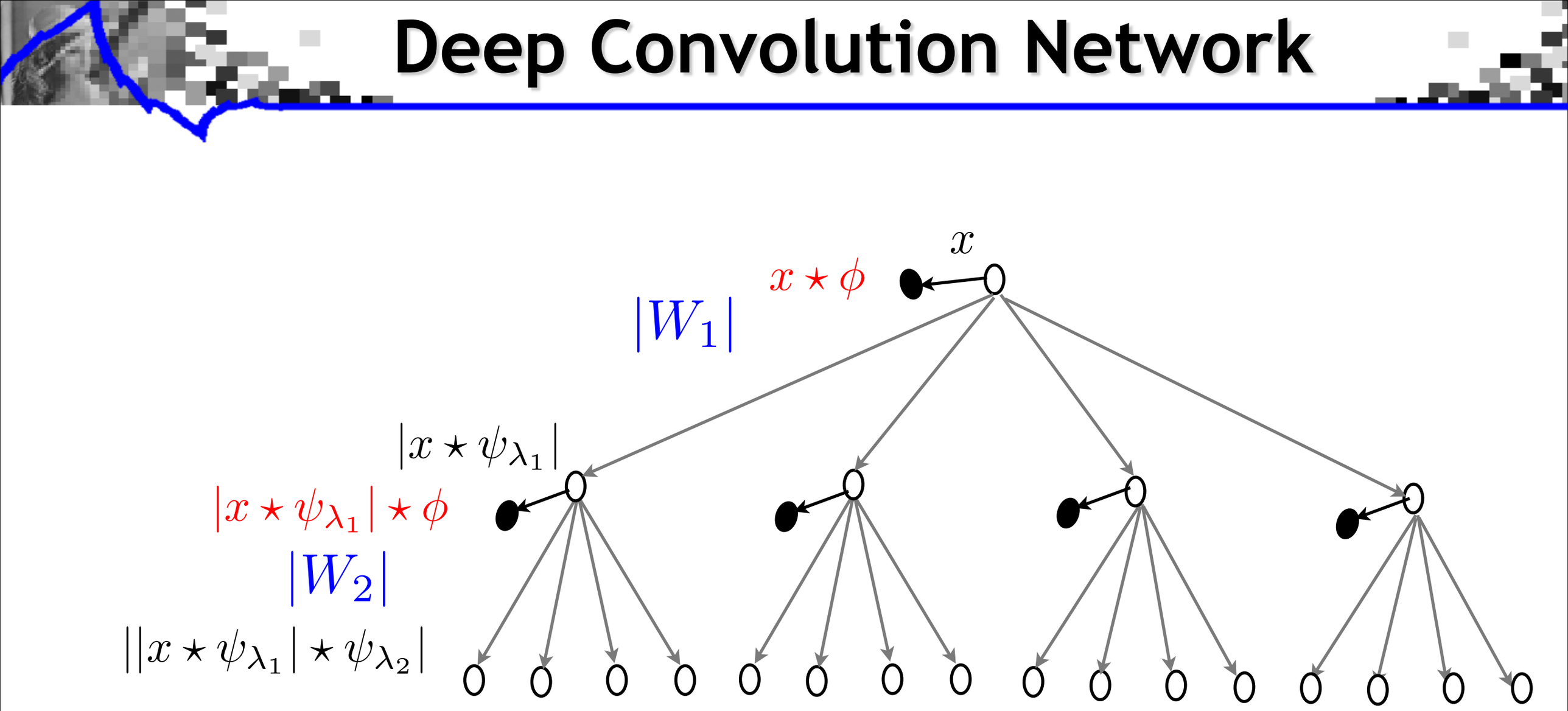
# Deep Convolution Network

$x_0$

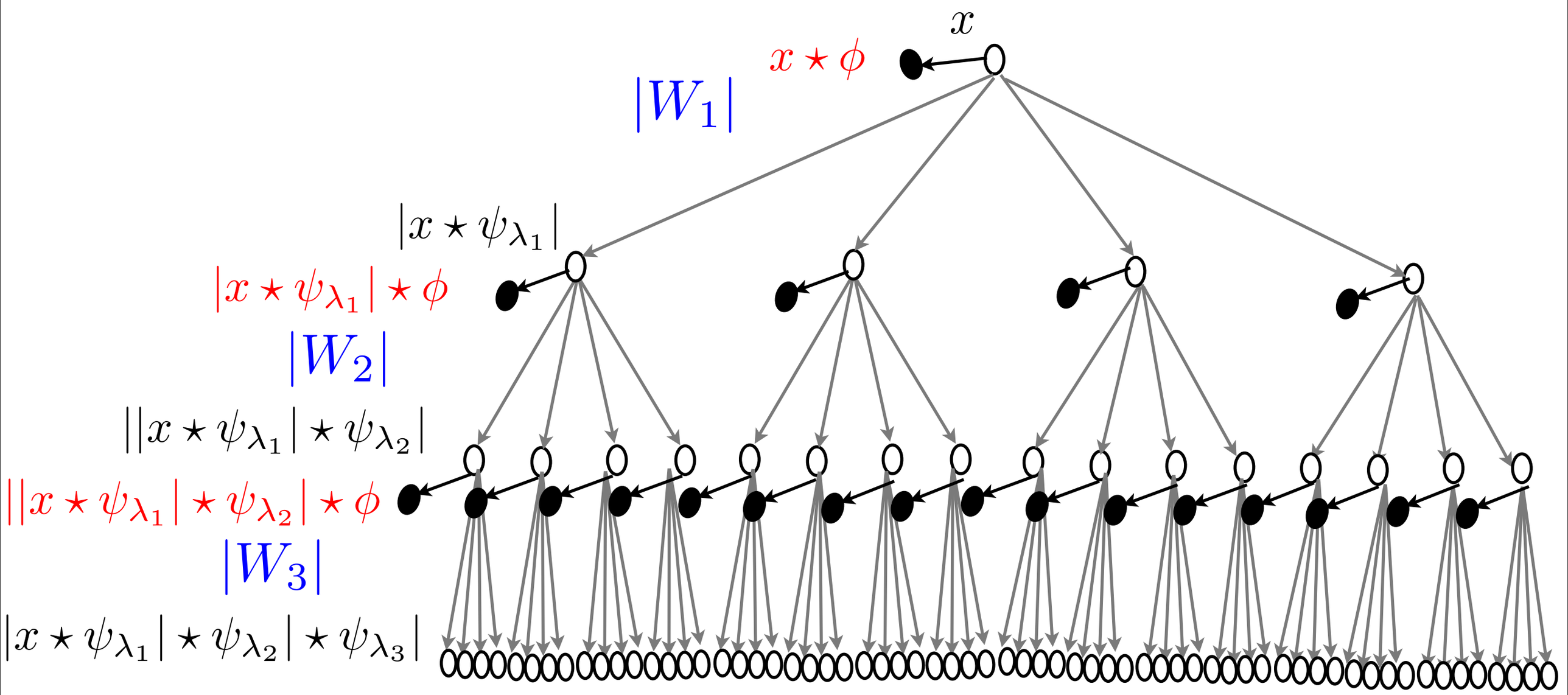
# Deep Convolution Network



# Deep Convolution Network



# Deep Convolution Network

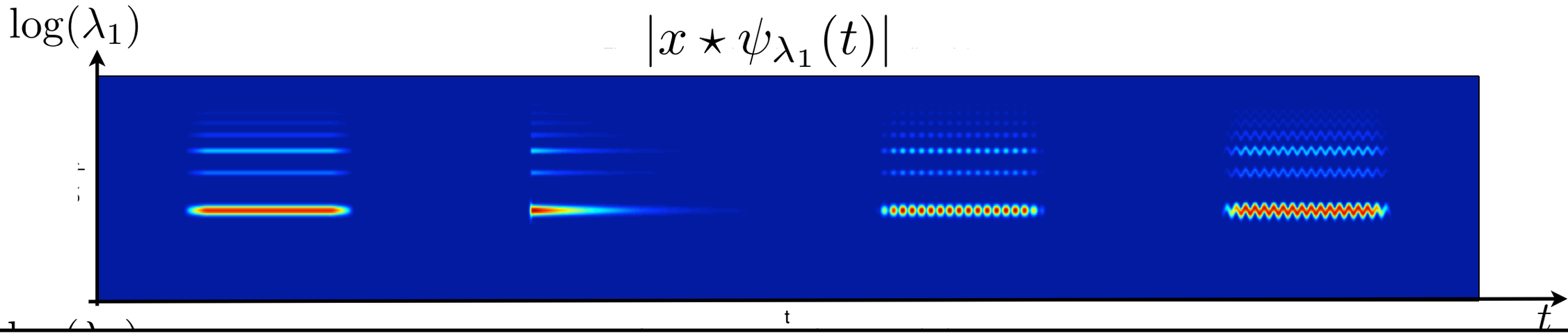


# Scattering Vector

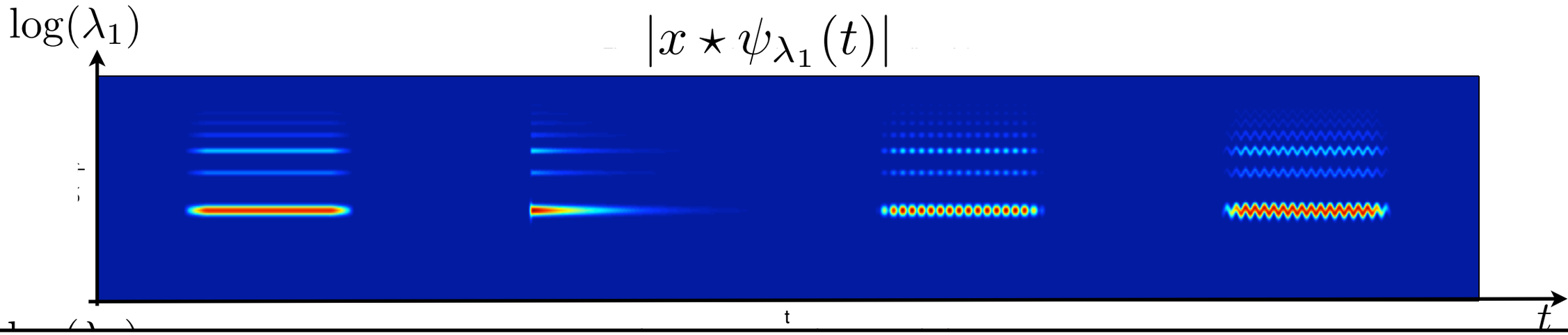
Network ouptut:

$$Sx = \begin{pmatrix} x \star\phi\,(u) \\ |x \star \psi_{\lambda_1}| \star \phi(u) \\ ||x \star\psi\ _{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(u) \\ |||x \star\psi\ _{\lambda_2}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}| \star \phi(u) \\ \dots \end{pmatrix}_{u,\lambda_1,\lambda_2,\lambda_3,\dots}$$

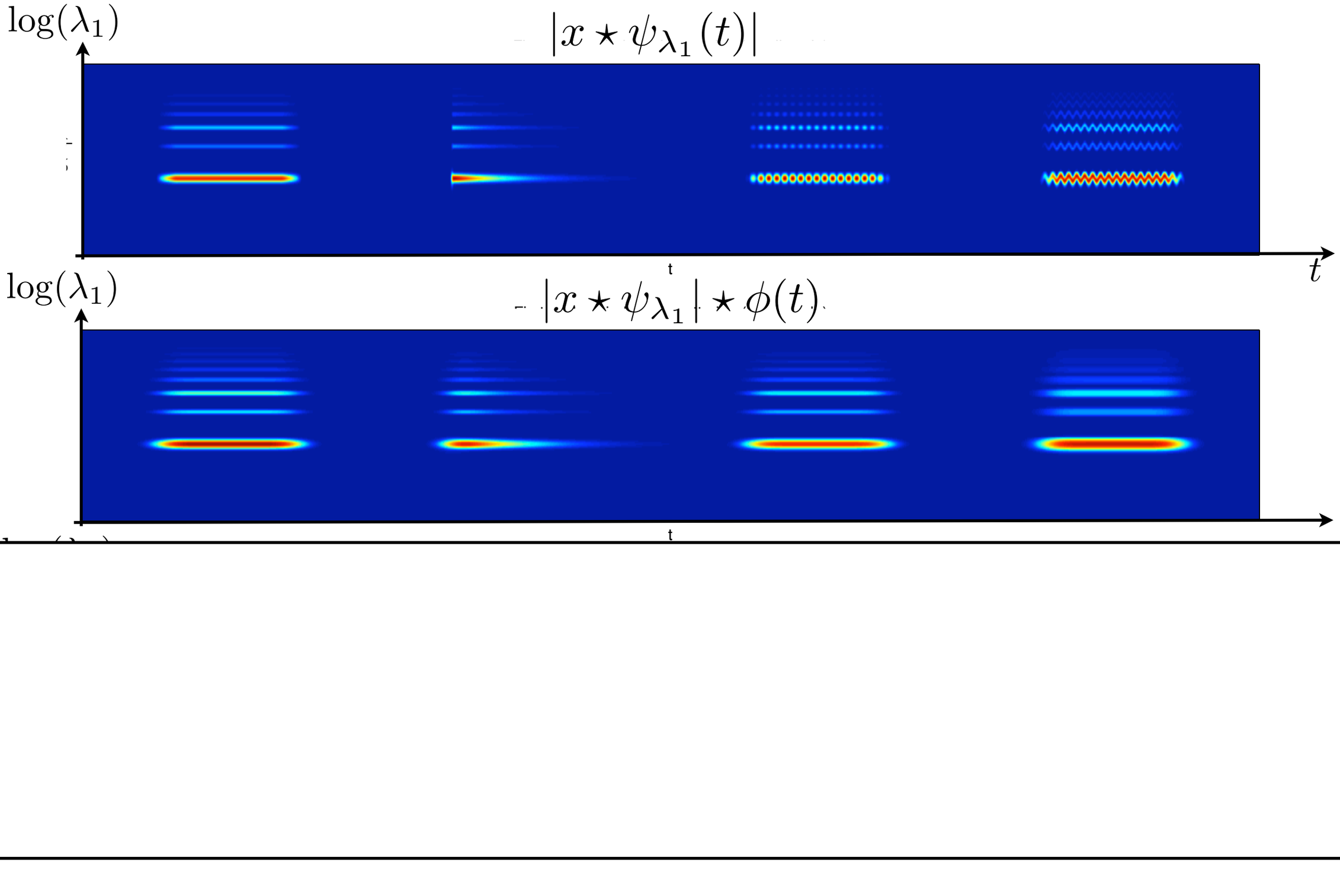
# Amplitude Modulation



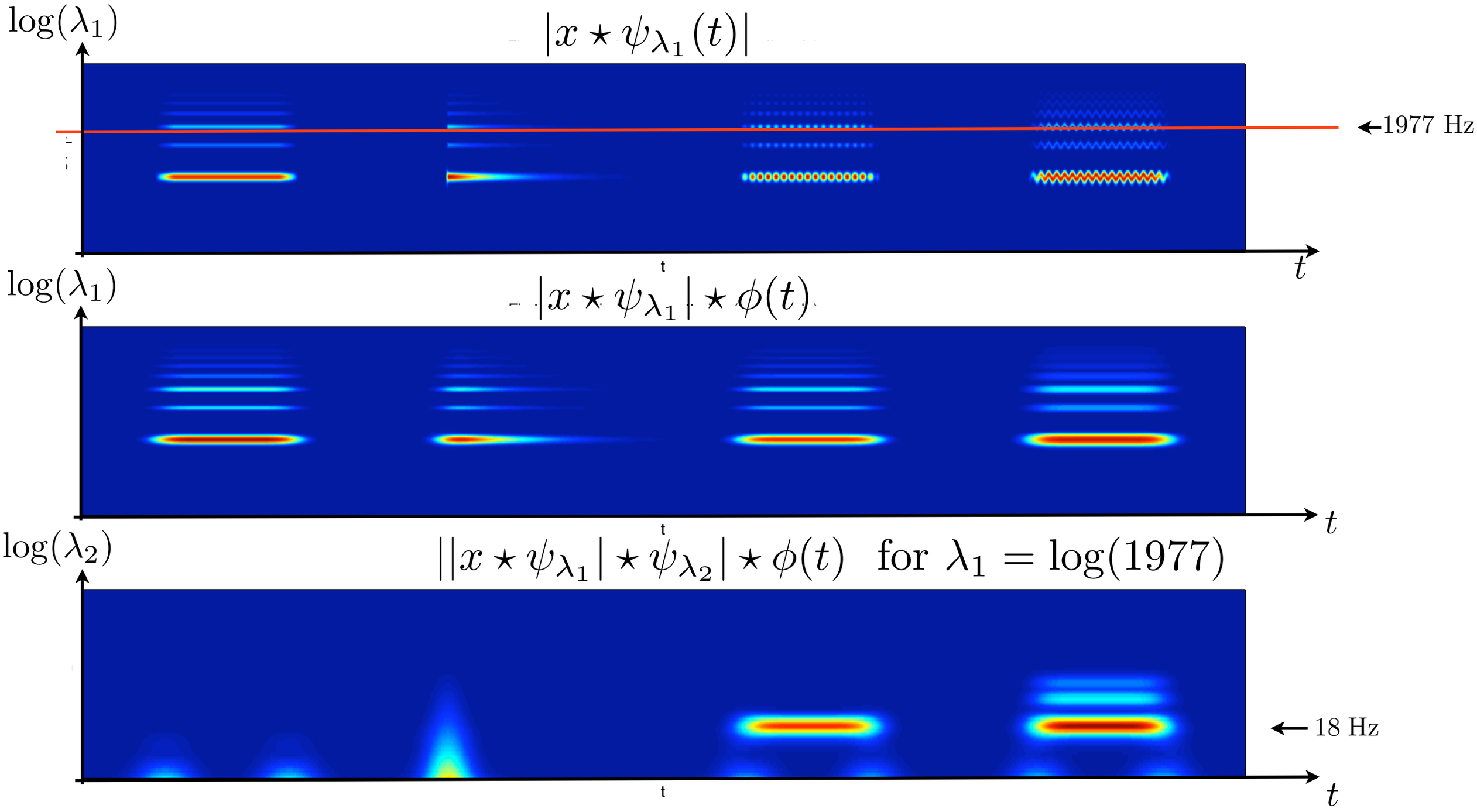
# Amplitude Modulation



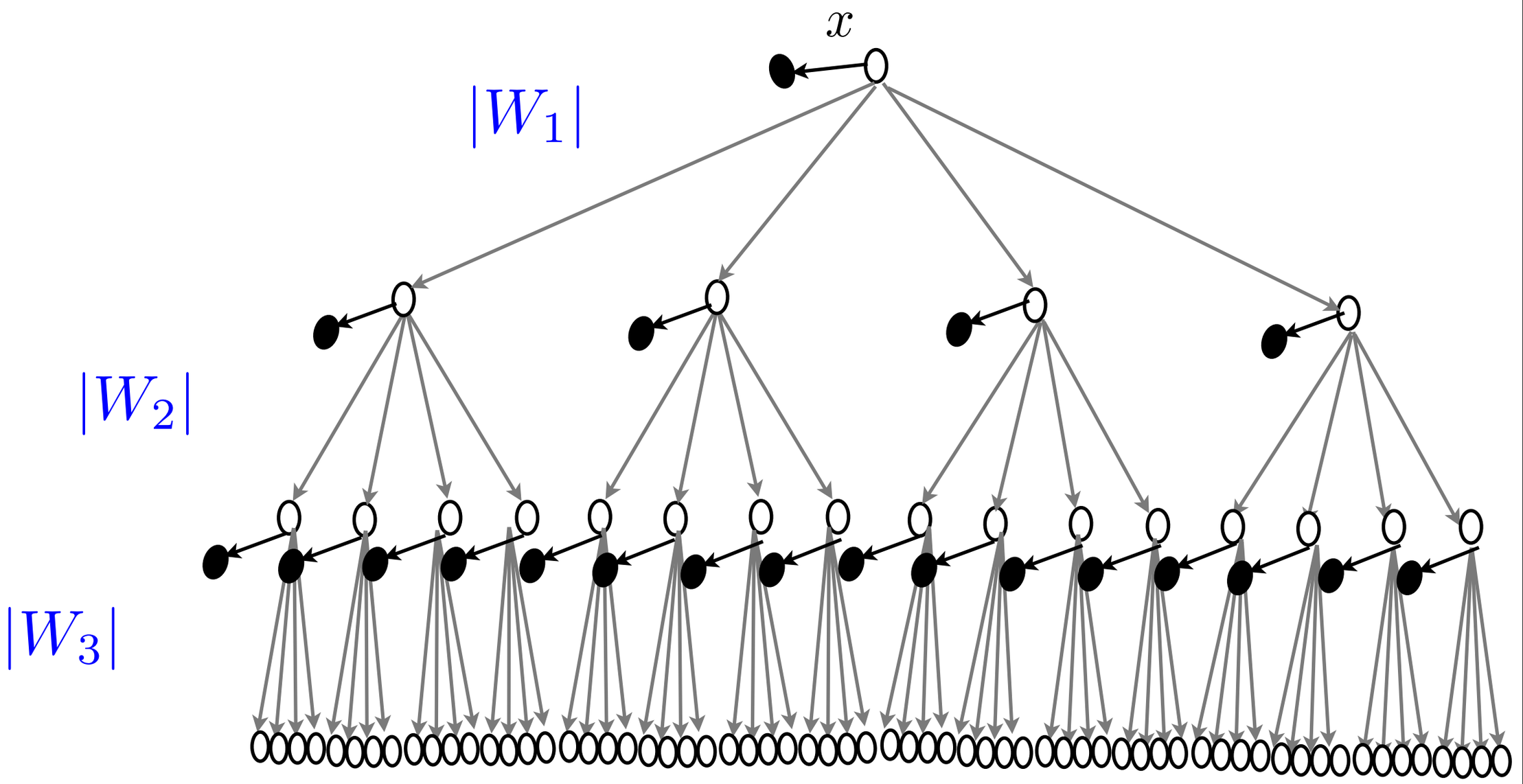
# Amplitude Modulation



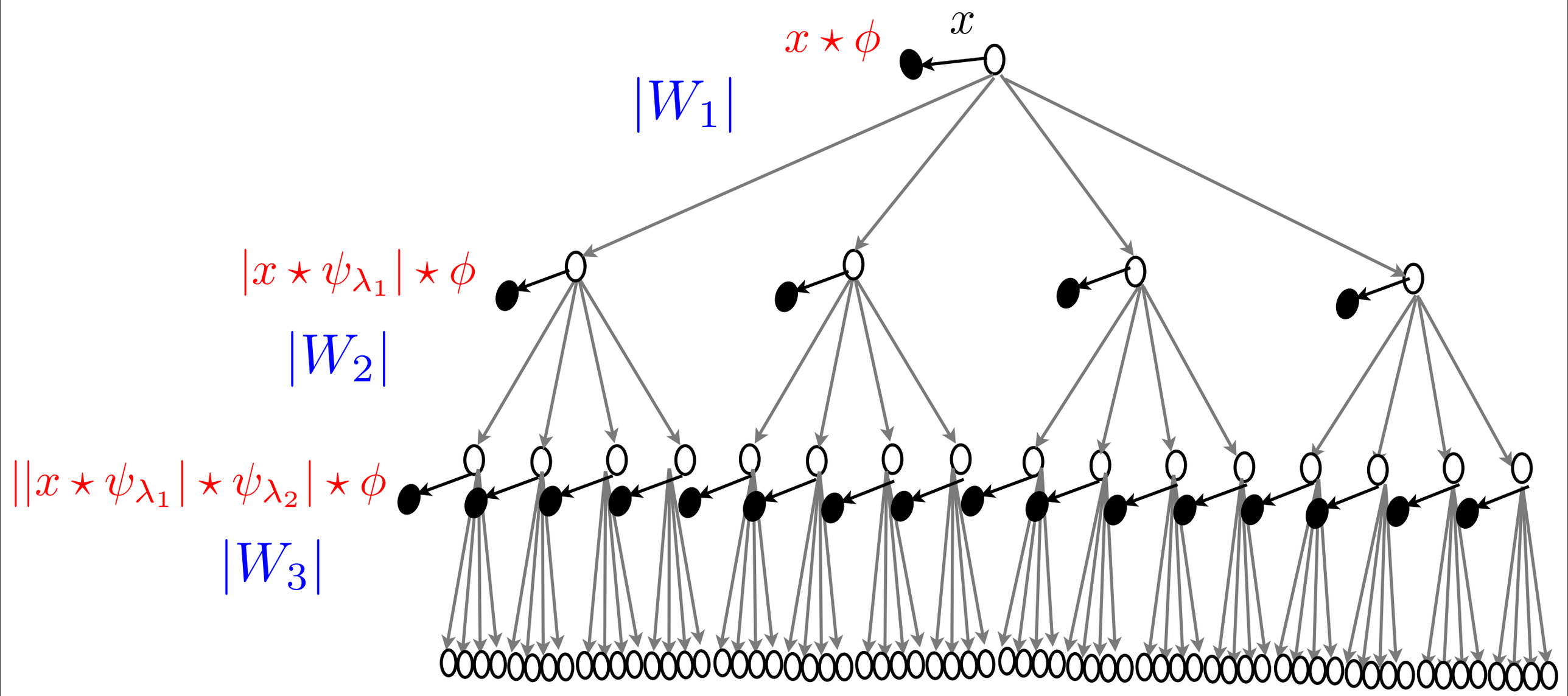
# Amplitude Modulation



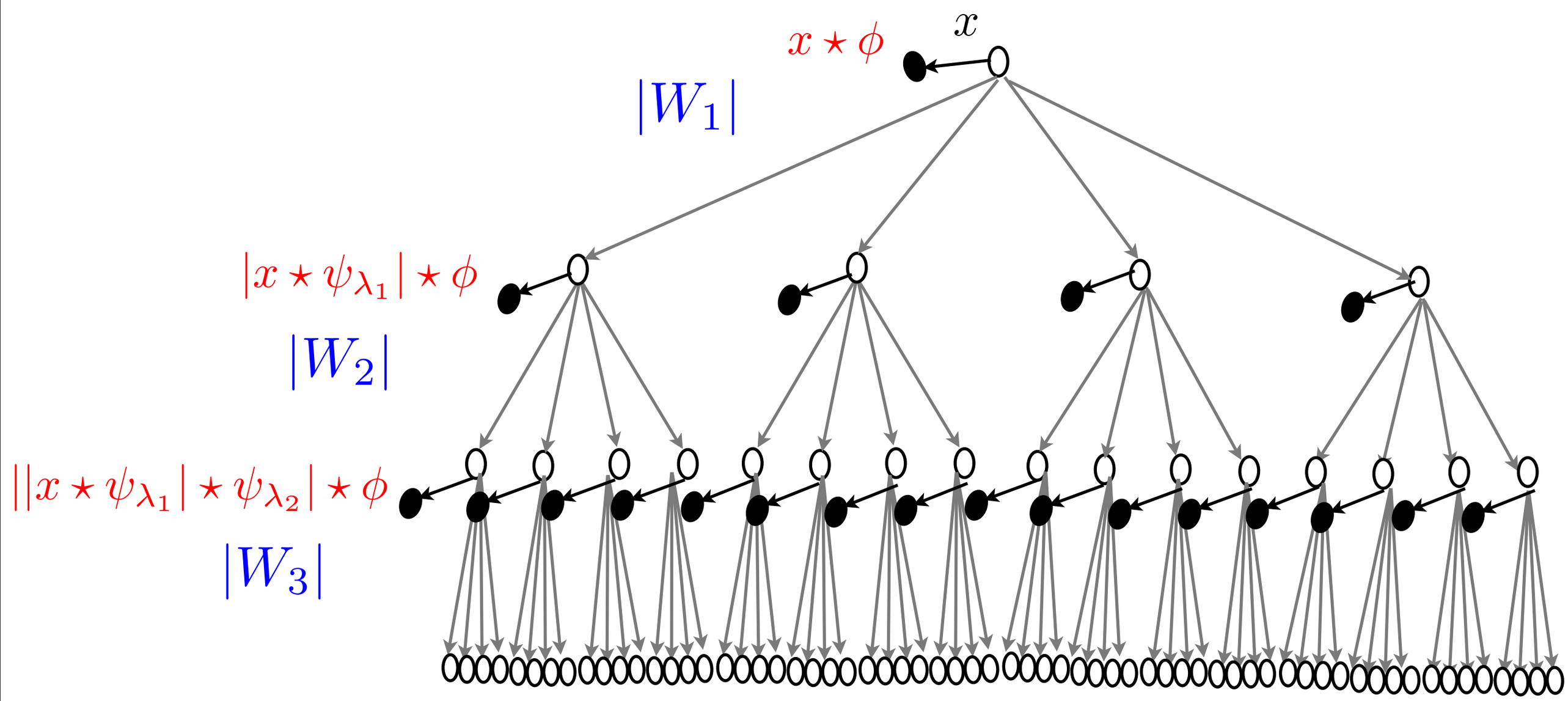
# Cascade of Contractions



# Cascade of Contractions



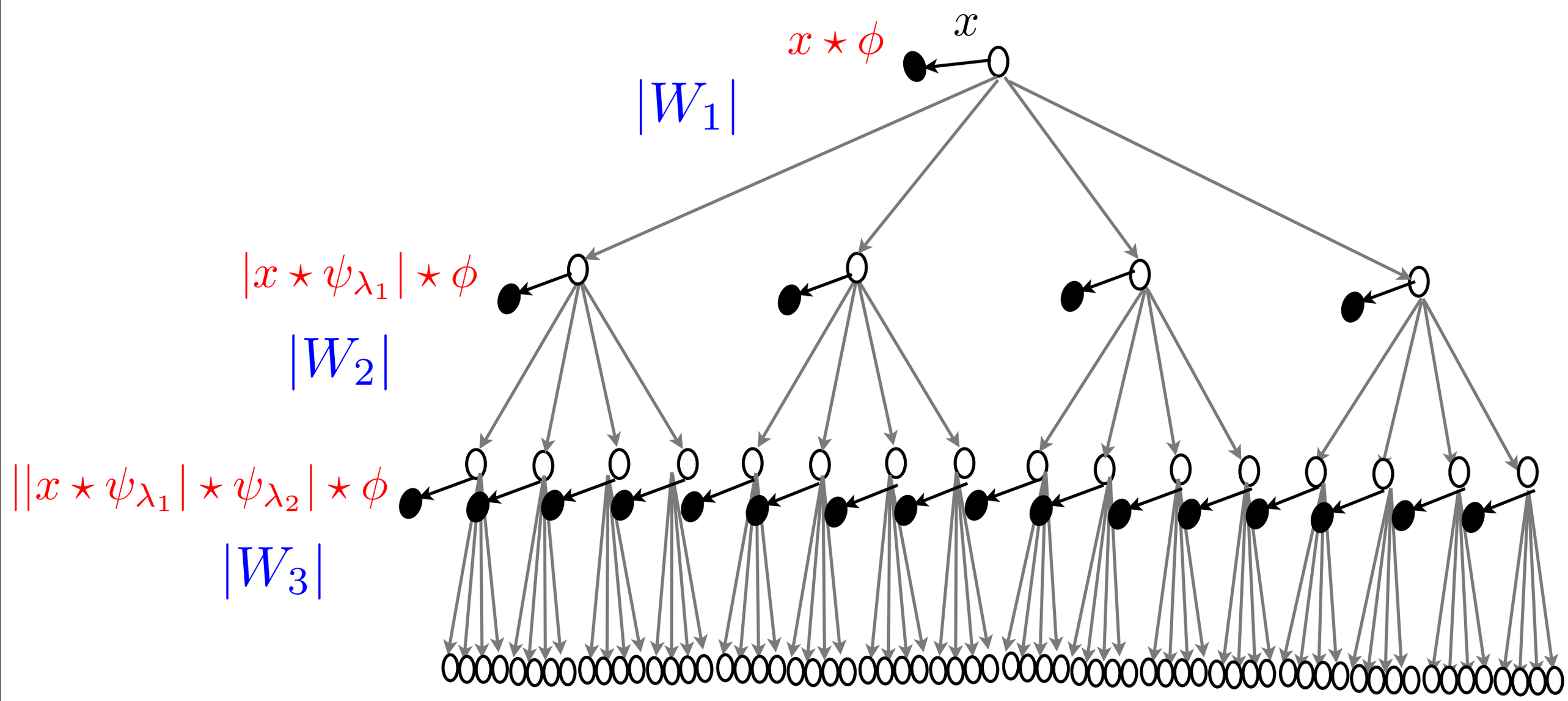
# Cascade of Contractions




- Cascade of contractive operators

$$\| |W_k|x - |W_k|x' \| \leq \|x - x'\|$$

# Cascade of Contractions




- Cascade of contractive operators

$$\| |W_k|x - |W_k|x' \| \leq \|x - x'\| \quad \text{with} \quad \| |W_k|x \| = \|x\| \ .$$

# Scattering Properties

$$Sx = \begin{pmatrix} x \star \phi\,(u) \\ |x \star \psi_{\lambda_1}| \star \phi(u) \\ ||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(u) \\ |||x \star \psi_{\lambda_2}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}| \star \phi(u) \\ \dots \end{pmatrix}_{u,\lambda_1,\lambda_2,\lambda_3,\dots}$$

**Theorem**: *For appropriate wavelets, a scattering is*

$$contractive \quad \|Sx - Sy\| \leq \|x - y\|$$

$$preserves\ norms \quad \|Sx\| = \|x\|$$

# Scattering Properties

$$Sx = \begin{pmatrix} x \star \phi\,(u) \\ |x \star \psi_{\lambda_1}| \star \phi(u) \\ ||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(u) \\ |||x \star \psi_{\lambda_2}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}| \star \phi(u) \\ \dots \end{pmatrix}_{u,\lambda_1,\lambda_2,\lambda_3,\dots}$$

**Theorem**: *For appropriate wavelets, a scattering is*

*contractive* $\|Sx - Sy\| \leq \|x - y\|$

*preserves norms* $\|Sx\| = \|x\|$

*stable to deformations* $x_\tau(t) = x(t - \tau(t))$

$$\|Sx - Sx_\tau\| \leq C \sup_t |\nabla \tau(t)|\, \|x\|$$

# Scattering Properties

$$Sx = \begin{pmatrix} x \star \phi\,(u) \\ |x \star \psi_{\lambda_1}| \star \phi(u) \\ ||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(u) \\ |||x \star \psi_{\lambda_2}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}| \star \phi(u) \\ \ldots \end{pmatrix}_{u,\lambda_1,\lambda_2,\lambda_3,\ldots}$$

**Theorem**: *For appropriate wavelets, a scattering is*

*contractive* $\|Sx - Sy\| \leq \|x - y\|$

*preserves norms* $\|Sx\| = \|x\|$

*stable to deformations* $x_\tau(t) = x(t - \tau(t))$

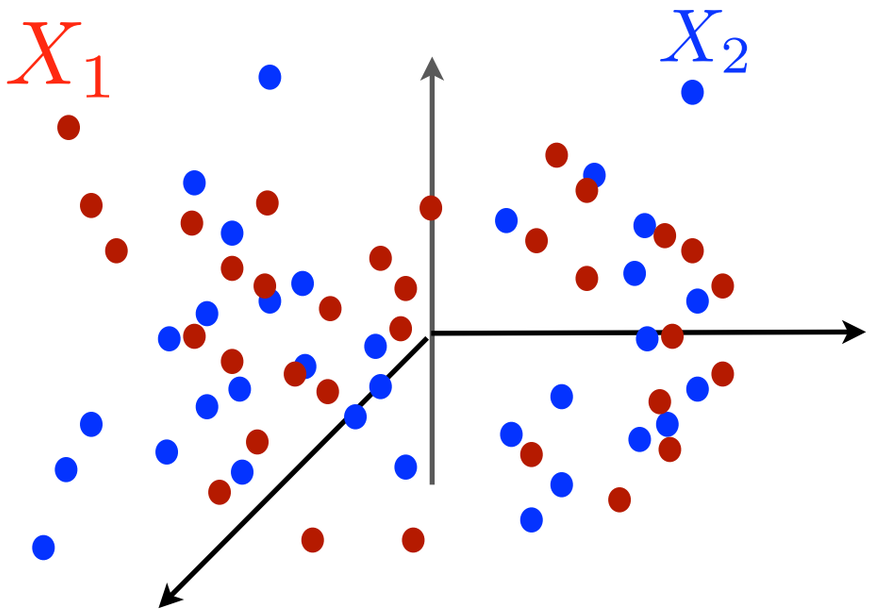
$$\|Sx - Sx_\tau\| \leq C \sup_t |\nabla \tau(t)|\, \|x\|$$

$\Rightarrow$ linear discriminative classification from $\Phi x = Sx$

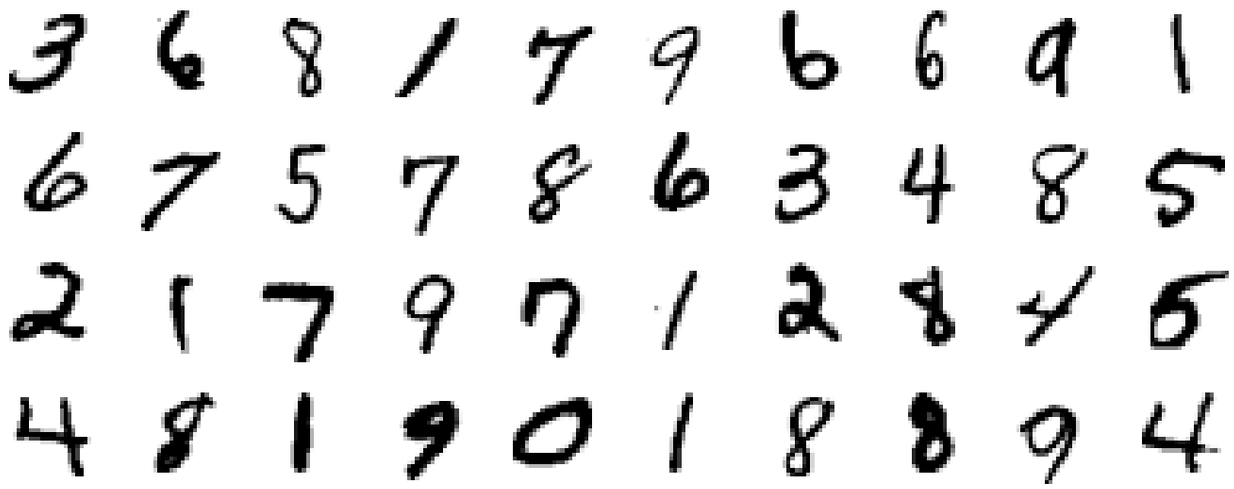
# Linearized Classification

*Joan Bruna*

computed with PCA.




MNIST data basis:

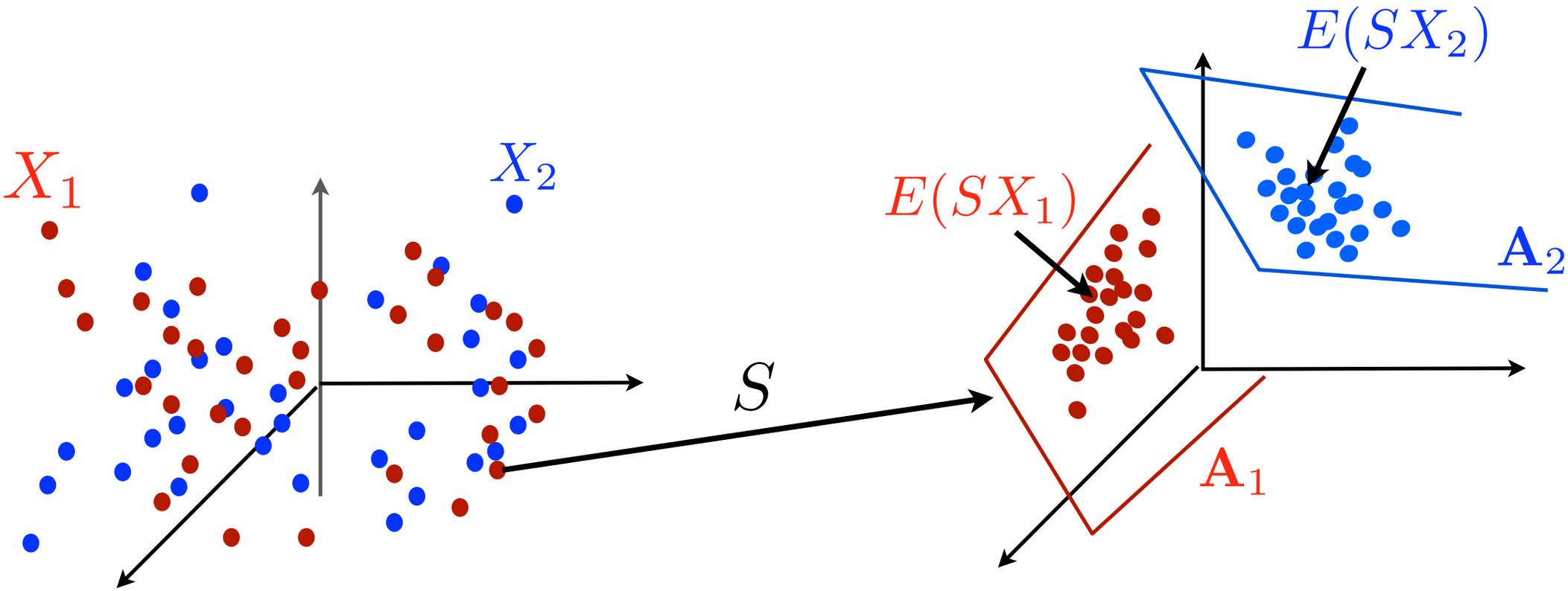


# Linearized Classification

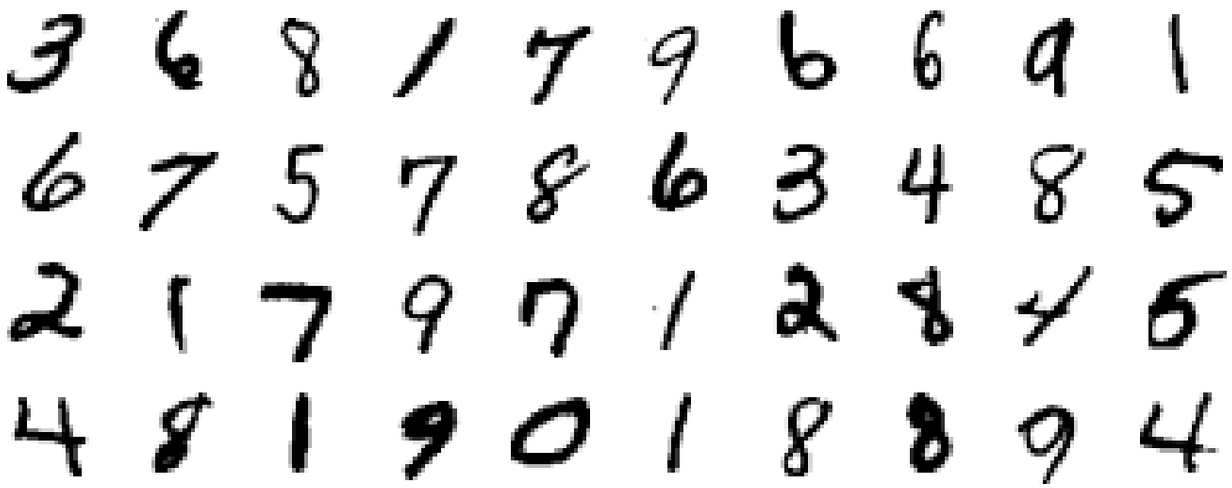
*Joan Bruna*

- Each class $X_k$ is represented by a scattering centroid $E(SX_k)$

  Affine space model $\mathbf{A}_k = E(SX_k) + \mathbf{V}_k$. computed with PCA.




MNIST data basis:

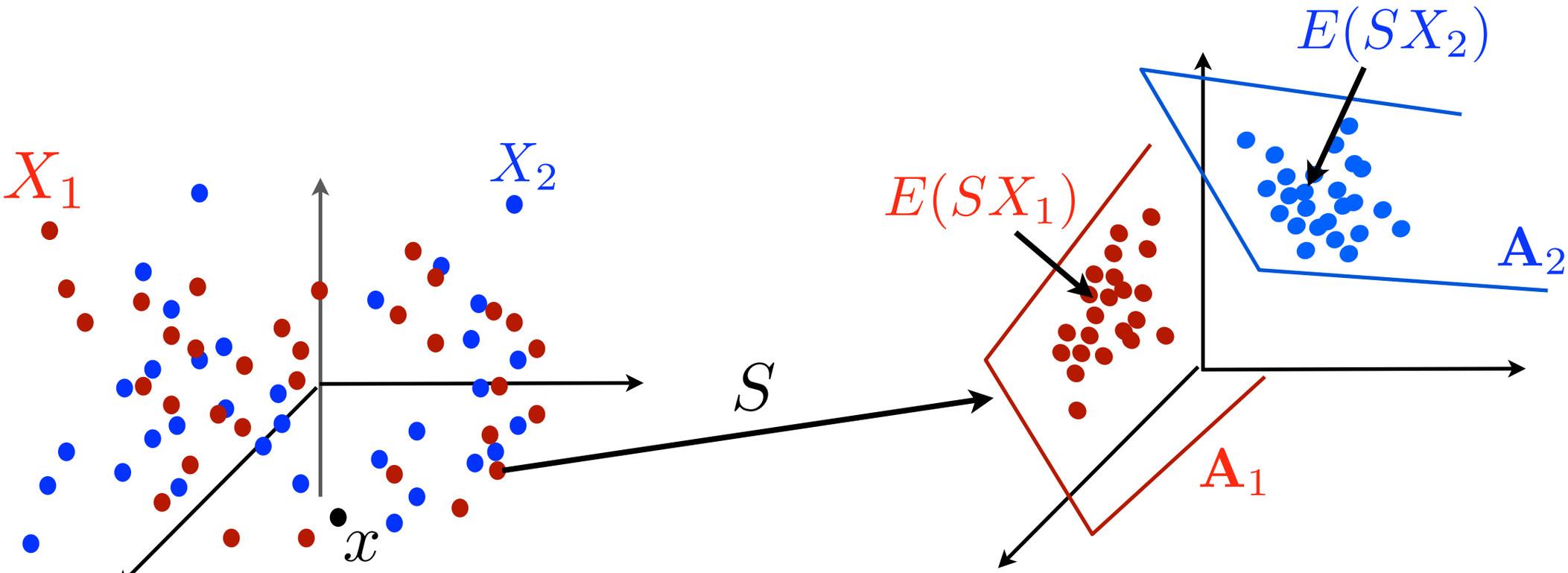


# Linearized Classification

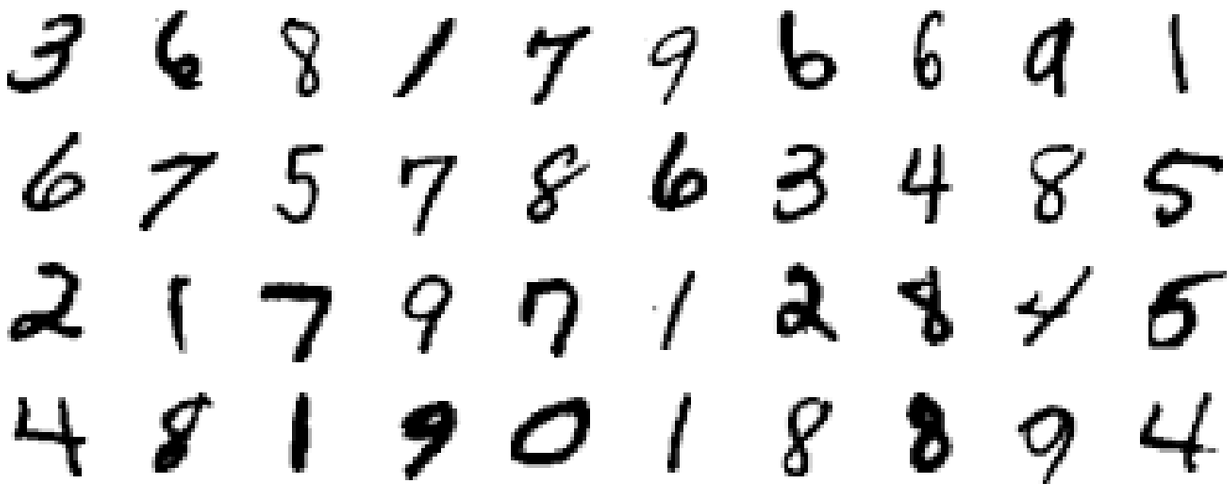
*Joan Bruna*

- Each class $X_k$ is represented by a scattering centroid $E(SX_k)$

  Affine space model $\mathbf{A}_k = E(SX_k) + \mathbf{V}_k$. computed with PCA.




MNIST data basis:

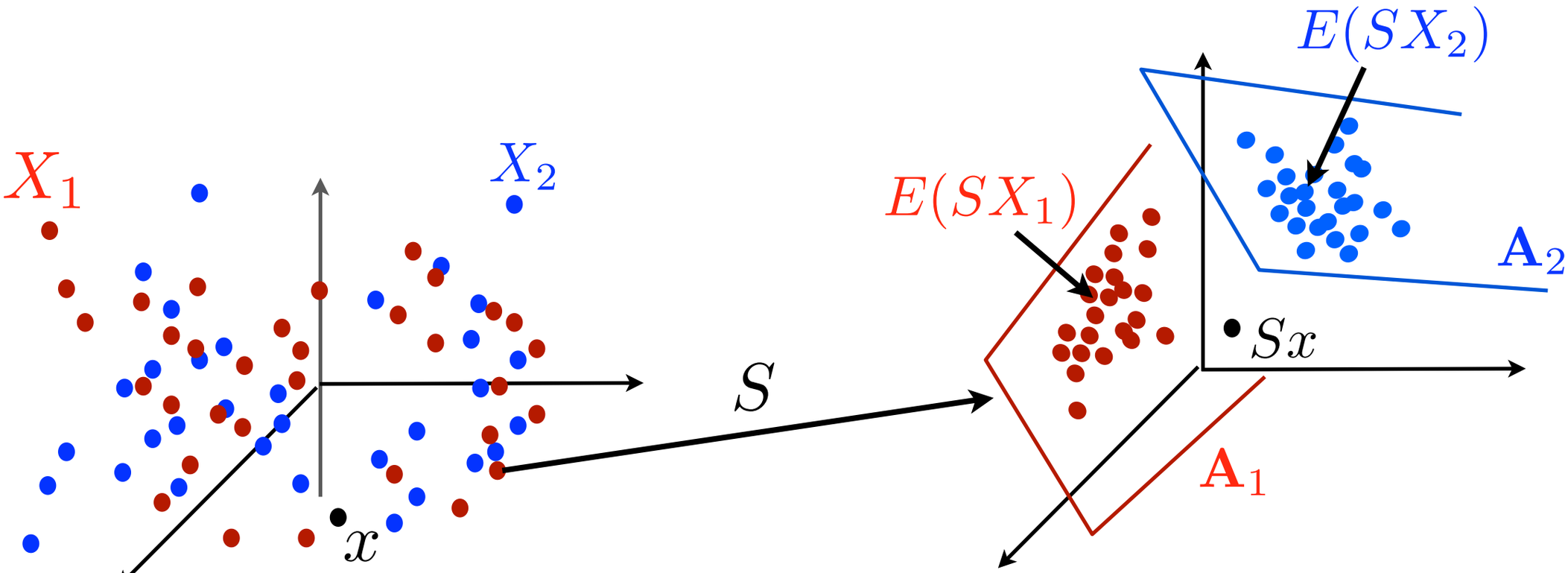


# Linearized Classification

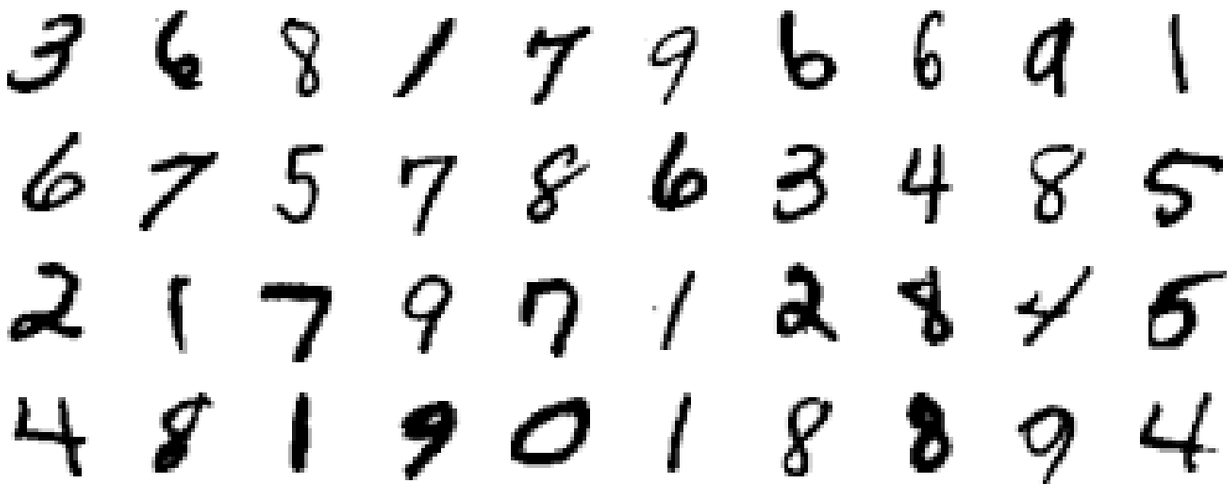
*Joan Bruna*

- Each class $X_k$ is represented by a scattering centroid $E(SX_k)$

  Affine space model $\mathbf{A}_k = E(SX_k) + \mathbf{V}_k$. computed with PCA.




MNIST data basis:

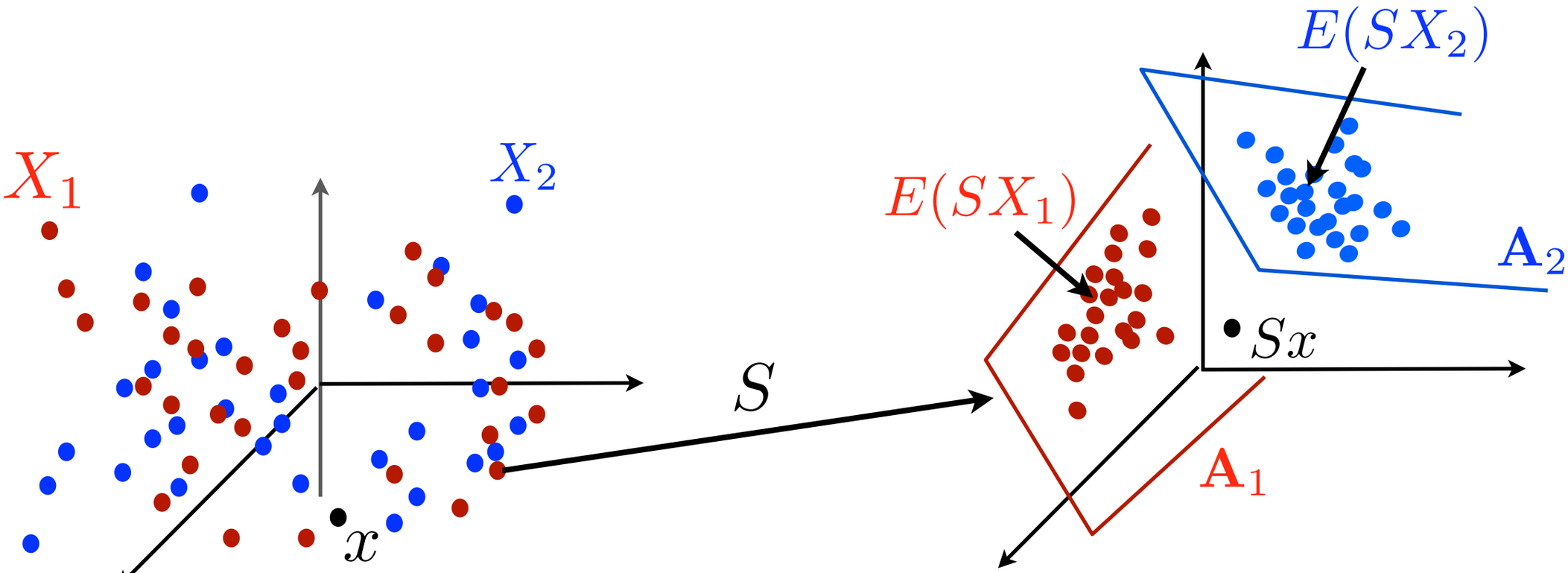


# Linearized Classification

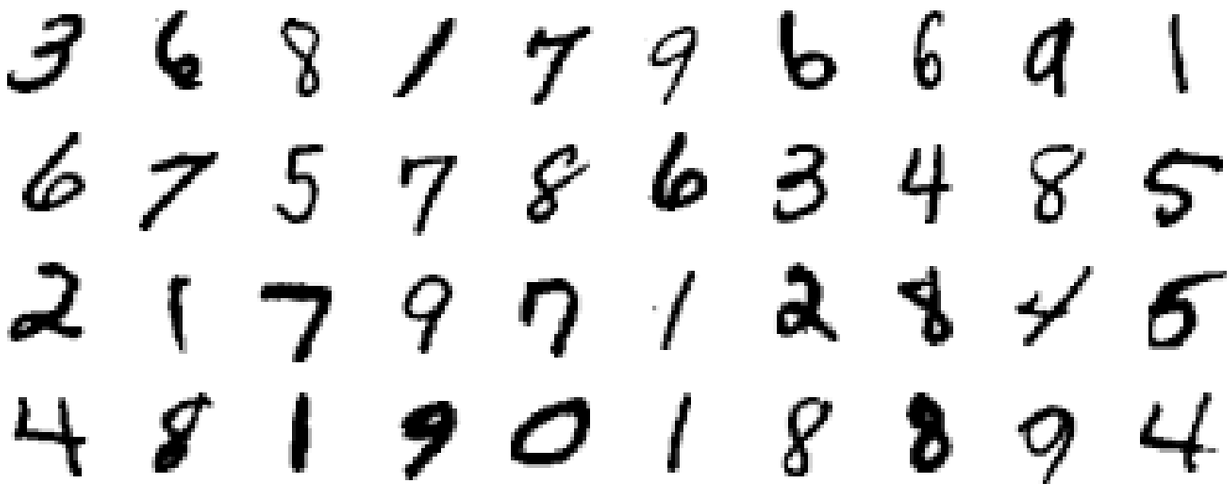
*Joan Bruna*

- Each class $X_k$ is represented by a scattering centroid $E(SX_k)$

  Affine space model $\mathbf{A}_k = E(SX_k) + \mathbf{V}_k$. computed with PCA.




MNIST data basis:



# Scattering Moments

The scattering transform of a stationary process $X(t)$

$$SX(t) = \begin{pmatrix} X \star \phi(t) \\ |X \star \psi_{\lambda_1}| \star \phi(t) \\ ||X \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(t) \\ |||X \star \psi_{\lambda_2}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}| \star \phi(t) \\ \dots \end{pmatrix}_{\lambda_1, \lambda_2, \lambda_3, \dots}$$

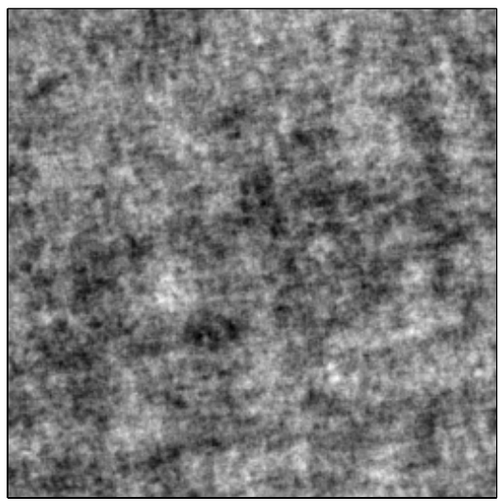
is an estimator of the expected scattering of $X(t)$

$$\overline{S}X = \begin{pmatrix} E(X) \\ E(|X \star \psi_{\lambda_1}|) \\ E(||X \star \psi_{\lambda_1}| \star \psi_{\lambda_2}|) \\ E(|||X \star \psi_{\lambda_2}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}|) \\ \dots \end{pmatrix}_{\lambda_1, \lambda_2, \lambda_3, \dots}$$

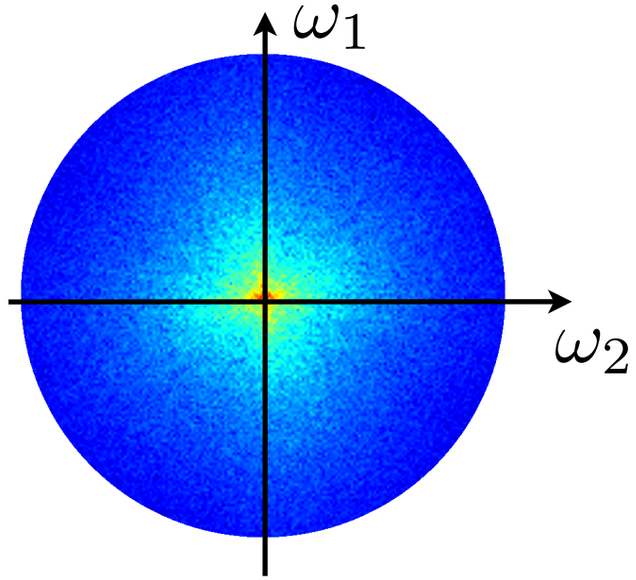
# Textures with Same Spectrum

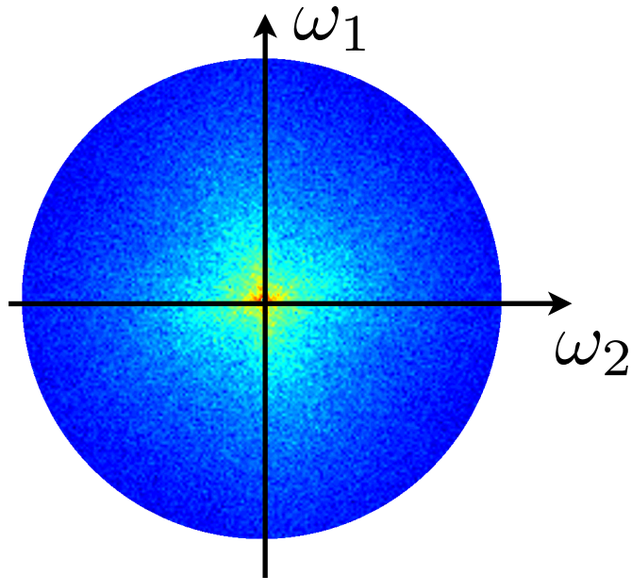
$x(t)$: stationary process

Textures
$x(t)$



Fourier
Power Spectrum

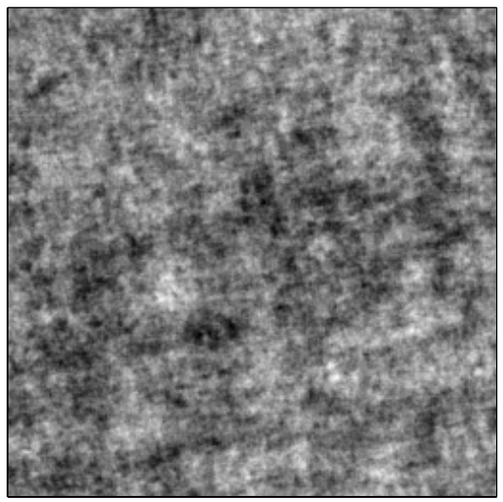


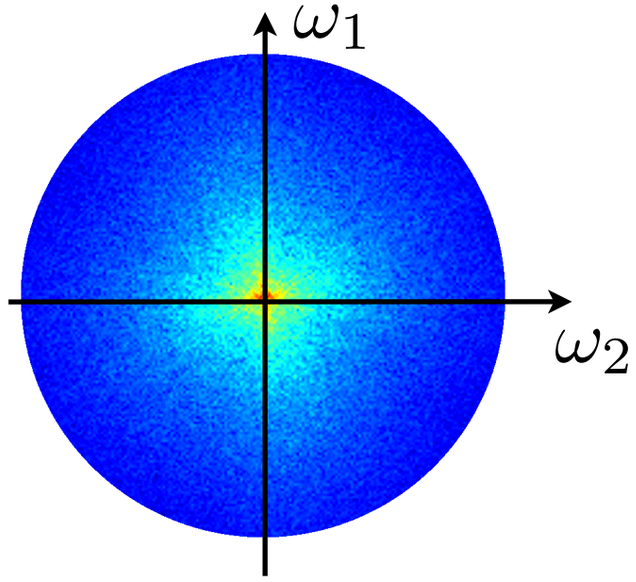
# Textures with Same Spectrum

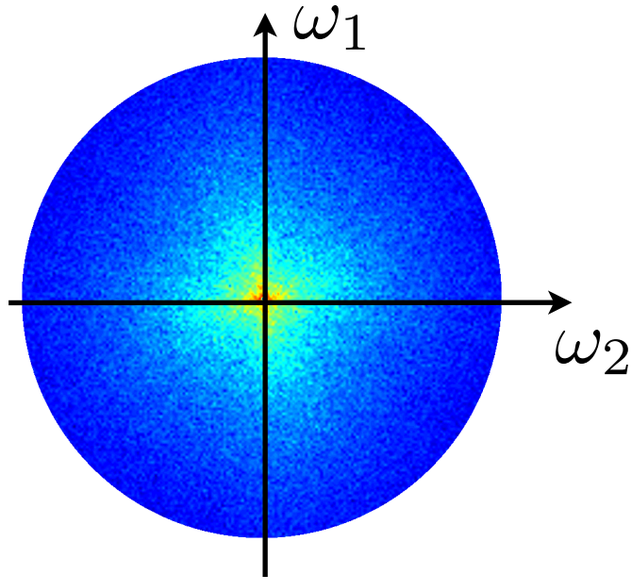
$x(t)$: stationary process

Textures
$x(t)$



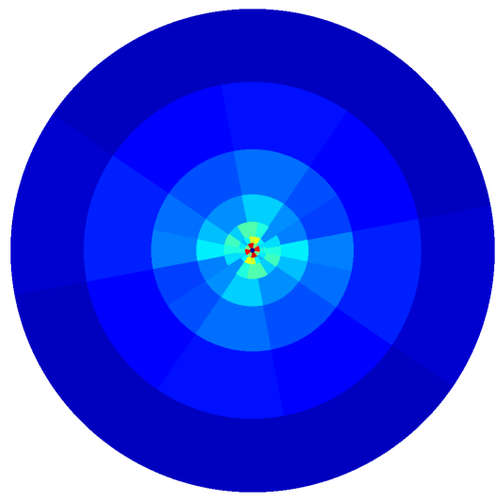
Fourier
Power Spectrum

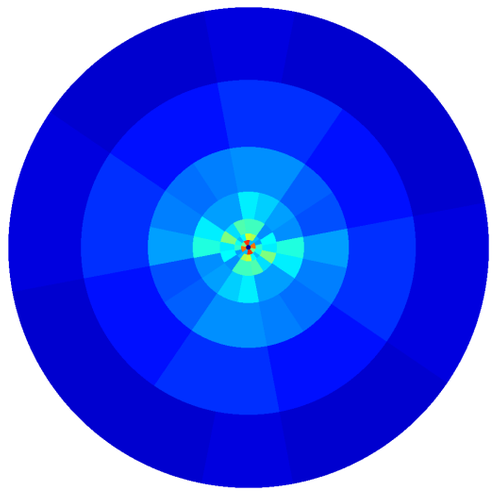



Wavelet Scattering

$|x \star \psi_{\lambda_1}| \star \phi$



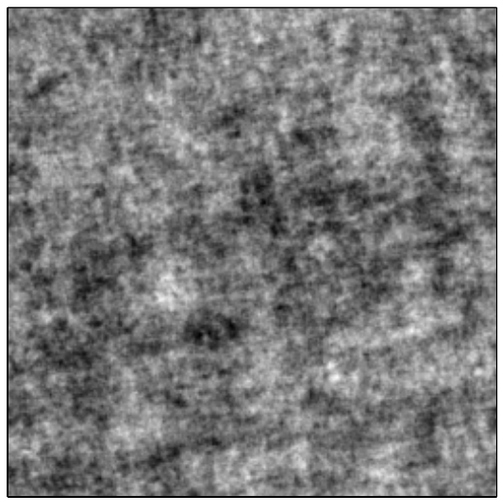


window size = image size

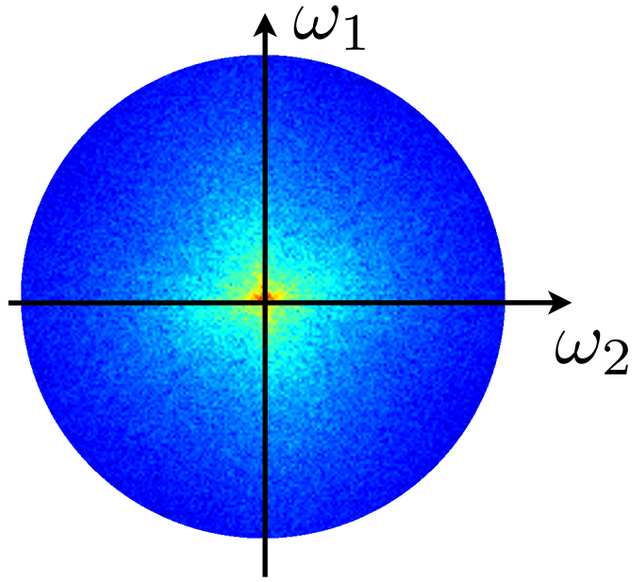
# Textures with Same Spectrum

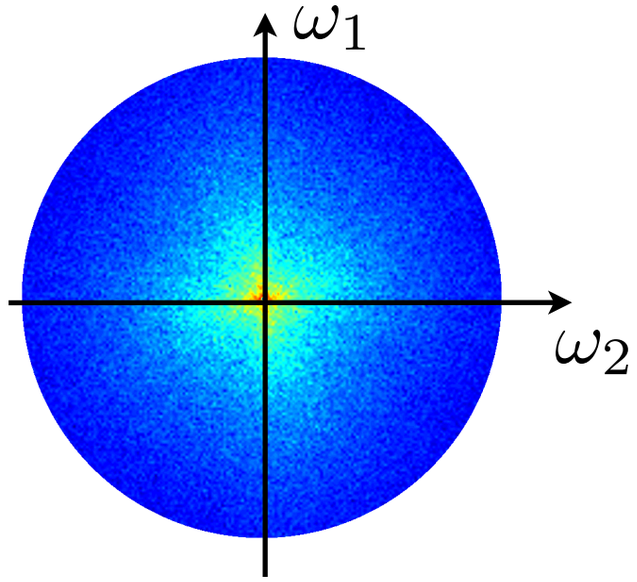
$x(t)$: stationary process

Textures
$x(t)$



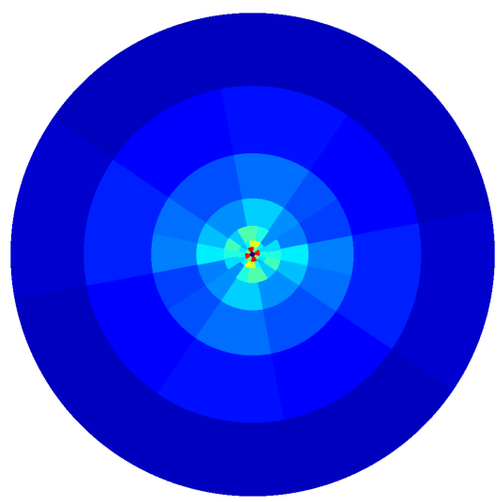
Fourier
Power Spectrum

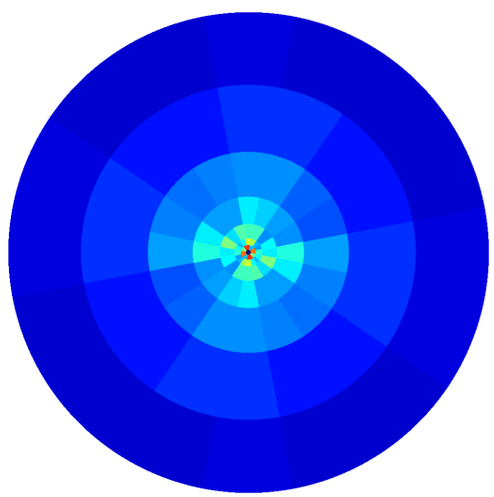


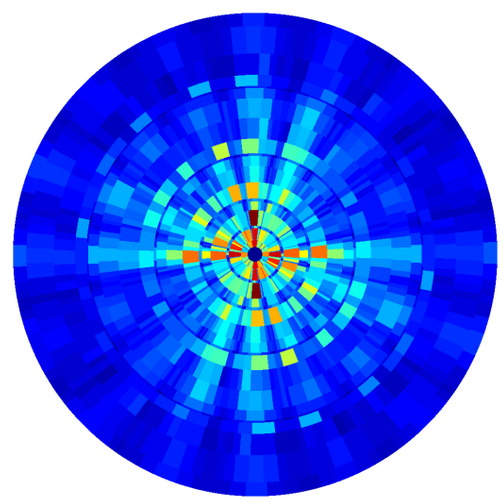

Wavelet Scattering

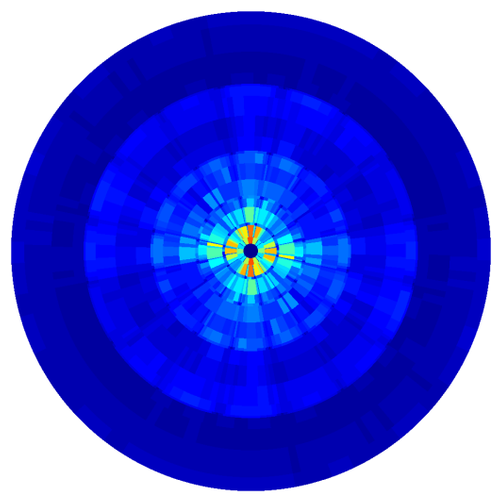
$|x \star \psi_{\lambda_1}| \star \phi$





$||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi$





window size = image size

# Sounds with Same Spectrum

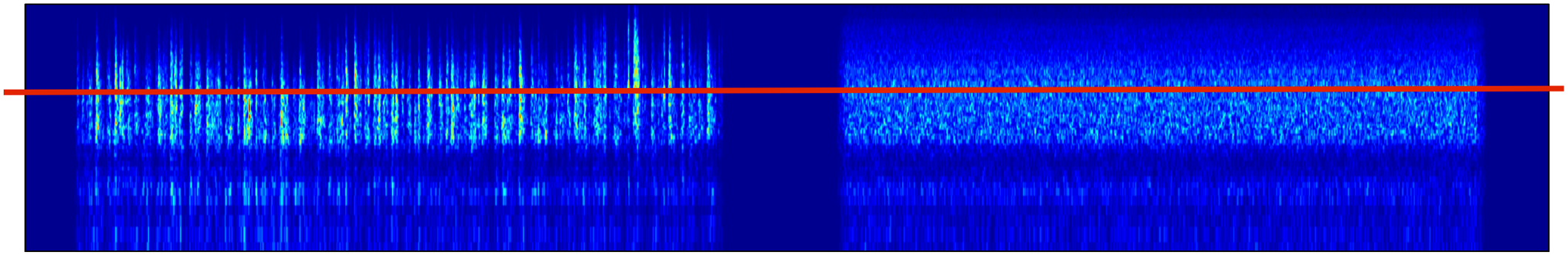
$X$: stationary process

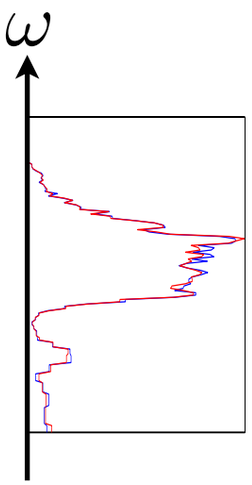
Fourier Spectrum

$\log(\lambda_1)$

J. McDermott

$|x \star \psi_{\lambda_1}|(t)$

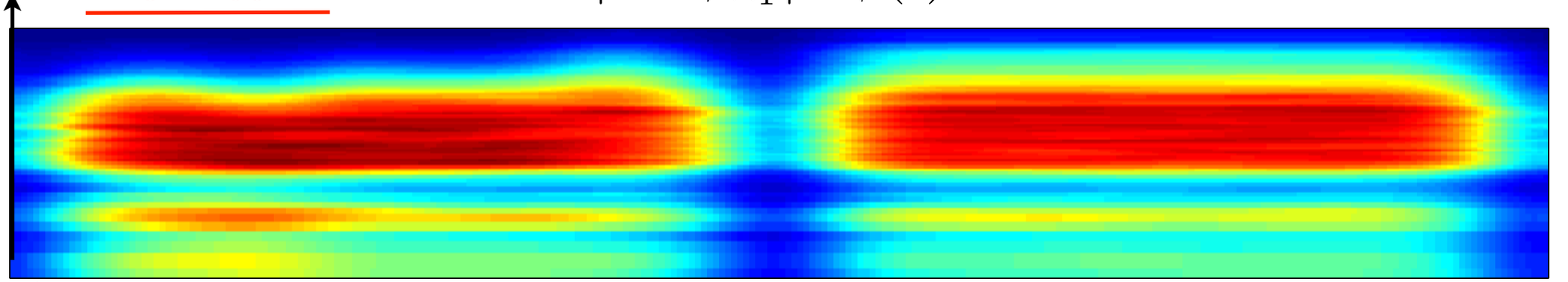





$\log(\lambda_1)$

2s window

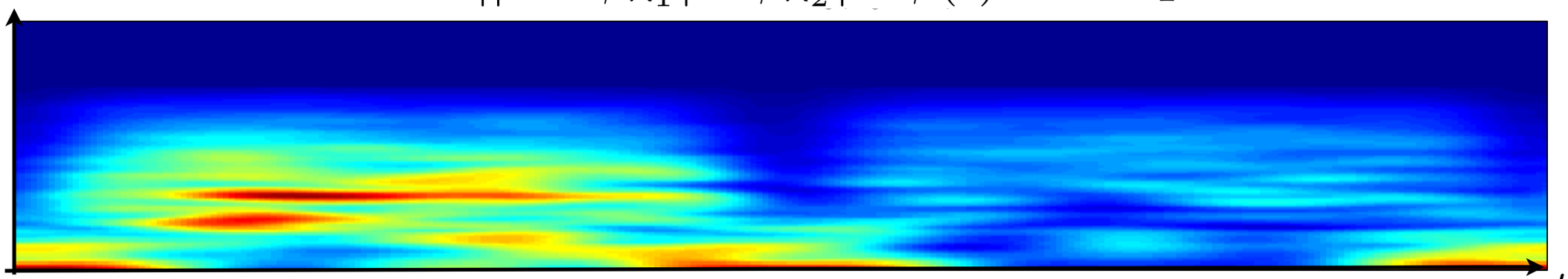
$|x \star \psi_{\lambda_1}| \star \phi(t)$



$\log(\lambda_2)$

$t$

$||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(t)$ for $\lambda_1 = 2000$



$t$

$t$

# Sounds with Same Spectrum

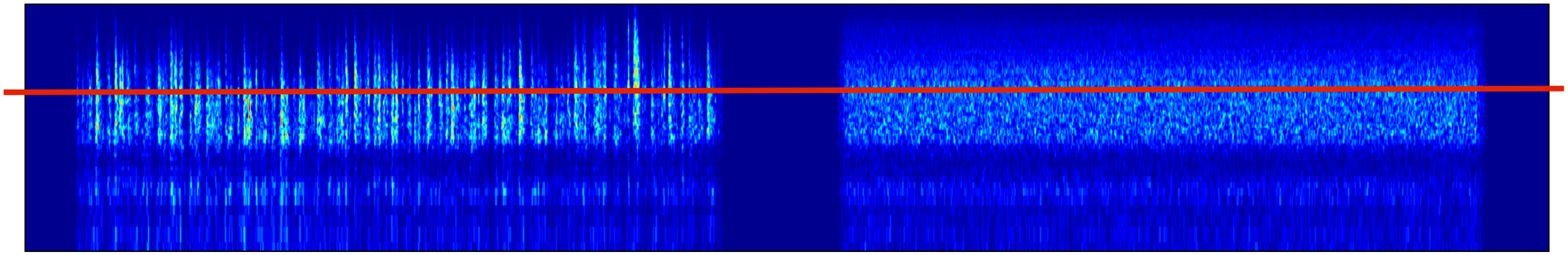
$X$: stationary process

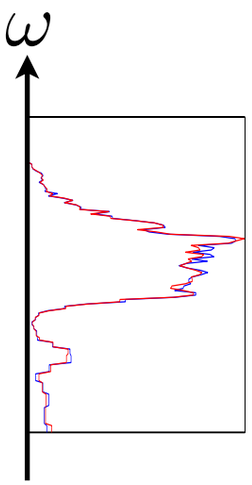
Fourier Spectrum

$\log(\lambda_1)$

J. McDermott

$|x \star \psi_{\lambda_1}|(t)$

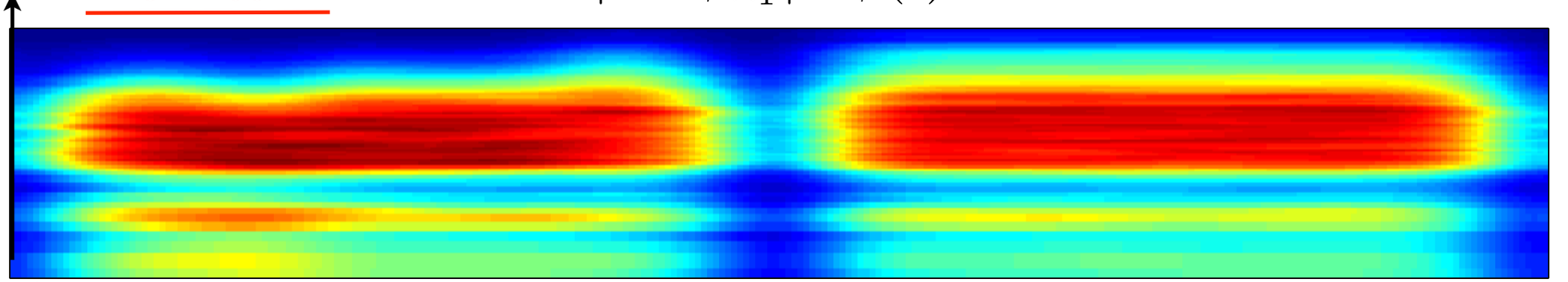





$\log(\lambda_1)$

2s window

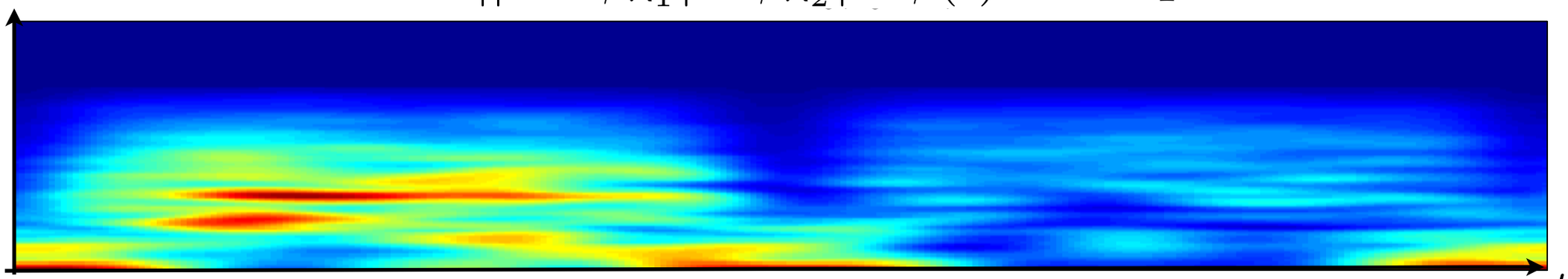
$|x \star \psi_{\lambda_1}| \star \phi(t)$



$\log(\lambda_2)$

$t$

$||x \star \psi_{\lambda_1}| \star \psi_{\lambda_2}| \star \phi(t)$ for $\lambda_1 = 2000$



$t$

$t$

# Representation of Random Processes

- An expected scattering is a non-complete representation

$$\overline{S}X = \begin{pmatrix} E(X) & = & E(U_0 X) \\ E(|X \star \psi_{\lambda_1}|) & = & E(U_1 X) \\ E(||X \star \psi_{\lambda_1}| \star \psi_{\lambda_2}|) & = & E(U_2 X) \\ E(|||X \star \psi_{\lambda_2}| \star \psi_{\lambda_2}| \star \psi_{\lambda_3}|) & = & E(U_3 X) \\ \ldots & & \end{pmatrix}_{\lambda_1, \lambda_2, \lambda_3, \ldots}$$

**Theorem** *(Boltzmann)* The distribution $p(x)$ which satisfies

$$\int_{\mathbb{R}^N} U_m x \; p(x)\, dx = E(U_m X)$$

and maximizes the entropy $-\int p(x) \; \log p(x)\, dx$

can be written: $$p(x) = \frac{1}{Z} \exp\Big( \sum_{m=1}^{\infty} \lambda_m \,.\, U_m x \Big)$$

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

Cocktail Party

# Synthesis from Second Order

*Joakim Anden  Joan Bruna*

- Maximum entropy estimation of $X(t)$ :
  - Gaussian model from $N$ power spectrum coefficients.
  - Scattering model from $(\log_2 N)^2/2$ 1st & 2nd orders.

J. McDermott textures

Original  Gaussian Model  Scattering Moments

JackHammer

Water

Applause

Paper

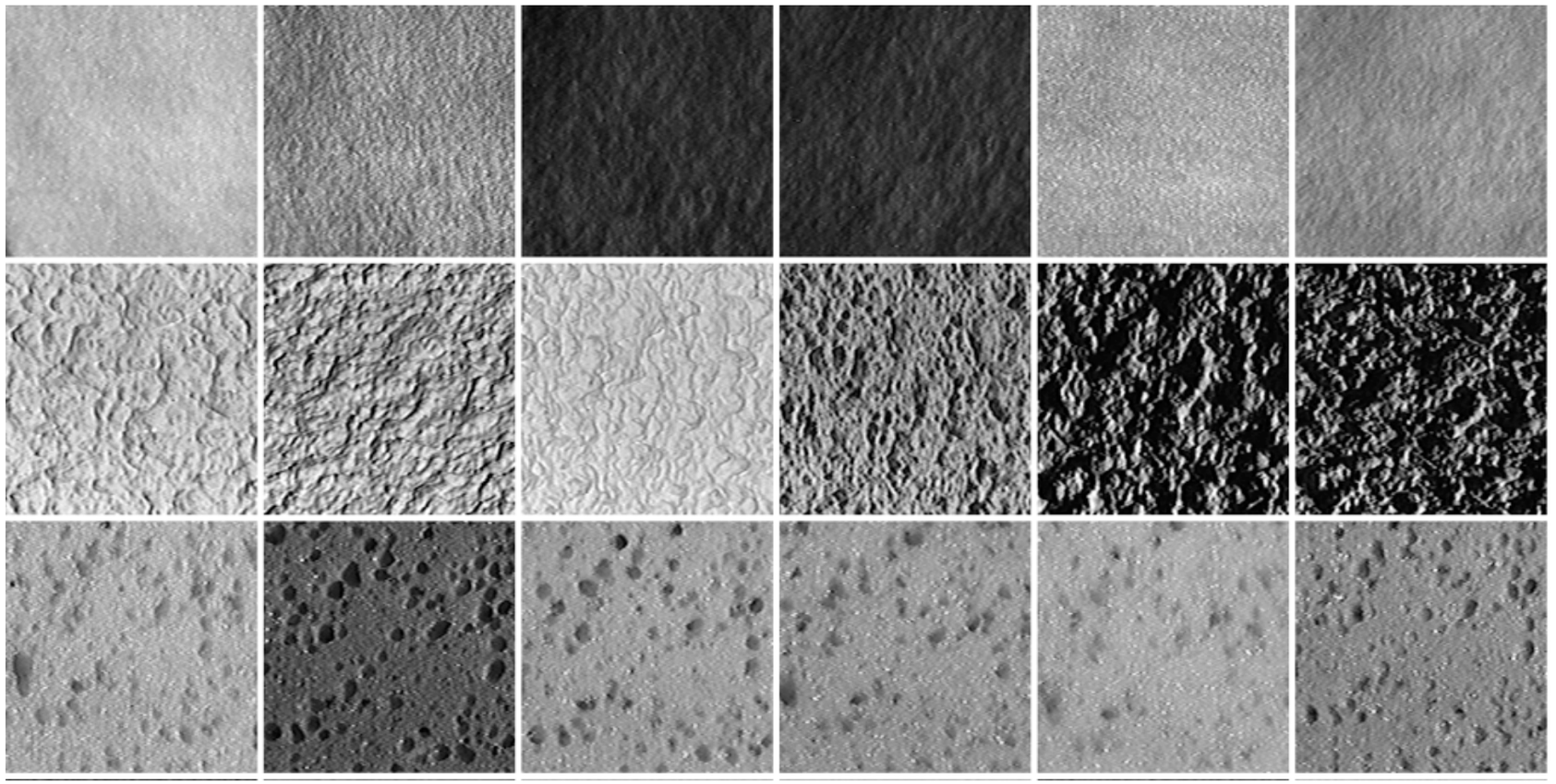
Cocktail Party

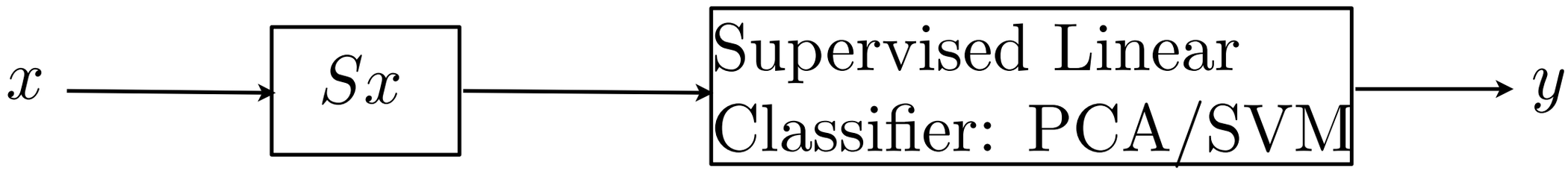
Not good for everything: learn from mistakes.

# Classification of Textures

*J. Bruna*

CUREt database
61 classes






| Training per class | Fourier Spectr. | Histogr. Features | Scattering |
|---|---|---|---|
| 46 | 1% | 1% | **0.2** % |

# Wavelet Transform on a Group

*Laurent Sifre*

- Roto-translation group $G = \{g = (r, t) \in SO(2) \times \mathbb{R}^2\}$

$$(r, t) \,.\, x(u) = x(r^{-1}(u - t))$$

# Wavelet Transform on a Group

*Laurent Sifre*

- Roto-translation group $G = \{g = (r, t) \in SO(2) \times \mathbb{R}^2\}$

$$(r, t) \,.\, x(u) = x(r^{-1}(u - t))$$

- Averaging on $G$: $\quad X \circledast \overline{\phi}(g) = \int_G X(g') \, \overline{\phi}(g'^{-1} g) \, dg'$

# Wavelet Transform on a Group

*Laurent Sifre*

- Roto-translation group $G = \{g = (r, t) \in SO(2) \times \mathbb{R}^2\}$

$$(r, t) \,.\, x(u) = x(r^{-1}(u - t))$$

- Averaging on $G$: $\quad X \circledast \overline{\phi}(g) = \int_G X(g')\, \overline{\phi}(g'^{-1} g)\, dg'$

- Wavelet transform on $G$: $\quad W_2 X = \begin{pmatrix} X \circledast \overline{\phi}(g) \\ X \circledast \overline{\psi}_{\lambda_2}(g) \end{pmatrix}_{\lambda_2, g}$ .

# Wavelet Transform on a Group

*Laurent Sifre*

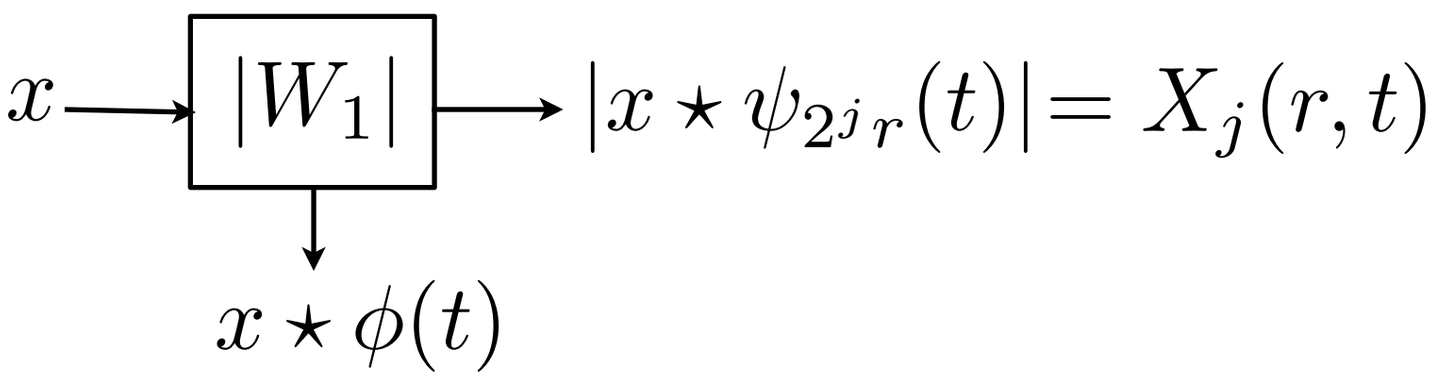
- Roto-translation group $G = \{g = (r, t) \in SO(2) \times \mathbb{R}^2\}$

$$(r, t) \,.\, x(u) = x(r^{-1}(u - t))$$

- Averaging on $G$: $\quad X \circledast \overline{\phi}(g) = \int_G X(g')\, \overline{\phi}(g'^{-1} g)\, dg'$

- Wavelet transform on $G$: $\quad W_2 X = \begin{pmatrix} X \circledast \overline{\phi}(g) \\ X \circledast \overline{\psi}_{\lambda_2}(g) \end{pmatrix}_{\lambda_2, g}$ .

translation



# Wavelet Transform on a Group

*Laurent Sifre*

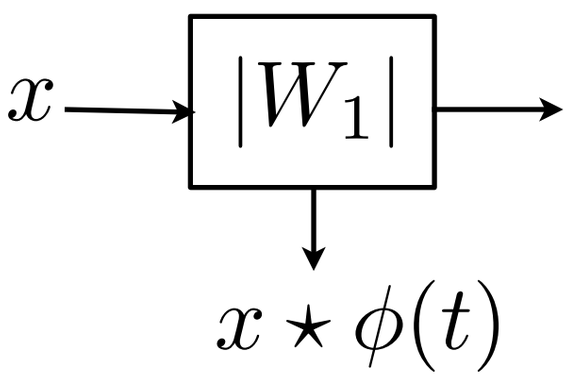
- Roto-translation group $G = \{g = (r, t) \in SO(2) \times \mathbb{R}^2\}$

$$(r, t) \,.\, x(u) = x(r^{-1}(u - t))$$

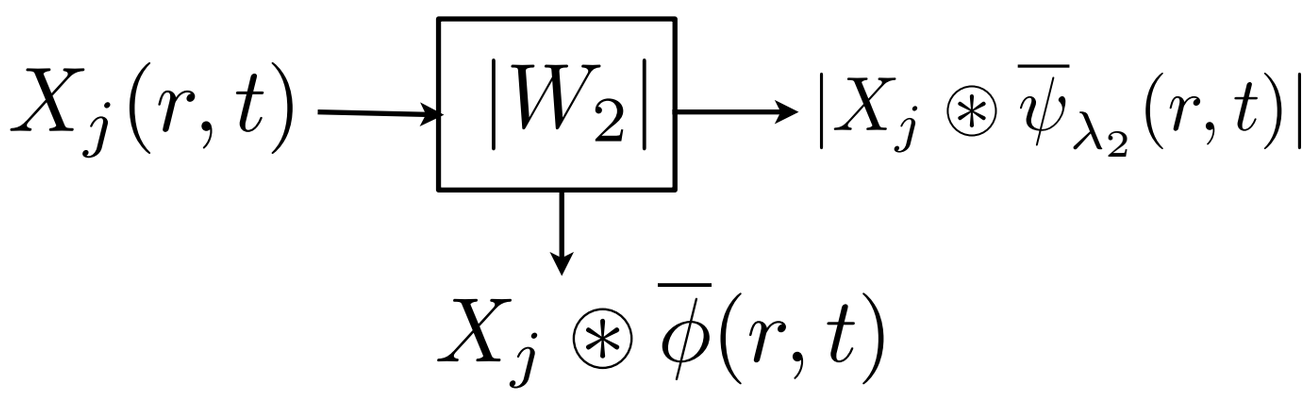
- Averaging on $G$: $\quad X \circledast \overline{\phi}(g) = \int_G X(g')\, \overline{\phi}(g'^{-1} g)\, dg'$

- Wavelet transform on $G$: $\quad W_2 X = \begin{pmatrix} X \circledast \overline{\phi}(g) \\ X \circledast \overline{\psi}_{\lambda_2}(g) \end{pmatrix}_{\lambda_2, g}$ .

translation




roto-translation



# Wavelet Transform on a Group

*Laurent Sifre*

- Roto-translation group $G = \{g = (r, t) \in SO(2) \times \mathbb{R}^2\}$

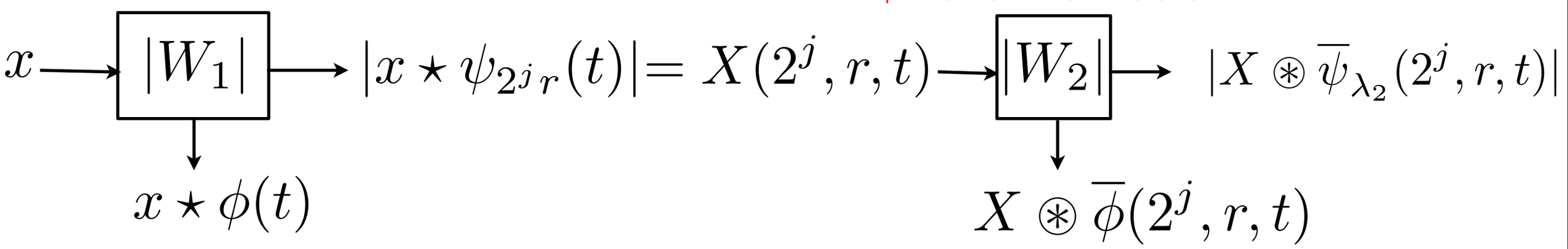
$$(r, t) \,.\, x(u) = x(r^{-1}(u - t))$$

- Averaging on $G$: $\quad X \circledast \overline{\phi}(g) = \int_G X(g')\, \overline{\phi}(g'^{-1} g)\, dg'$

- Wavelet transform on $G$: $\quad W_2 X = \begin{pmatrix} X \circledast \overline{\phi}(g) \\ X \circledast \overline{\psi}_{\lambda_2}(g) \end{pmatrix}_{\lambda_2, g}$ .

translation

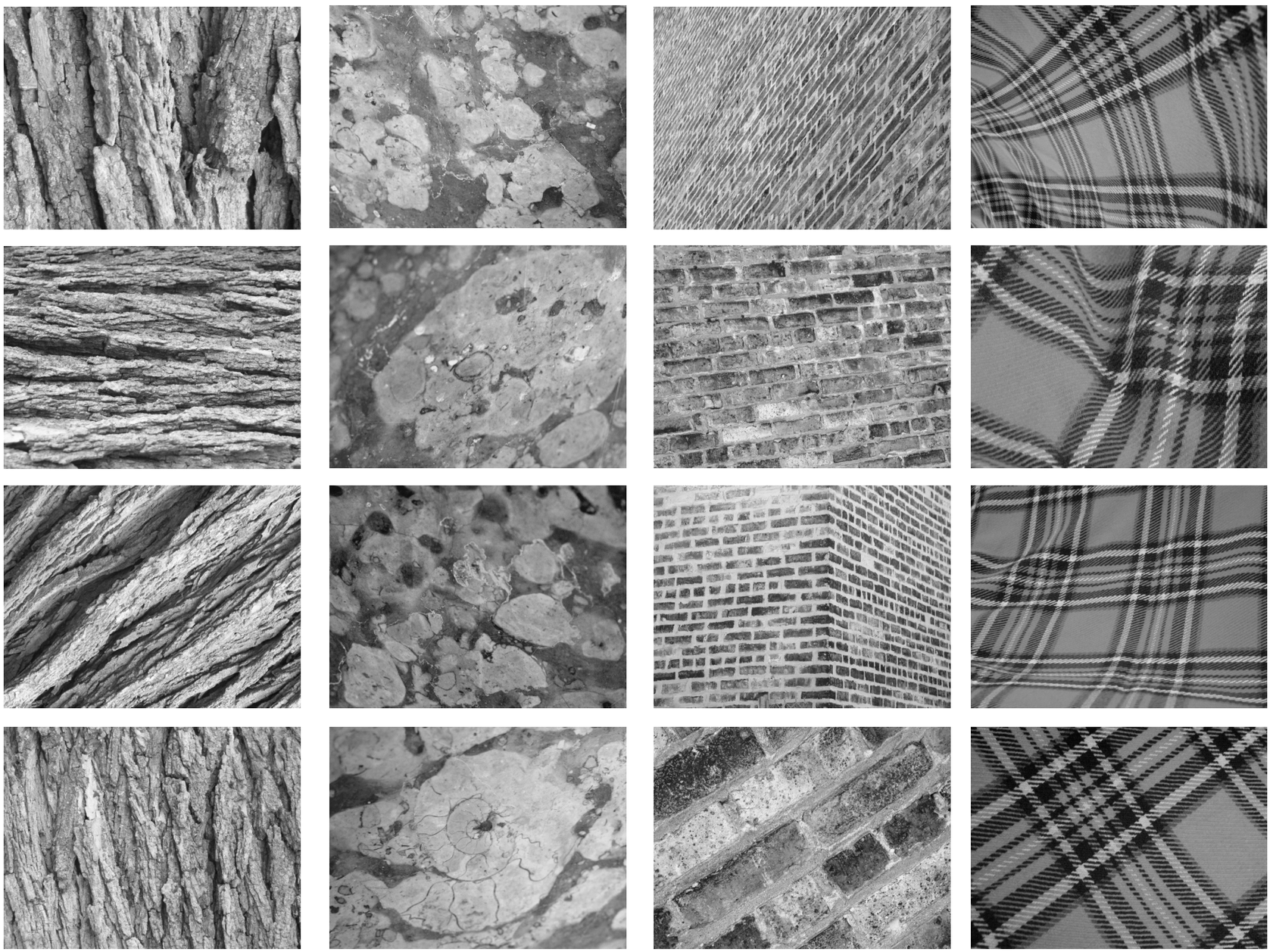
scalo-roto-translation
+ renormalization



# Rotation and Scaling Invariance

*Laurent Sifre*

UIUC database:
25 classes



Scattering classification errors

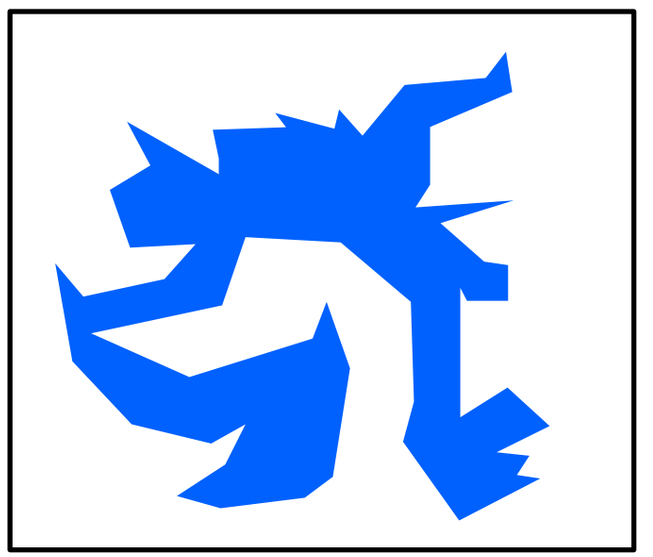
| Training | Translation | Transl + Rotation | + Scaling |
|---|---|---|---|
| 20 | 20 % | 2% | **0.6**% |

# Unsupervised Learning

Unsupervised Learning
Representation

$$x \in \mathbb{R}^d \longrightarrow \boxed{\Phi} \longrightarrow \Phi x \in \mathbb{R}^D$$

- **Unsupervised learning** of $\Phi$ from unlabeled examples $\{x_i\}$:
  - model the $\{x_i\}_i$ as realization of a random vector $X \in \mathbb{R}^d$
  - estimate a representation of $\Phi X$ of $p(x)$



# Unsupervised Learning

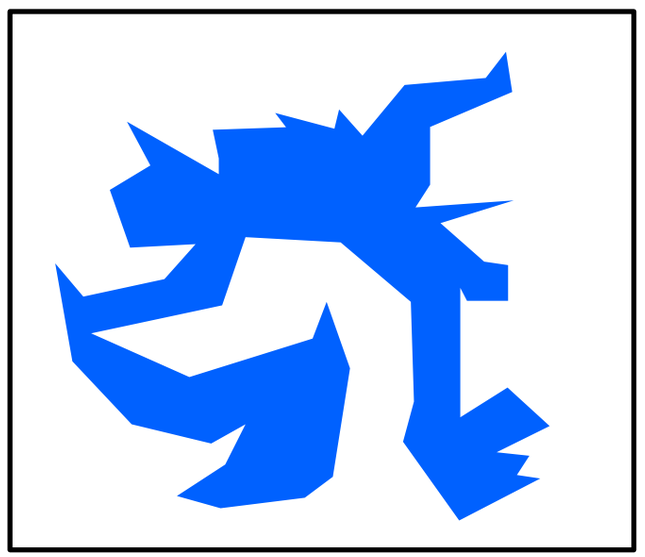
Unsupervised Learning
Representation

$$x \in \mathbb{R}^d \longrightarrow \boxed{\Phi} \longrightarrow \Phi x \in \mathbb{R}^D$$

- **Unsupervised learning** of $\Phi$ from unlabeled examples $\{x_i\}$:
  - model the $\{x_i\}_i$ as realization of a random vector $X \in \mathbb{R}^d$
  - estimate a representation of $\Phi X$ of $p(x)$

- Classical approaches: clustering and Gaussian mixture models

  decomposition in ellipsoids
  not feasible in high dimensions



$p(x)$

# Unsupervised Learning

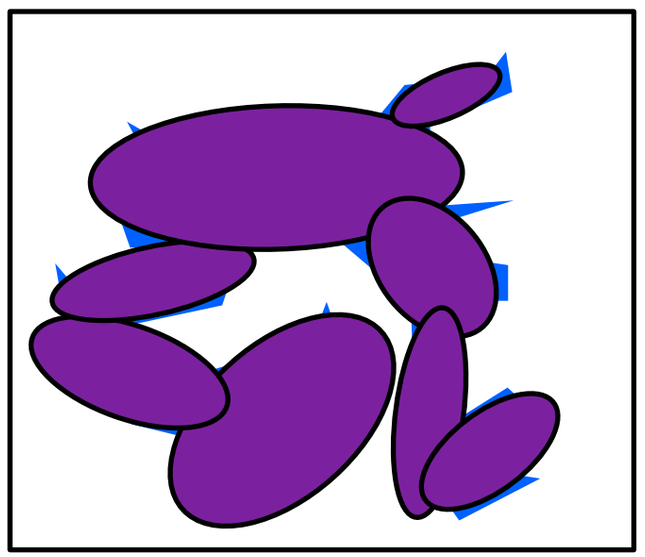
Unsupervised Learning
Representation

$$x \in \mathbb{R}^d \longrightarrow \boxed{\Phi} \longrightarrow \Phi x \in \mathbb{R}^D$$

- **Unsupervised learning** of $\Phi$ from unlabeled examples $\{x_i\}$:
  - model the $\{x_i\}_i$ as realization of a random vector $X \in \mathbb{R}^d$
  - estimate a representation of $\Phi X$ of $p(x)$

- Classical approaches: clustering and Gaussian mixture models

decomposition in ellipsoids
not feasible in high dimensions



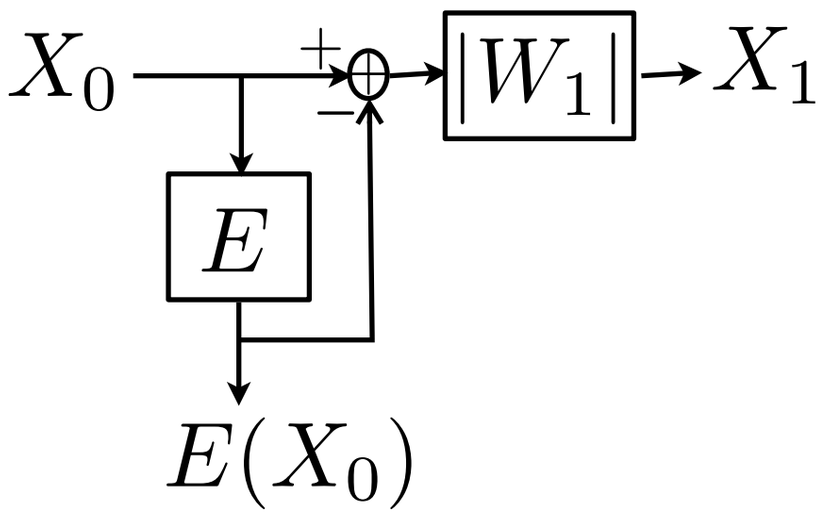
$p(x)$

# Generalized Scattering

$X_0$

# Generalized Scattering

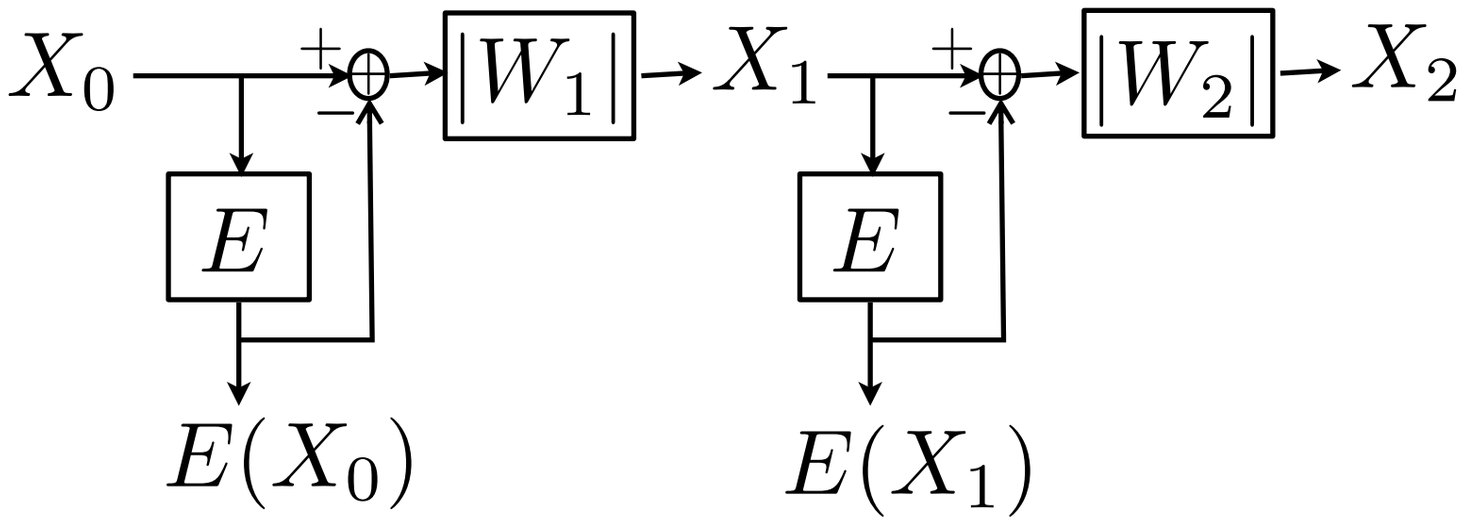
$$X_1 = |W_1(X_0 - E(X_0)| \quad \text{with} \quad W_1^* W_1 = Id$$



# Generalized Scattering

$$X_1 = |W_1(X_0 - E(X_0)| \quad \text{with} \quad W_1^* W_1 = Id$$

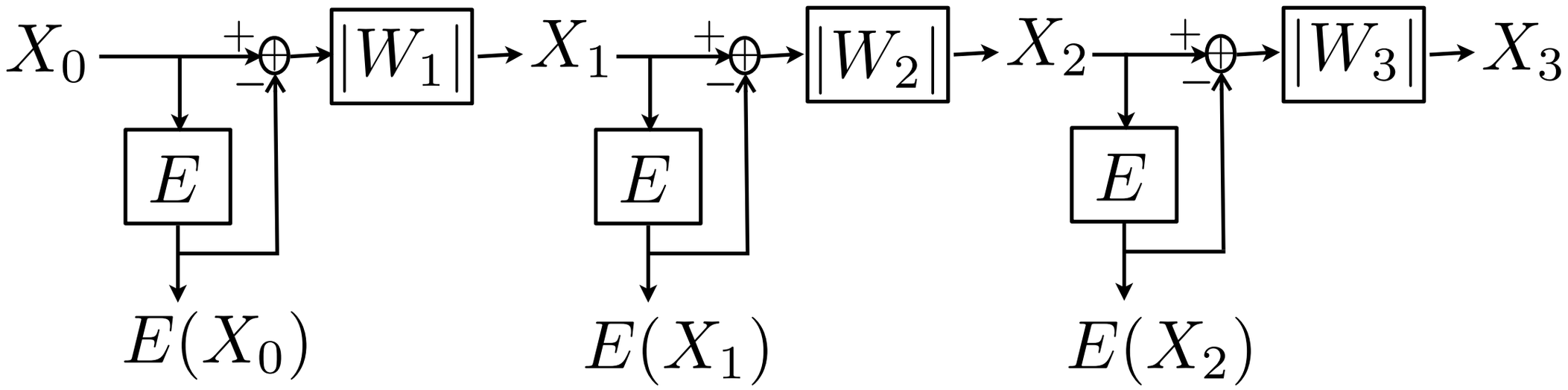
$$\forall m \quad X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with} \quad W_m^* W_m = Id$$



# Generalized Scattering

$$X_1 = |W_1(X_0 - E(X_0)| \quad \text{with} \quad W_1^* W_1 = Id$$

$$\forall m \quad X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with} \quad W_m^* W_m = Id$$

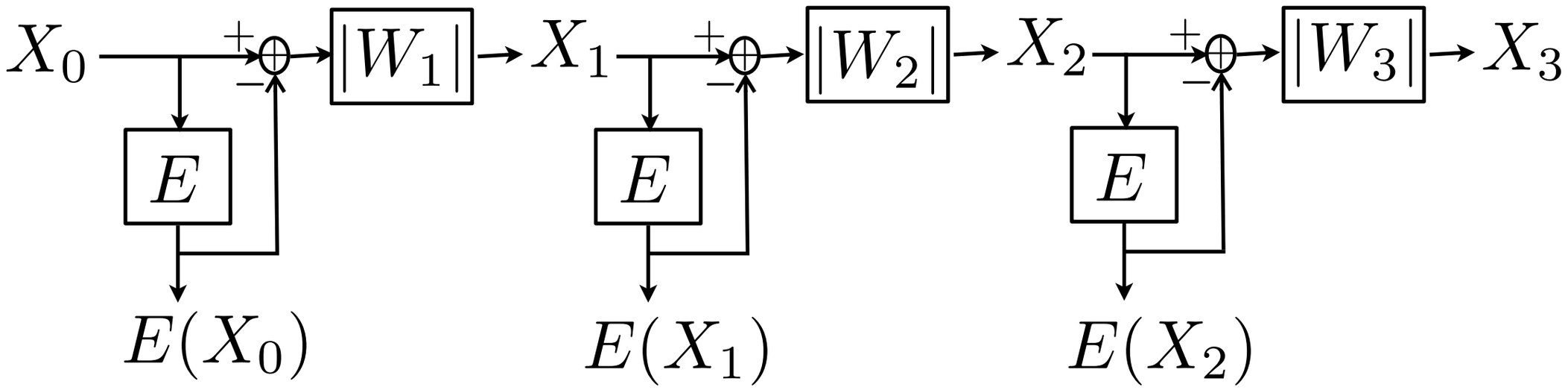



- Expected scattering transform: $\overline{S}X = \{E(X_m)\}_{m \in \mathbb{N}}$

# Generalized Scattering

$$X_1 = |W_1(X_0 - E(X_0)| \quad \text{with} \quad W_1^* W_1 = Id$$

$$\forall m \quad X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with} \quad W_m^* W_m = Id$$

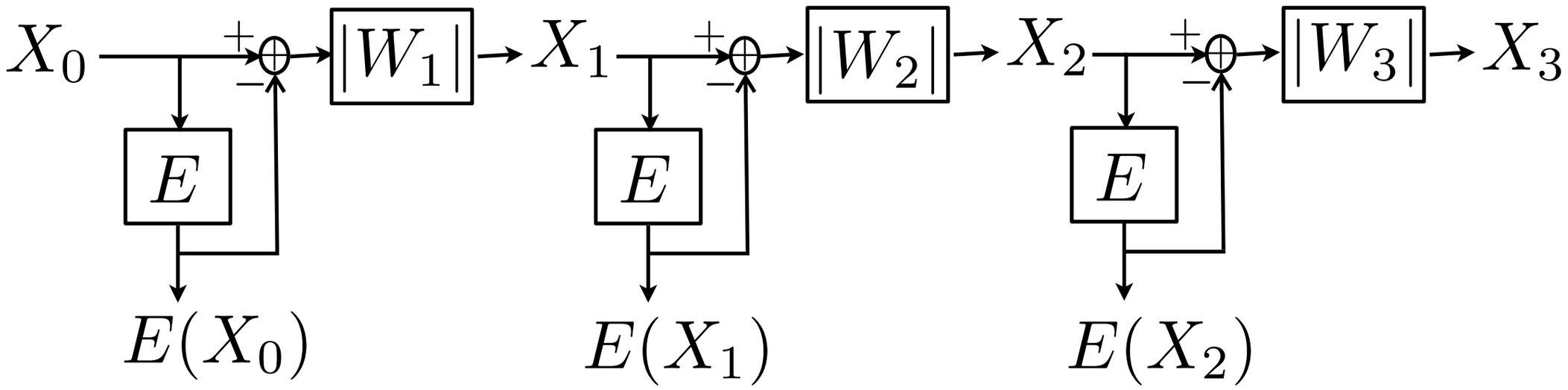



- Expected scattering transform: $\overline{S}X = \{E(X_m)\}_{m \in \mathbb{N}}$ represents the probability density of $X$.

# Generalized Scattering

$$X_1 = |W_1(X_0 - E(X_0)| \quad \text{with} \quad W_1^* W_1 = Id$$

$$\forall m \quad X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with} \quad W_m^* W_m = Id$$



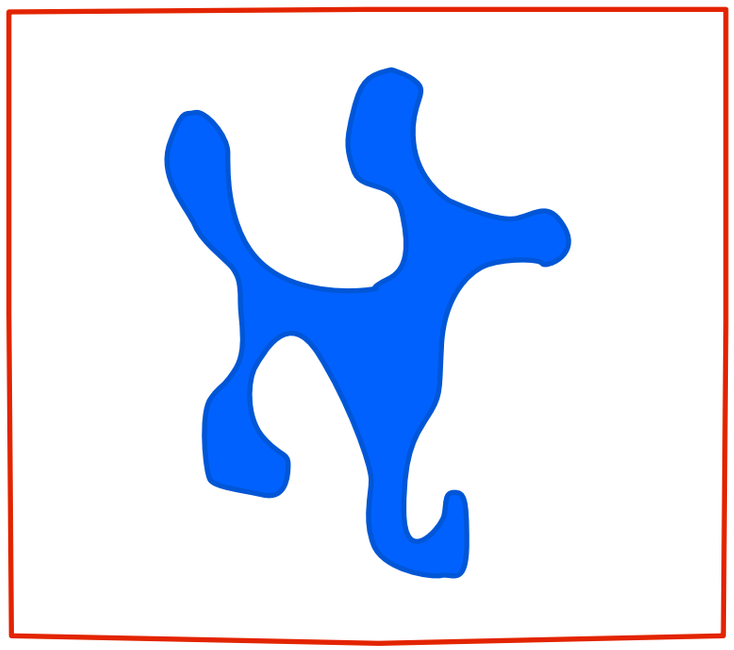

- Expected scattering transform: $\overline{S}X = \{E(X_m)\}_{m \in \mathbb{N}}$ represents the probability density of $X$.

**Theorem**: $$\|\overline{S}X - \overline{S}Y\| \leq E(\|X - Y\|^2)$$

$$\|\overline{S}X\|^2 = E(\|X\|^2)$$

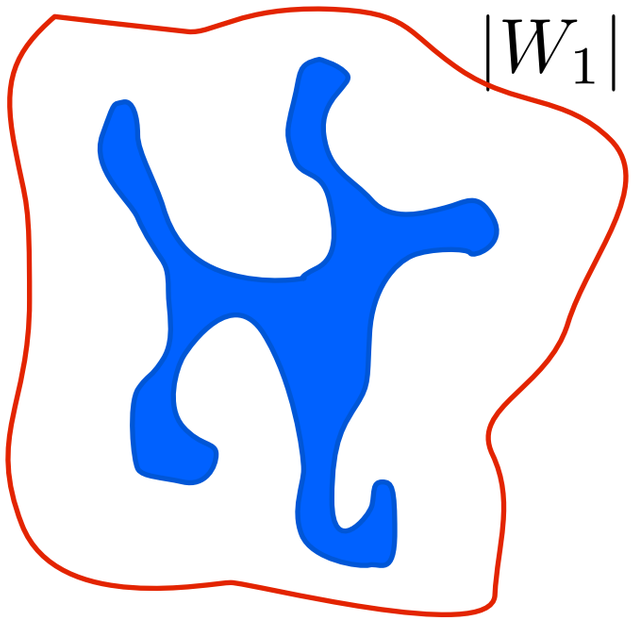
# Unsupervised Learning by Scattering

- Deep scattering perform adaptive space contractions
- Squeeze the space while minimizing the data volume reduction



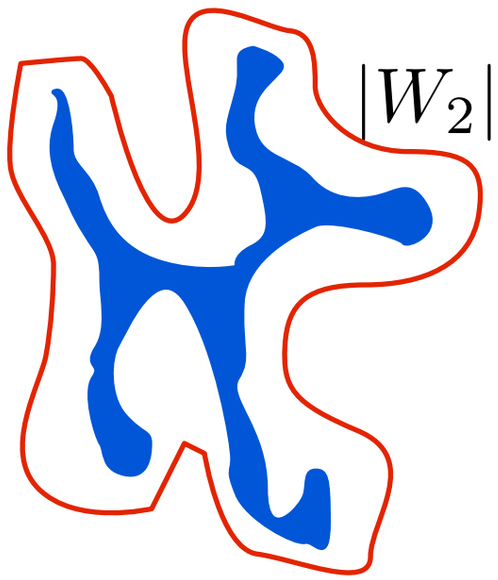
# Unsupervised Learning by Scattering

- Deep scattering perform adaptive space contractions
- Squeeze the space while minimizing the data volume reduction



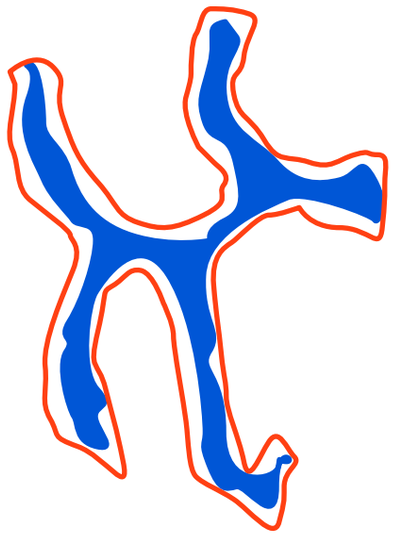
# Unsupervised Learning by Scattering

- Deep scattering perform adaptive space contractions
- Squeeze the space while minimizing the data volume reduction



# Unsupervised Learning by Scattering

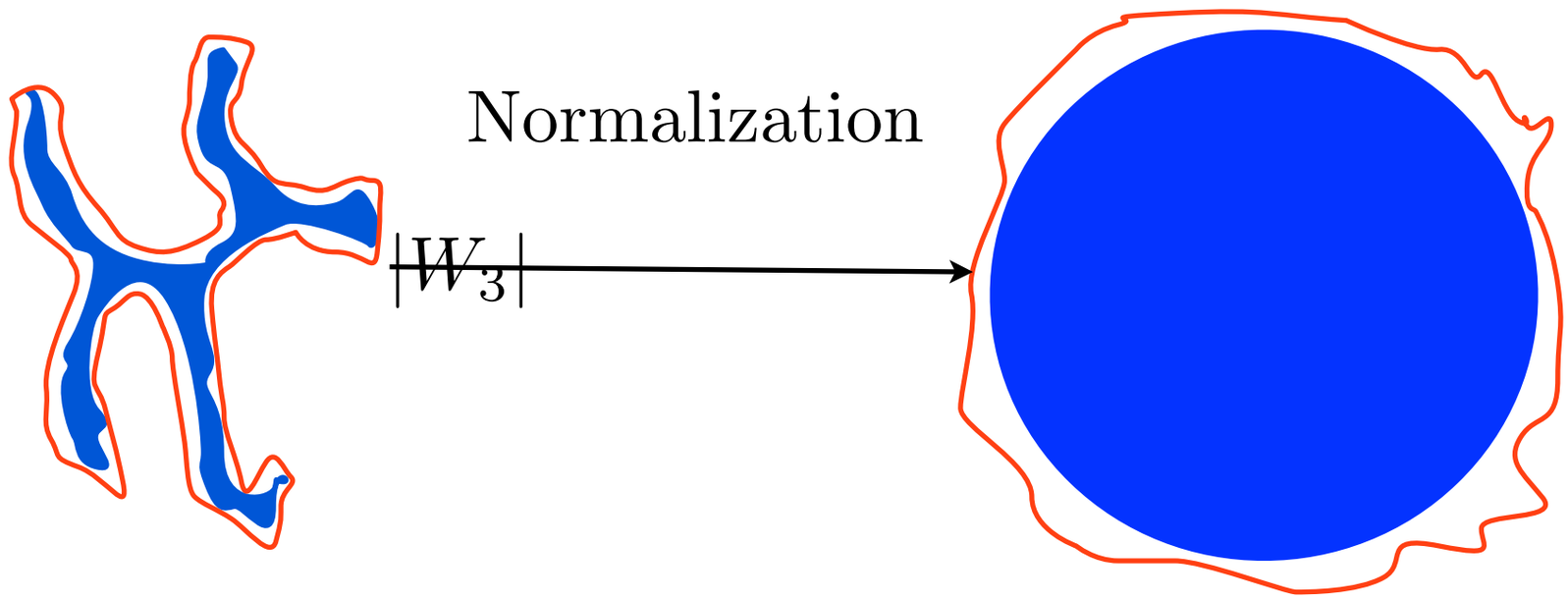
- Deep scattering perform adaptive space contractions
- Squeeze the space while minimizing the data volume reduction



$|W_3|$

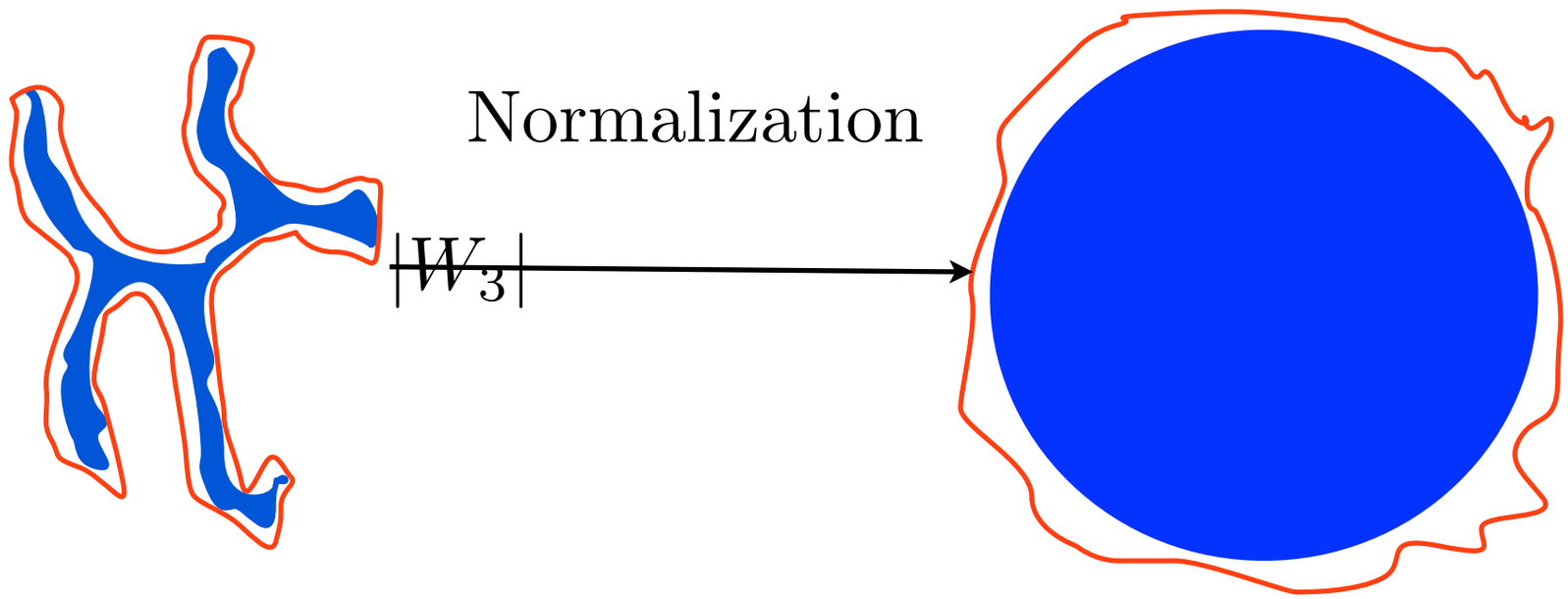
# Unsupervised Learning by Scattering

- Deep scattering perform adaptive space contractions
- Squeeze the space while minimizing the data volume reduction



# Unsupervised Learning by Scattering

- Deep scattering perform adaptive space contractions
- Squeeze the space while minimizing the data volume reduction

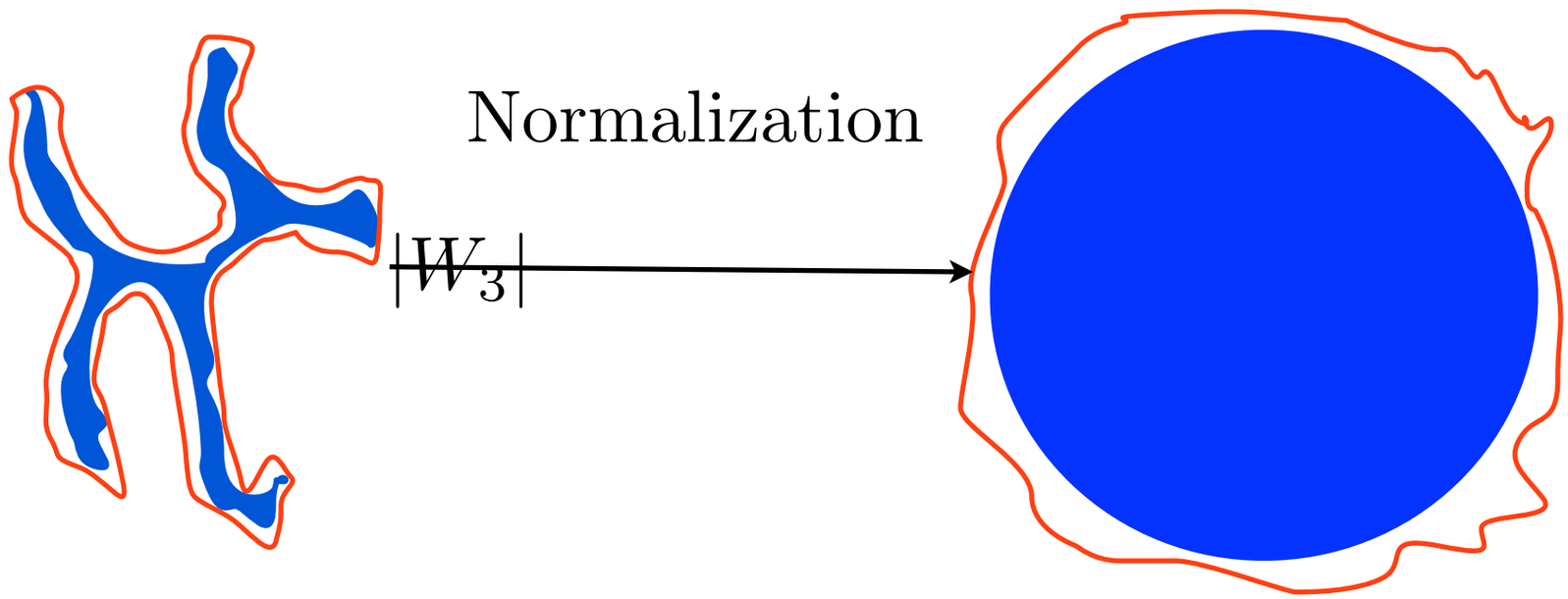



**Proposition:** The data volume reduction at layer $m$ is

$$E(\|X_{m-1} - E(X_{m-1})\|^2) - E(\|X_m - E(X_m)\|^2) = \|E(X_m)\|^2$$

# Unsupervised Learning by Scattering

- Deep scattering perform adaptive space contractions
- Squeeze the space while minimizing the data volume reduction



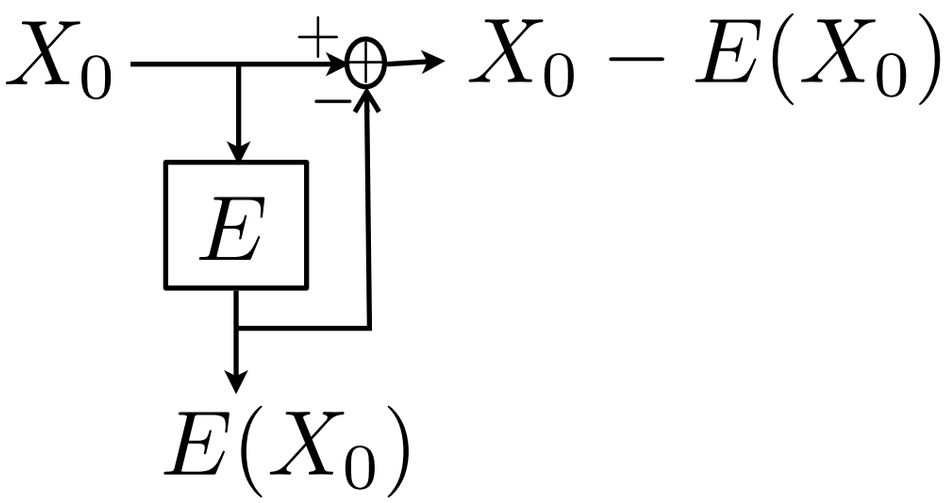

**Proposition:** The data volume reduction at layer $m$ is

$$E(\|X_{m-1} - E(X_{m-1})\|^2) - E(\|X_m - E(X_m)\|^2) = \|E(X_m)\|^2$$

$$\Rightarrow \text{ for all } m \text{ minimize } \|E(X_m)\| \ .$$

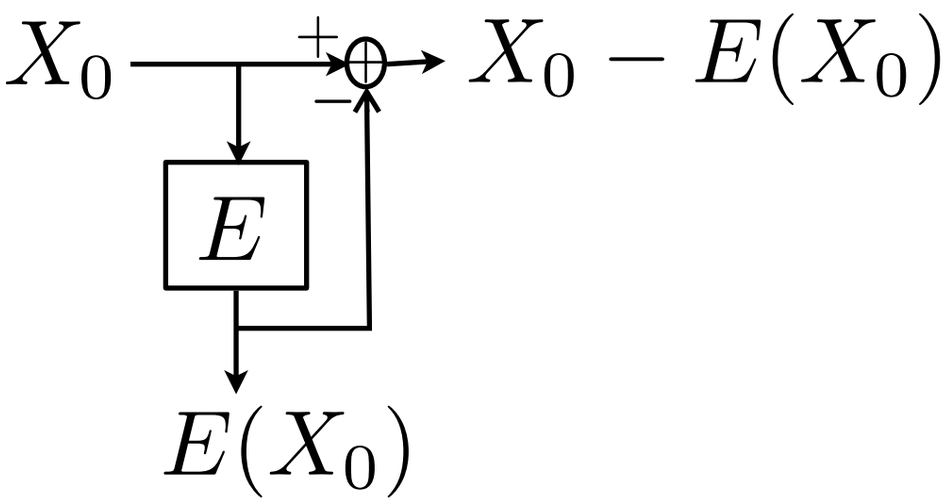
# Sparse Layerwise Learning

$$X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with } W_m^* W_m = Id.$$



# Sparse Layerwise Learning

$$X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with } W_m^* W_m = Id.$$

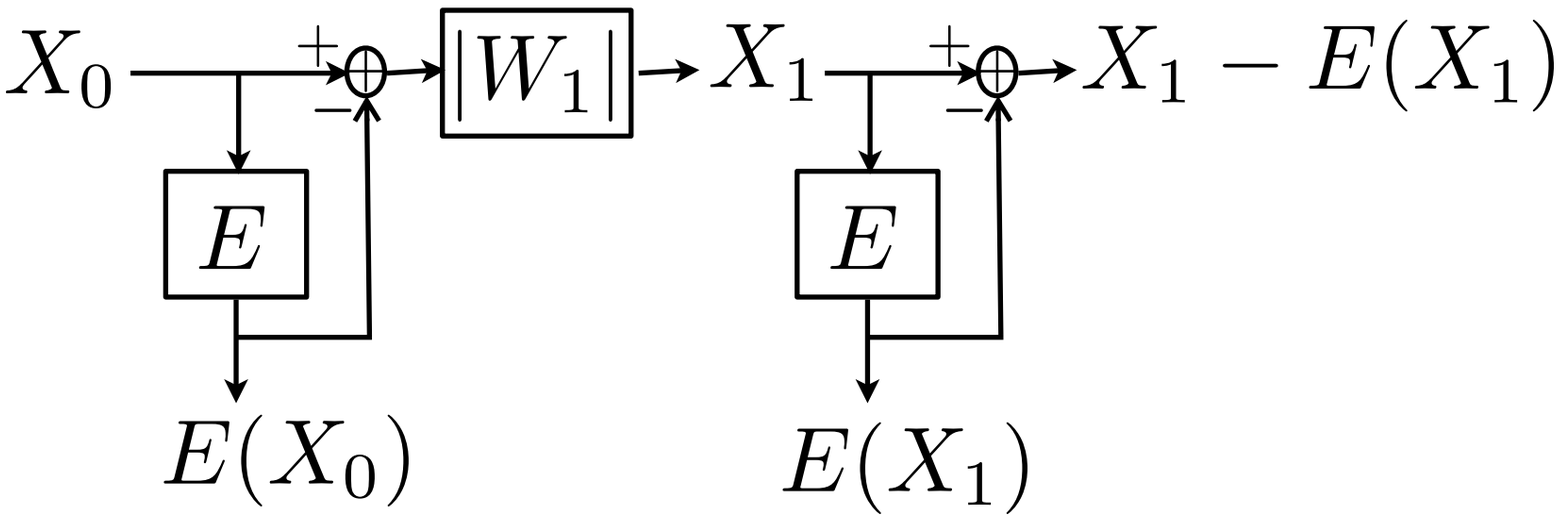



- Given $X_{m-1} - E(X_{m-1})$ we compute $W_m$ by minimizing

$$\|E(X_m)\| = \Big\| \underbrace{E\Big(|W_m(X_{m-1} - E(X_{m-1})|\Big)}_{\mathbf{l^1} \text{ norm across realizations}} \Big\|$$

# Sparse Layerwise Learning

$$X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with } W_m^* W_m = Id.$$

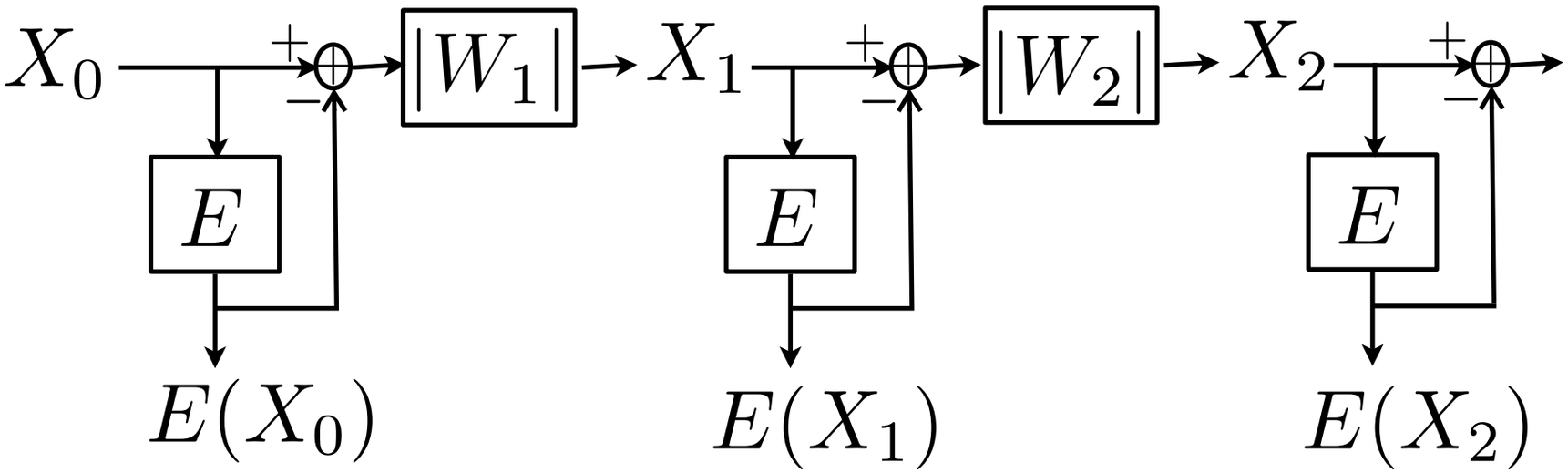



- Given $X_{m-1} - E(X_{m-1})$ we compute $W_m$ by minimizing

$$\|E(X_m)\| = \Big\| \underbrace{E\Big(|W_m(X_{m-1} - E(X_{m-1})|\Big)}_{\mathbf{l^1} \text{ norm across realizations}} \Big\|$$

# Sparse Layerwise Learning

$$X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with } W_m^* W_m = Id.$$

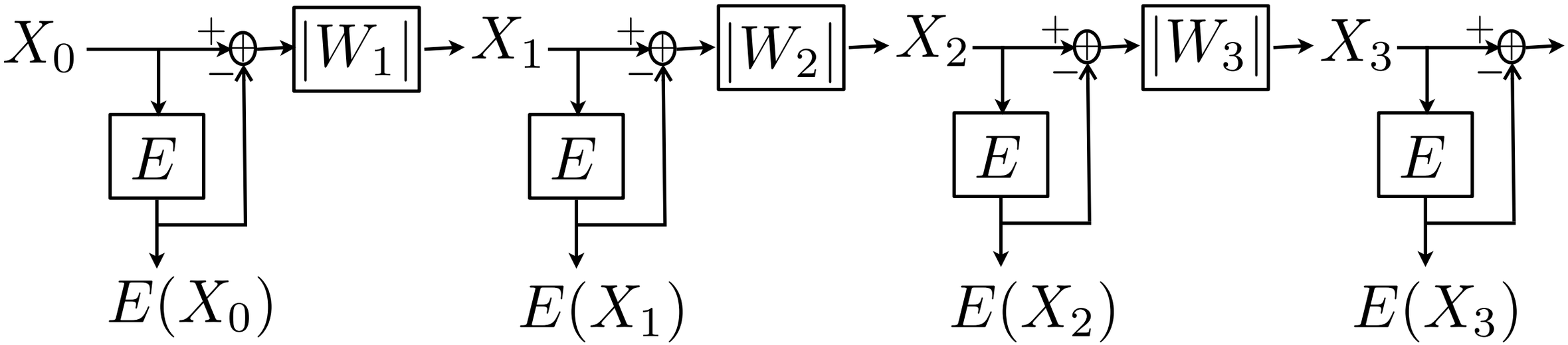



- Given $X_{m-1} - E(X_{m-1})$ we compute $W_m$ by minimizing

$$\|E(X_m)\| = \Big\| \underbrace{E\Big(|W_m(X_{m-1} - E(X_{m-1})|\Big)}_{\mathbf{l^1} \text{ norm across realizations}} \Big\|$$

# Sparse Layerwise Learning

$$X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with } W_m^* W_m = Id.$$

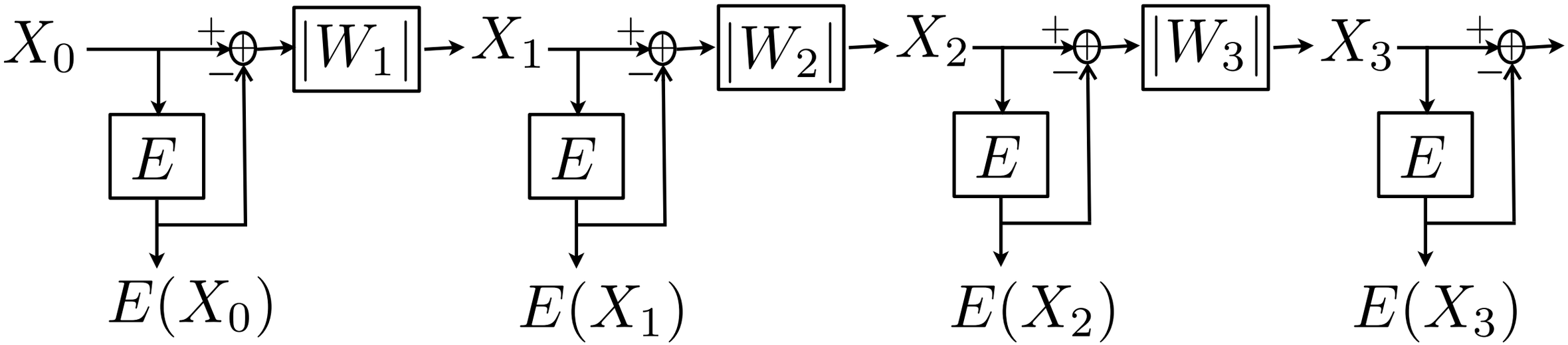



- Given $X_{m-1} - E(X_{m-1})$ we compute $W_m$ by minimizing

$$\|E(X_m)\| = \Big\| \underbrace{E\Big(|W_m(X_{m-1} - E(X_{m-1})|\Big)}_{\mathbf{l^1} \text{ norm across realizations}} \Big\|$$

# Sparse Layerwise Learning

$$X_m = |W_m(X_{m-1} - E(X_{m-1})| \quad \text{with } W_m^* W_m = Id.$$

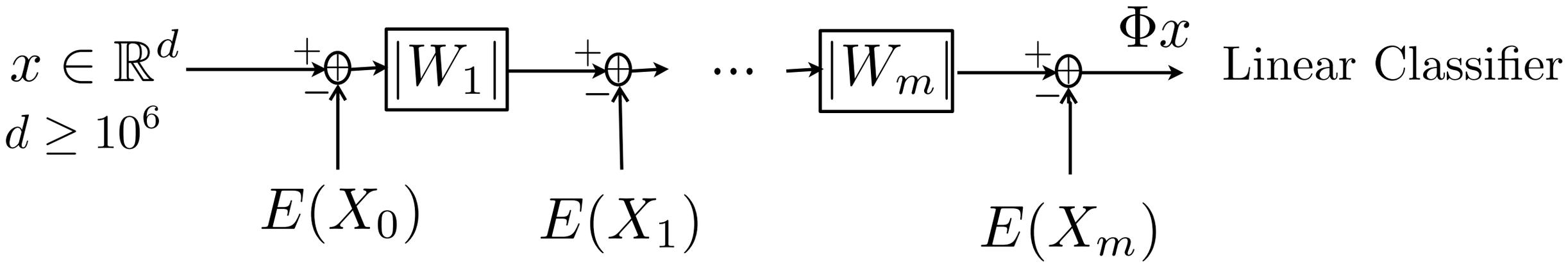



- Given $X_{m-1} - E(X_{m-1})$ we compute $W_m$ by minimizing

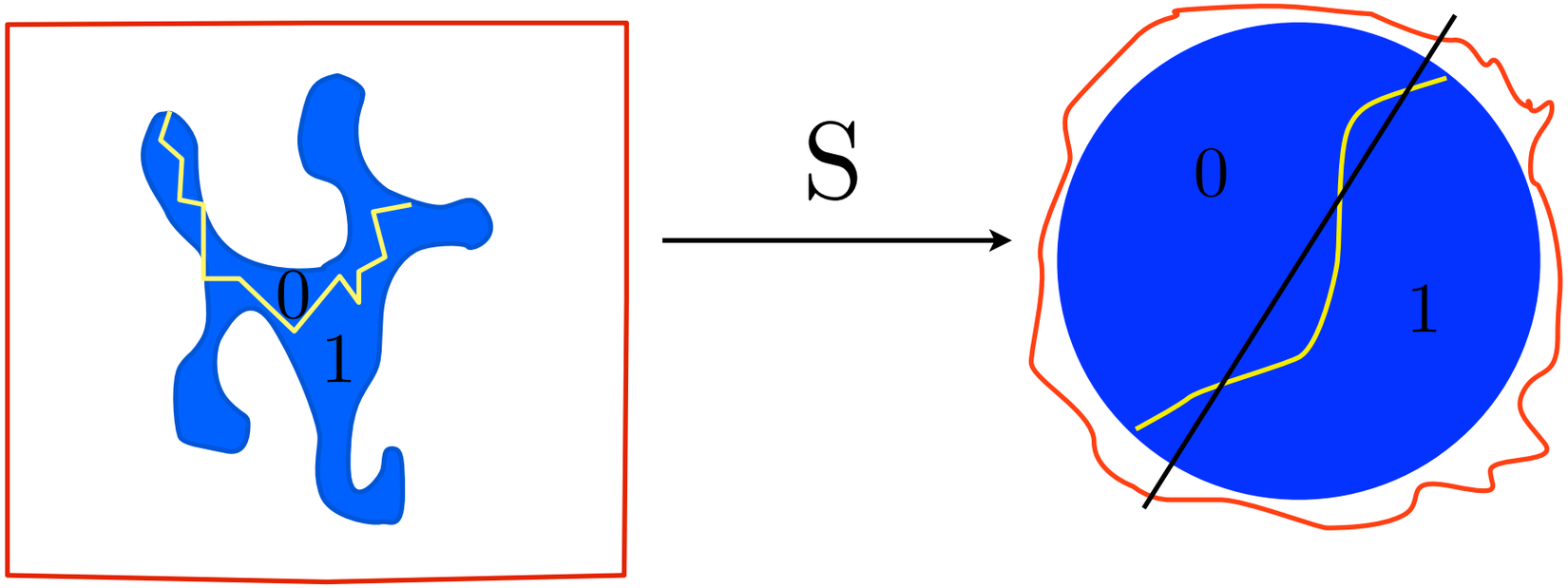
$$\|E(X_m)\| = \Big\| \underbrace{E\Big(|W_m(X_{m-1} - E(X_{m-1})|\Big)}_{\mathbf{l^1} \text{ norm across realizations}} \Big\|$$

$\Rightarrow W_m$ defines a sparse representation of $X_{m-1} - E(X_{m-1})$

# Supervised Linear Classifiers

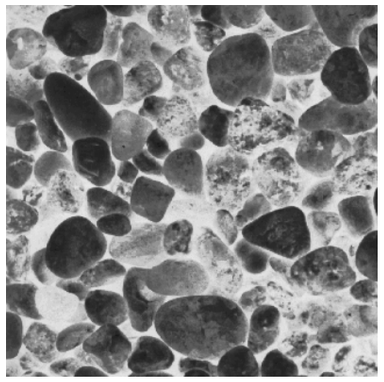



Binary classification
$y(x) = 0$ or $1$



# Problem: Clutter

- Average pooling is ok without clutter:

- Clutter requires detection: max pooling



# Conclusion

- Wavelets are good for deformations and because of sparsity
- Scattering defines contractive deep neural networks which can be analyzed mathematically
- Unsupervised learning can be optimized with sparse contractions
- What about **max** and clutter ?
- Analysis of non-linear PDE : turbulence and Navier-Stokes
- Papers and softwares: www.di.ens.fr/scattering