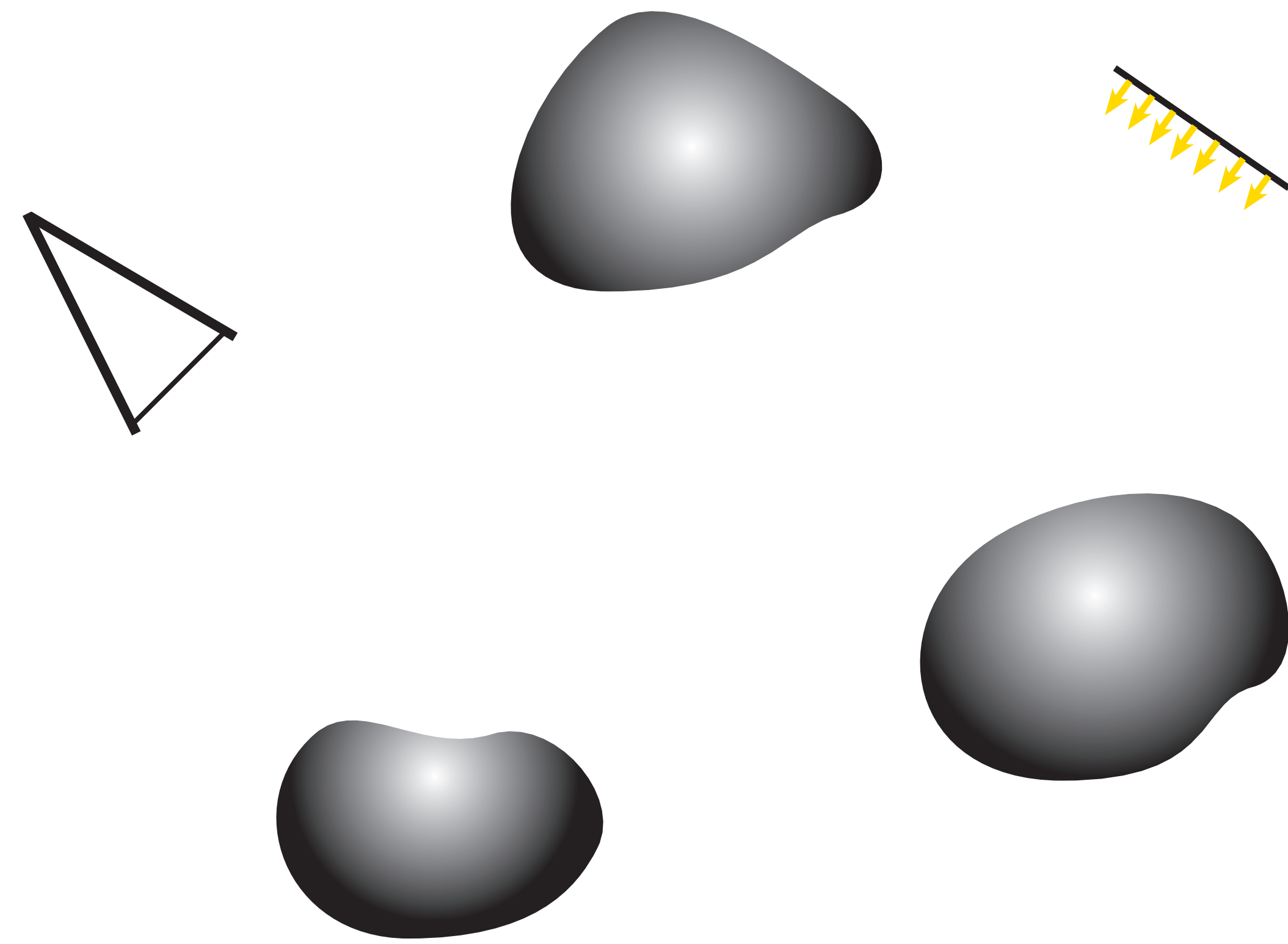


How many rays do we need?

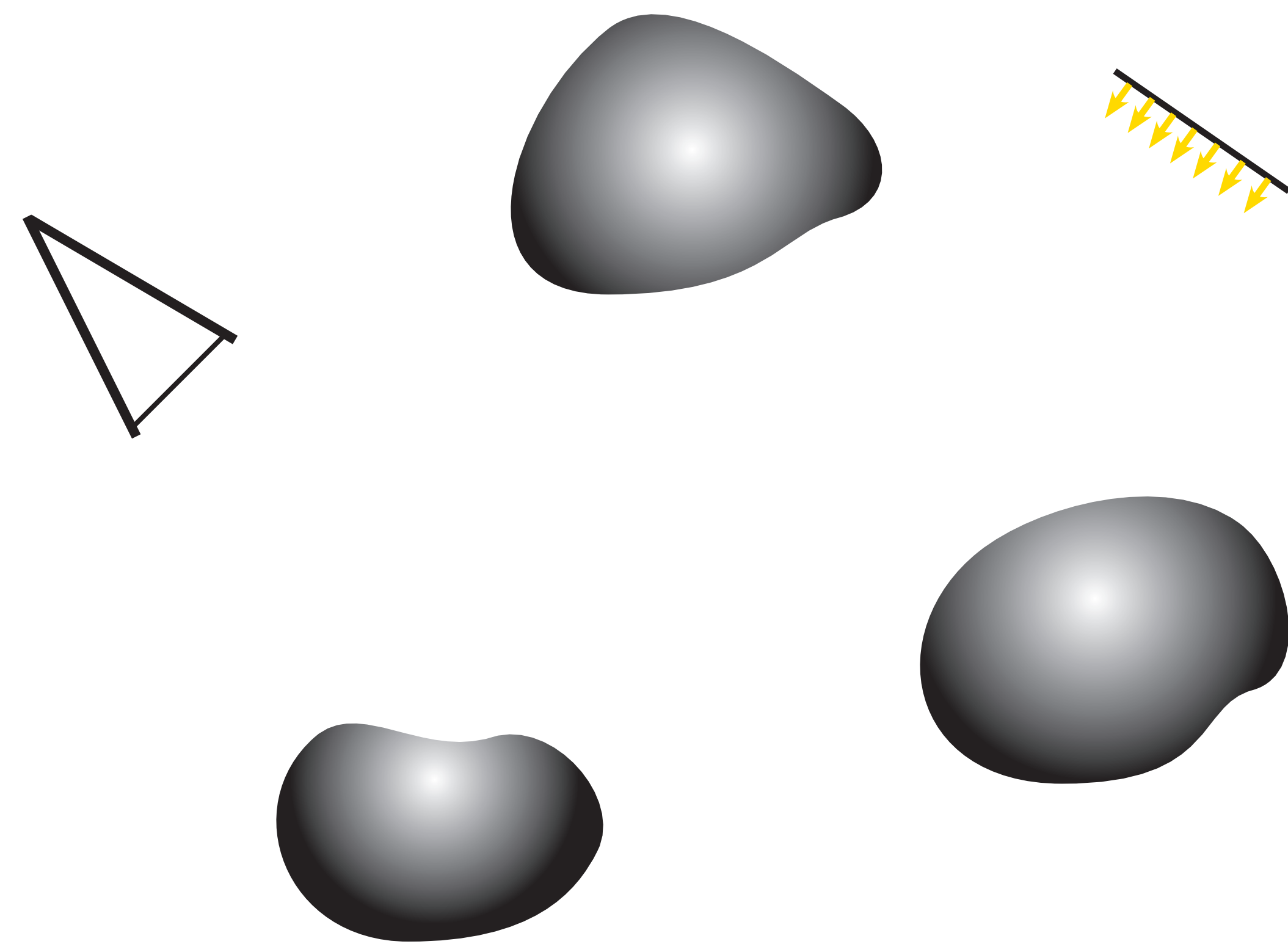
Christian Lessig

Otto-von-Guericke-Universität Magdeburg

Objective

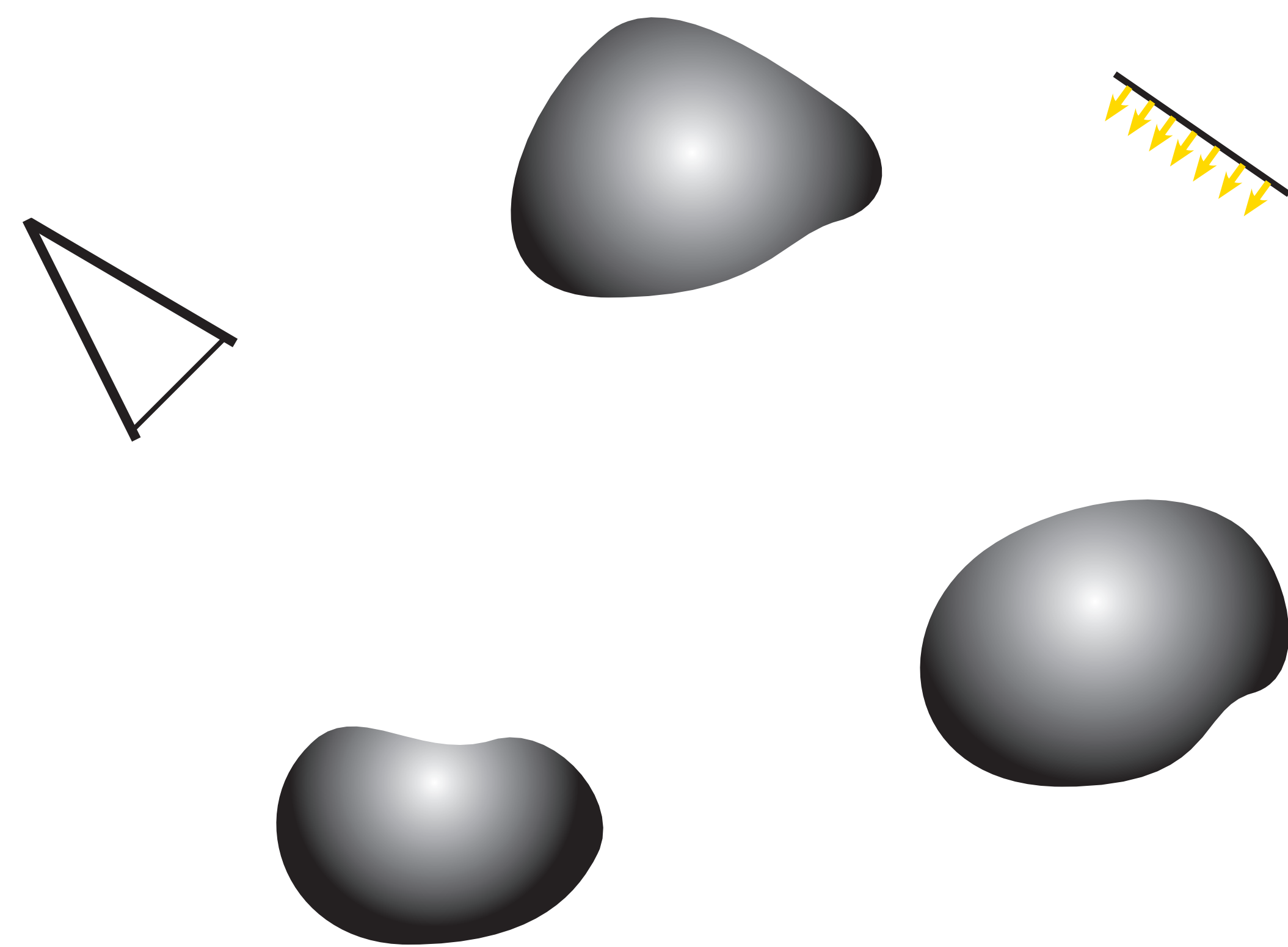


Objective



Rectilinear transport in
known scene

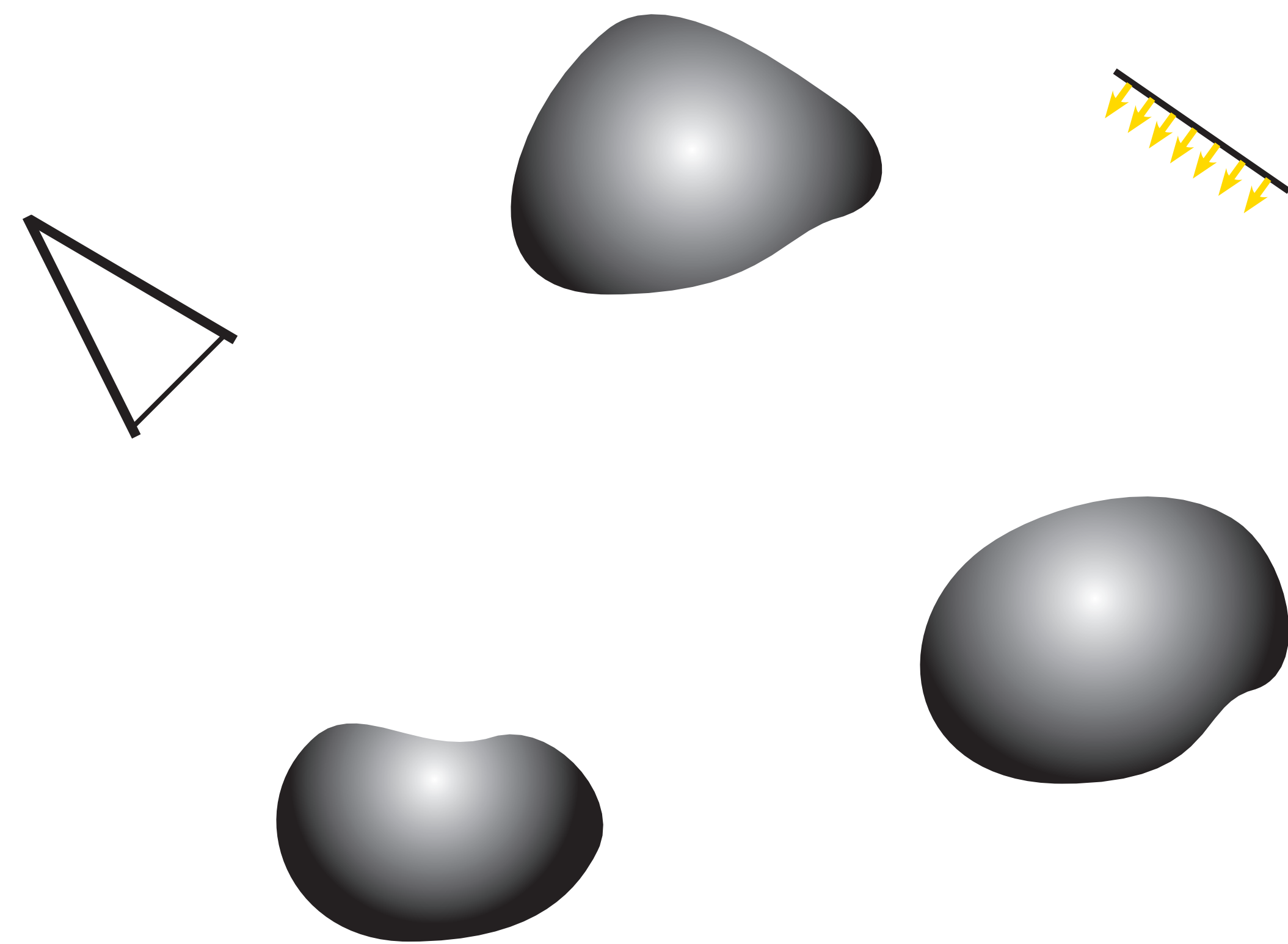
Objective



Rectilinear transport in
known scene



Objective



Rectilinear transport in
known scene

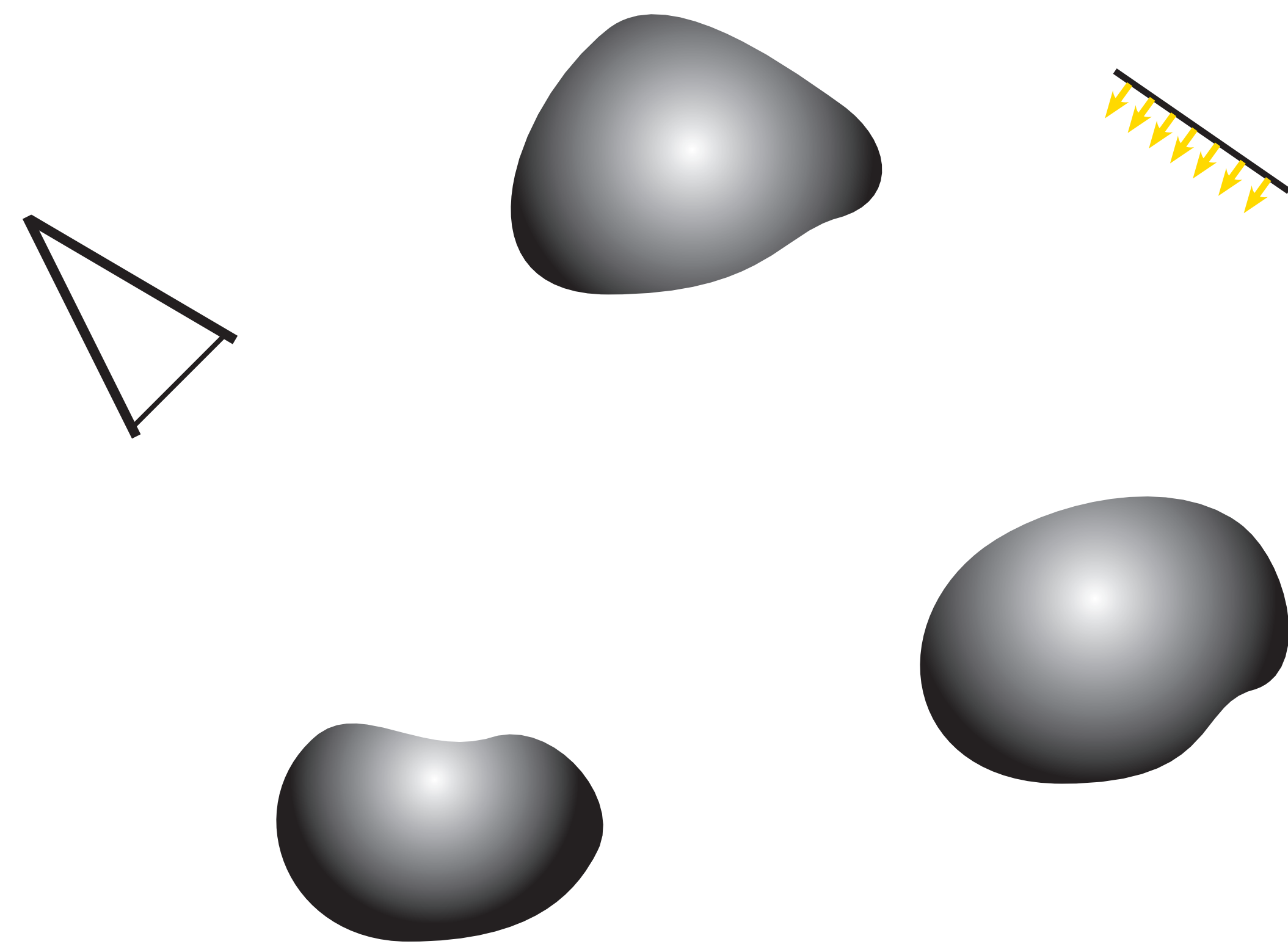


$$\hat{f} \in L_2([0, 1]^2)$$



$$f \in \mathcal{H} \subset L_2, \dim(\mathcal{H}) < \infty$$

Objective



Rectilinear transport in
known scene

$$\xrightarrow{G}$$

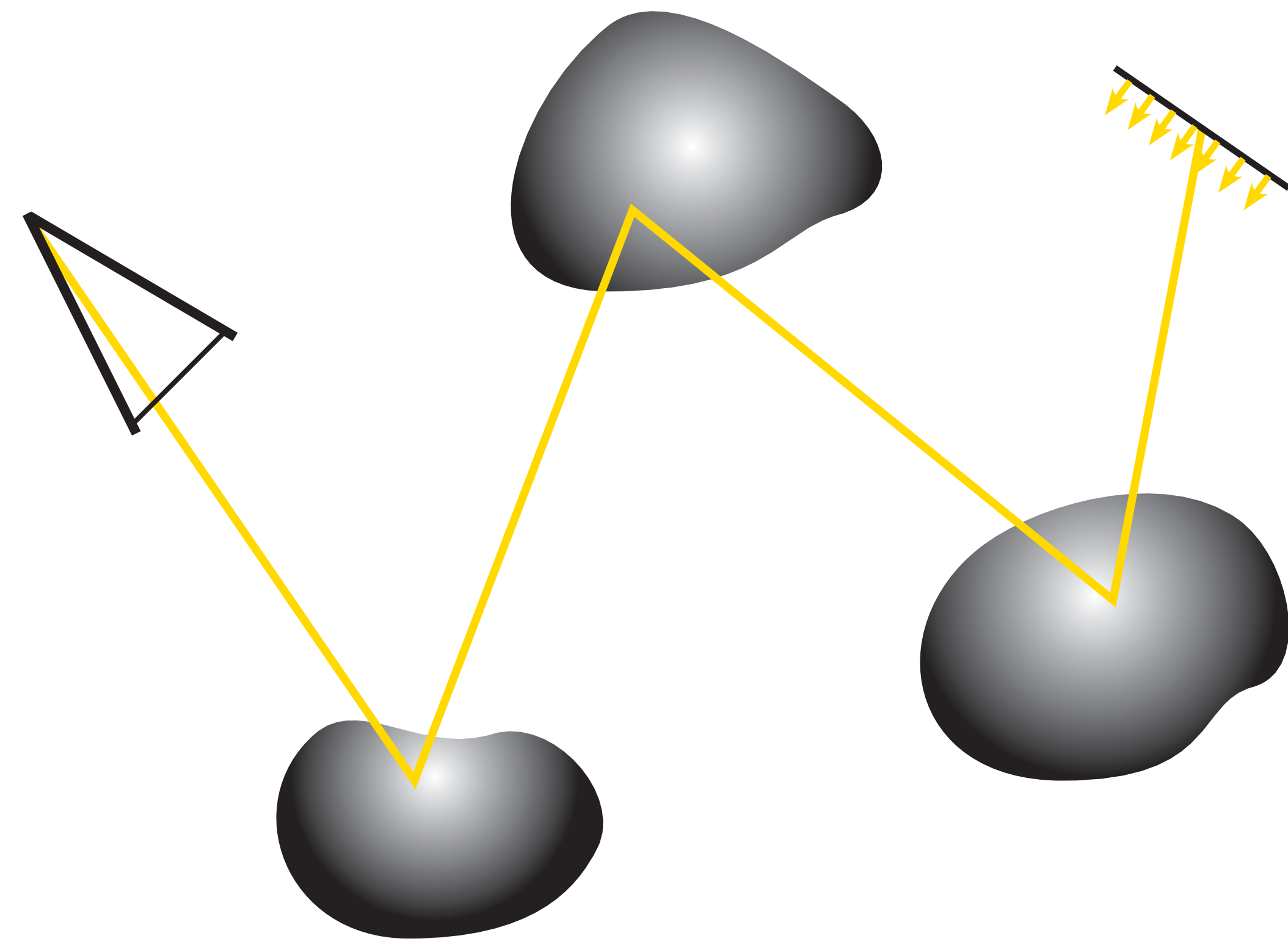
(close-to-) optimal for
 $\|\hat{f} - f\| < \epsilon$

$$\hat{f} \in L_2([0, 1]^2)$$



$$f \in \mathcal{H} \subset L_2, \dim(\mathcal{H}) < \infty$$

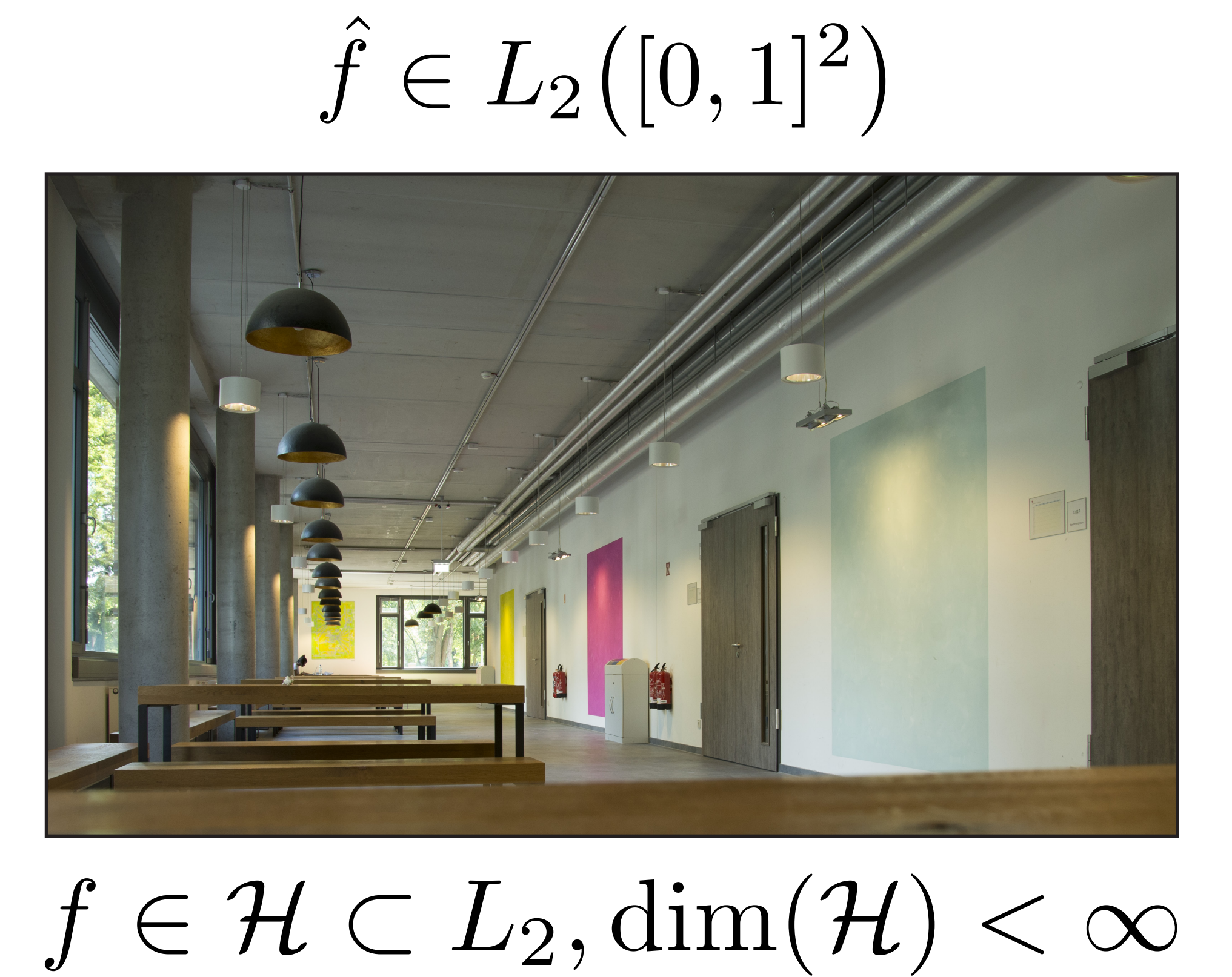
Objective



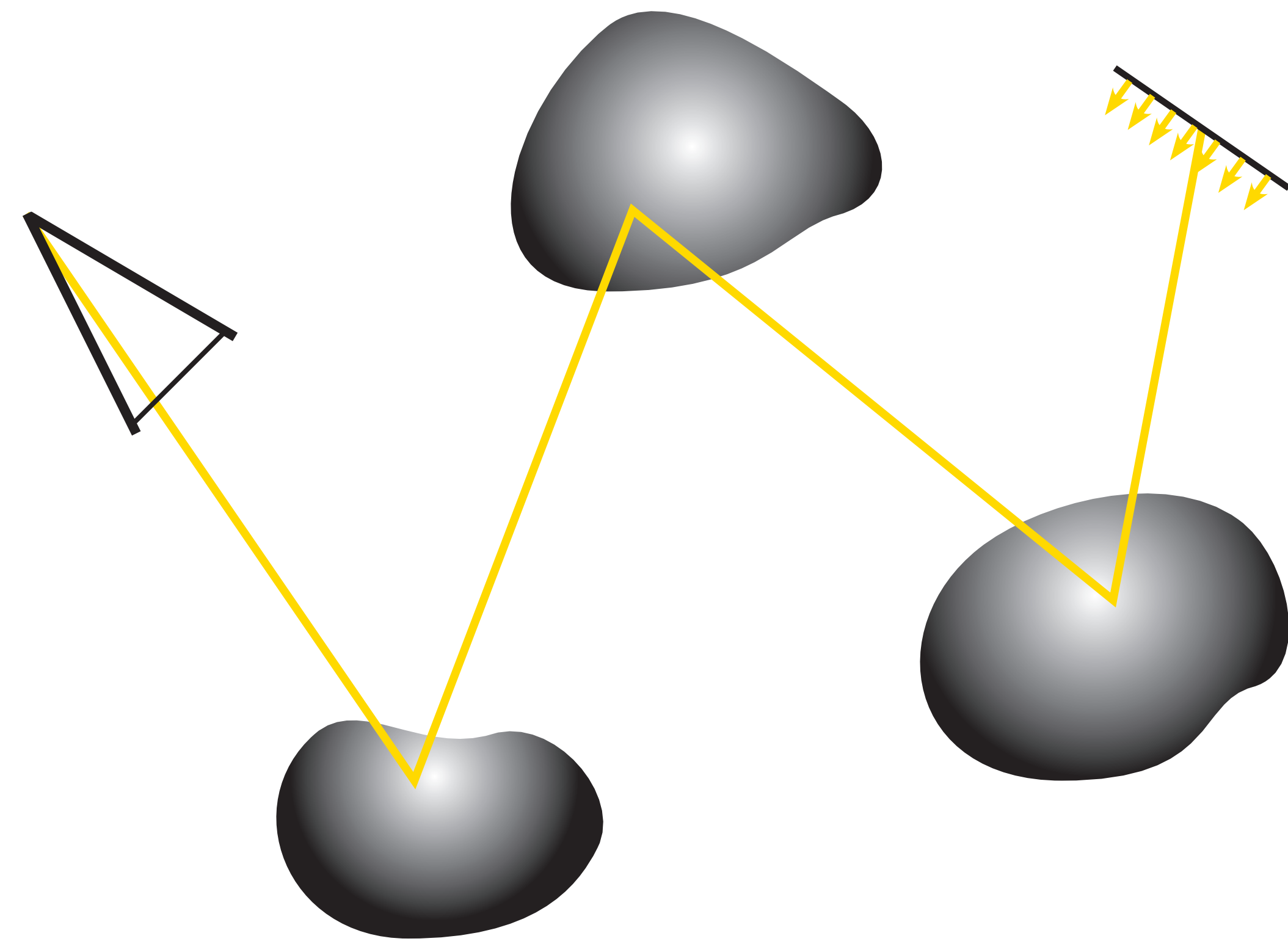
Rectilinear transport in
known scene

G

minimal # of rays for
 $\|\hat{f} - f\| < \epsilon$



Objective



Rectilinear transport in
known scene

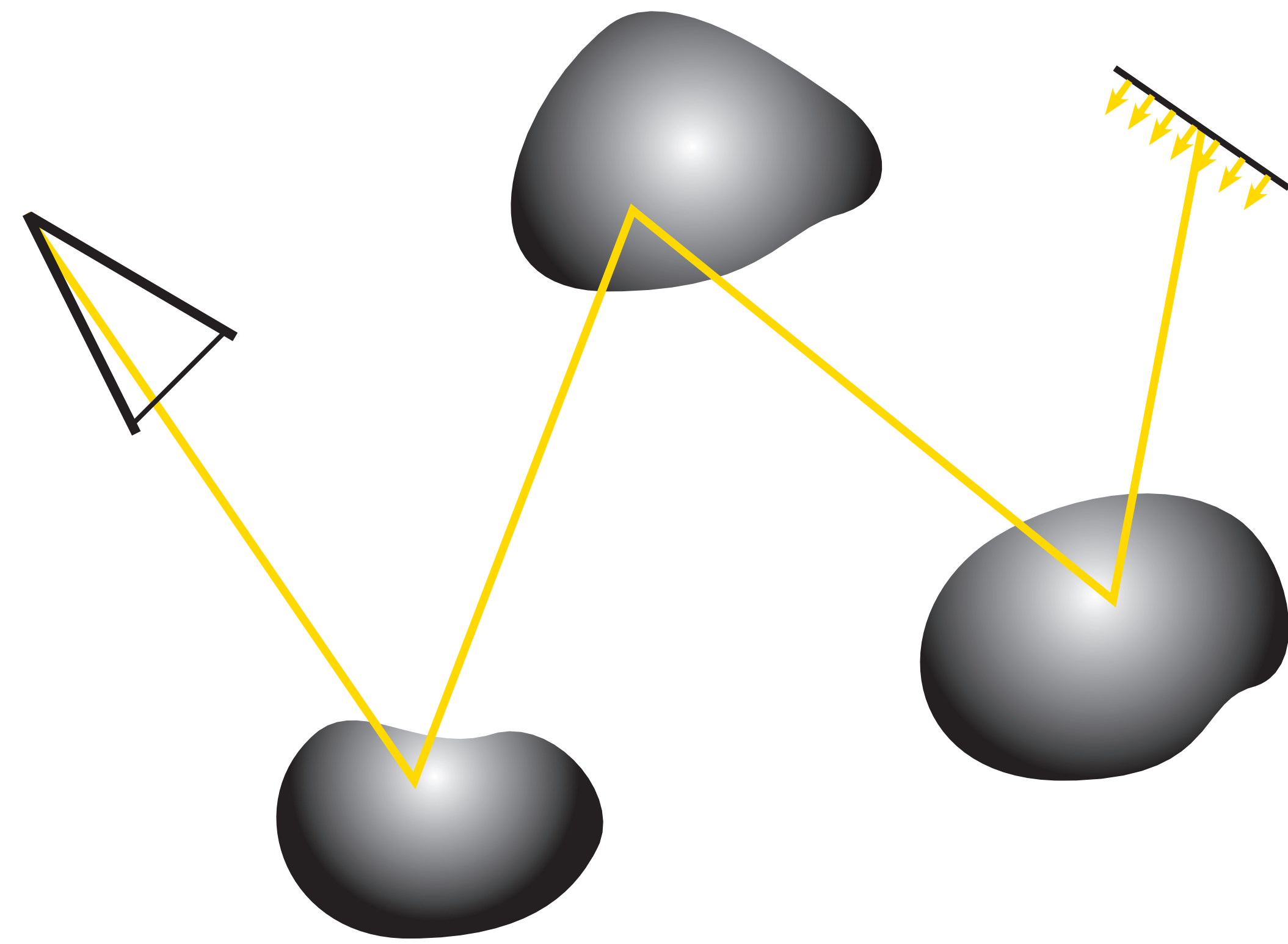
$$\xrightarrow[\text{minimal \# of rays for}]{G} \|\hat{f} - f\| \lesssim N^{-\alpha}$$

$$\hat{f} \in L_2([0, 1]^2)$$



$$f \in \mathcal{H} \subset L_2, \dim(\mathcal{H}) < \infty$$

Objective



Rectilinear transport in
known scene

G

minimal # of rays for
 $\|\hat{f} - f\| \lesssim N^{-\alpha}$

- Superresolution
- Progressive
- Varying resolution
- Parallelizable

$$\hat{f} \in L_2([0, 1]^2)$$



$$f \in \mathcal{H} \subset L_2, \dim(\mathcal{H}) < \infty$$

Computational costs

$$C = N \cdot C_p$$

/ \

Number of Cost per

pixels pixel

Computational costs

$$C = N \cdot C_p$$

/ \

Number of Cost per

pixels pixel

**last
bounce**

Number of “pixels”



Number of “pixels”



compression



Number of “pixels”



compression
 $\epsilon_{rel} = 6.44 \times 10^{-4}$



100%



8.48%

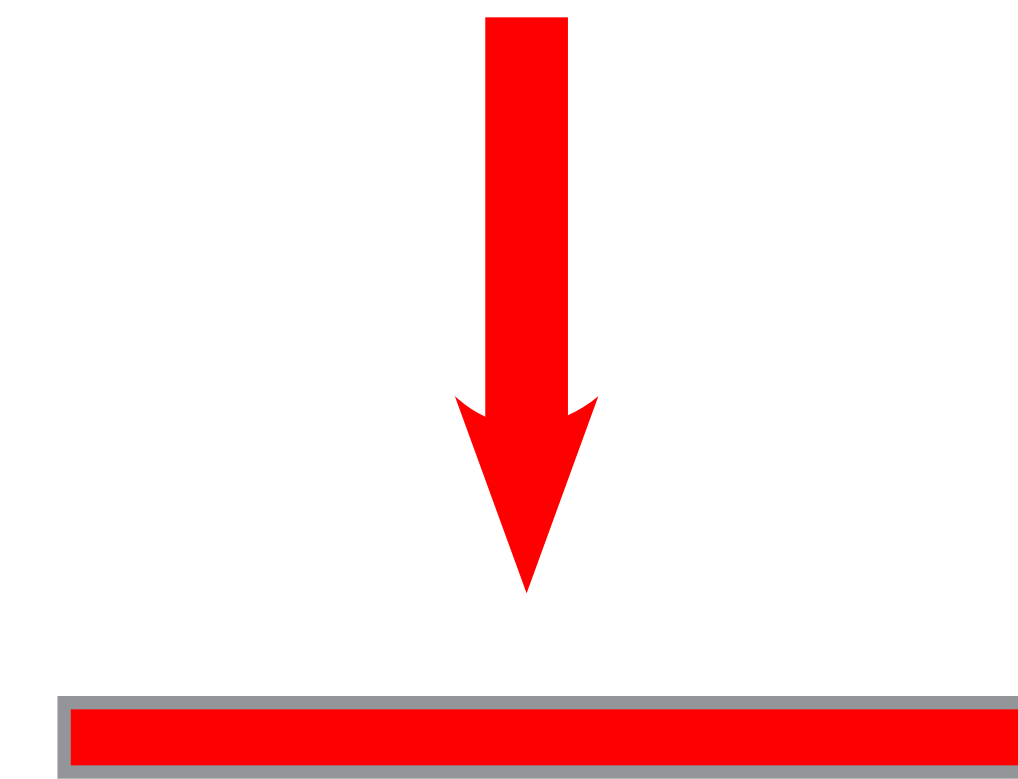
Number of “pixels”



compression



100%



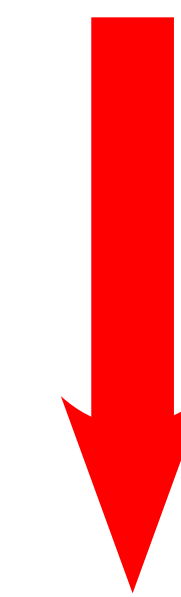
Number of “pixels”



compression



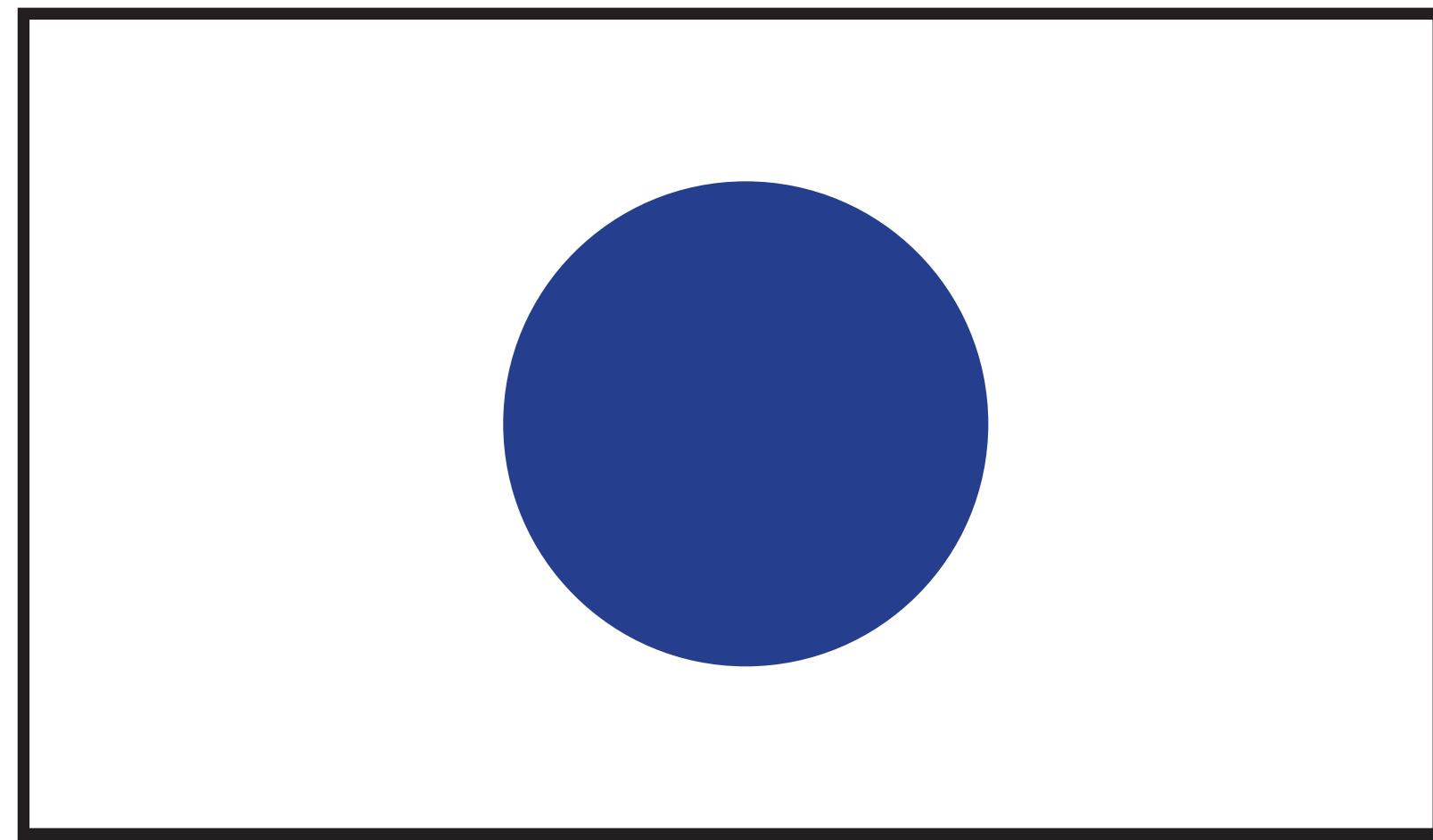
100%



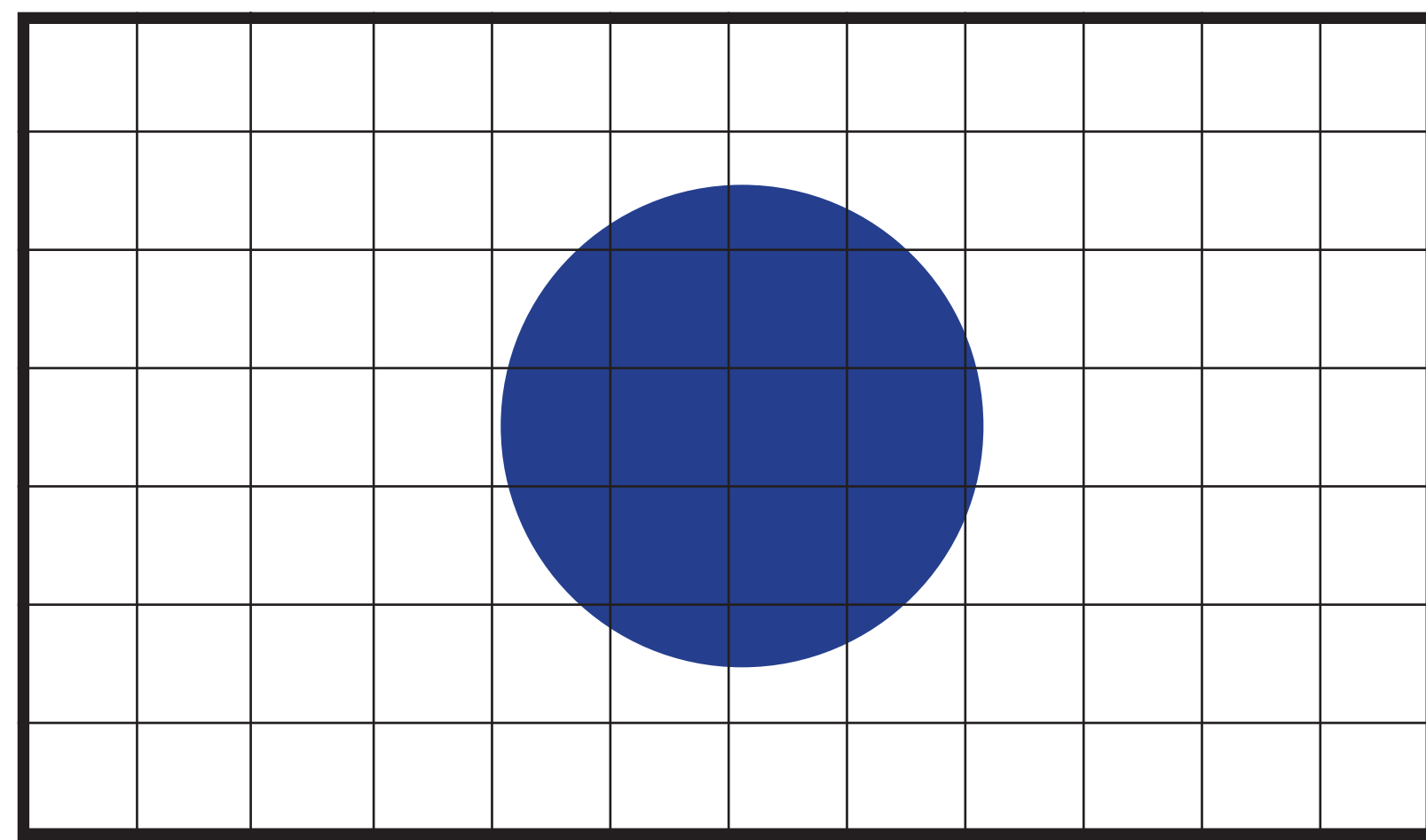
- wavelets (quasi)
- curvelets
- shearlets
- contourlets

$$\|\hat{f} - f\| \lesssim N^{-\alpha}$$

Number of “pixels”

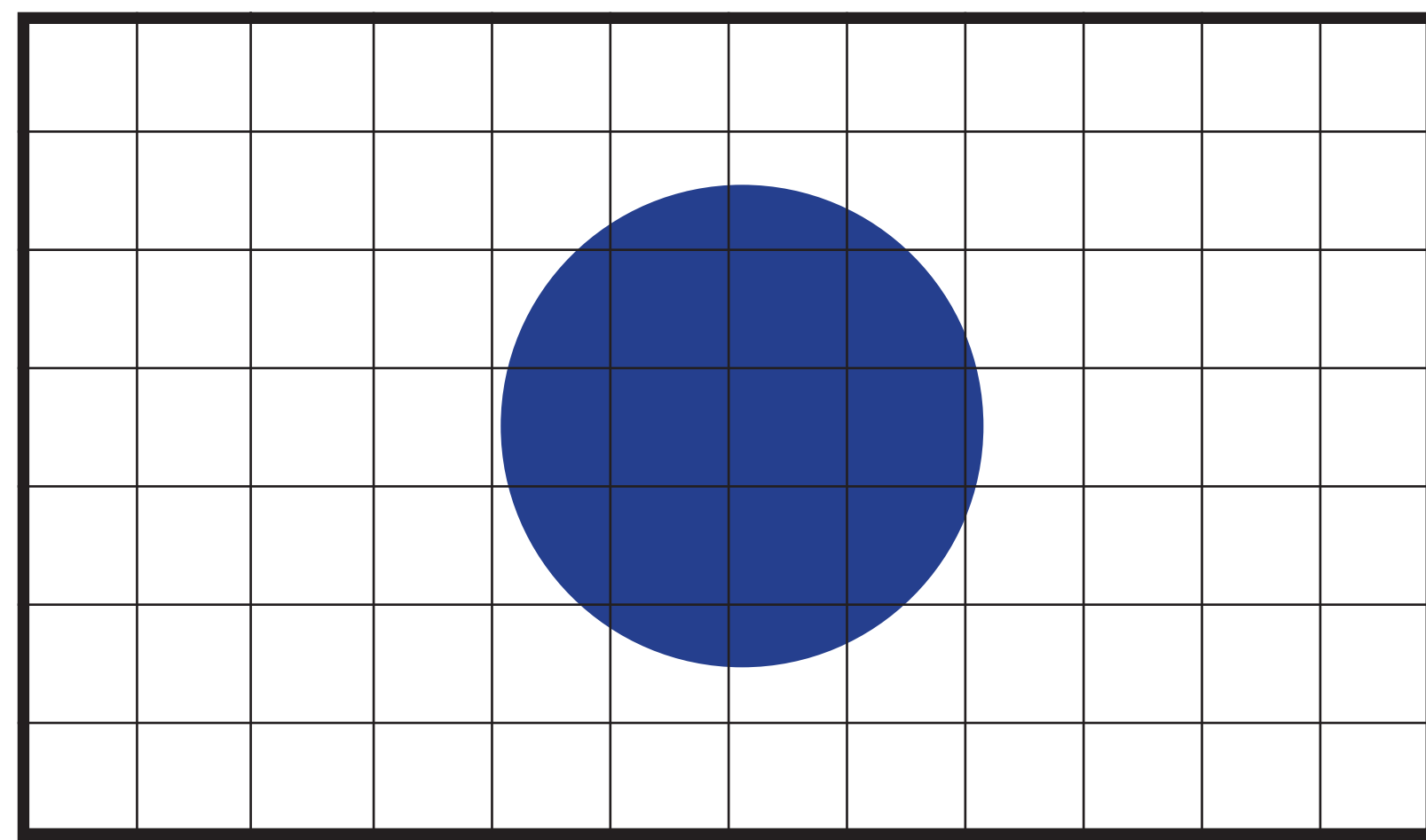


Number of “pixels”

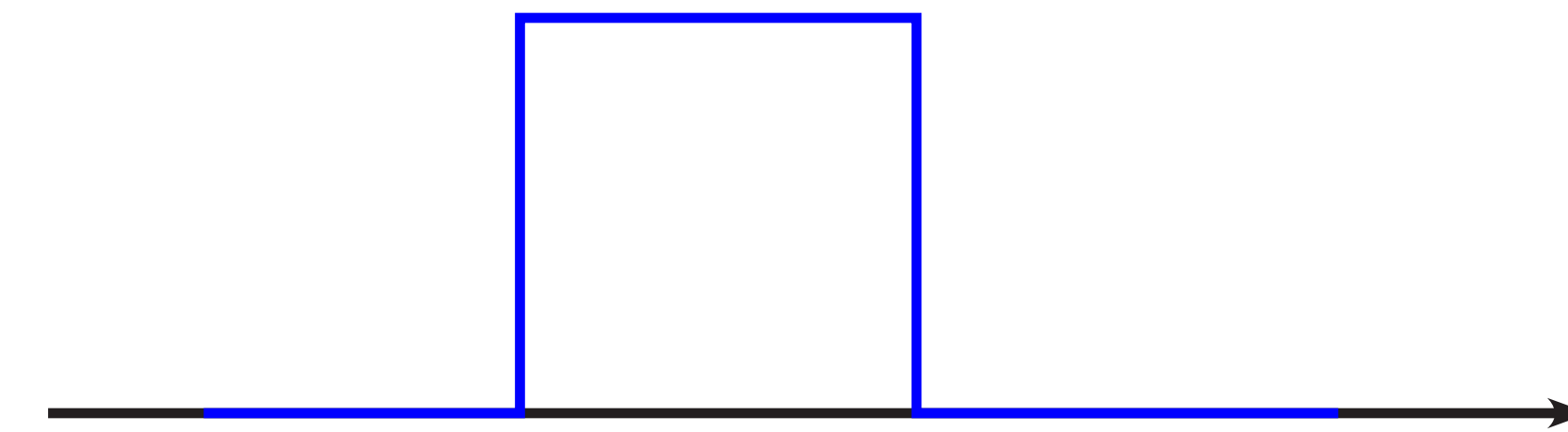


$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$

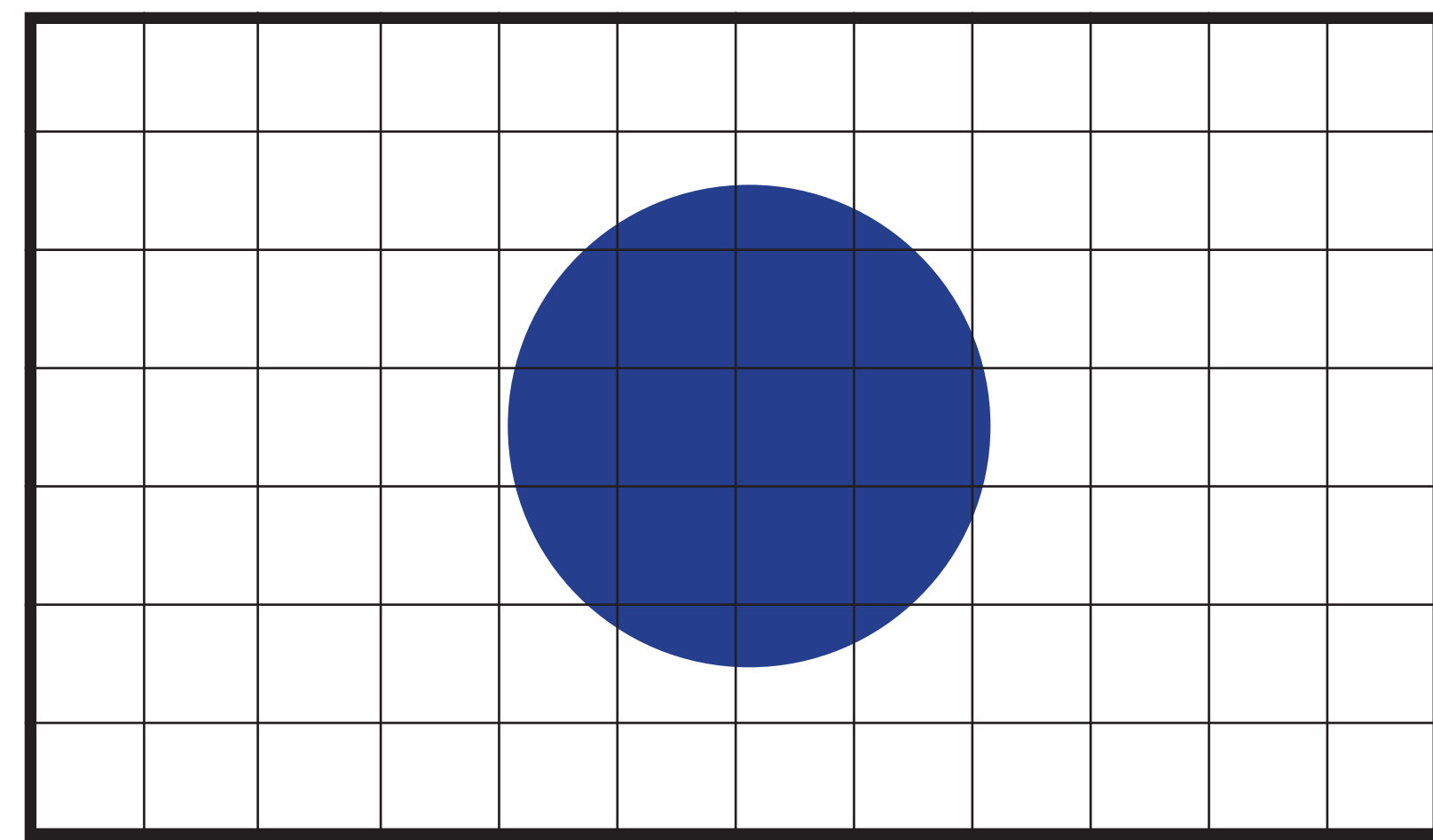
Number of “pixels”



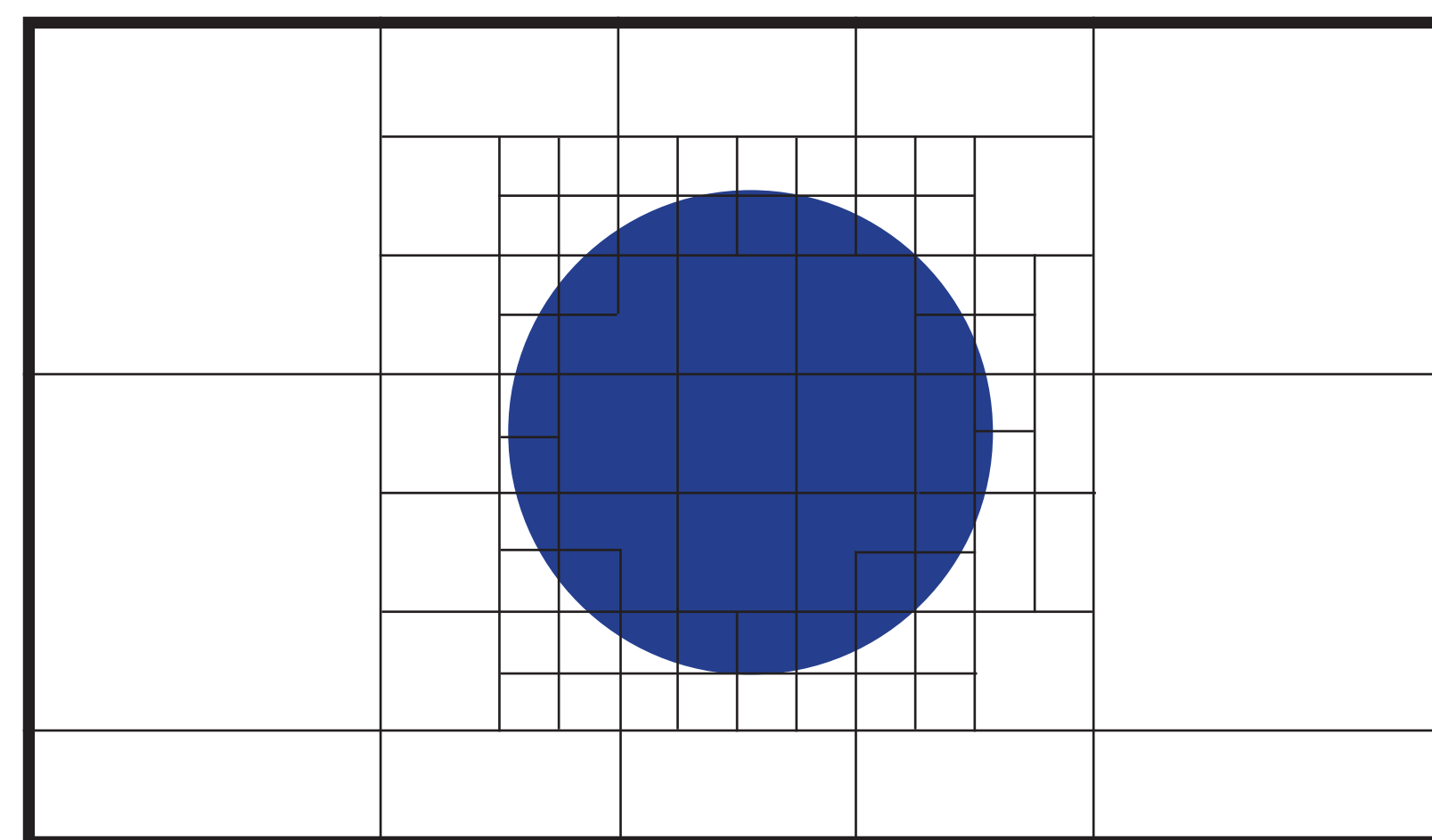
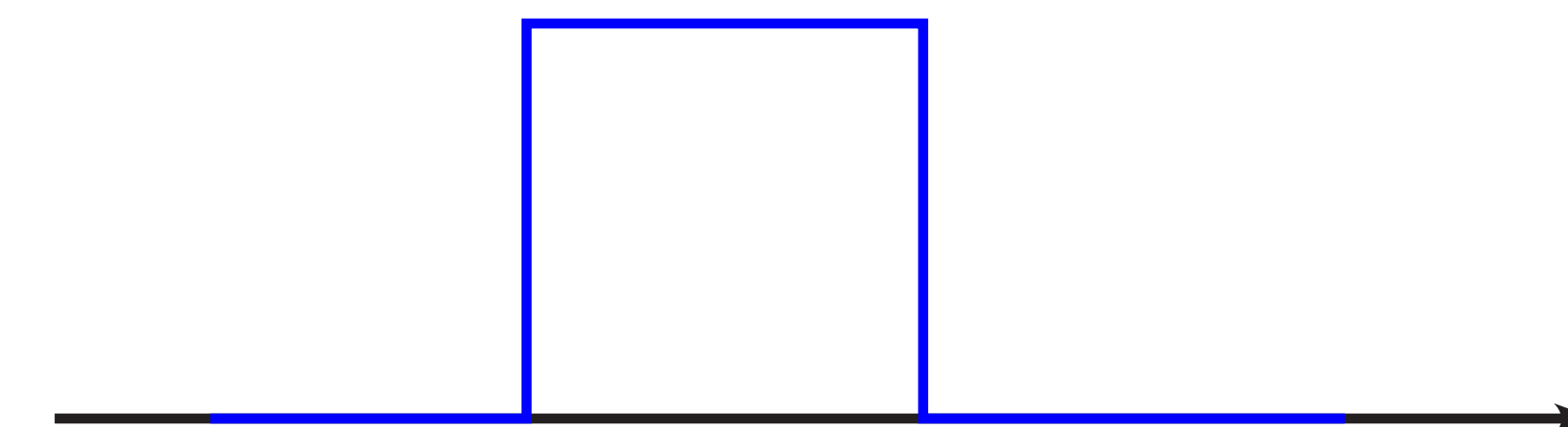
$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$



Number of “pixels”

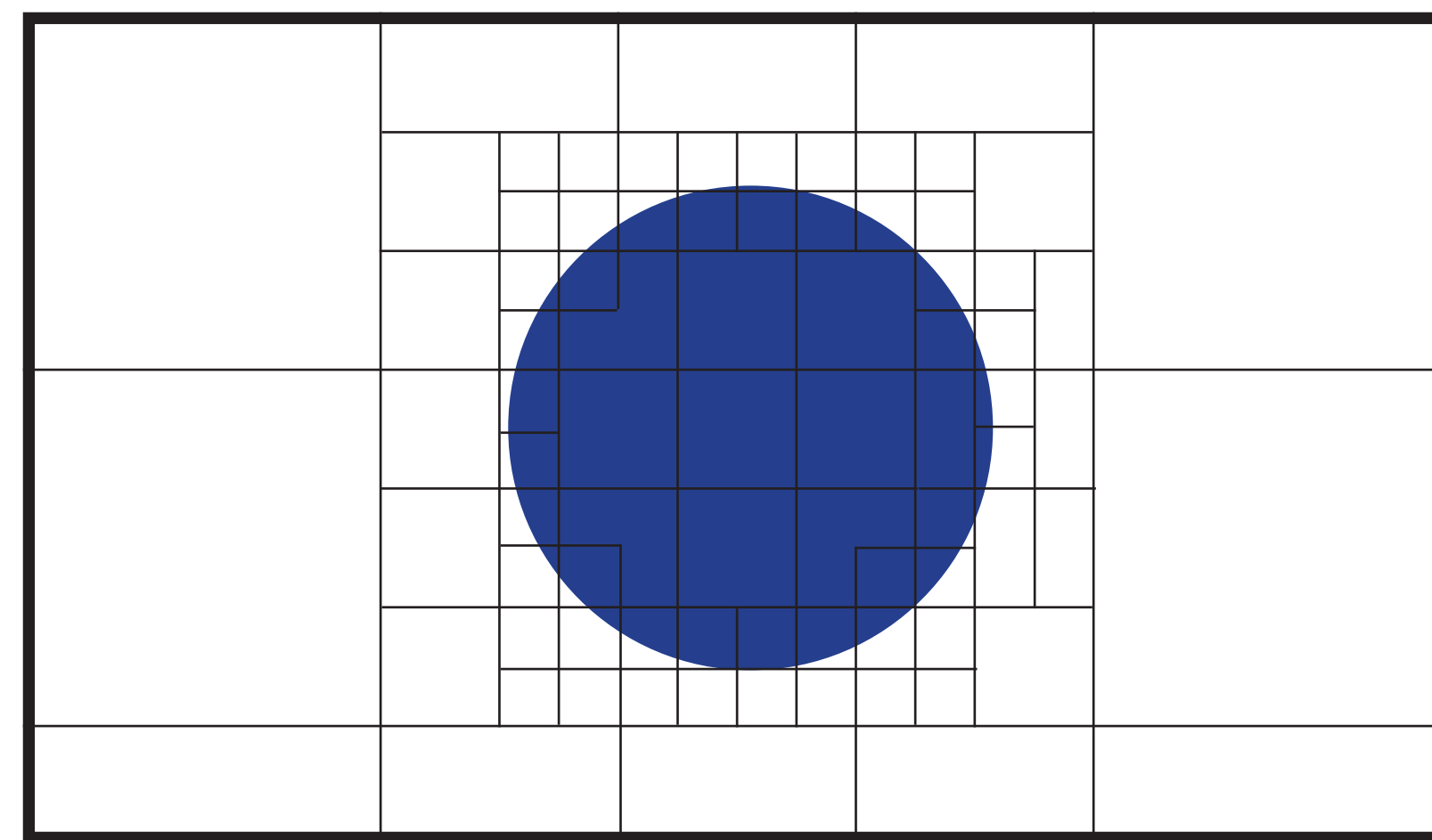
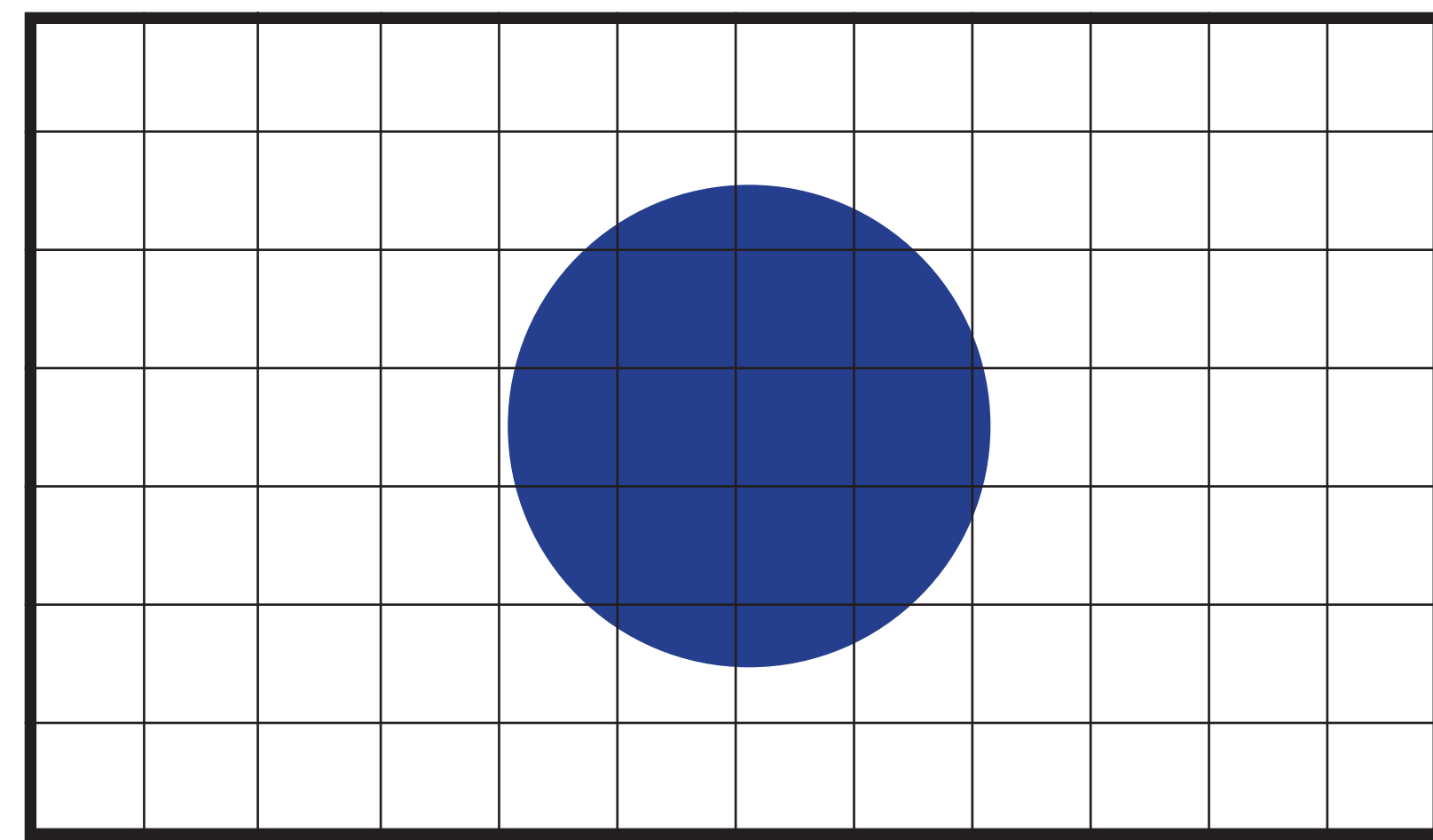


$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$

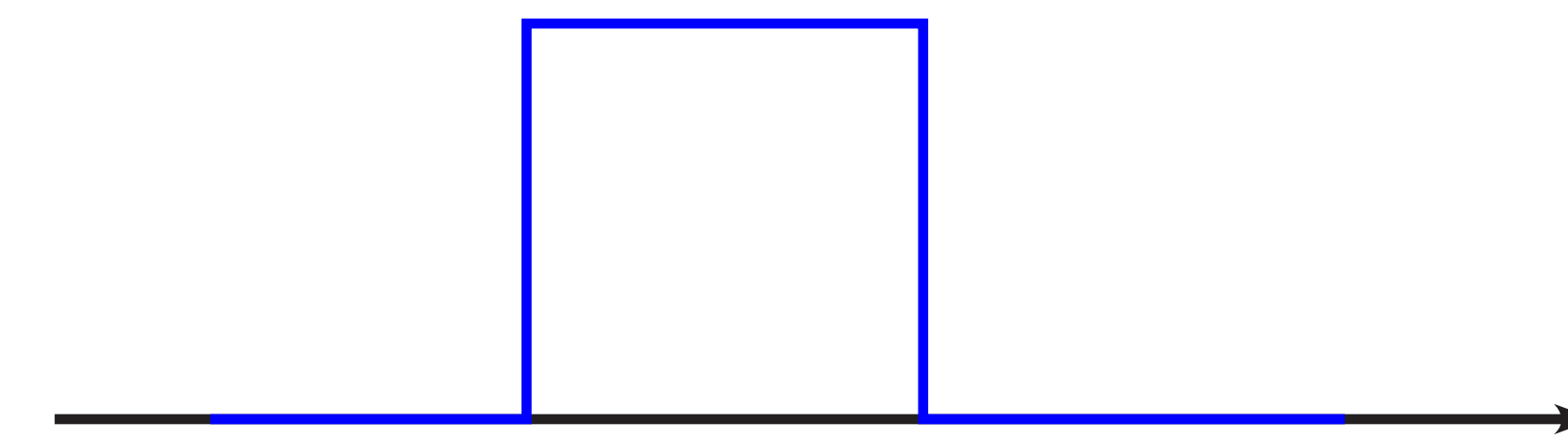


$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x)$$

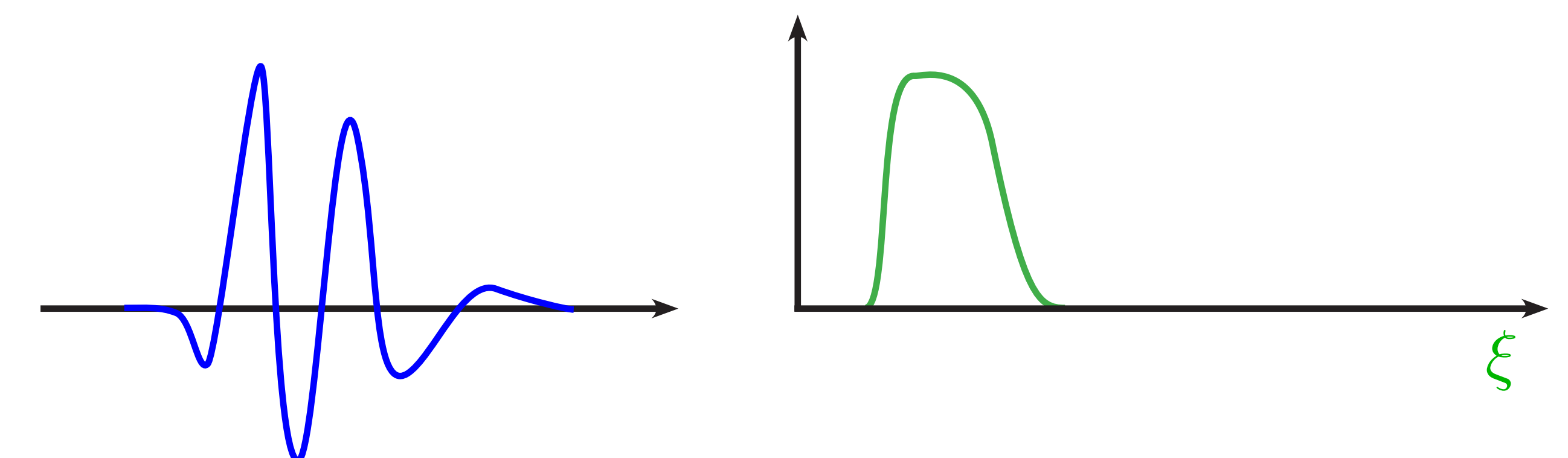
Number of “pixels”



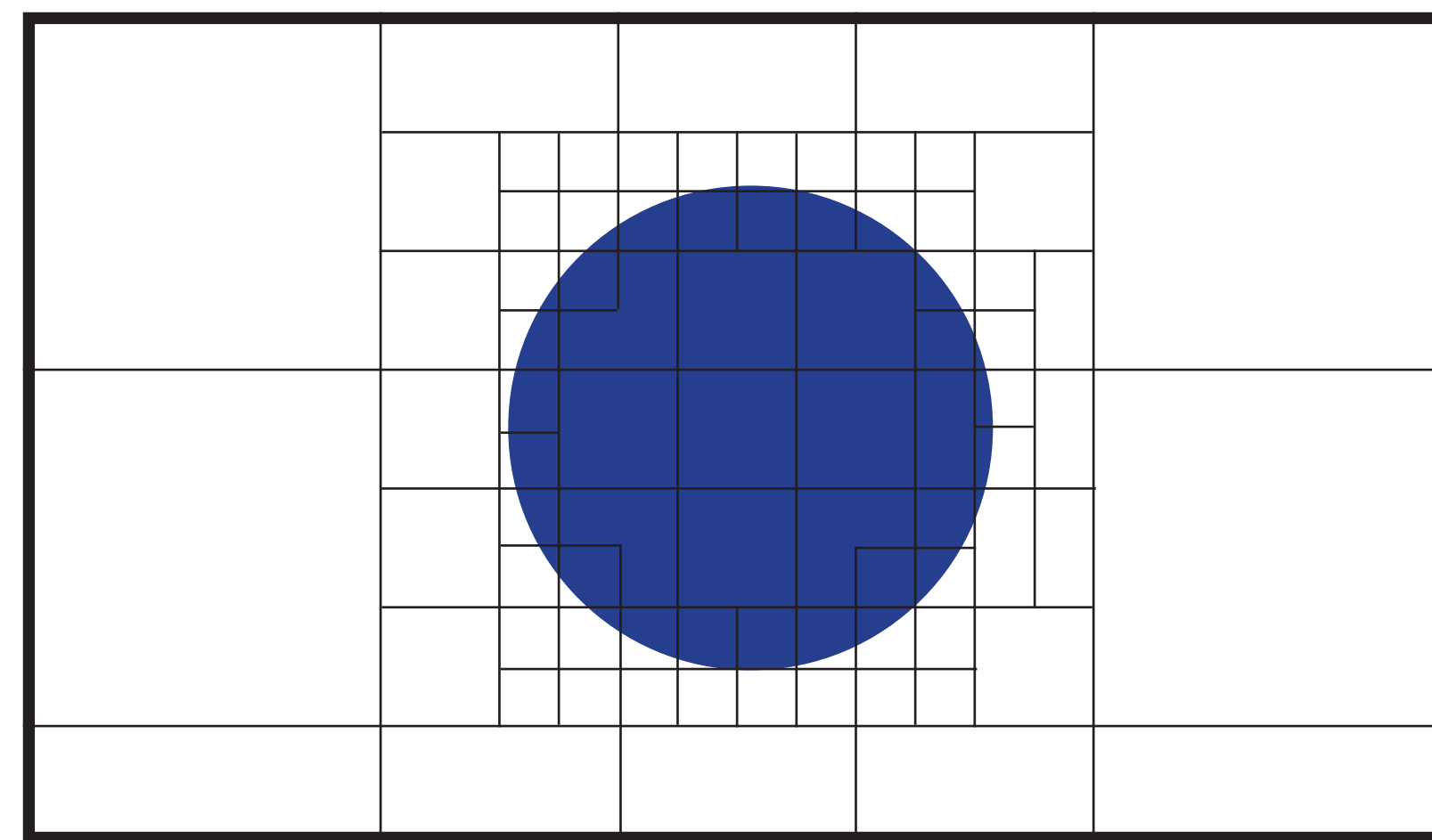
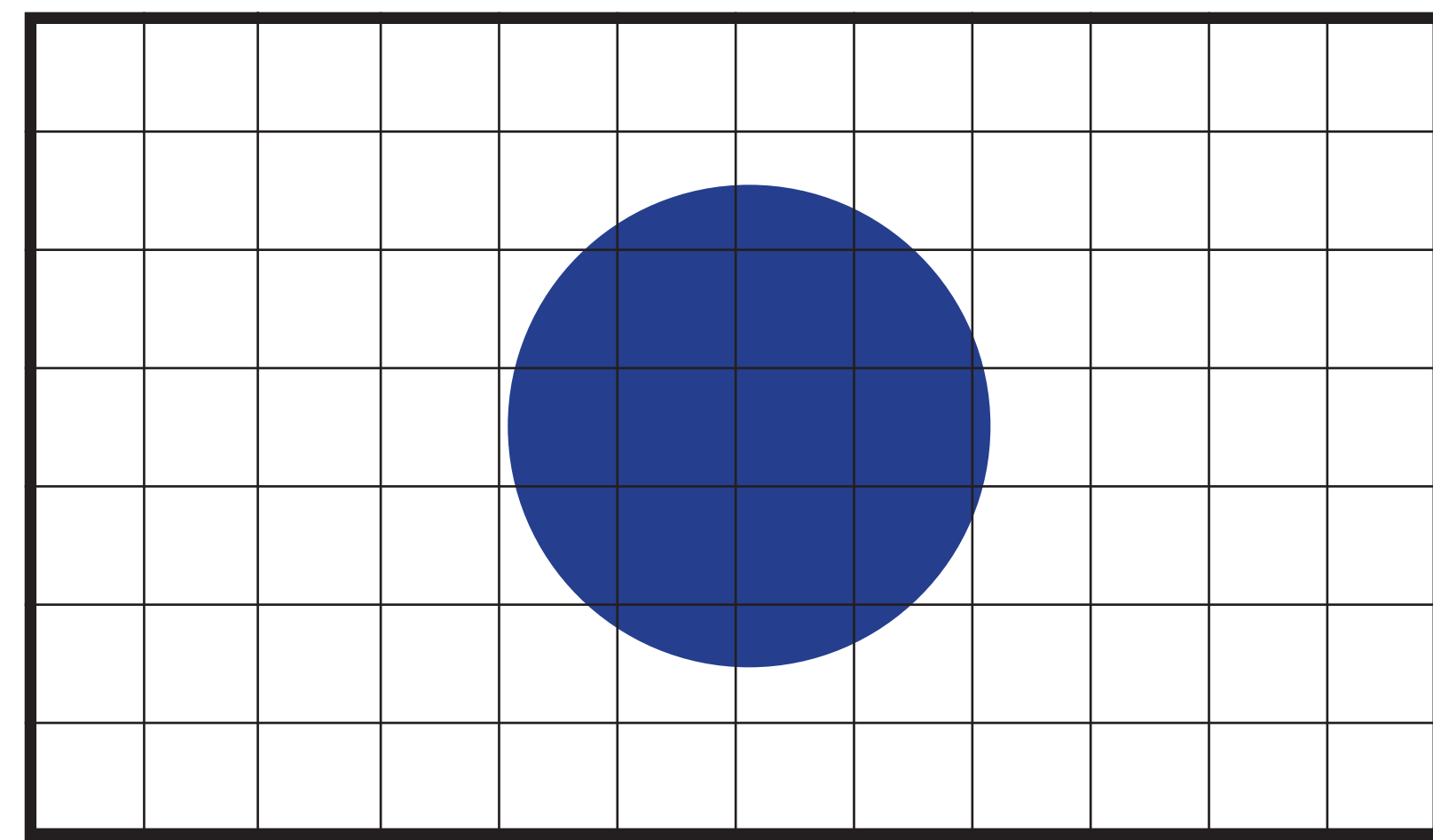
$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$



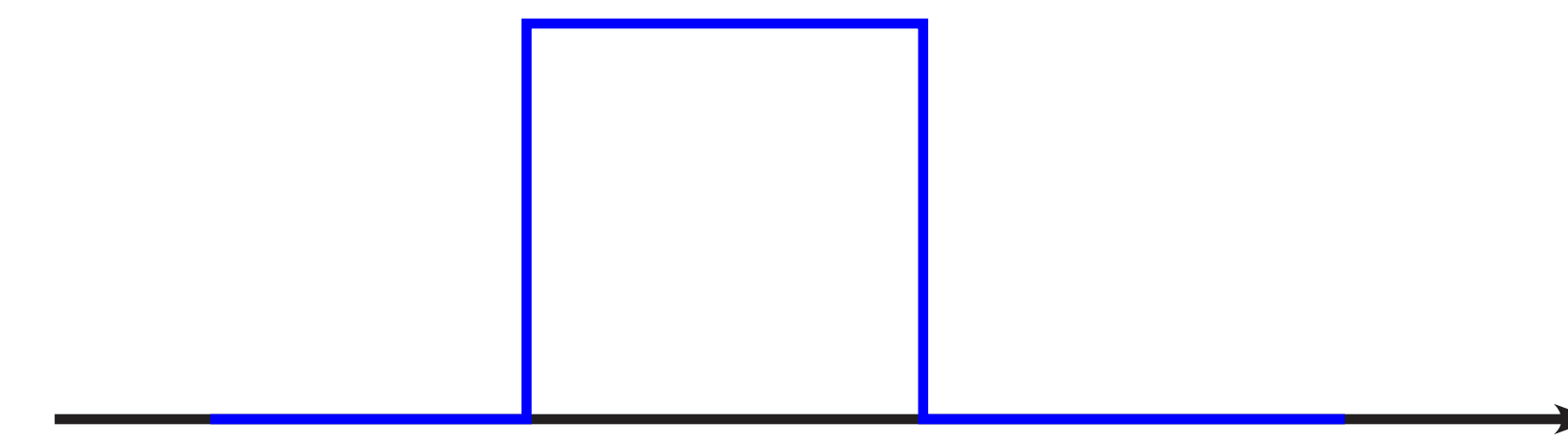
$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l} k, 2^l \xi)$$



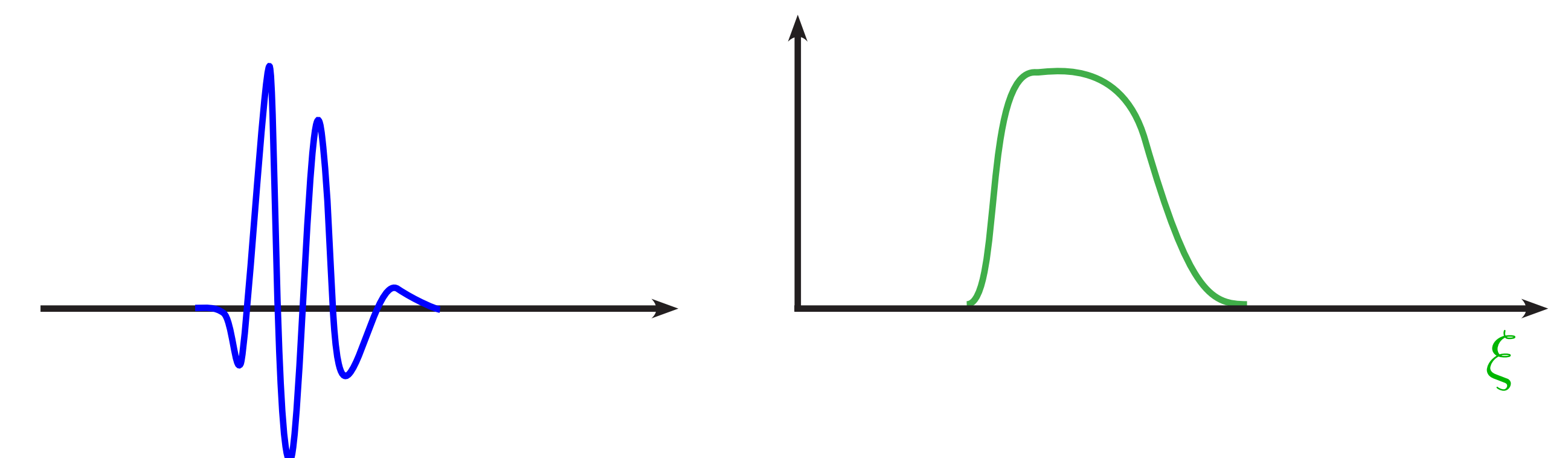
Number of “pixels”



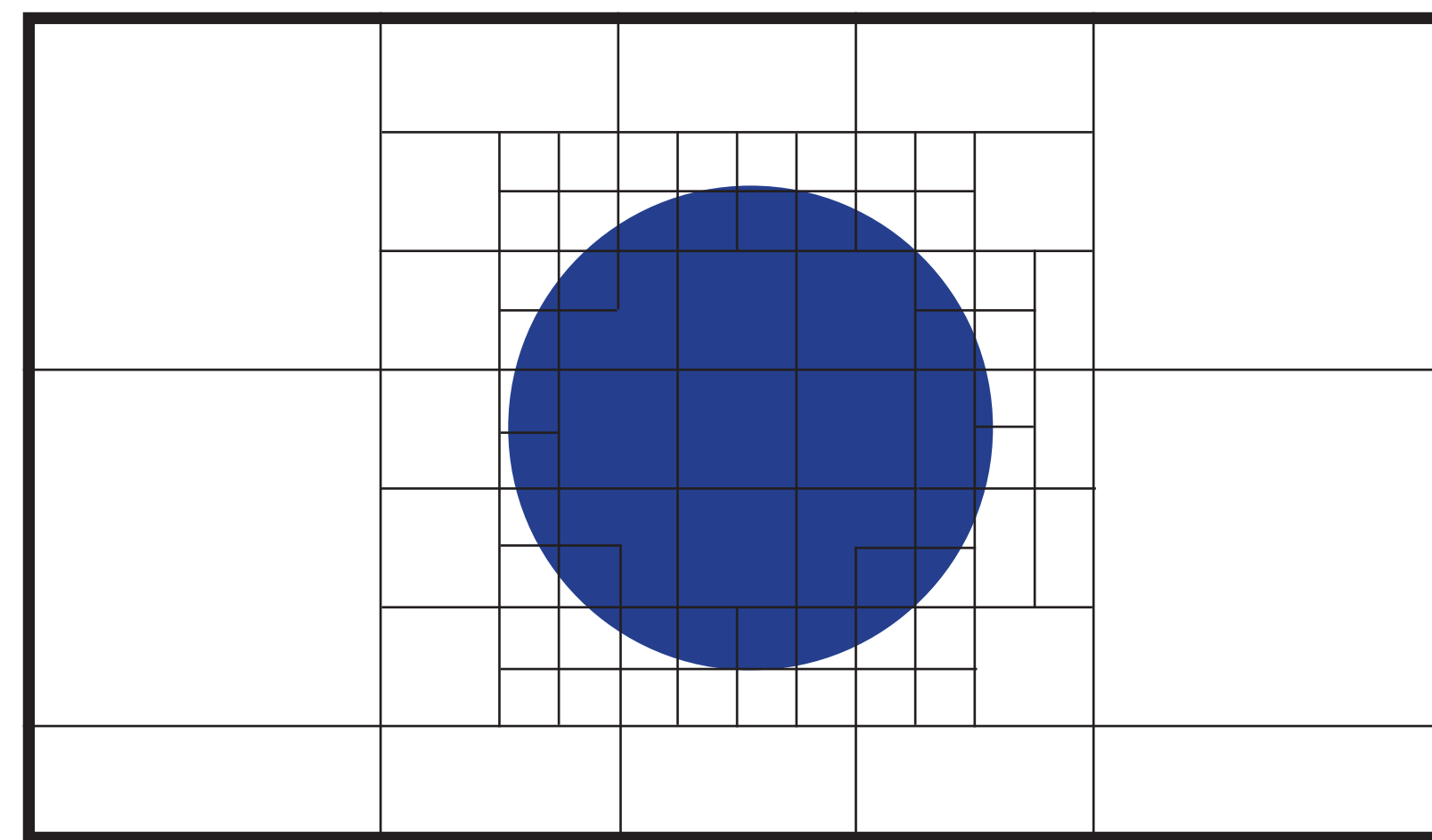
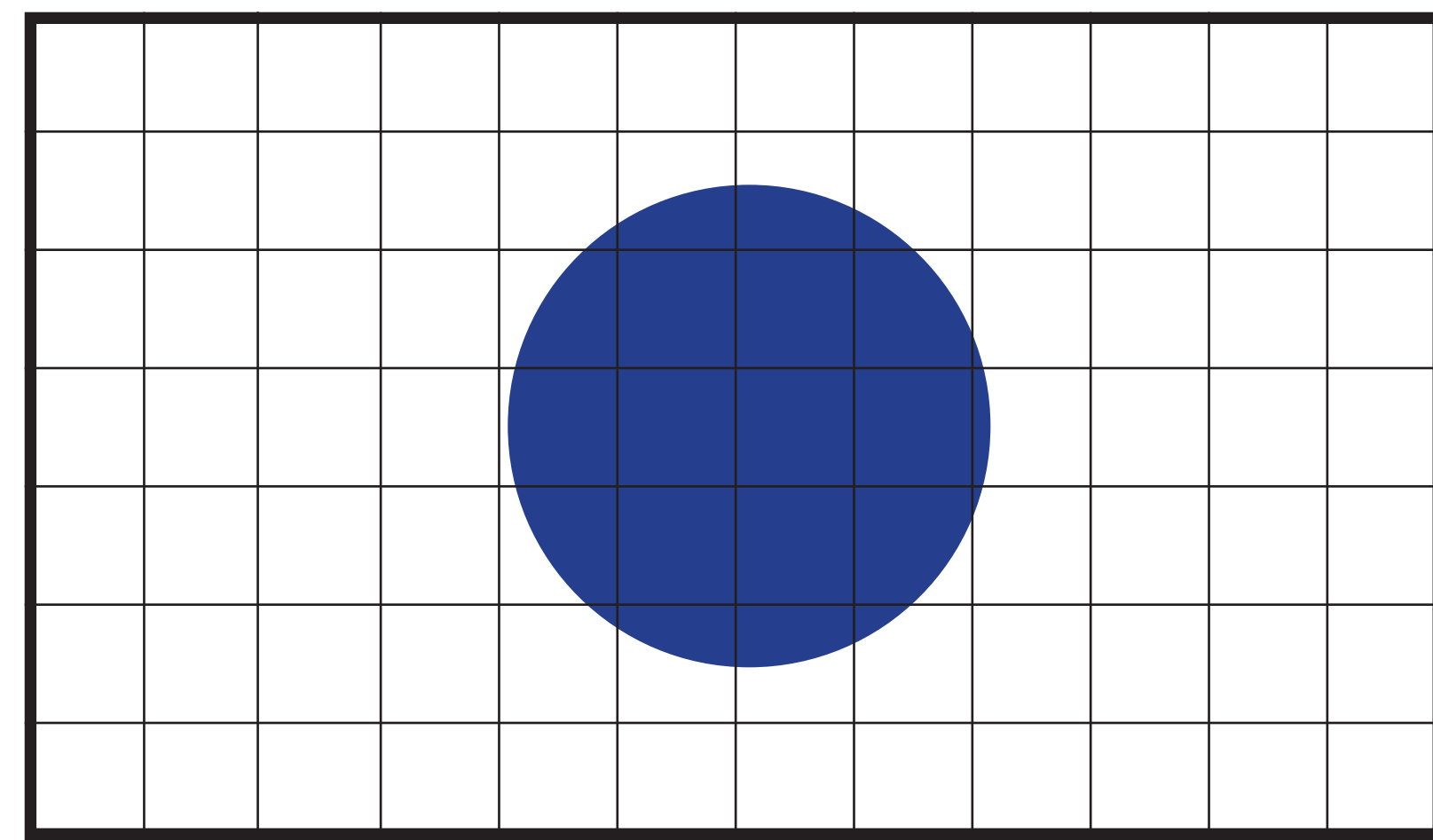
$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$



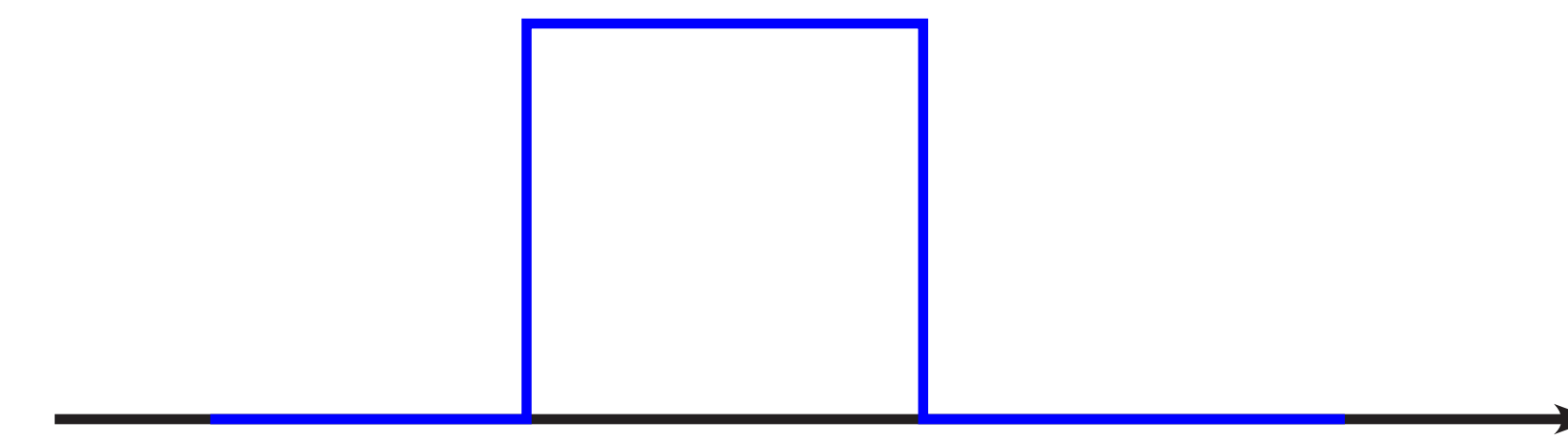
$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l}k, 2^l\xi)$$



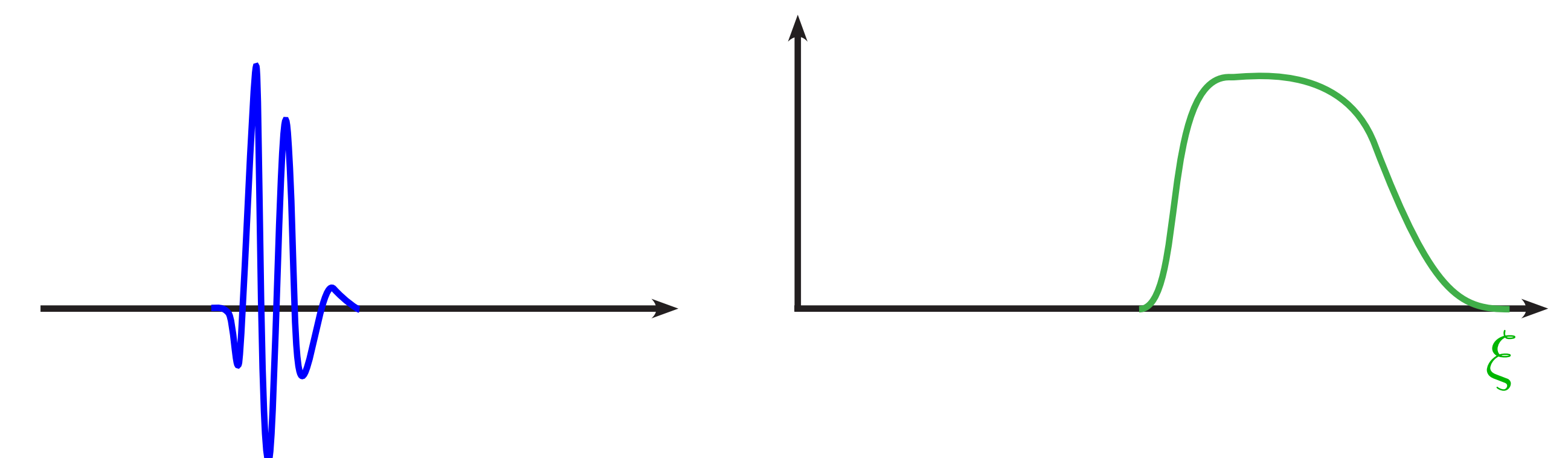
Number of “pixels”



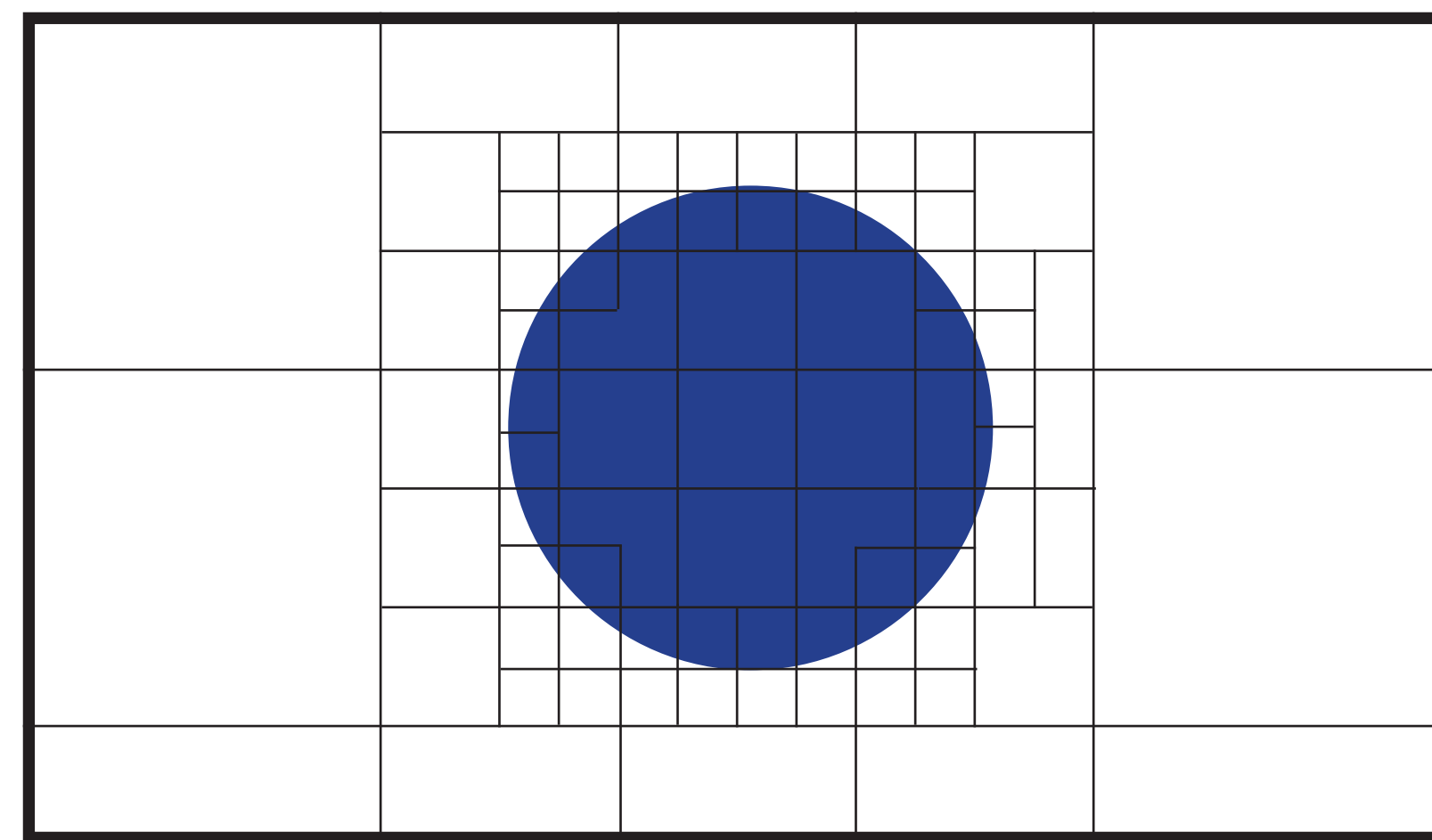
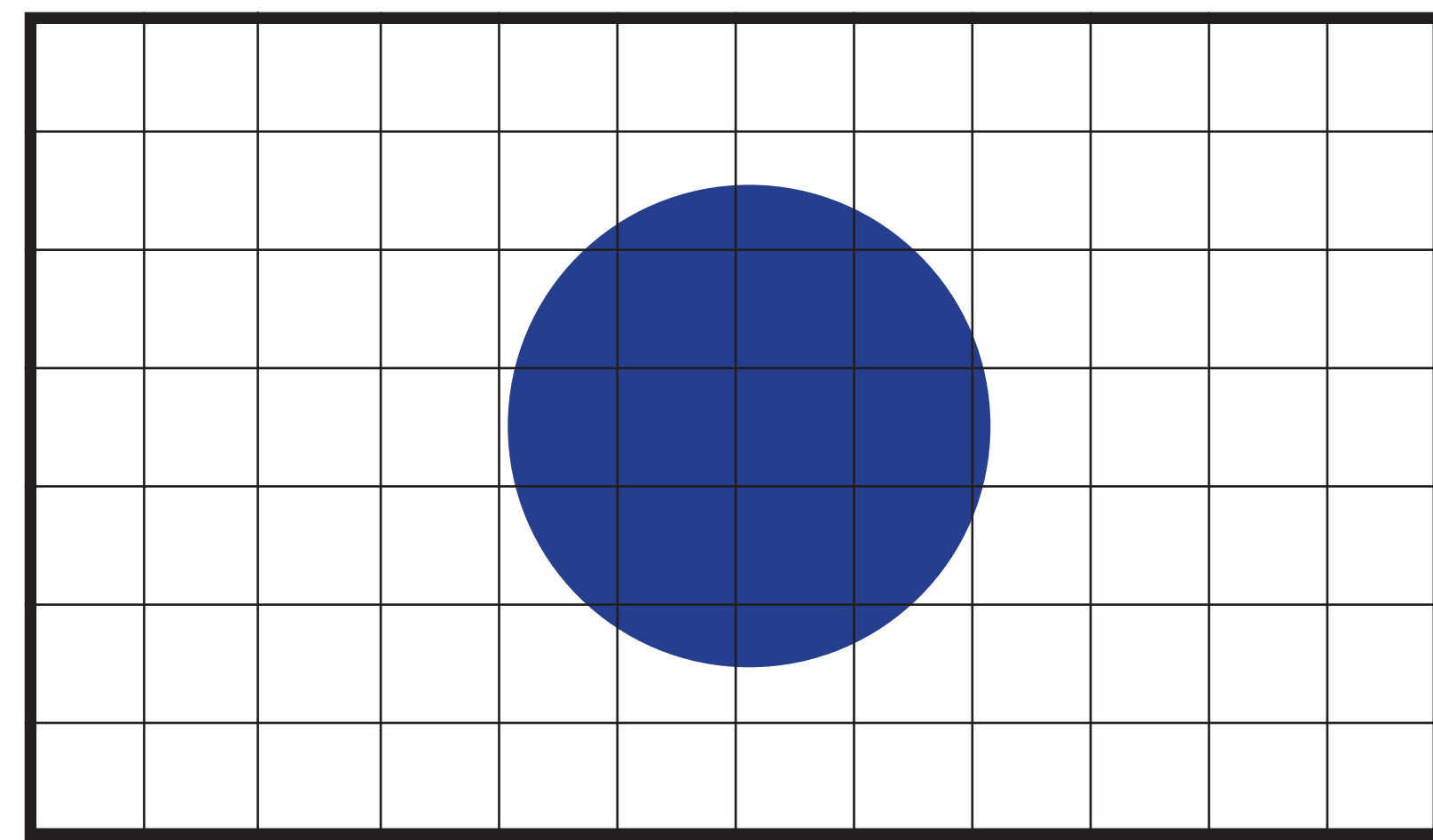
$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$



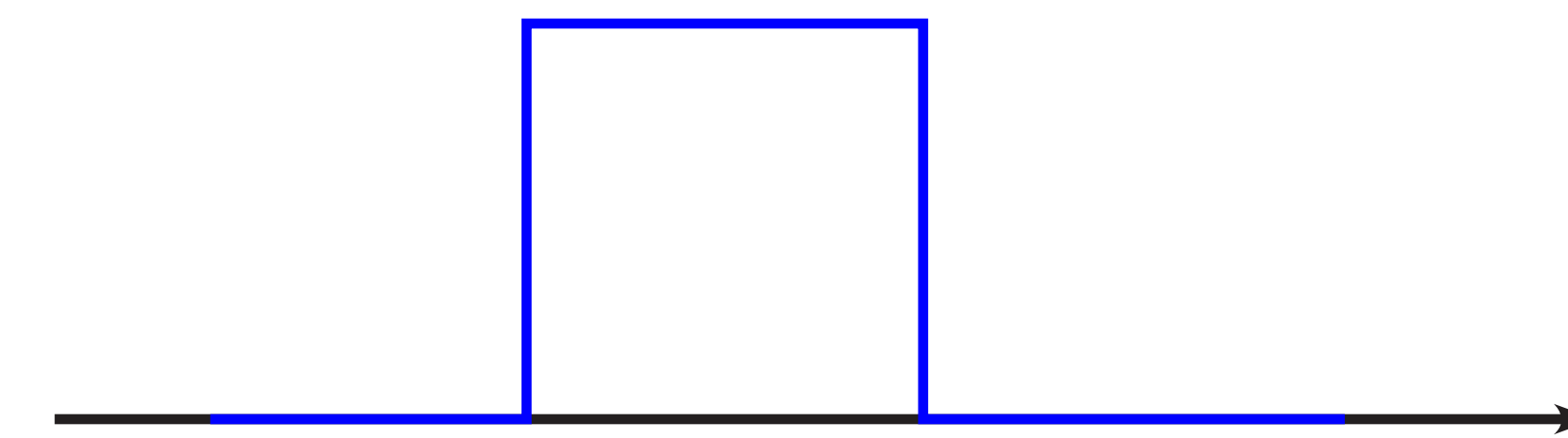
$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l}k, 2^l\xi)$$



Number of “pixels”

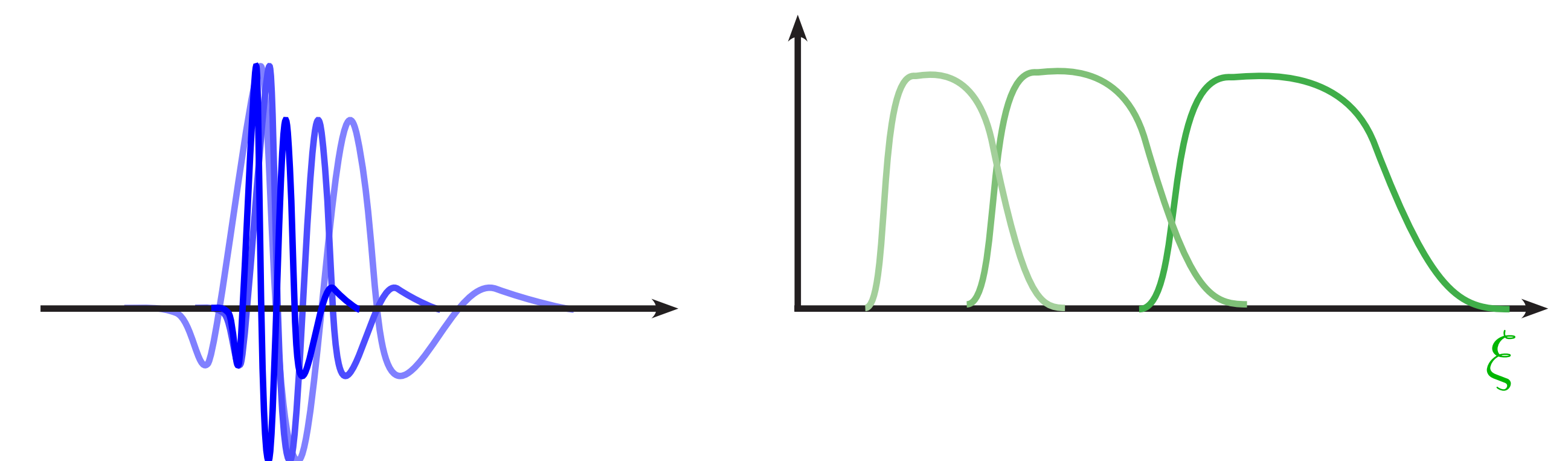


$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$

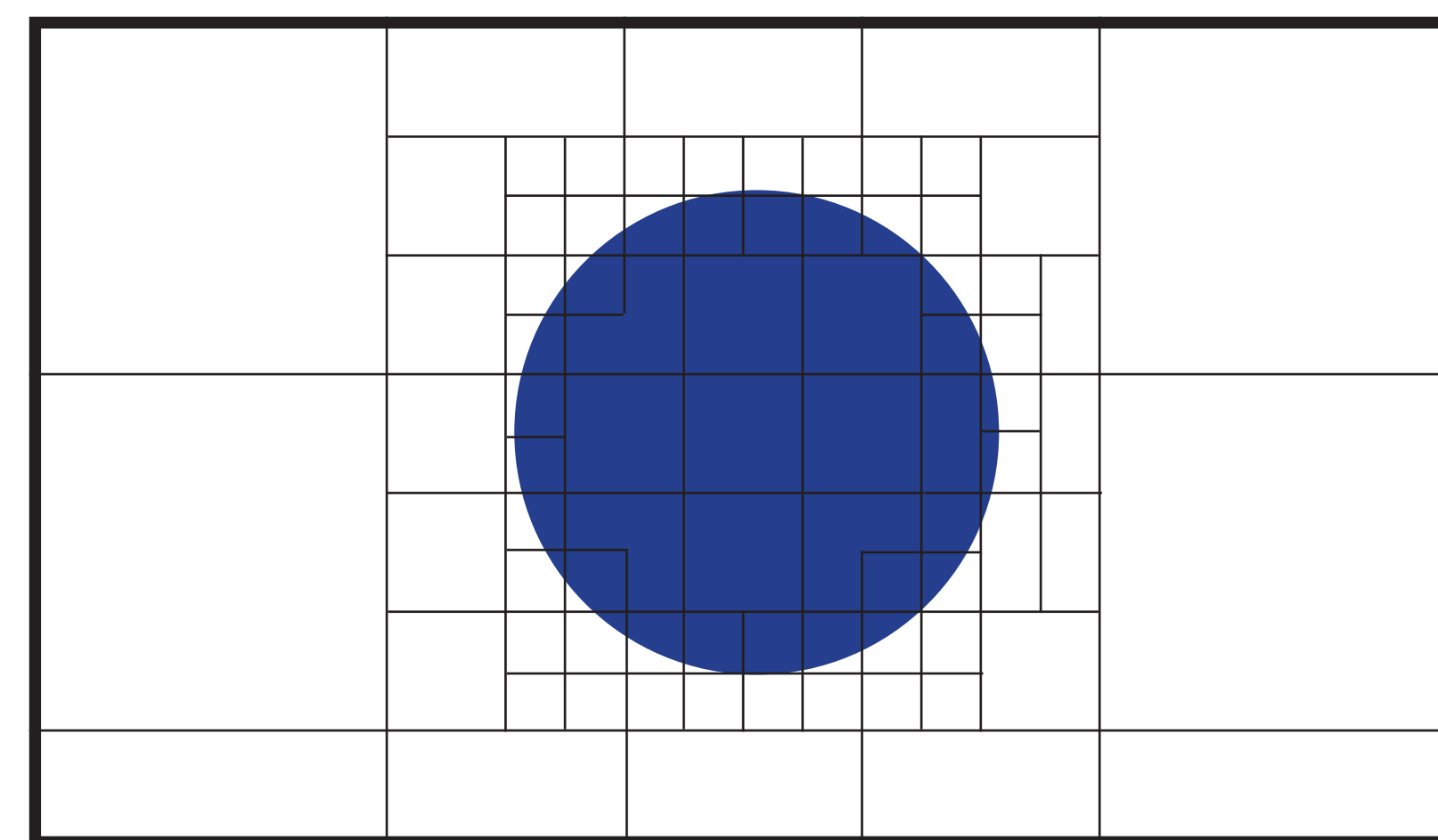
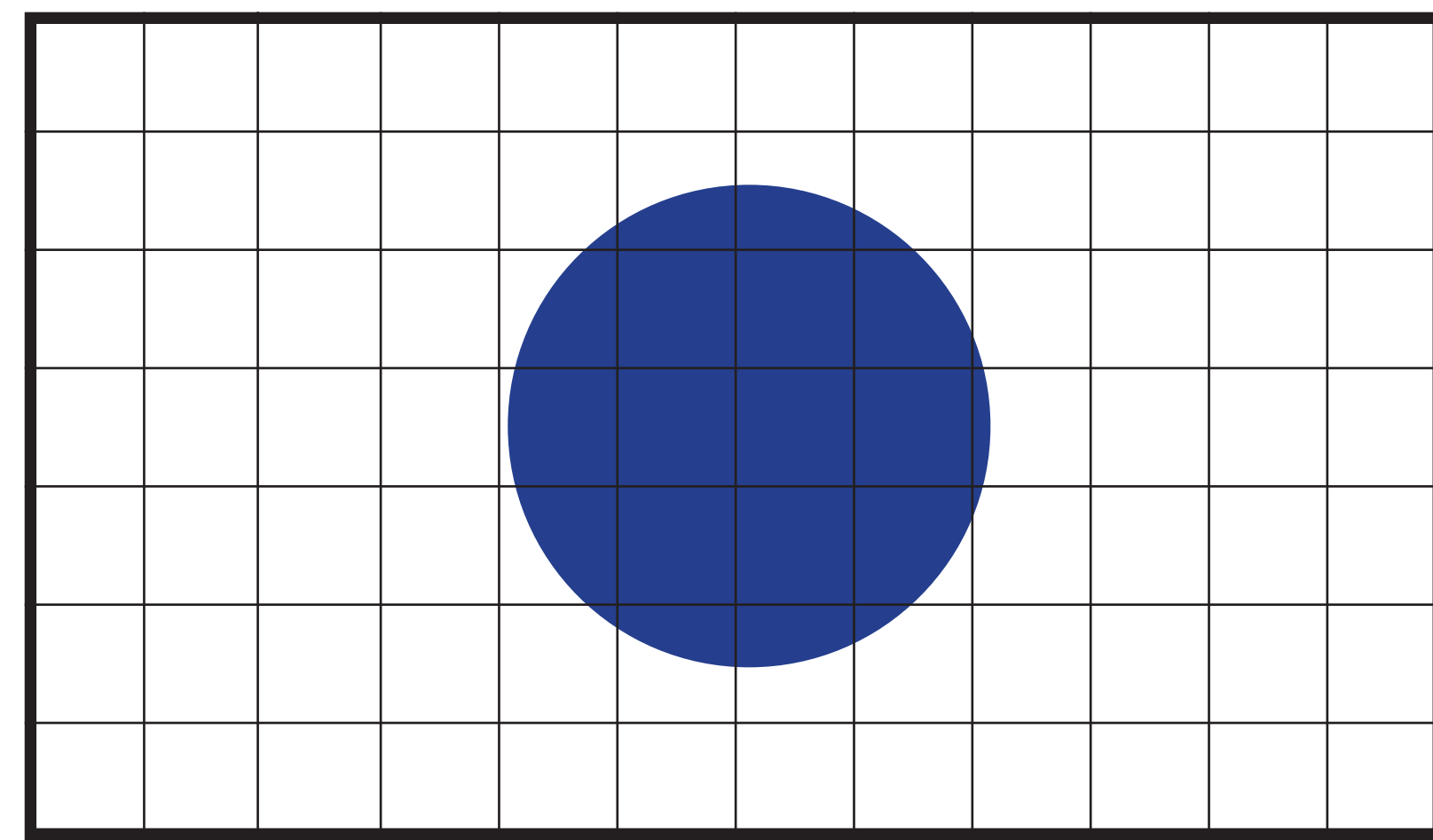


$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l} k, 2^l \xi)$$

Hierarchical:
progressive

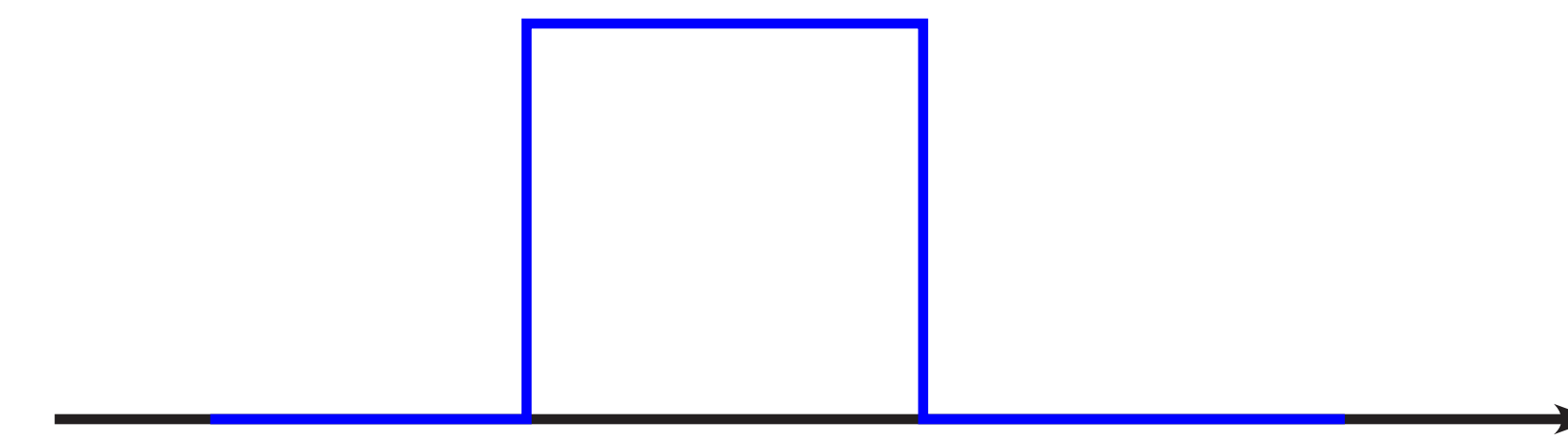


Number of “pixels”



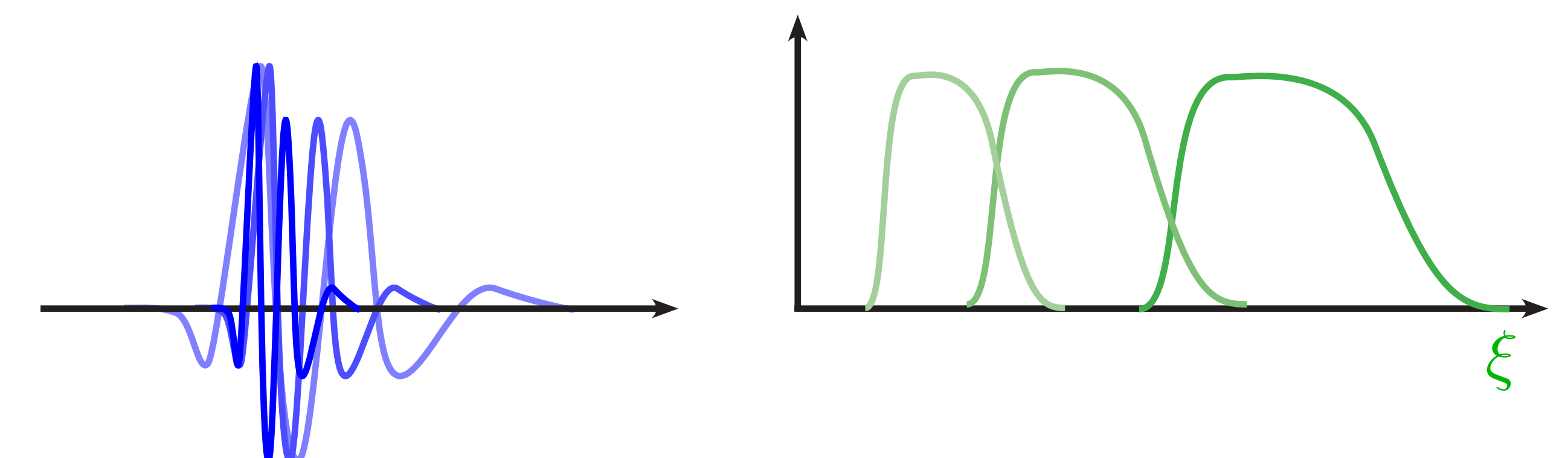
Locality: varying resolution

$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$

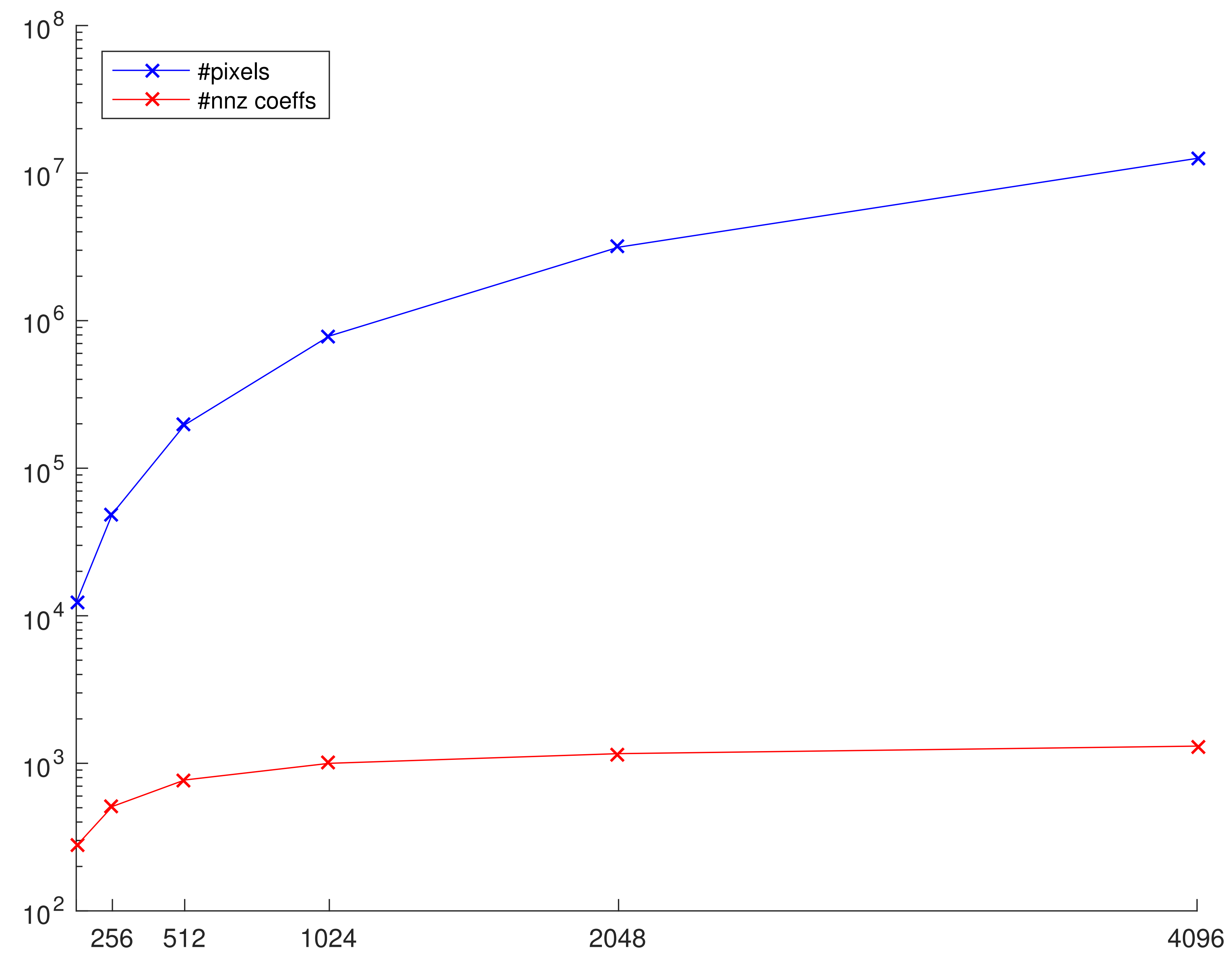


$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l}k, 2^l\xi)$$

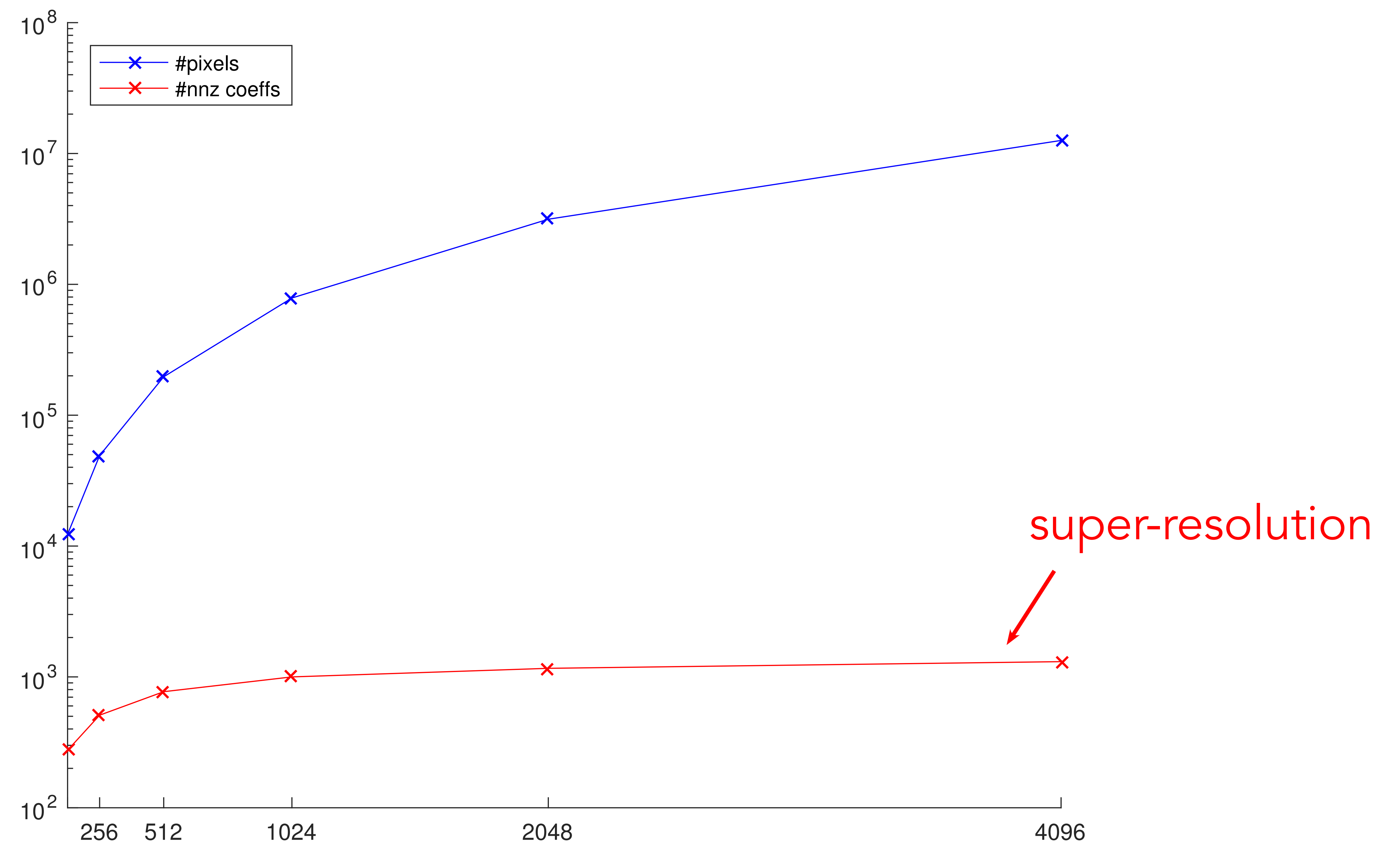
Hierarchical:
progressive



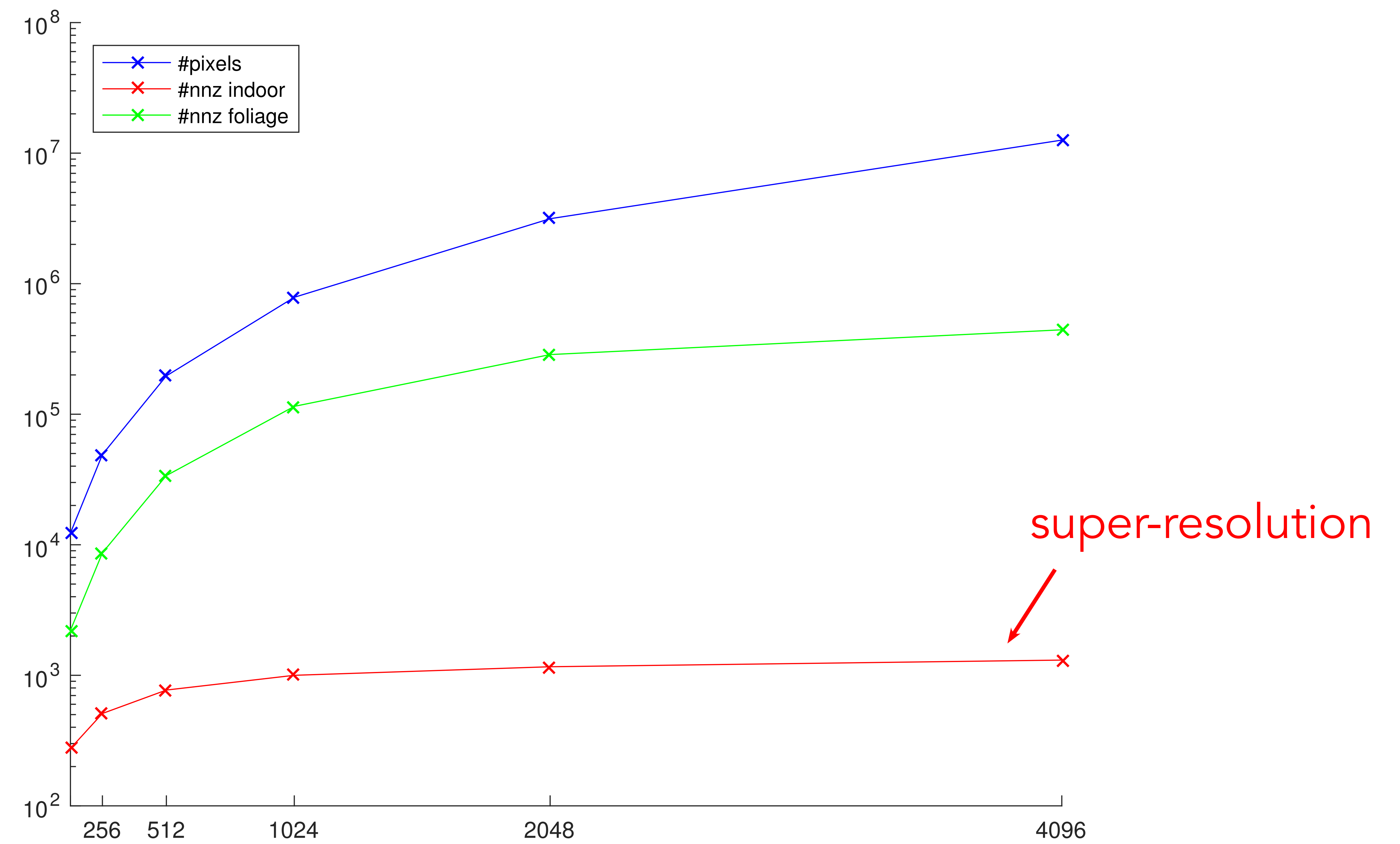
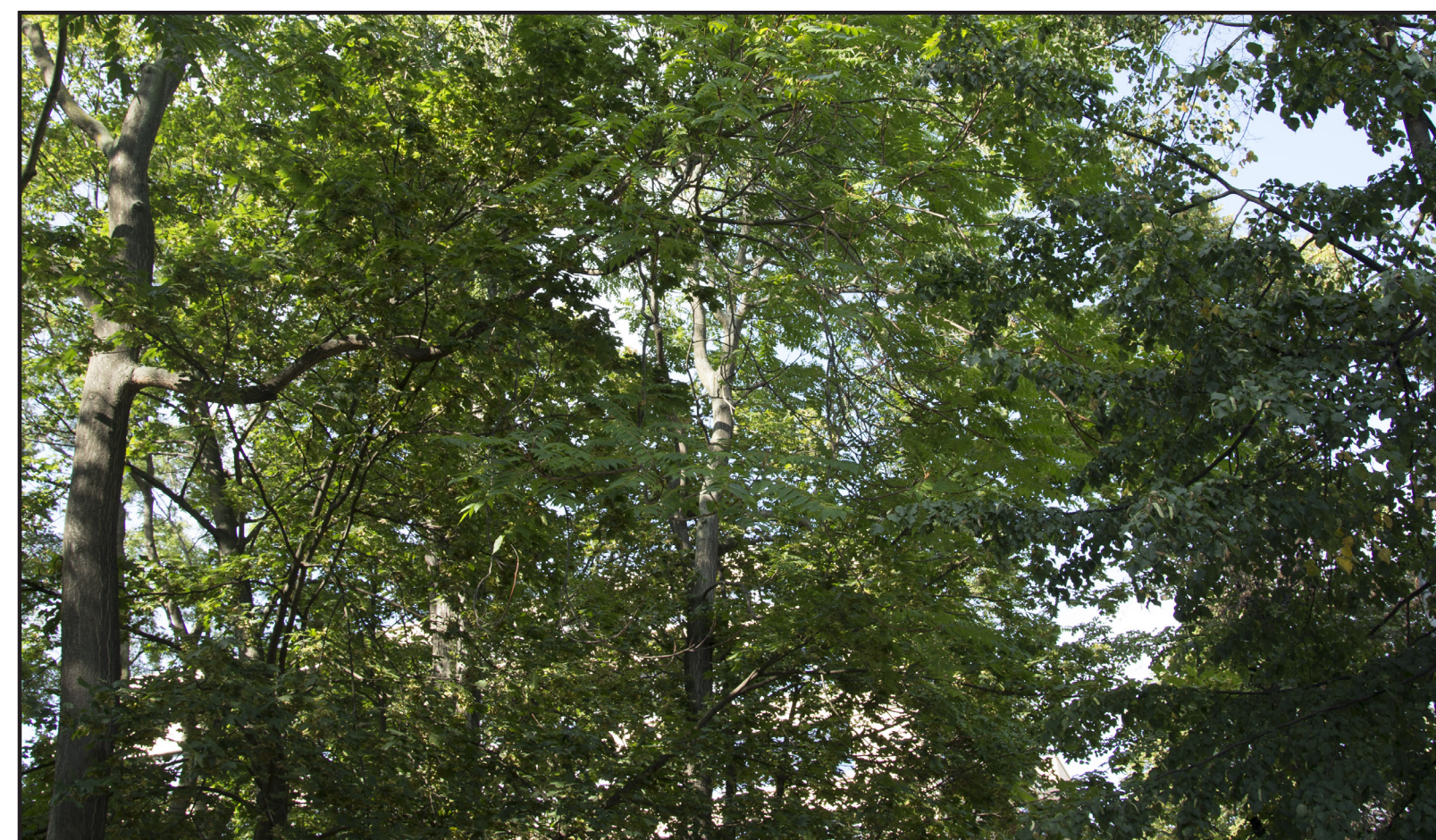
Number of “pixels”



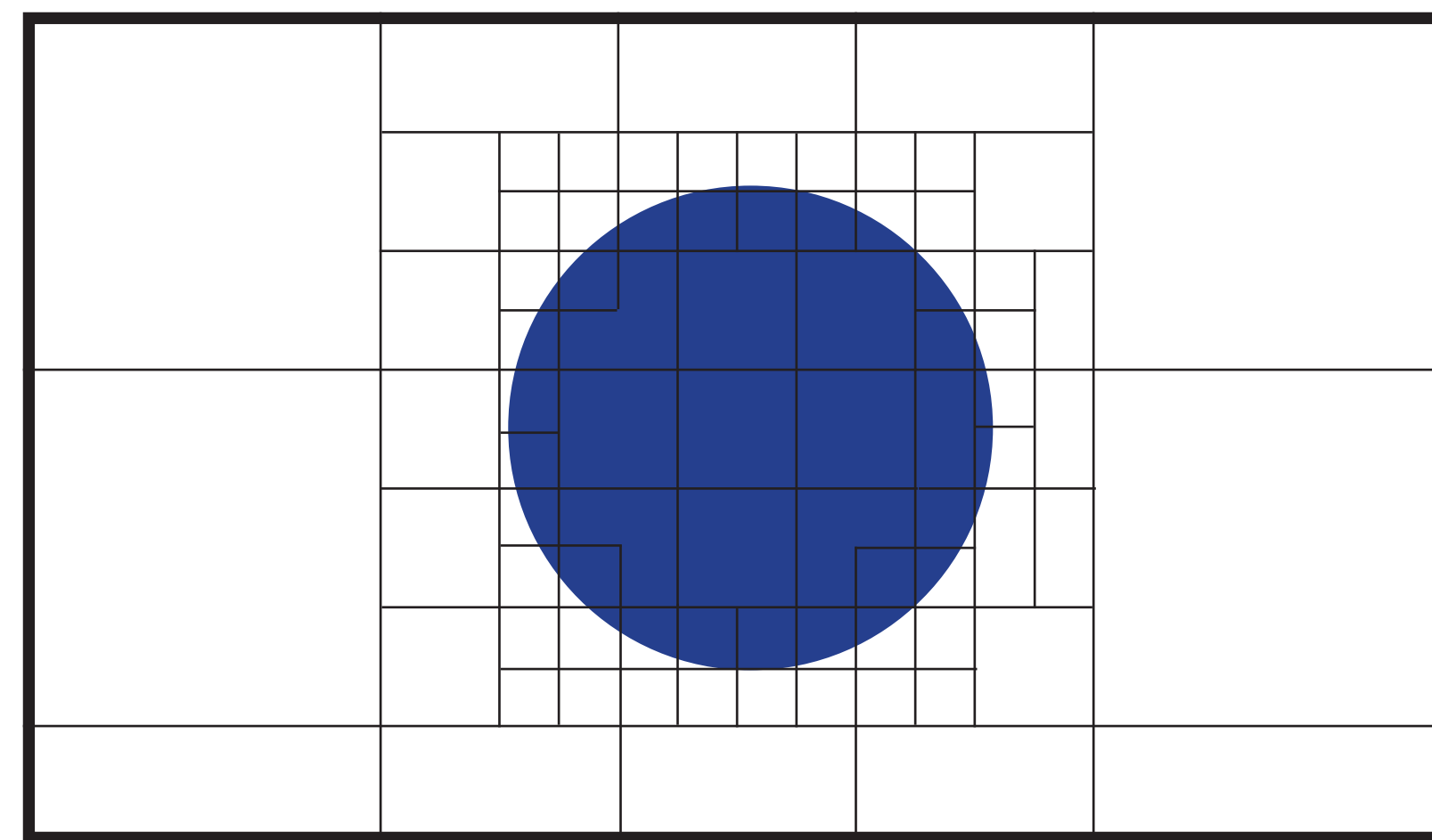
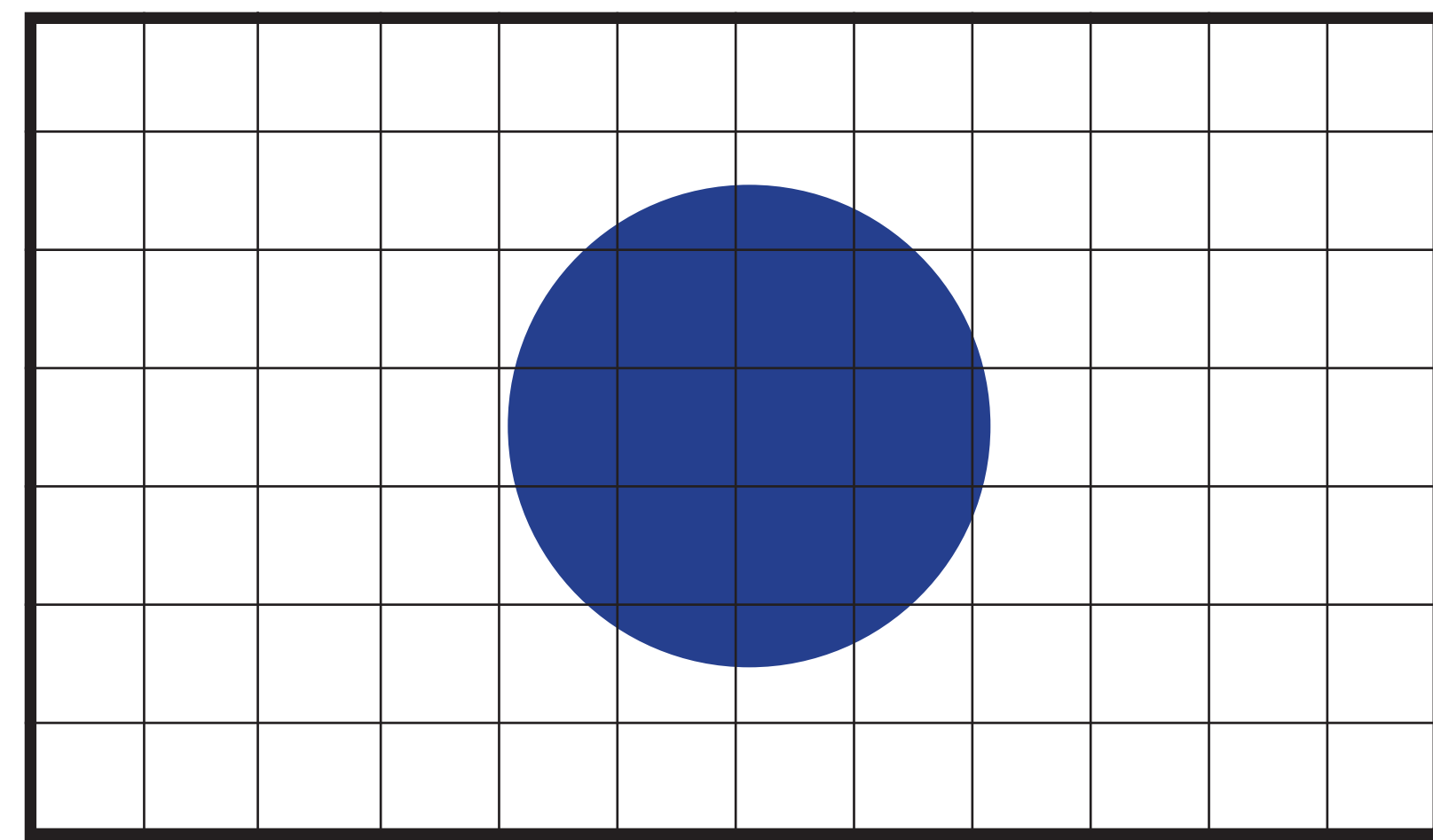
Number of “pixels”



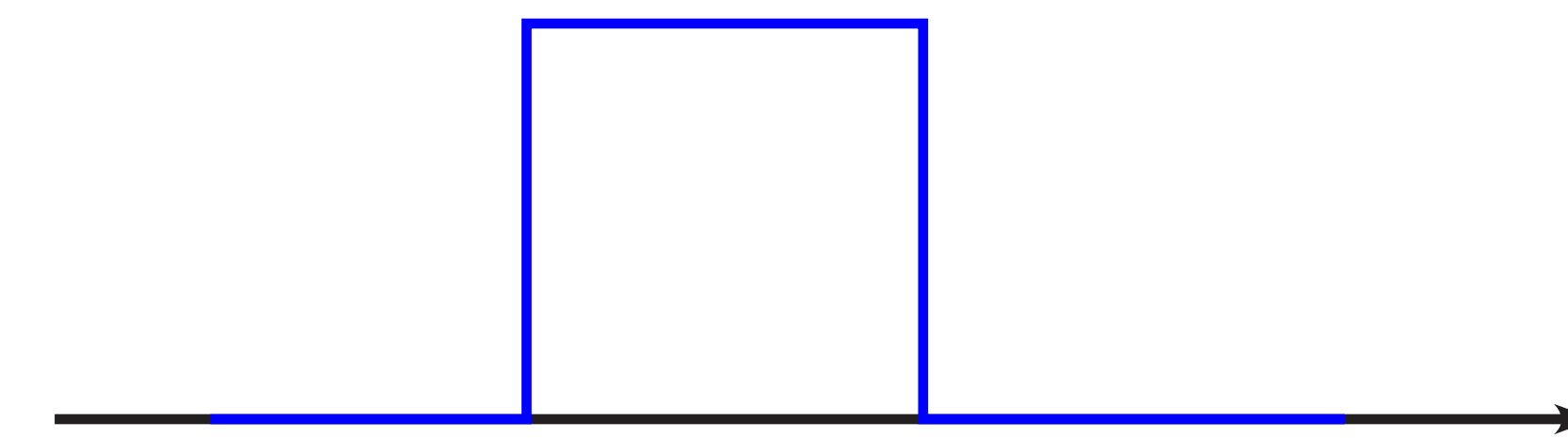
Number of “pixels”



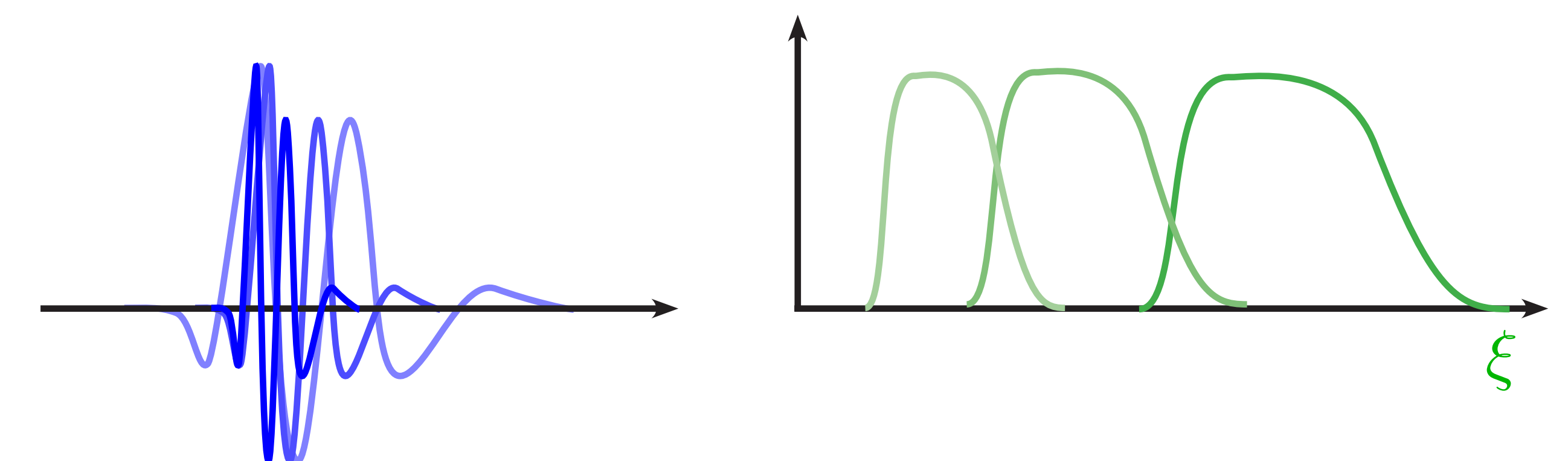
Number of “pixels”



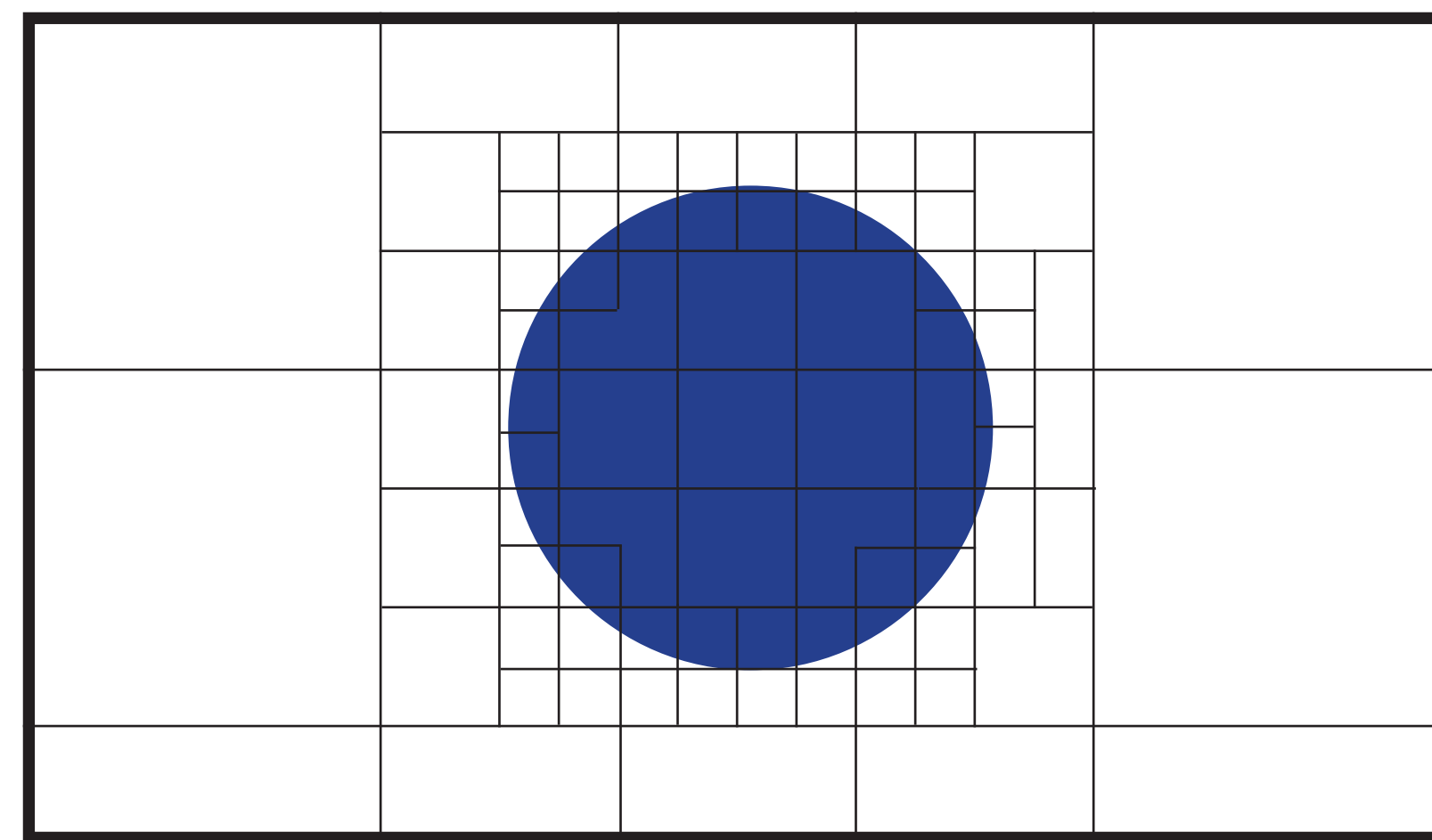
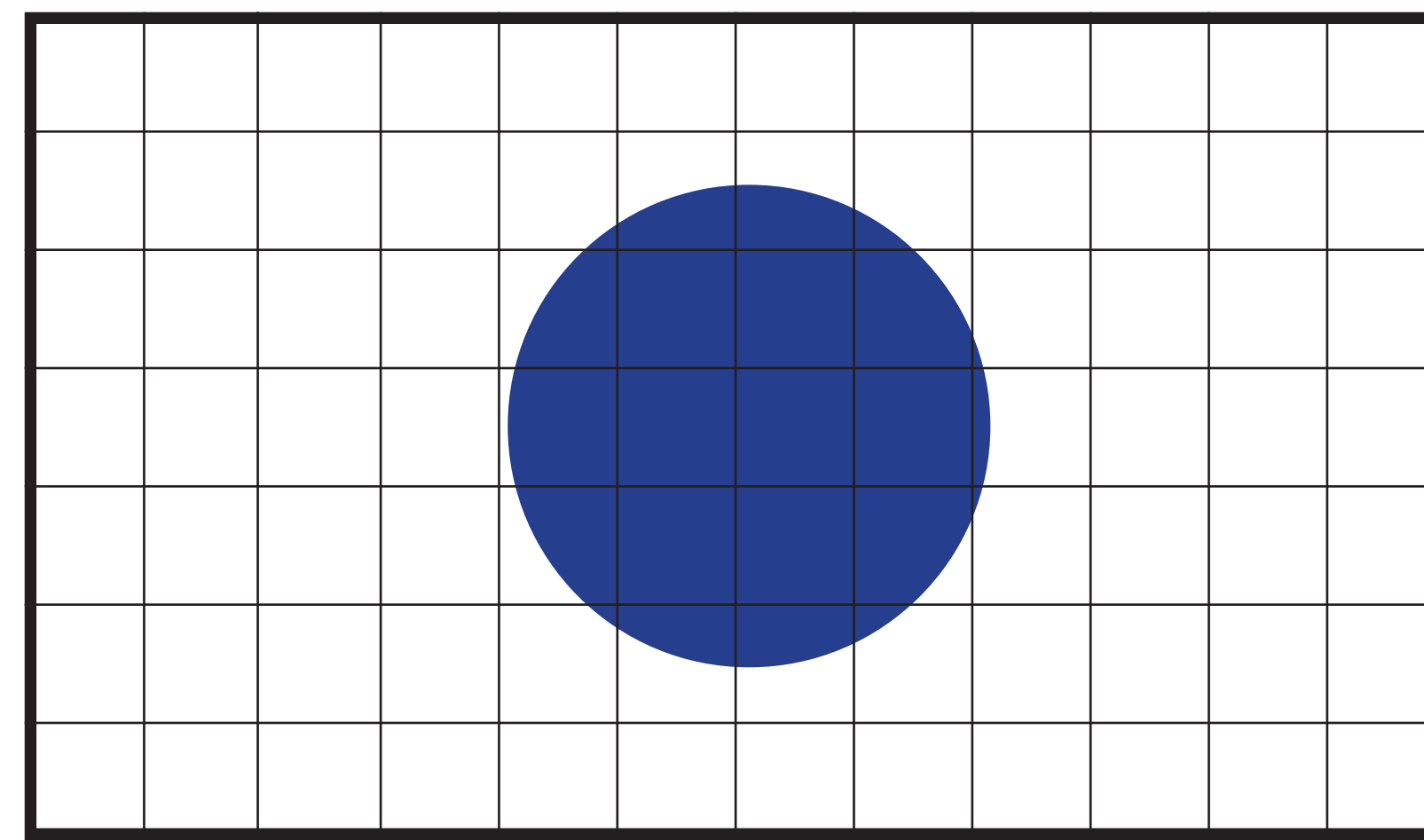
$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$



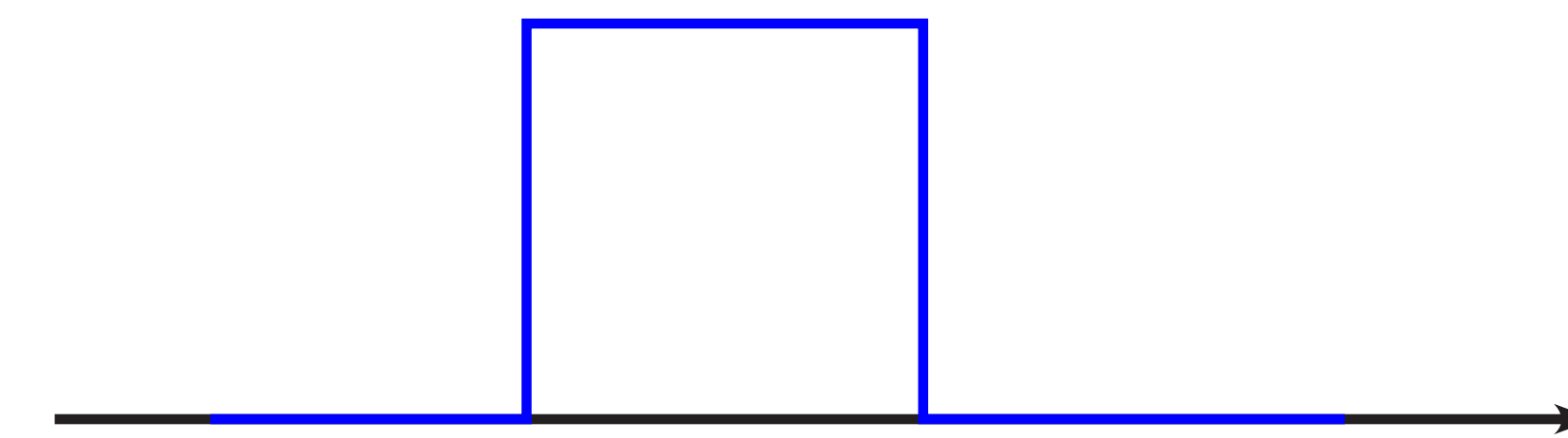
$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l}k, 2^l\xi)$$



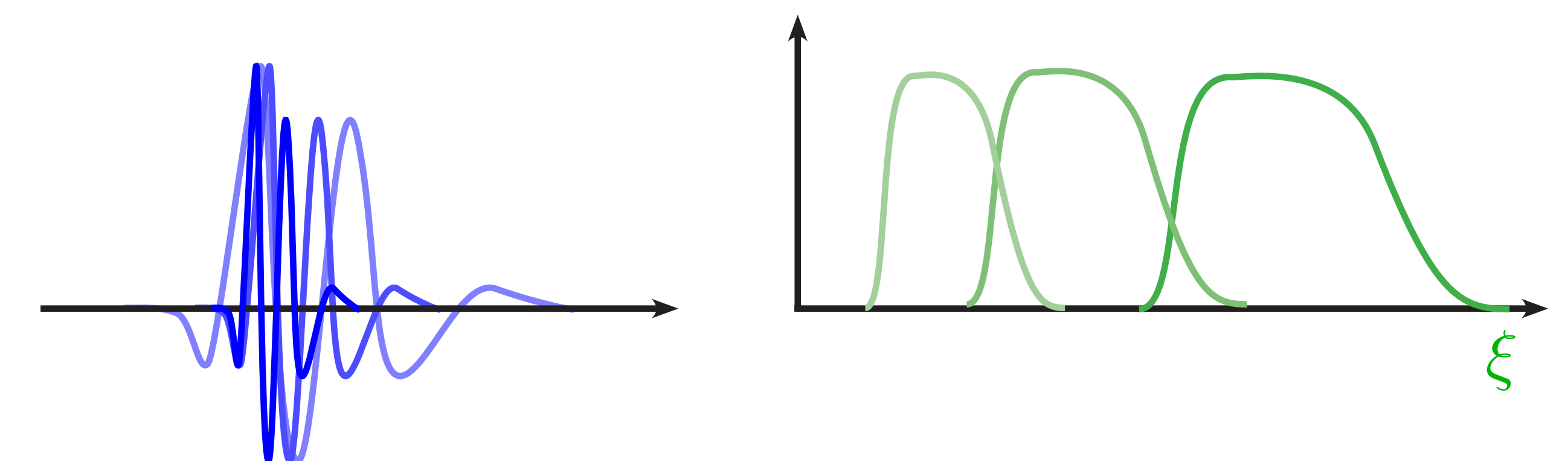
Number of “pixels”



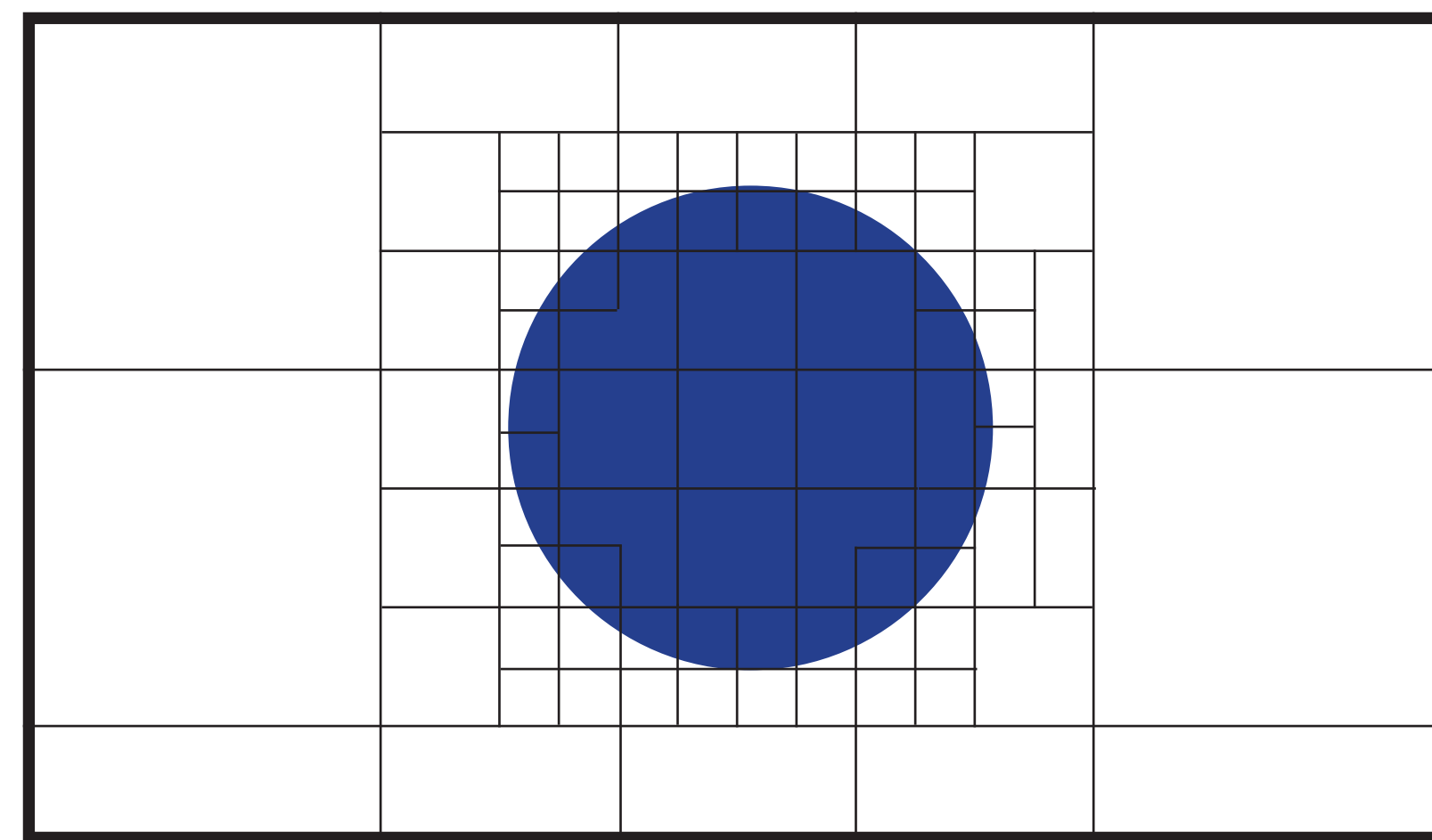
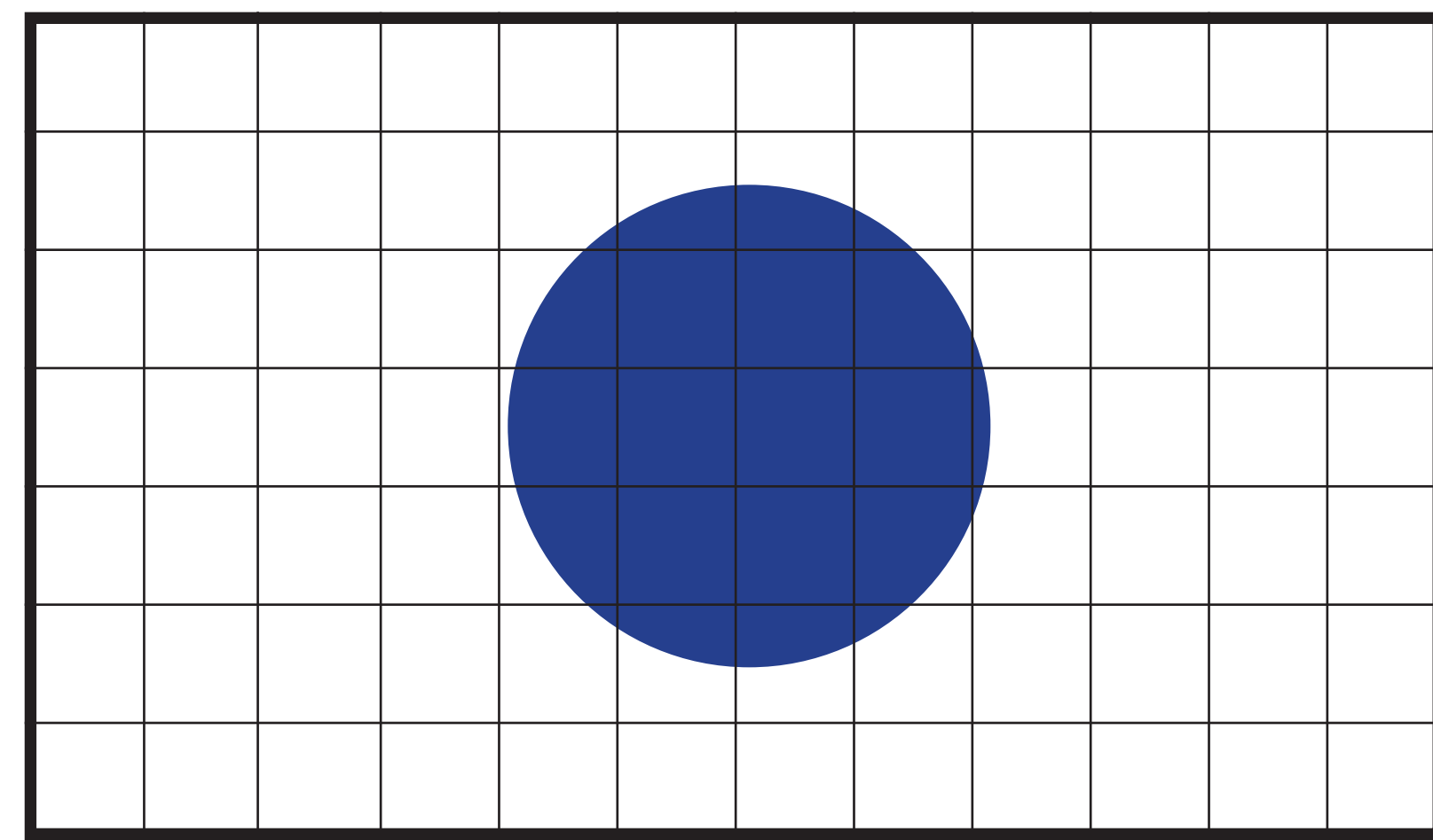
$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$



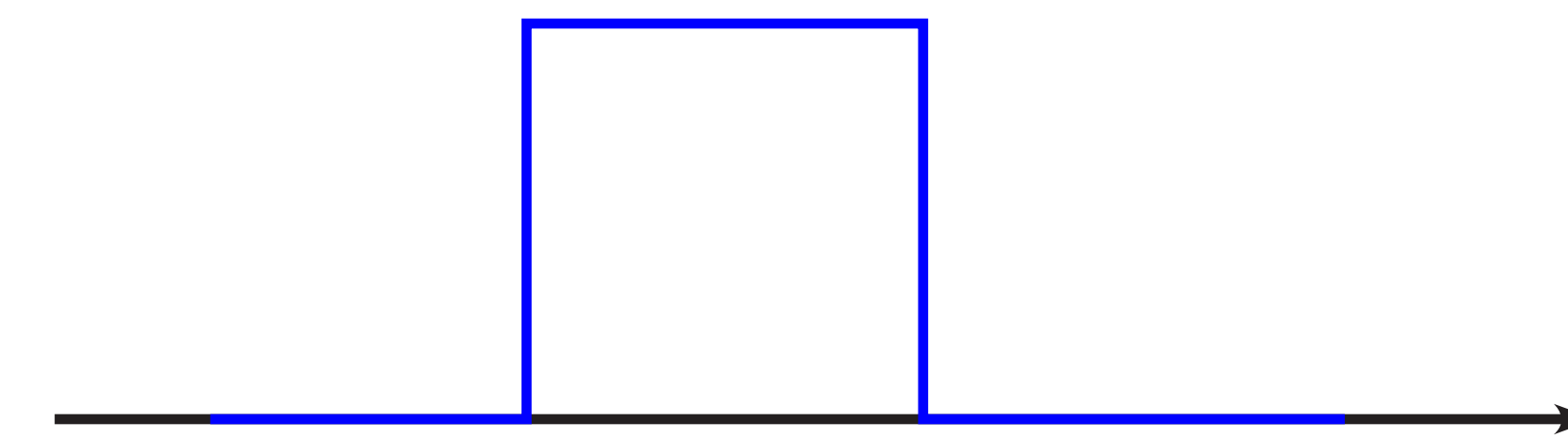
$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l}k, 2^l\xi)$$



Number of “pixels”

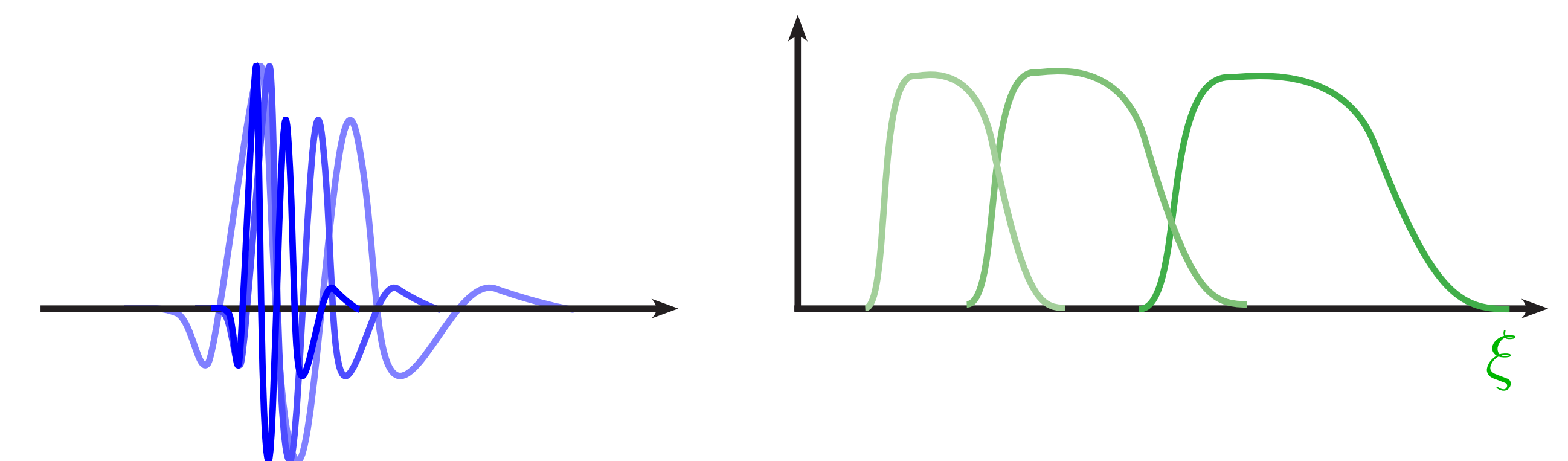


$$f(x) = \sum_{i \in U \subset r\mathbb{Z}^2} f_i \chi_i(x)$$



$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l}k, 2^l\xi)$$

depends on
image function



Computational costs

$$C = N \cdot C_p$$

/ \

Number of Cost per

"pixels" pixel

Computational costs

$$C = N \cdot C_p$$

/ \

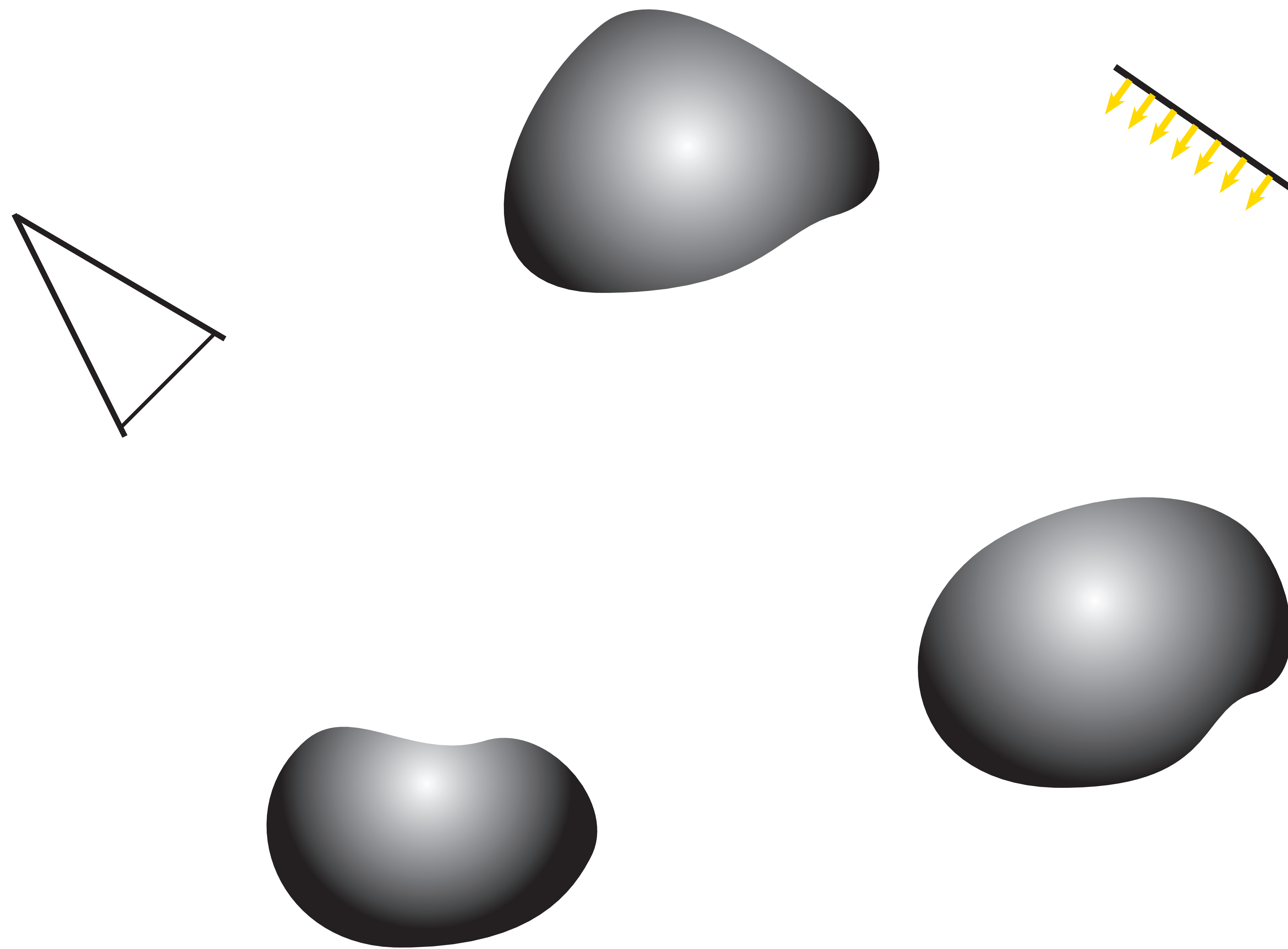
Number of Cost per

"pixels" pixel

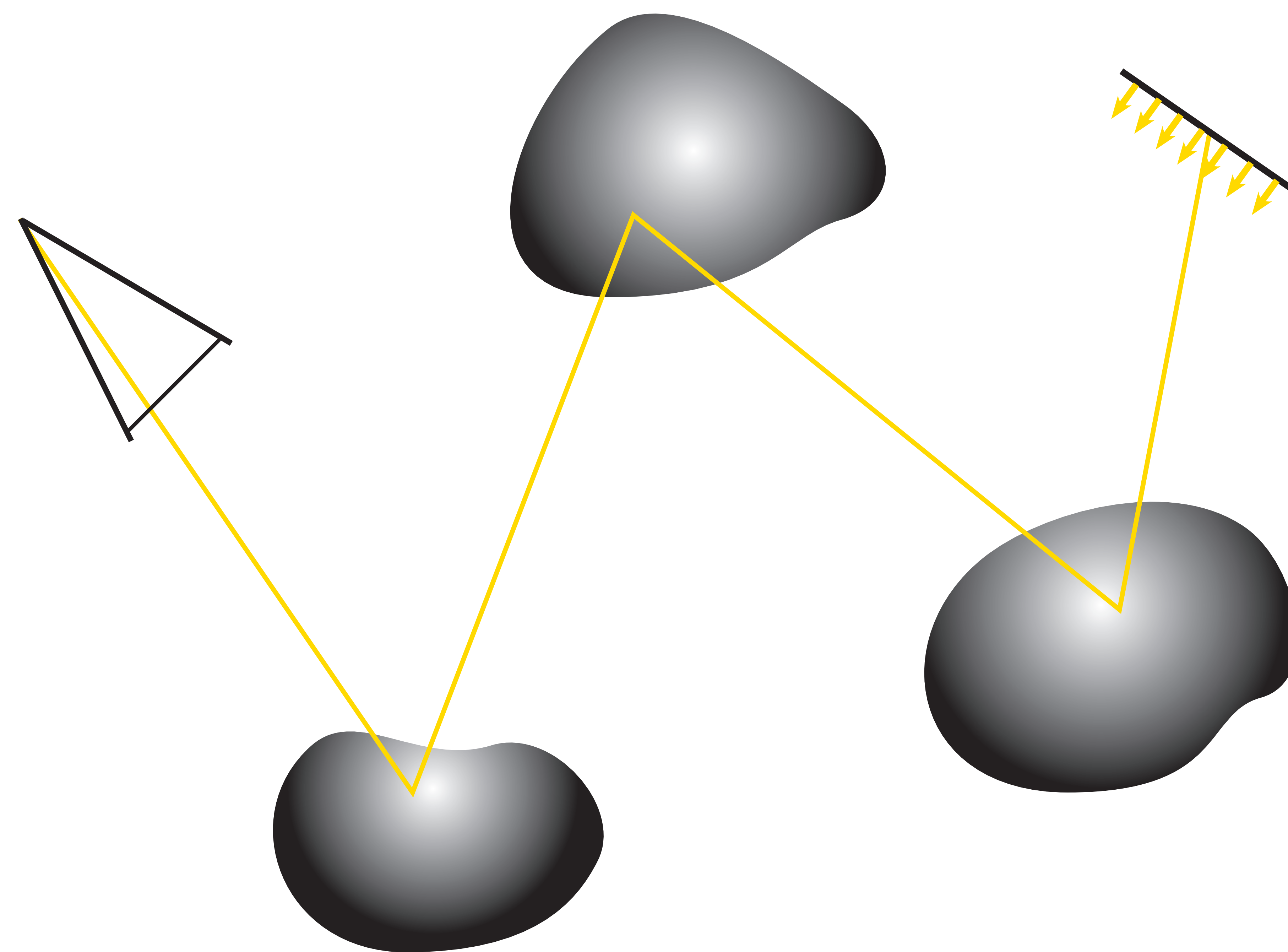
↓ minimize by

- Use sparse, adaptive image representation

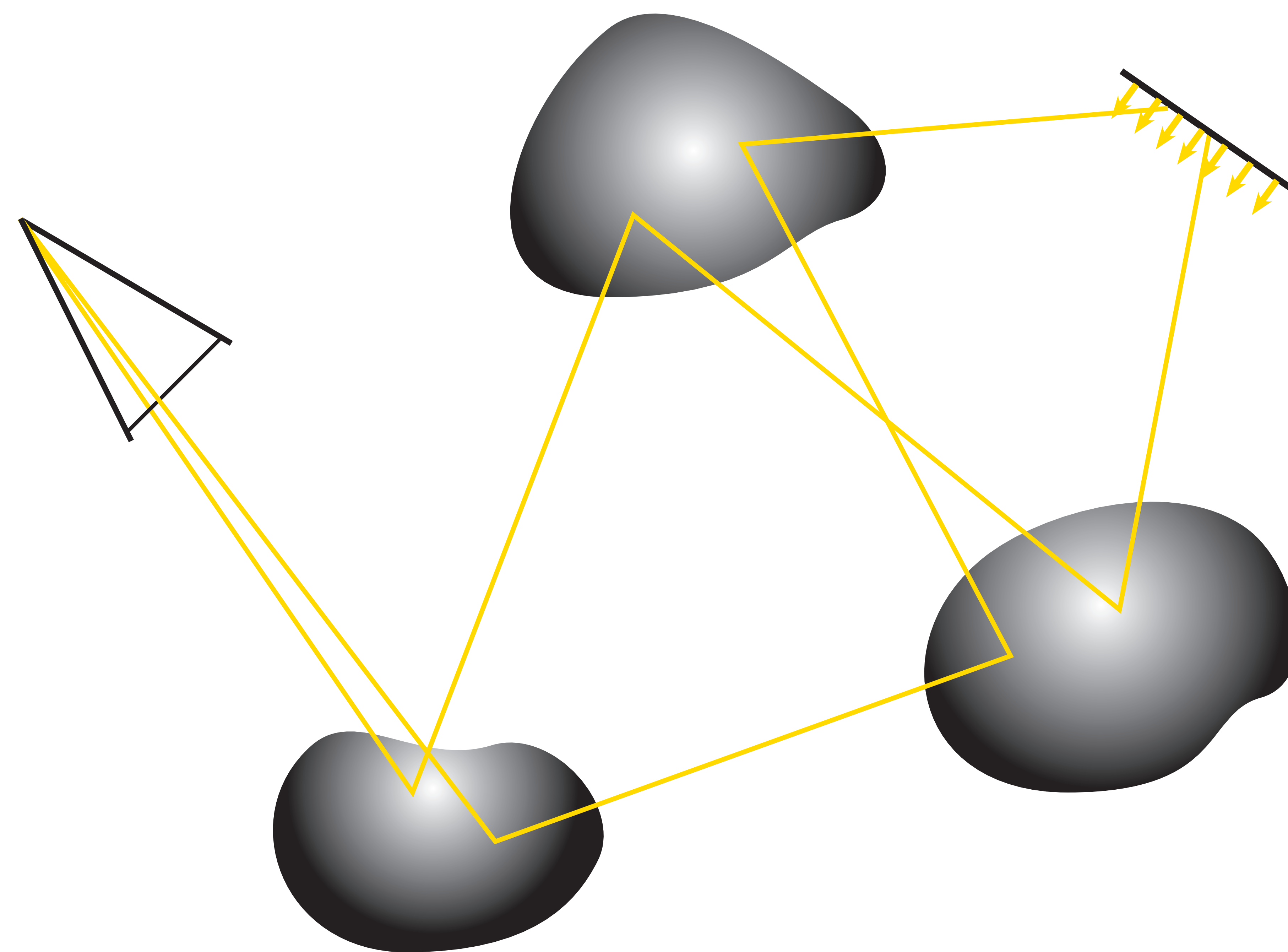
Cost per “pixel”



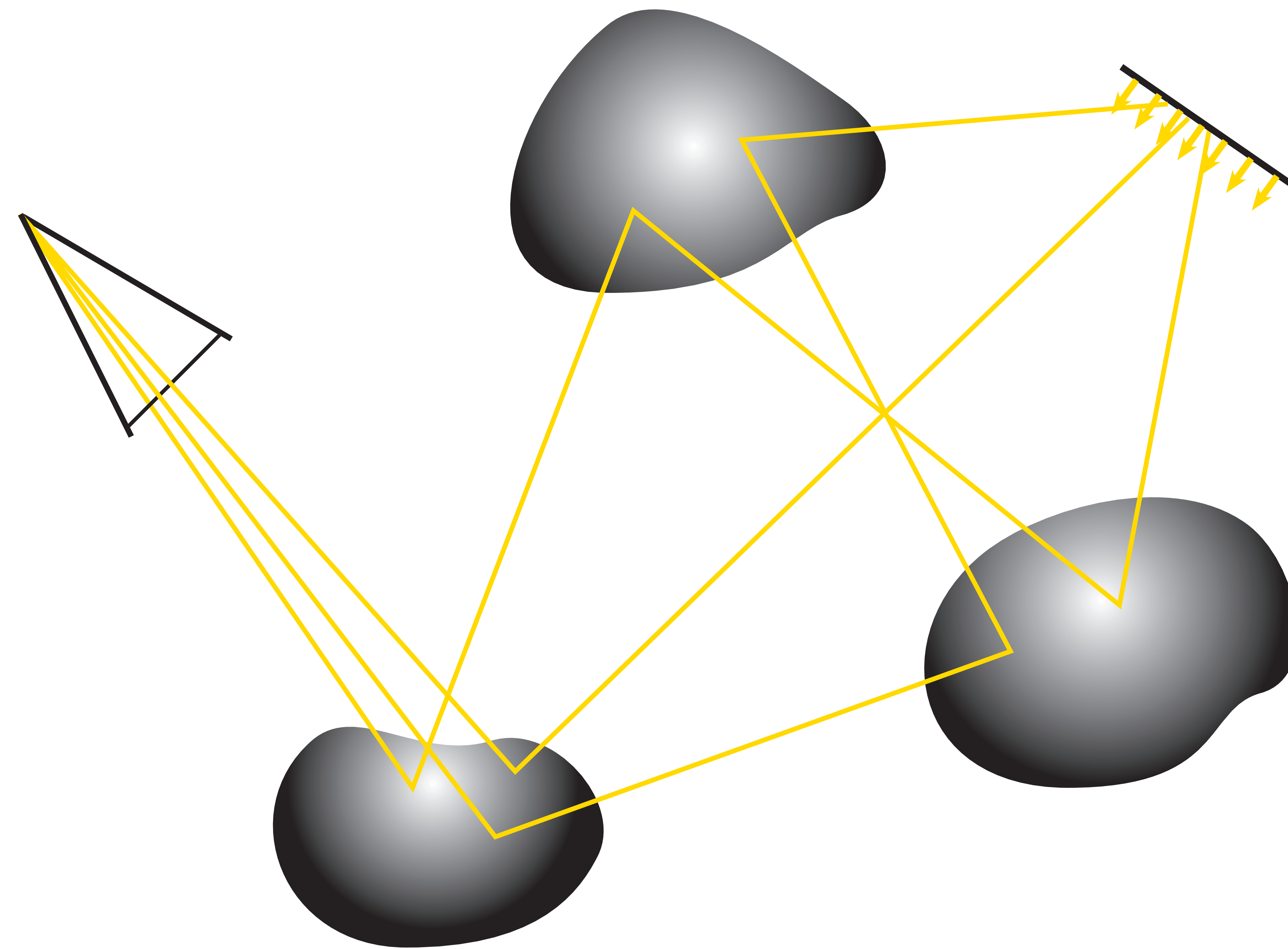
Cost per “pixel”



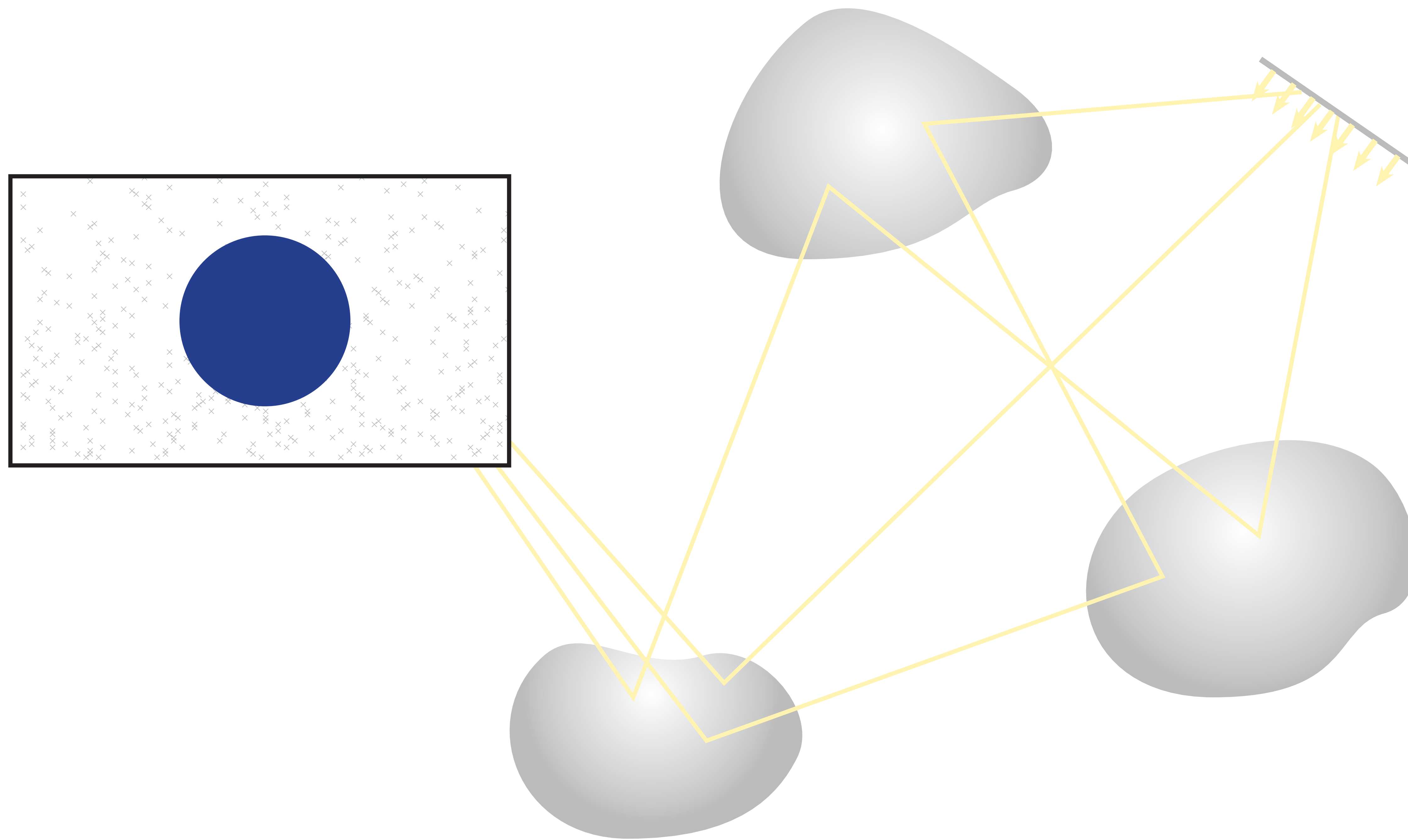
Cost per “pixel”



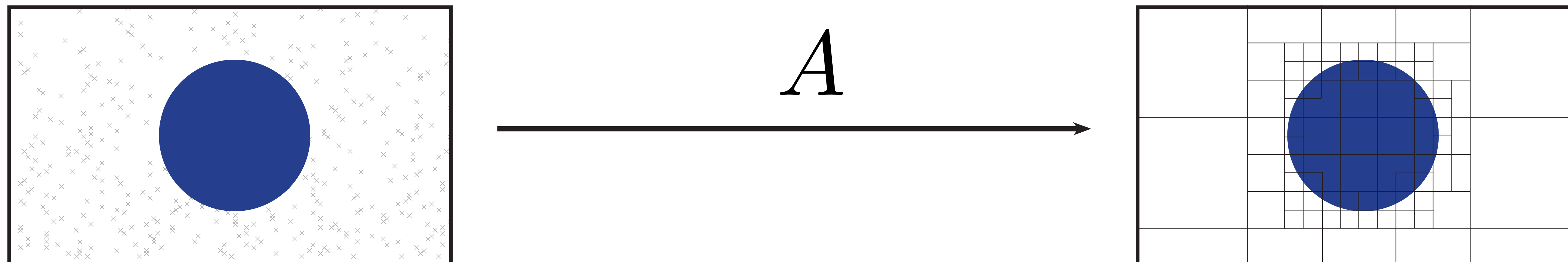
Cost per "pixel"



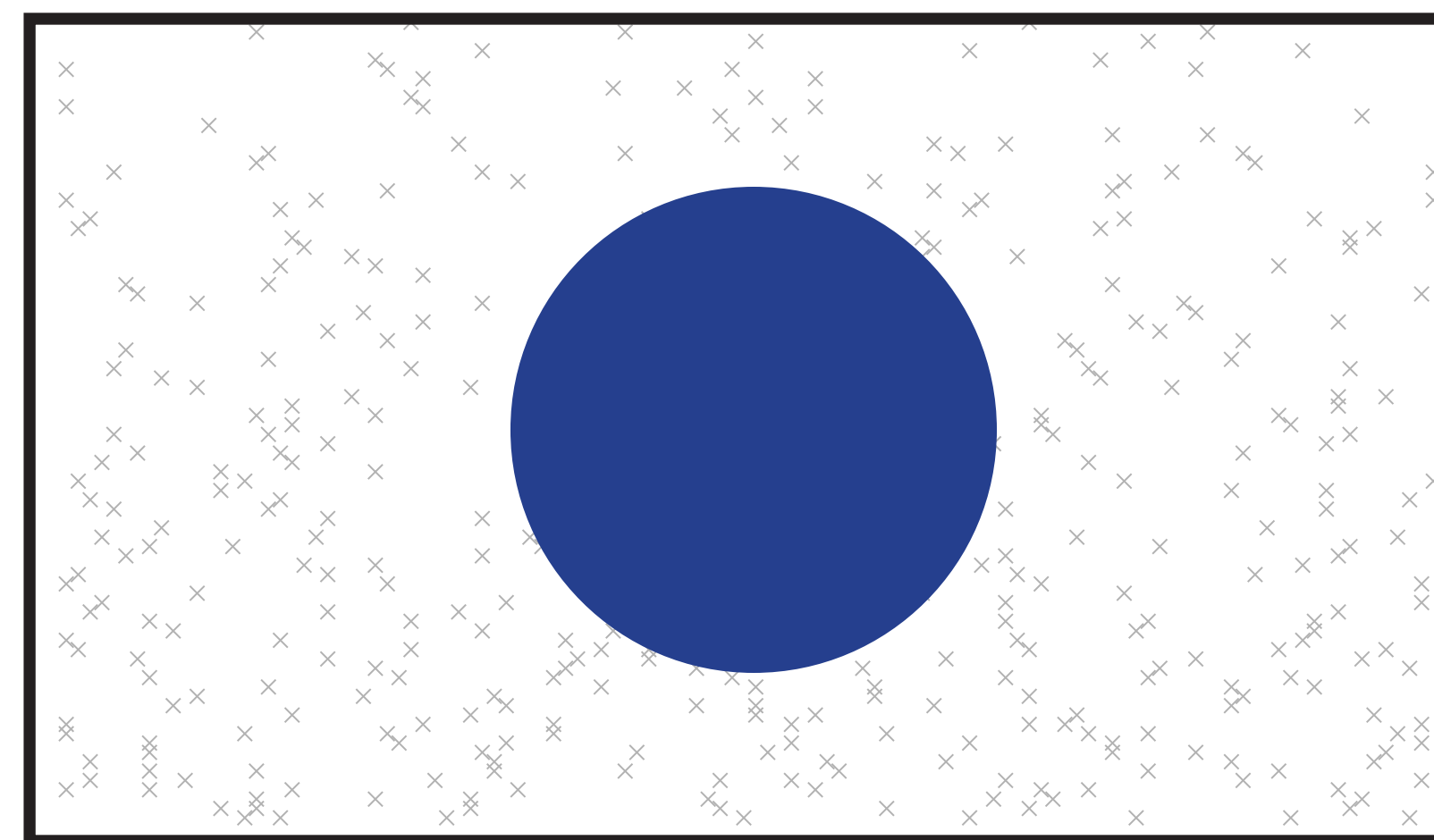
Cost per “pixel”



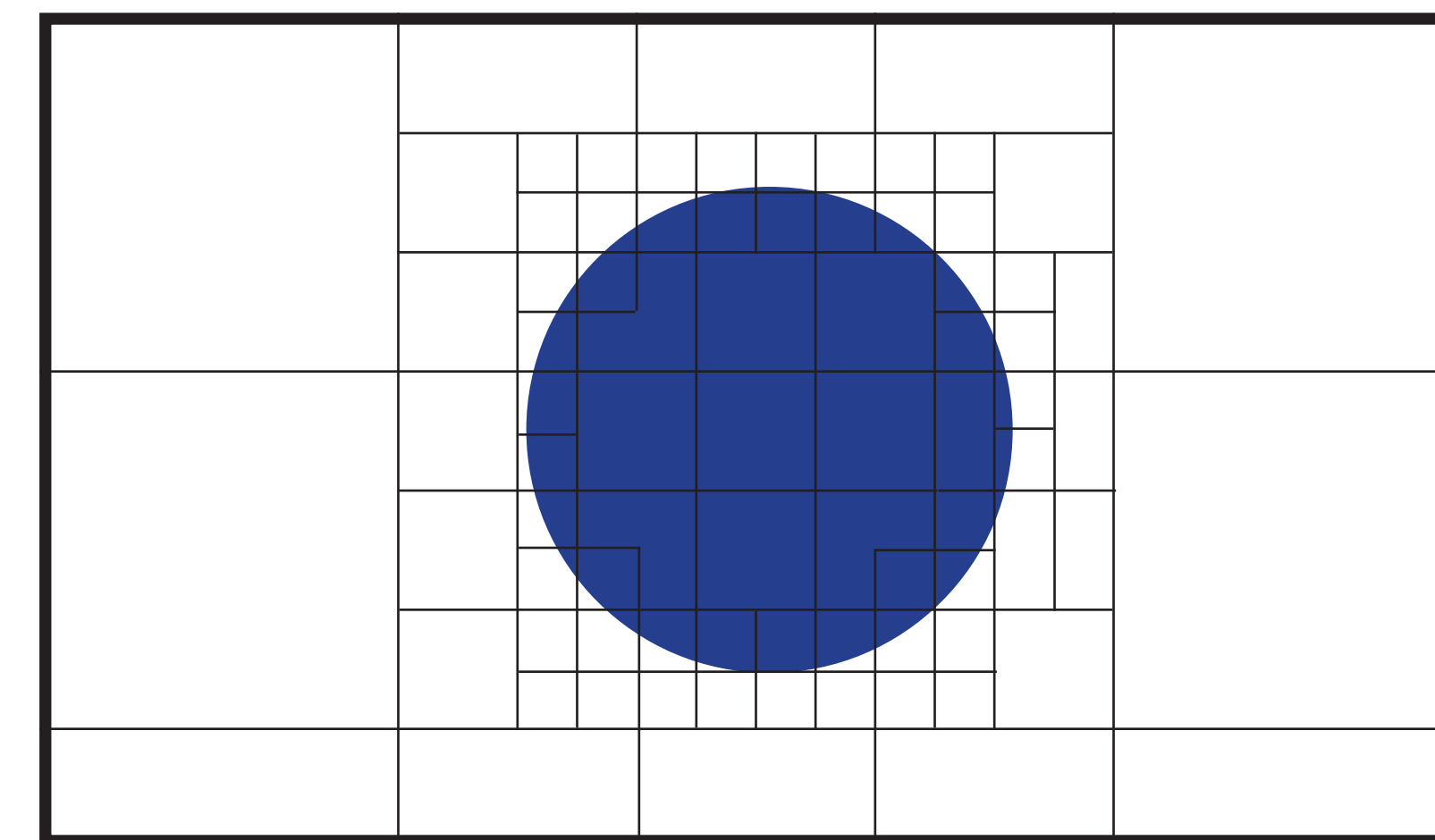
Cost per “pixel”



Cost per “pixel”

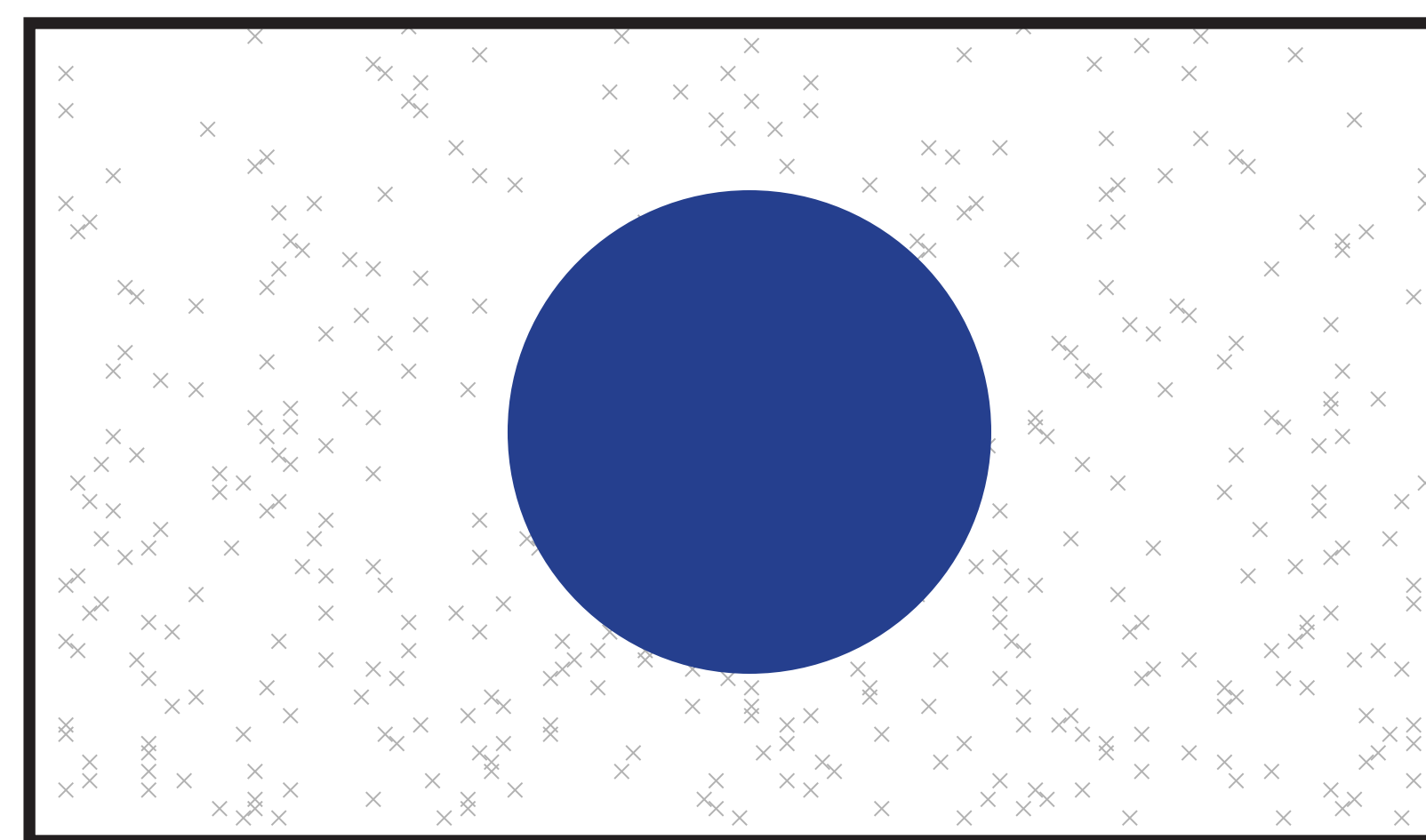


$$\{f(x_i)\} \in \mathbb{R}^m$$



$$\{f_i\} \in \mathbb{R}^n$$

Cost per “pixel”

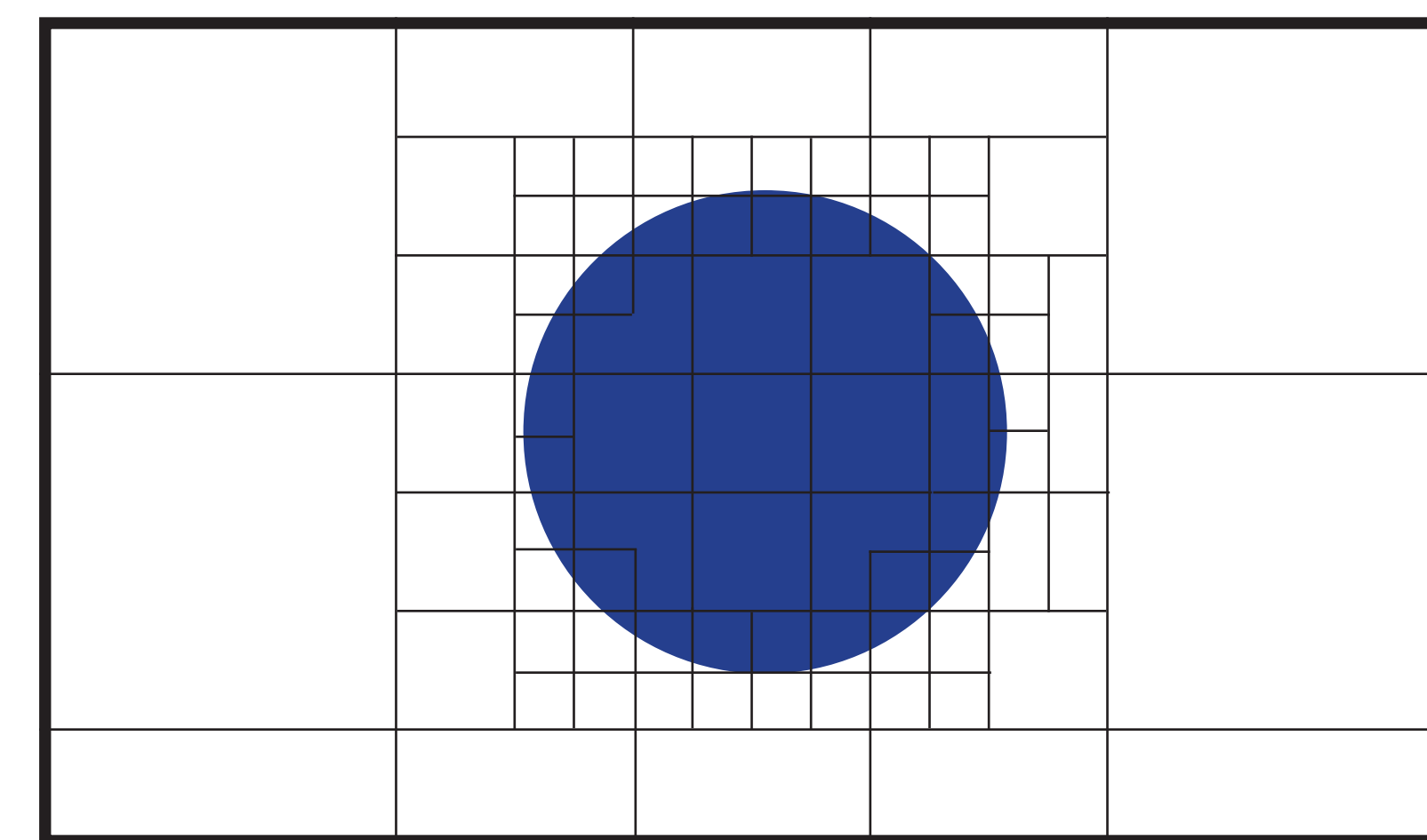


$$\{f(x_i)\} \in \mathbb{R}^m$$

A



$$f(x_j) = \sum_{i \in \mathcal{I}} f_i \phi_i(x_j)$$



$$\{f_i\} \in \mathbb{R}^n$$

Cost per “pixel”

$$f(x_j) = \sum_{i \in \mathcal{I}} f_i \phi_i(x_j)$$

Cost per “pixel”

$$f(x_j) = \sum_{i \in \mathcal{I}} f_i \phi_i(x_j)$$



$$\begin{pmatrix} f(x_1) \\ \vdots \\ f(x_m) \end{pmatrix} = \begin{pmatrix} \phi_1(x_1) & \phi_2(x_1) & \dots & & \\ & \phi_2(x_2) & \phi_3(x_1) & \dots & \\ & & \ddots & \ddots & \\ & & \dots & \phi_{n-1}(x_m) & \phi_n(x_m) \end{pmatrix} \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix}$$

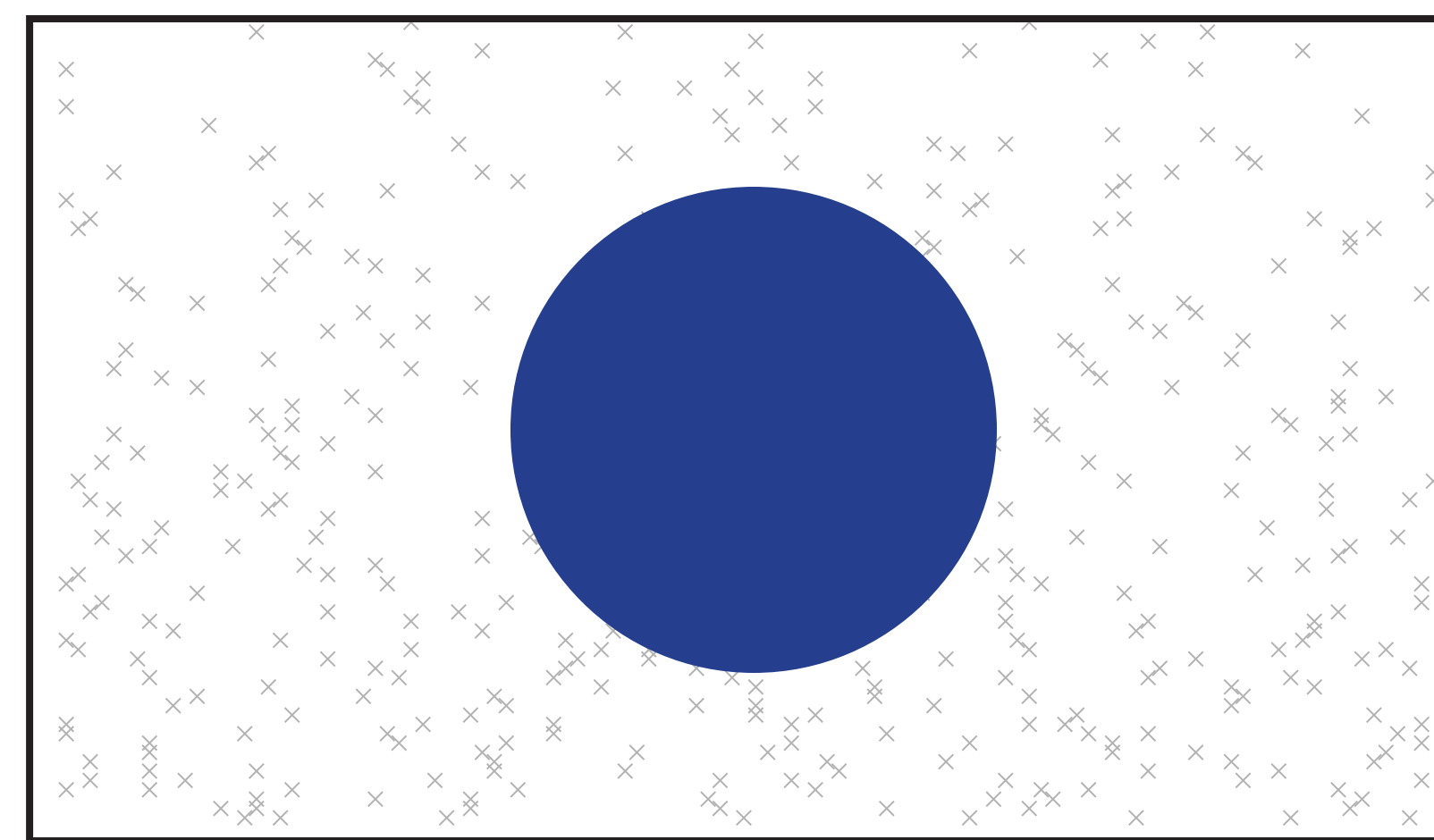
Cost per “pixel”

$$f(x_j) = \sum_{i \in \mathcal{I}} f_i \phi_i(x_j)$$



$$\underbrace{\begin{pmatrix} f(x_1) \\ \vdots \\ f(x_m) \end{pmatrix}}_{\bar{f}(x_i)} = \begin{pmatrix} \phi_1(x_1) & \phi_2(x_1) & \dots & & \\ & \phi_2(x_2) & \phi_3(x_1) & \dots & \\ & & \ddots & \ddots & \\ & & & \phi_{n-1}(x_m) & \phi_n(x_m) \end{pmatrix} \underbrace{\begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix}}_{\bar{f}_i}$$

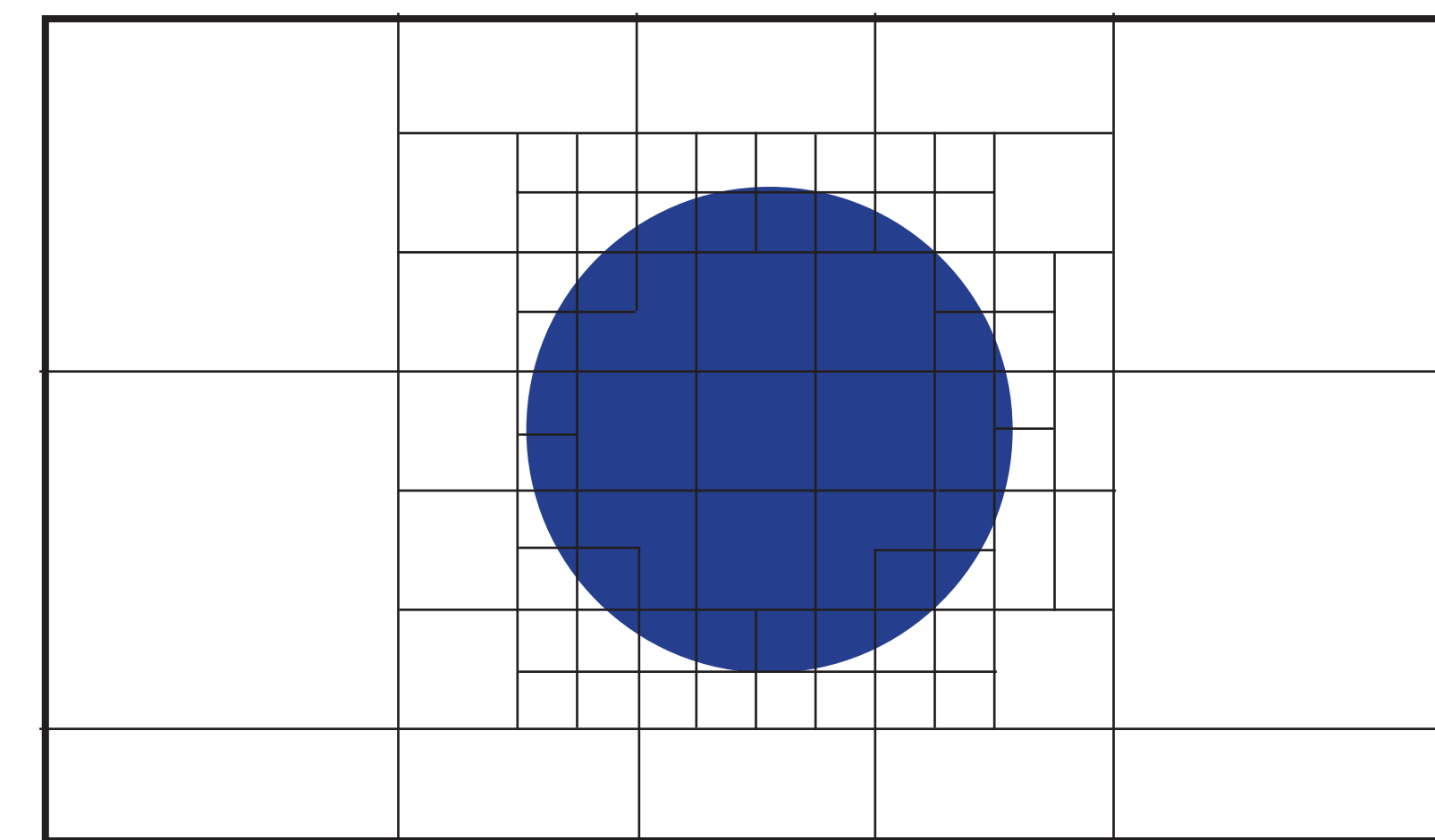
Cost per “pixel”



$$\{f(x_i)\} \in \mathbb{R}^m$$

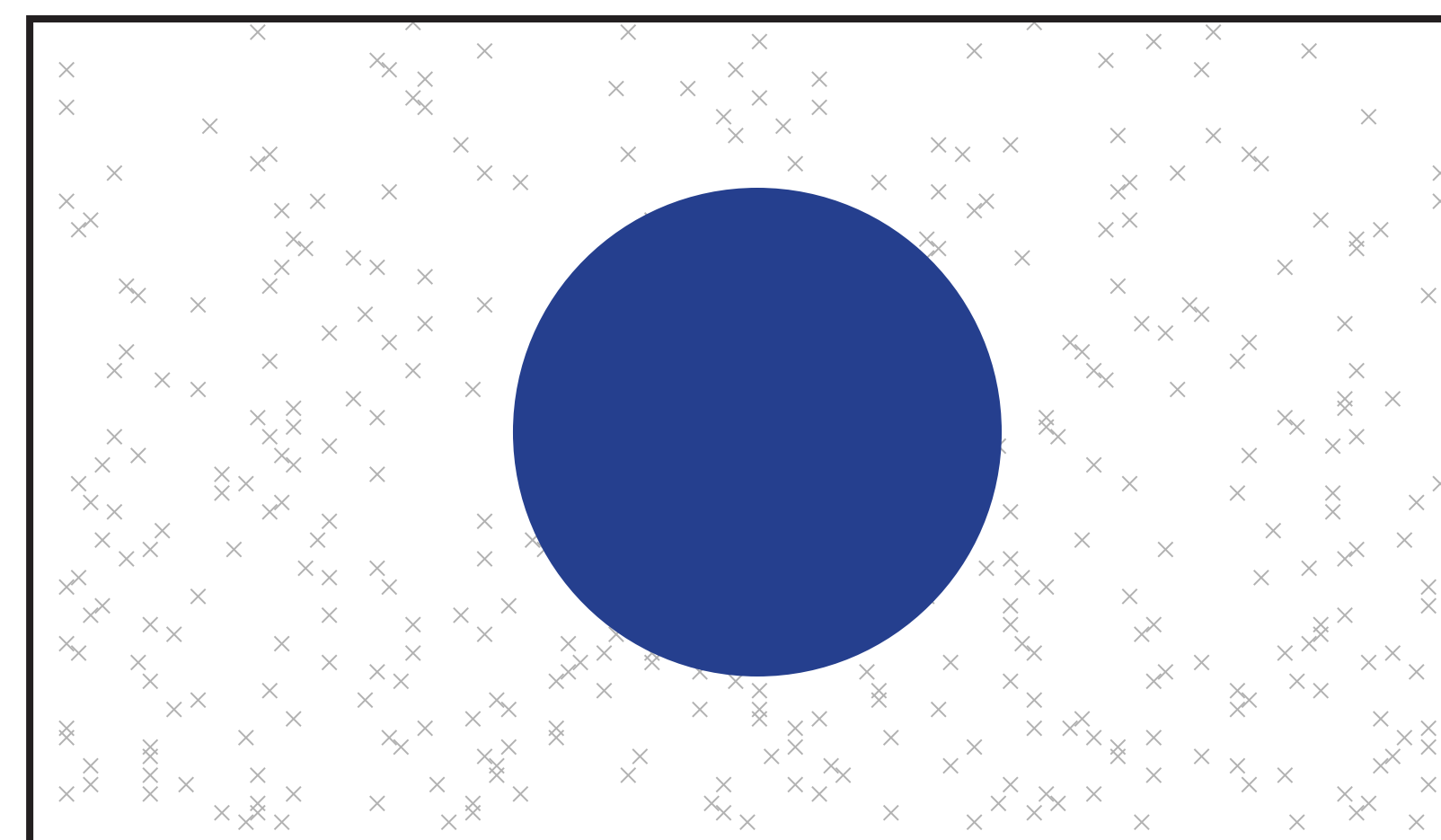


$$\bar{f}(x_i) = B \bar{f}_i$$



$$\{f_i\} \in \mathbb{R}^n$$

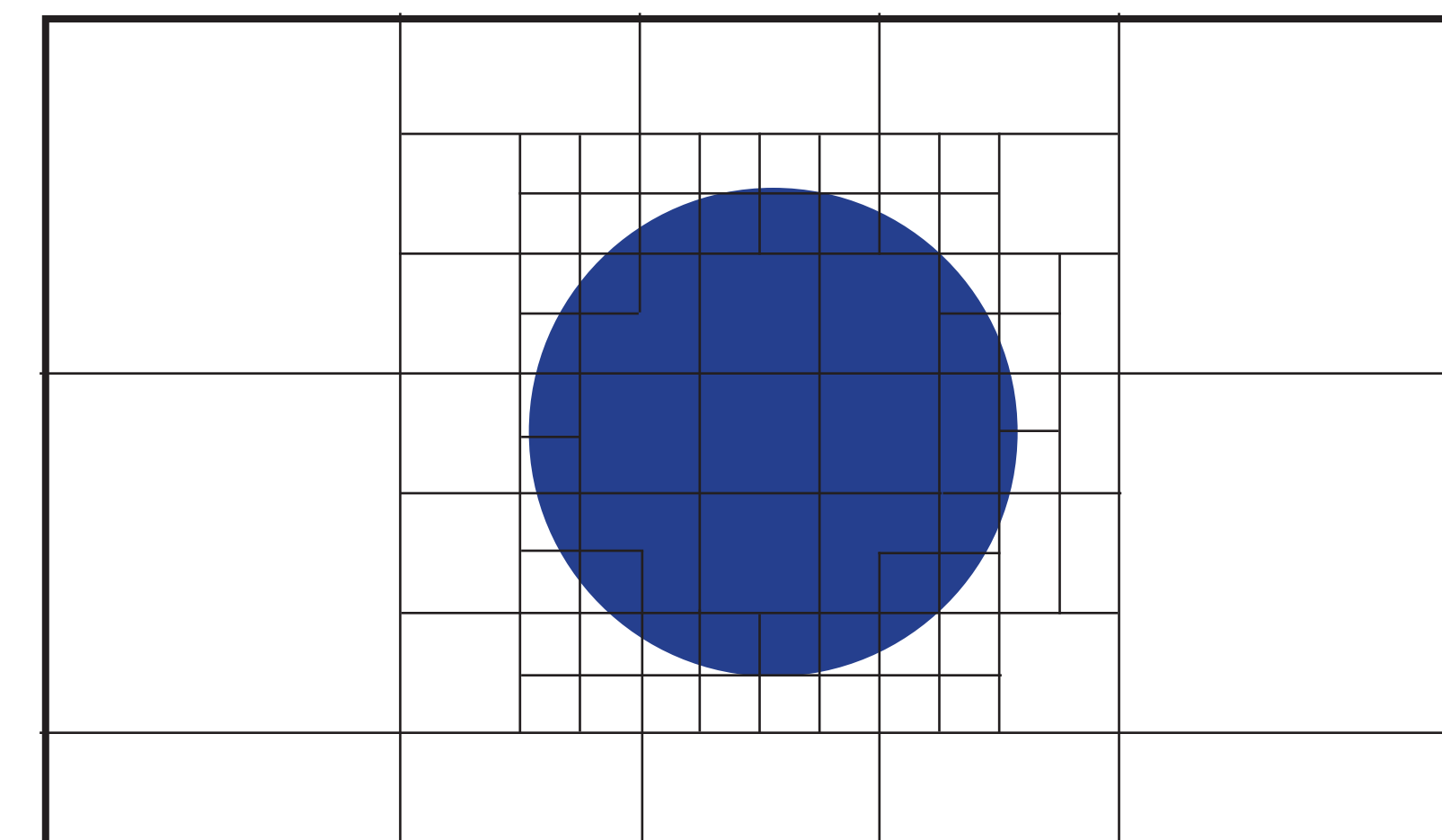
Cost per “pixel”



$$\{f(x_i)\} \in \mathbb{R}^m$$

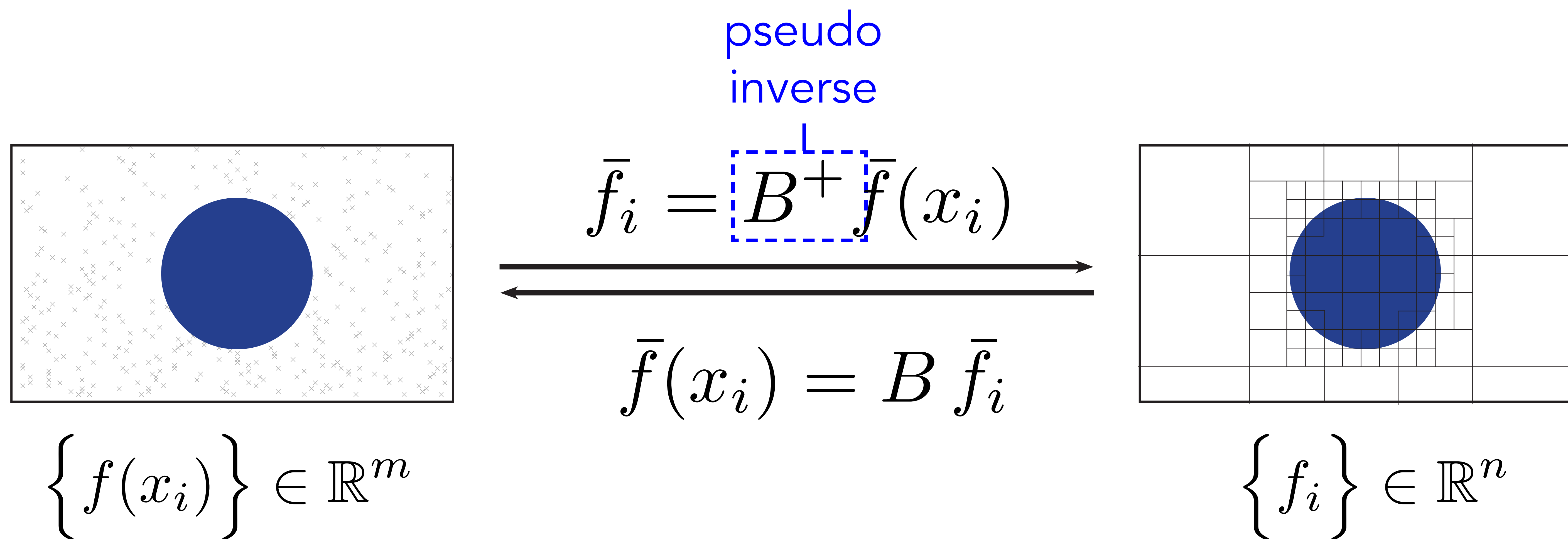
$$\bar{f}_i = B^+ \bar{f}(x_i)$$

$$\bar{f}(x_i) = B \bar{f}_i$$



$$\{f_i\} \in \mathbb{R}^n$$

Cost per “pixel”



Cost per “pixel”

Example: pixel basis

$$B = \begin{pmatrix} \phi_1(x_1) & \phi_2(x_1) & \dots & & \\ & \phi_2(x_2) & \phi_3(x_1) & \dots & \\ & & \ddots & \ddots & \\ & & & \phi_{n-1}(x_m) & \phi_n(x_m) \end{pmatrix}$$

Cost per “pixel”

Example: pixel basis

$$B = \begin{pmatrix} \chi_1(x_1) & \chi_2(x_1) & \dots & & \\ & \chi_2(x_2) & \chi_3(x_1) & \dots & \\ & & \ddots & \ddots & \\ & & & \chi_{n-1}(x_m) & \chi_n(x_m) \end{pmatrix}$$

Cost per “pixel”

Example: pixel basis

$$B = \begin{pmatrix} 1 & 0 & \dots & & \\ 1 & 0 & \dots & & \\ & & \ddots & \ddots & \\ & & & 0 & 1 \\ & & & 0 & 1 \end{pmatrix}$$

Cost per “pixel”

Example: pixel basis

$$B = \begin{pmatrix} \text{gray} & 0 & \dots & & & \\ & 0 & \dots & & & \\ & & \text{gray} & & & \\ & & & \cdot & \cdot & \cdot \\ & & & & \cdot & \cdot \\ & & & & & 0 \\ & & & & & 0 & \text{gray} \end{pmatrix}$$

Cost per “pixel”

Example: pixel basis

$$B^+ = \begin{pmatrix} \text{[gray box]} & \dots & & \\ & \text{[gray box]} & & \\ & & \ddots & \\ & & & \text{[gray box]} \end{pmatrix}$$

Cost per “pixel”

Example: pixel basis

$$B^+ = \begin{pmatrix} 1/N_1 & 1/N_1 & \dots & & & \\ & 1/N_2 & 1/N_2 & & & \\ & & \ddots & \ddots & & \\ & & & 1/N_n & 1/N_n & \end{pmatrix}$$

Cost per “pixel”

Example: pixel basis

$$B^+ = \begin{pmatrix} 1/N_1 & 1/N_1 & \dots & & \\ & 1/N_2 & 1/N_2 & & \\ & & \ddots & \ddots & \\ & & & 1/N_n & 1/N_n \end{pmatrix}$$

$$\Rightarrow f_i = \frac{1}{N_i} \sum_{j=1}^{N_i} f(x_j)$$

Cost per “pixel”

Example: wavelet basis

$$B = \begin{pmatrix} \psi_1(x_1) & \psi_2(x_1) & \dots & & \\ & \psi_2(x_2) & \psi_3(x_1) & \dots & \\ & & \ddots & \ddots & \\ & & \dots & \psi_{n-1}(x_m) & \psi_n(x_m) \end{pmatrix}$$

Cost per “pixel”

Example: wavelet basis

$$B = \left(\begin{array}{c} \text{[Diagram of three wavelet basis functions: a vertical line, a diagonal line, and a horizontal line]} \end{array} \right)$$

Cost per “pixel”

Example: wavelet basis

$$B = \begin{pmatrix} \text{---} \end{pmatrix}$$

$$B^+ \not\approx f_i = \frac{1}{N} \sum_{i=1}^N f(x_i) \psi(x_i)$$

Cost per “pixel”

Example: wavelet basis

$$B = \begin{pmatrix} \text{---} & & & \\ & \text{---} & & \\ & & \text{---} & \\ & & & \text{---} \end{pmatrix}$$

Multi-res structure: can be solved in $O(n)$ time using multi-grid.

Cost per “pixel”

Example: wavelet basis

$$B = \begin{pmatrix} \text{---} & & & \\ & \text{---} & & \\ & & \text{---} & \\ & & & \text{---} \end{pmatrix}$$

Multi-res structure: can be solved in $\boxed{O(n)}$ time using multi-grid.
optimal

Computational costs

$$C = N \cdot C_p$$

/ \

Number of Cost per

"pixels" pixel

↓ minimize by

- Use sparse, adaptive image representation

Computational costs

$$C = N \cdot C_p$$

/ \

Number of Cost per

"pixels" pixel

minimize by

- Use sparse, adaptive image representation
- Use at least n samples
- Solve using multi-grid

Computational costs

$$C = N \cdot C_p$$

/ \

Number of Cost per

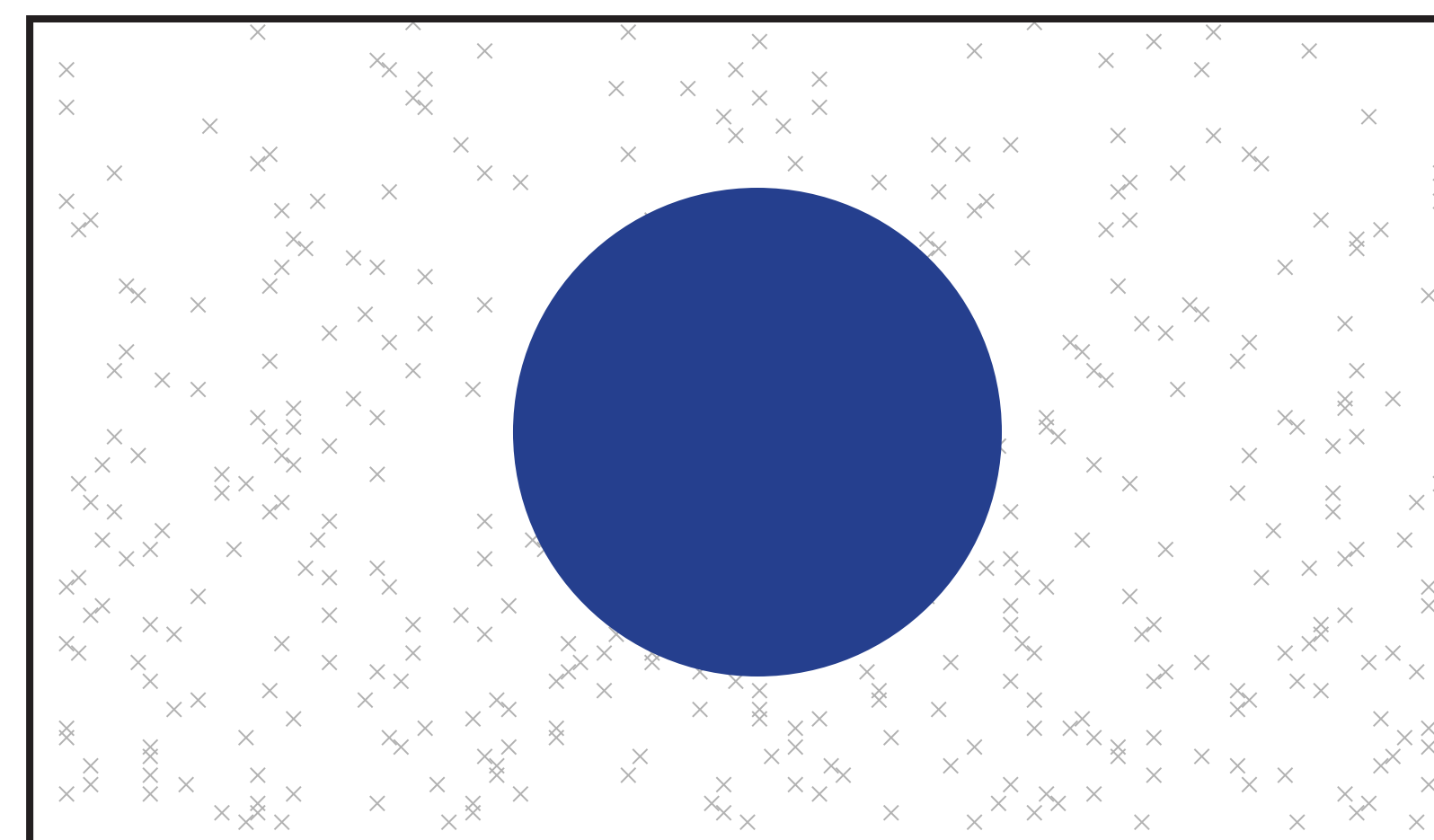
"pixels" pixel

minimize by

- Use sparse, adaptive image representation
- Use at least n samples
- Solve using multi-grid

**last
bounce**

Another perspective

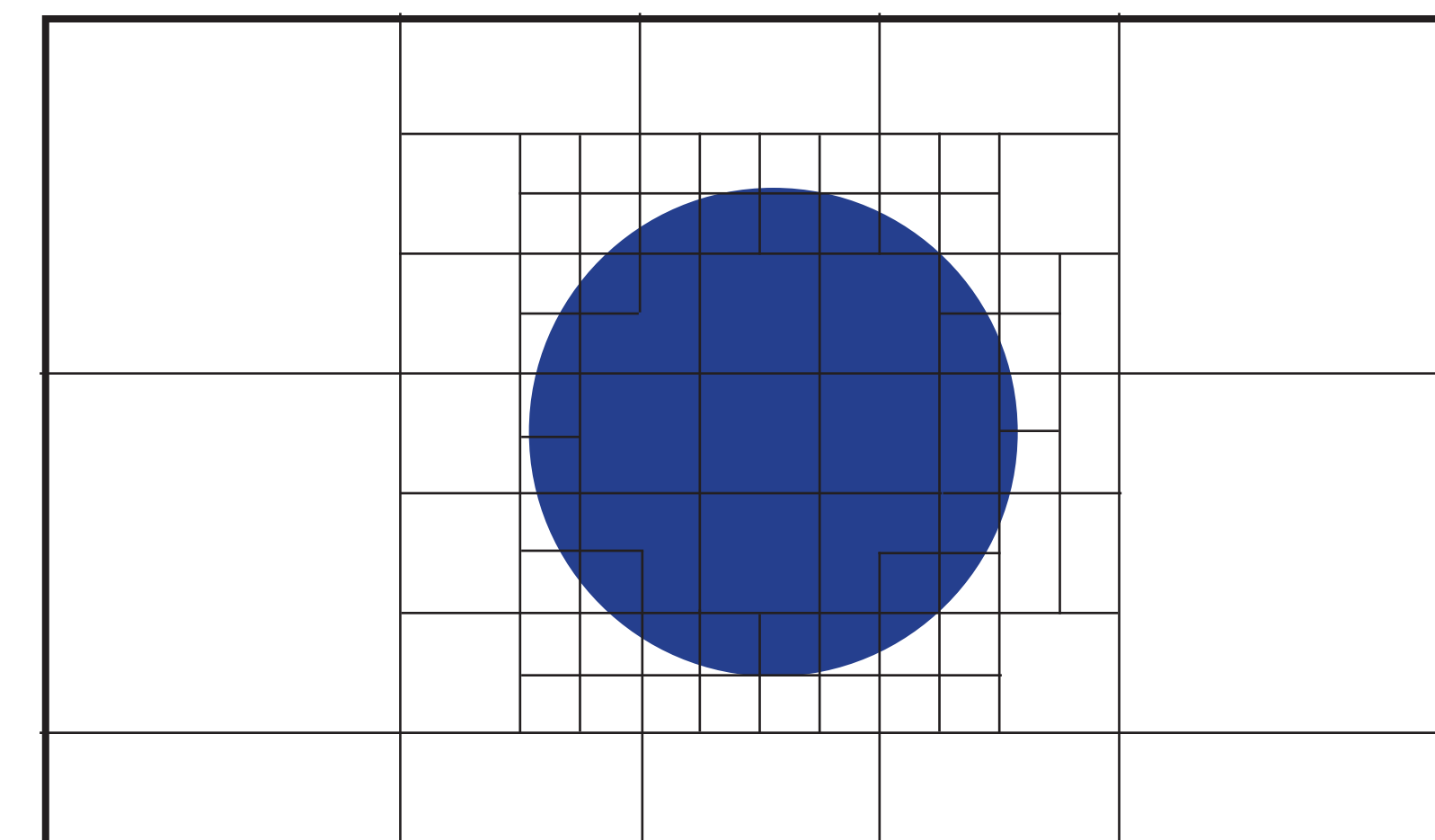


$$\{f(x_i)\} \in \mathbb{R}^m$$

$$\bar{f}_i = B^+ \bar{f}(x_i)$$

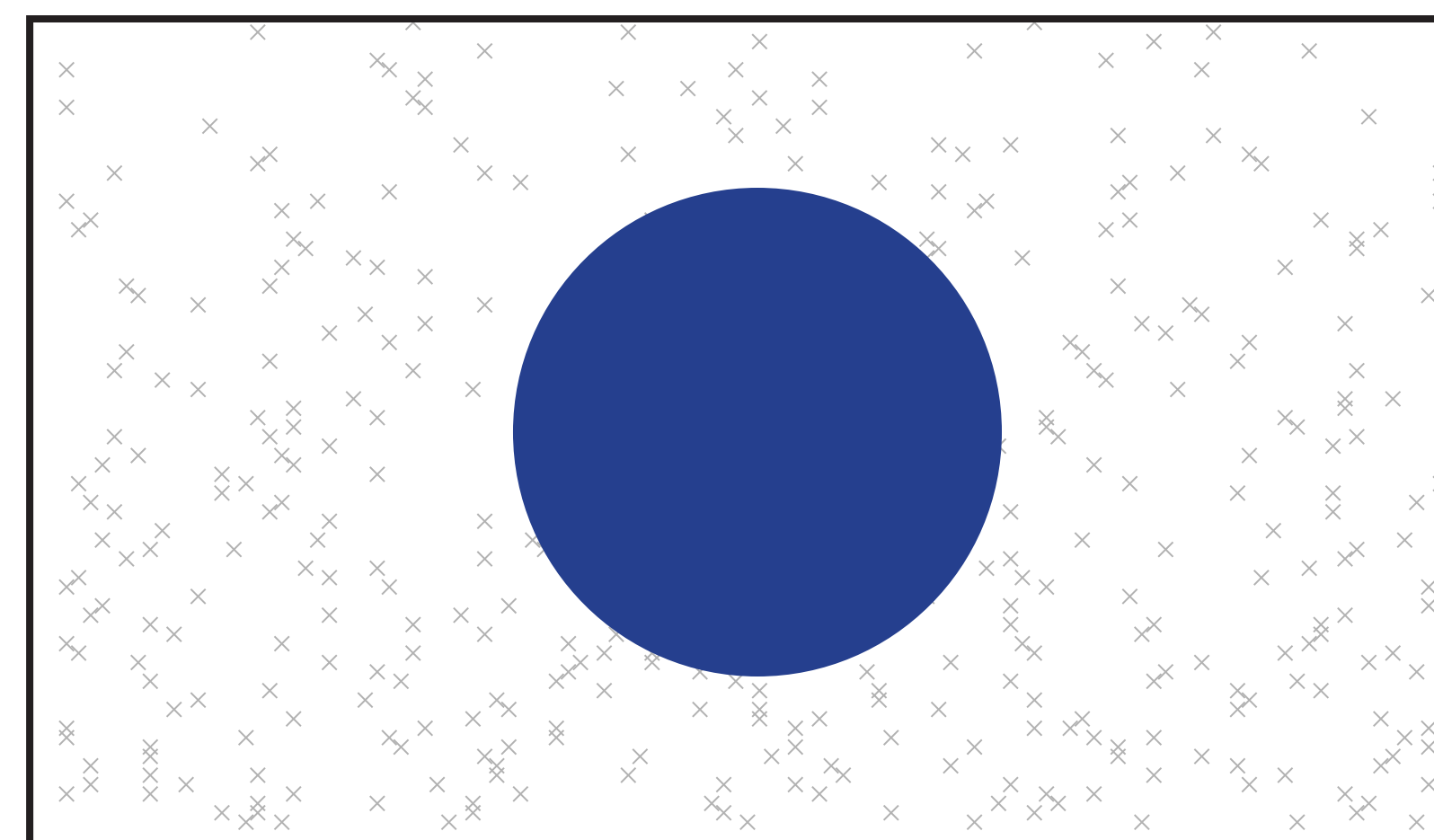


$$\bar{f}(x_i) = B \bar{f}_i$$



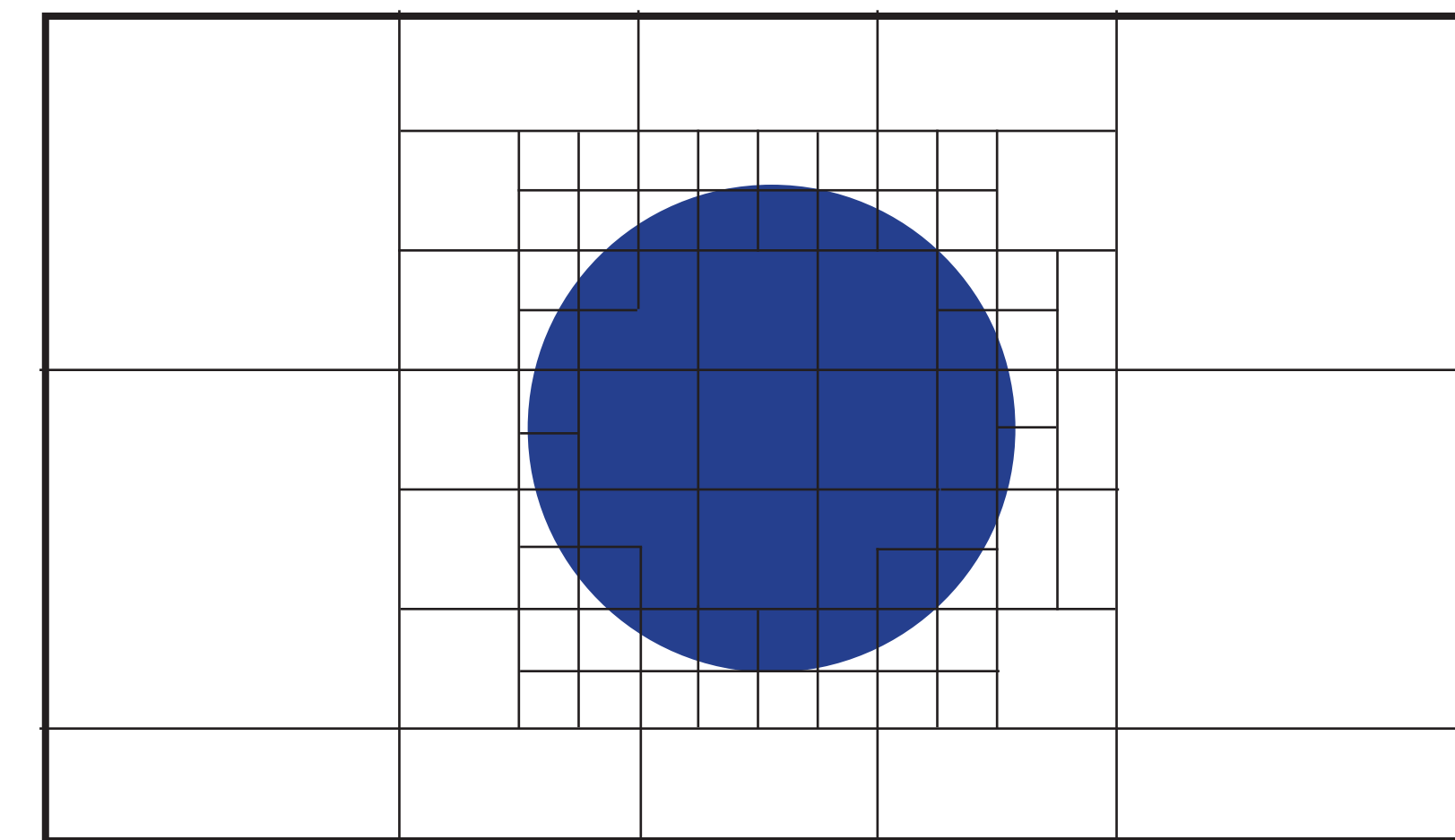
$$\{f_i\} \in \mathbb{R}^n$$

Another perspective



$$\boxed{\{f(x_i)\}} \in \mathbb{R}^m$$

$$\begin{aligned} \bar{f}_i &= B^+ \bar{f}(x_i) \\ \bar{f}(x_i) &= B \bar{f}_i \end{aligned}$$



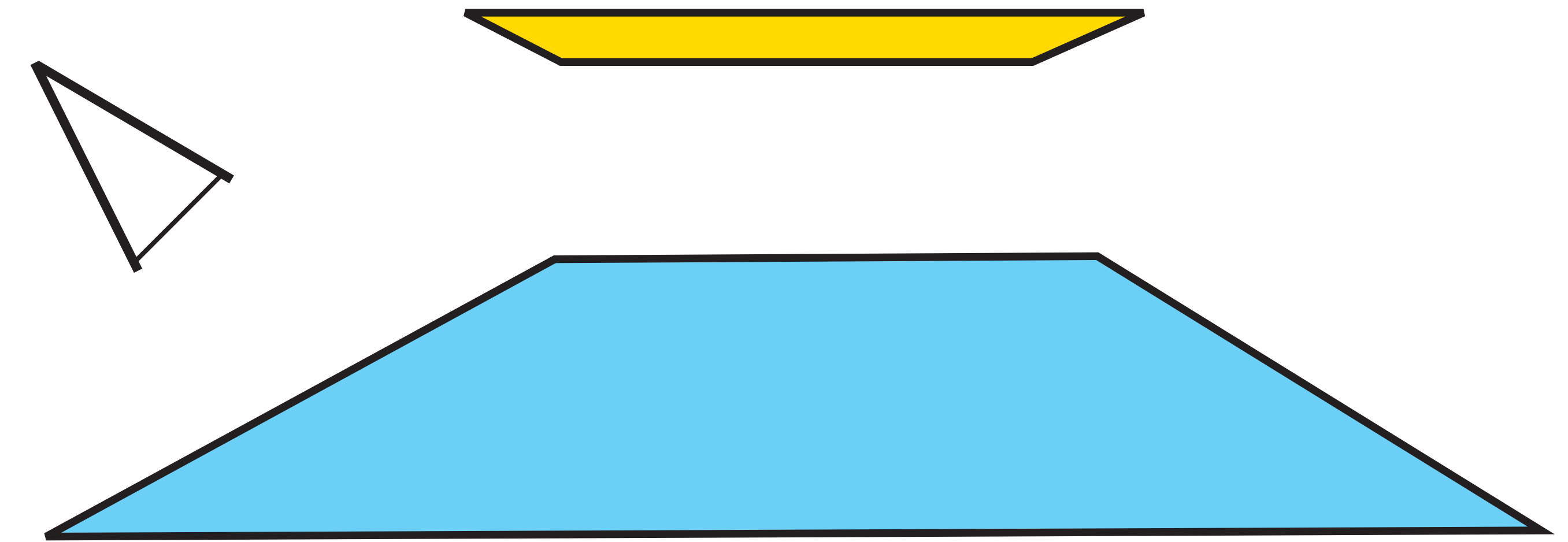
$$\{f_i\} \in \mathbb{R}^n$$

Another perspective

$$\left\{ f(x_i) \right\} = \left\{ \delta_{x_i}(f) \right\}$$

Another perspective

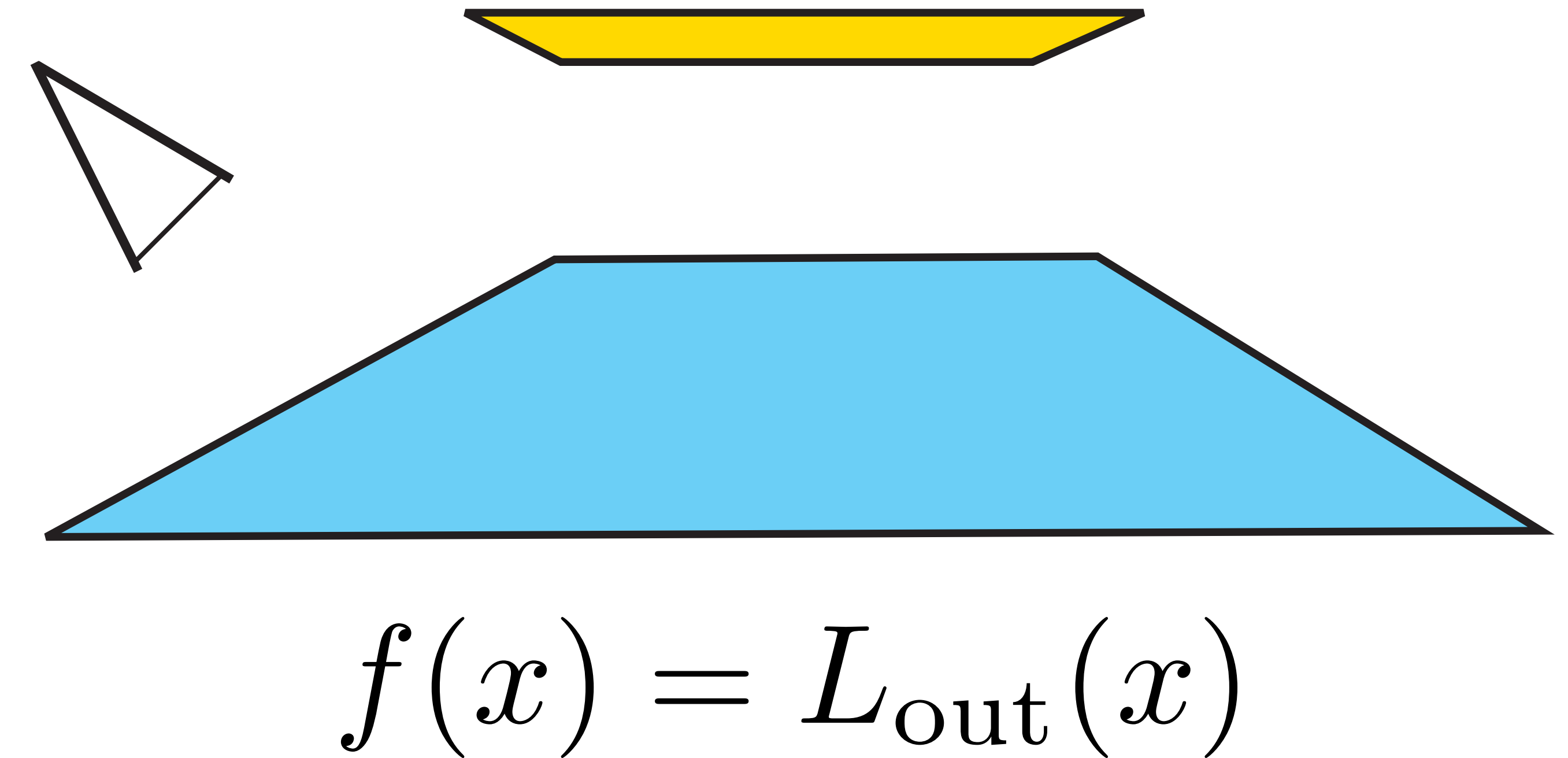
$$\left\{ f(x_i) \right\} = \left\{ \delta_{x_i}(f) \right\}$$



$$f(x) = L_{\text{out}}(x)$$

Another perspective

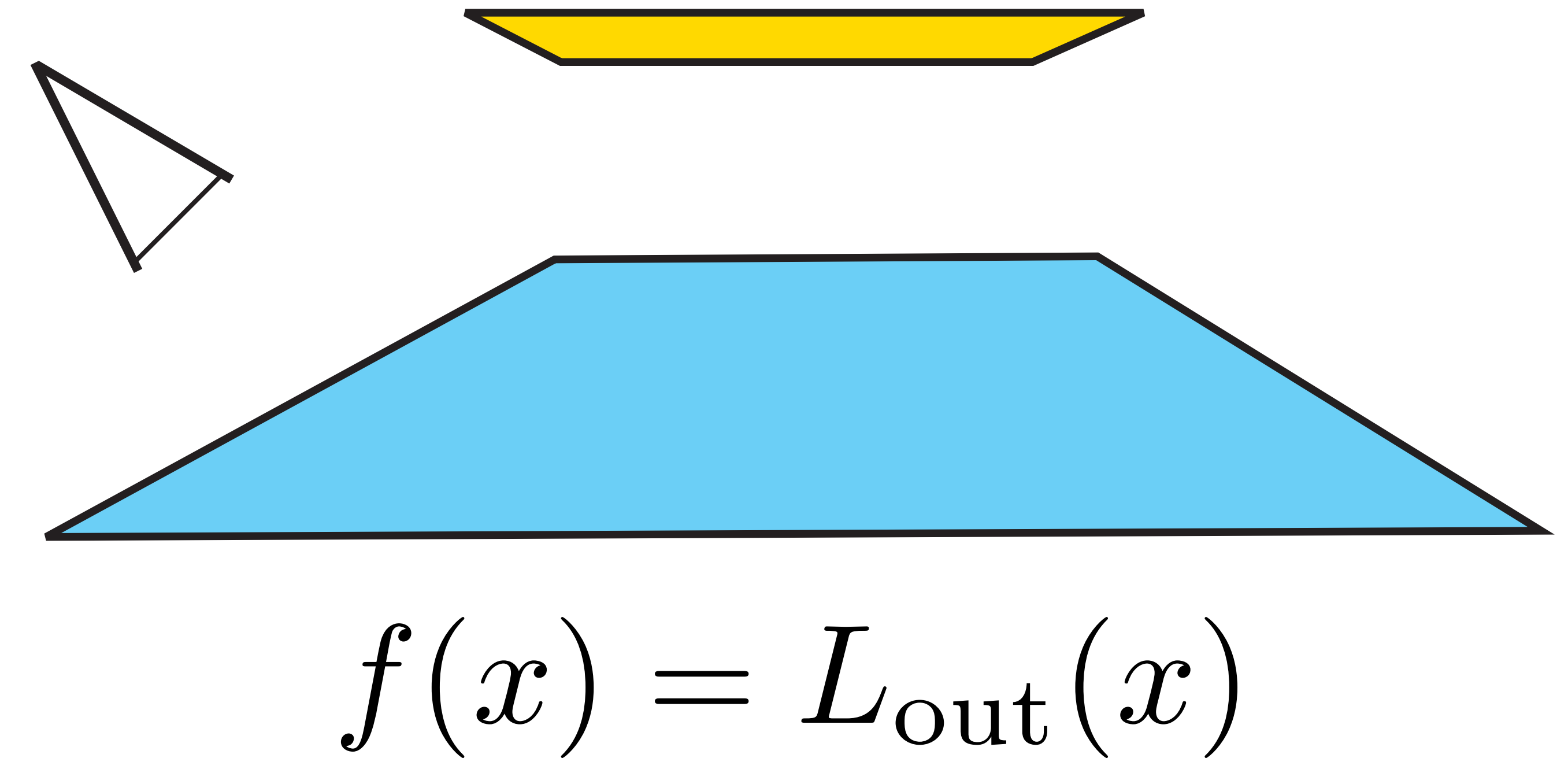
$$\underbrace{\left\{ f(x_i) \right\}}_{f = \hat{f} + \epsilon} = \left\{ \delta_{x_i}(f) \right\}$$



Another perspective

$$\underbrace{\left\{ f(x_i) \right\}}_{f = \hat{f} + \epsilon} = \left\{ \delta_{x_i}(f) \right\}$$

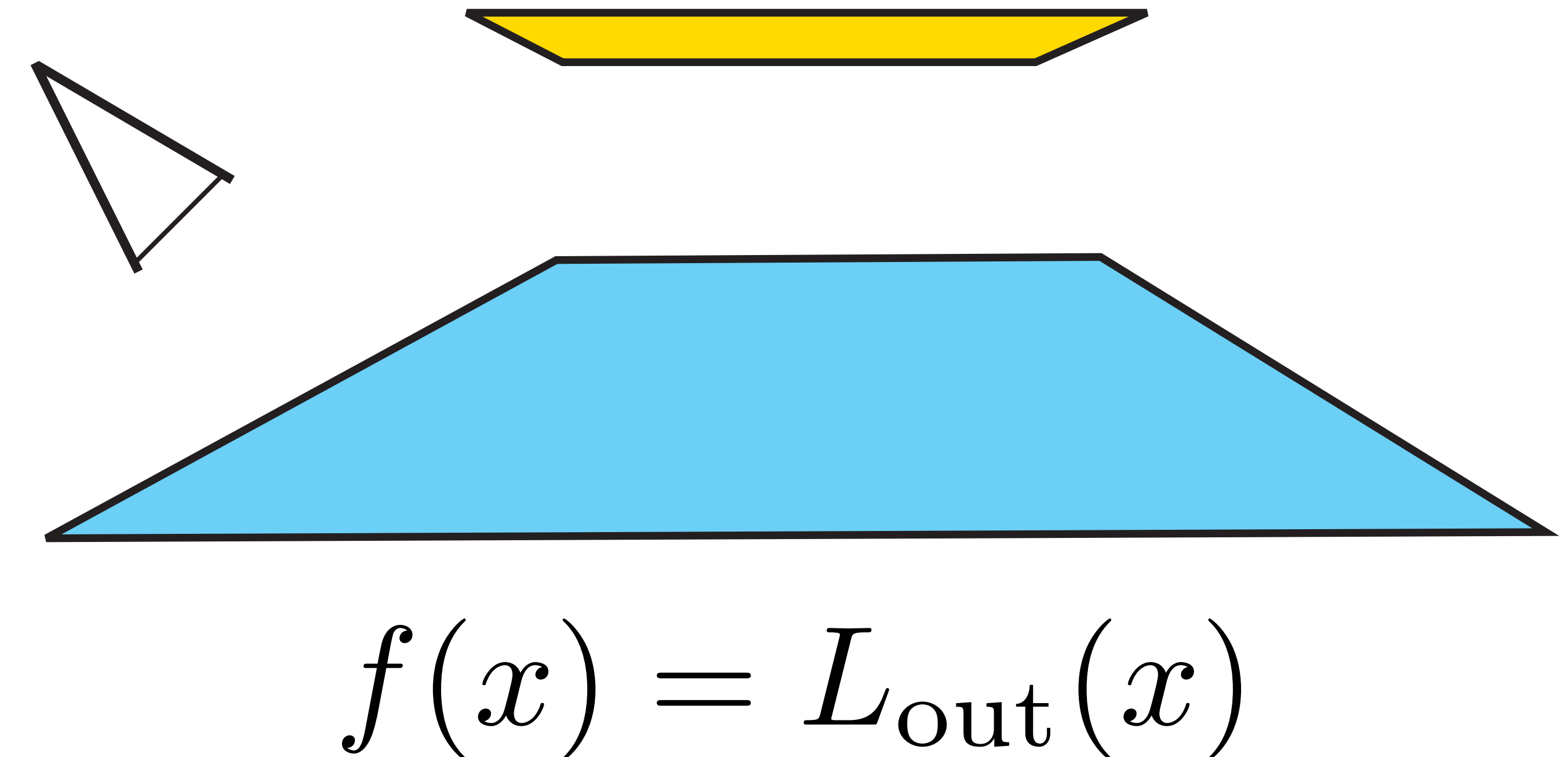
$$\Rightarrow \delta_{x_i}(f) \approx \delta_{x_i}(\hat{f})$$



Another perspective

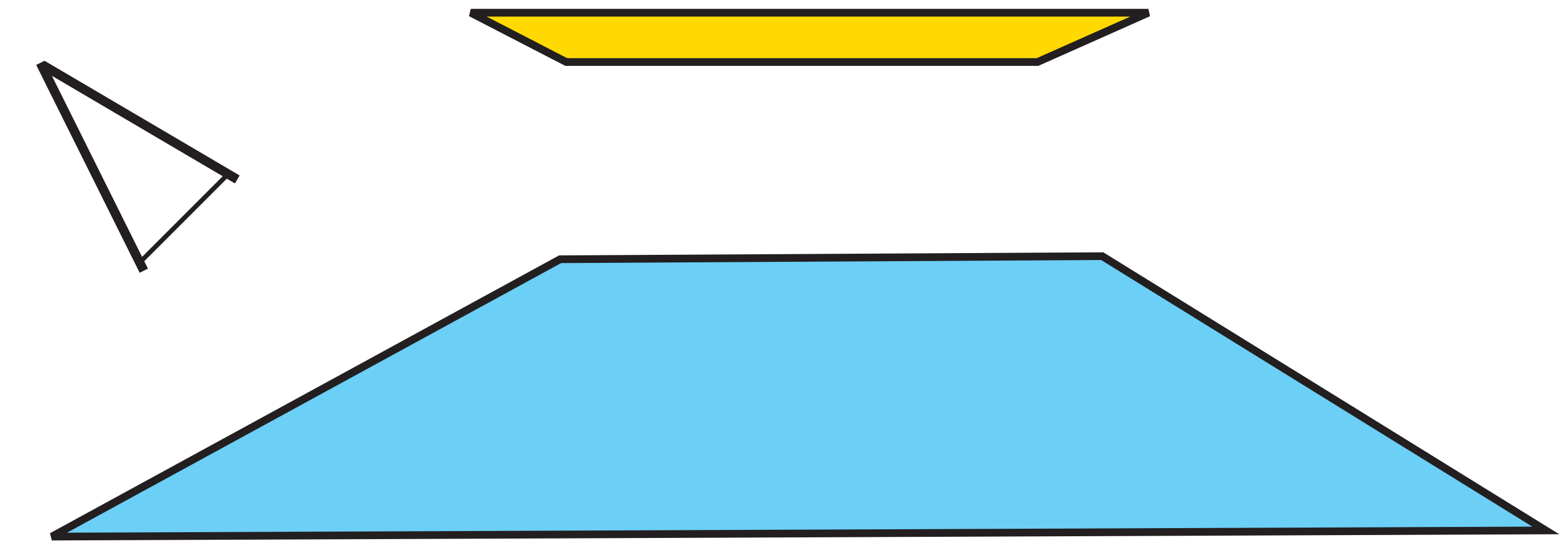
$$\underbrace{\left\{ f(x_i) \right\}}_{f = \hat{f} + \epsilon} = \left\{ \delta_{x_i}(f) \right\}$$

$$\Rightarrow \boxed{\delta_{x_i}(f) \approx \delta_{x_i}(\hat{f})} \quad \delta_x \text{ is a continuous functional}$$



Another perspective

$$\underbrace{\left\{ f(x_i) \right\}}_{f = \hat{f} + \epsilon} = \left\{ \delta_{x_i}(f) \right\}$$



$$f(x) = L_{\text{out}}(x)$$

$$\Rightarrow \boxed{\delta_{x_i}(f) \approx \delta_{x_i}(\hat{f})} \quad \delta_x \text{ is a continuous functional}$$

$$\Rightarrow f \in \mathcal{H} \text{ with } \mathcal{H} \text{ a "reasonable" function space}$$

Reproducing kernel Hilbert spaces

$\mathcal{H}(X)$ is Hilbert and $\delta_{x_i}(f)$ continuous, then

Reproducing kernel Hilbert spaces

$\mathcal{H}(X)$ is Hilbert and $\delta_{x_i}(f)$ continuous, then

$$\forall \bar{x} \in X, \exists k_{\bar{x}}(x) \in \mathcal{H}(X) : \left\langle k_{\bar{x}}(x), f(x) \right\rangle = f(\bar{x})$$

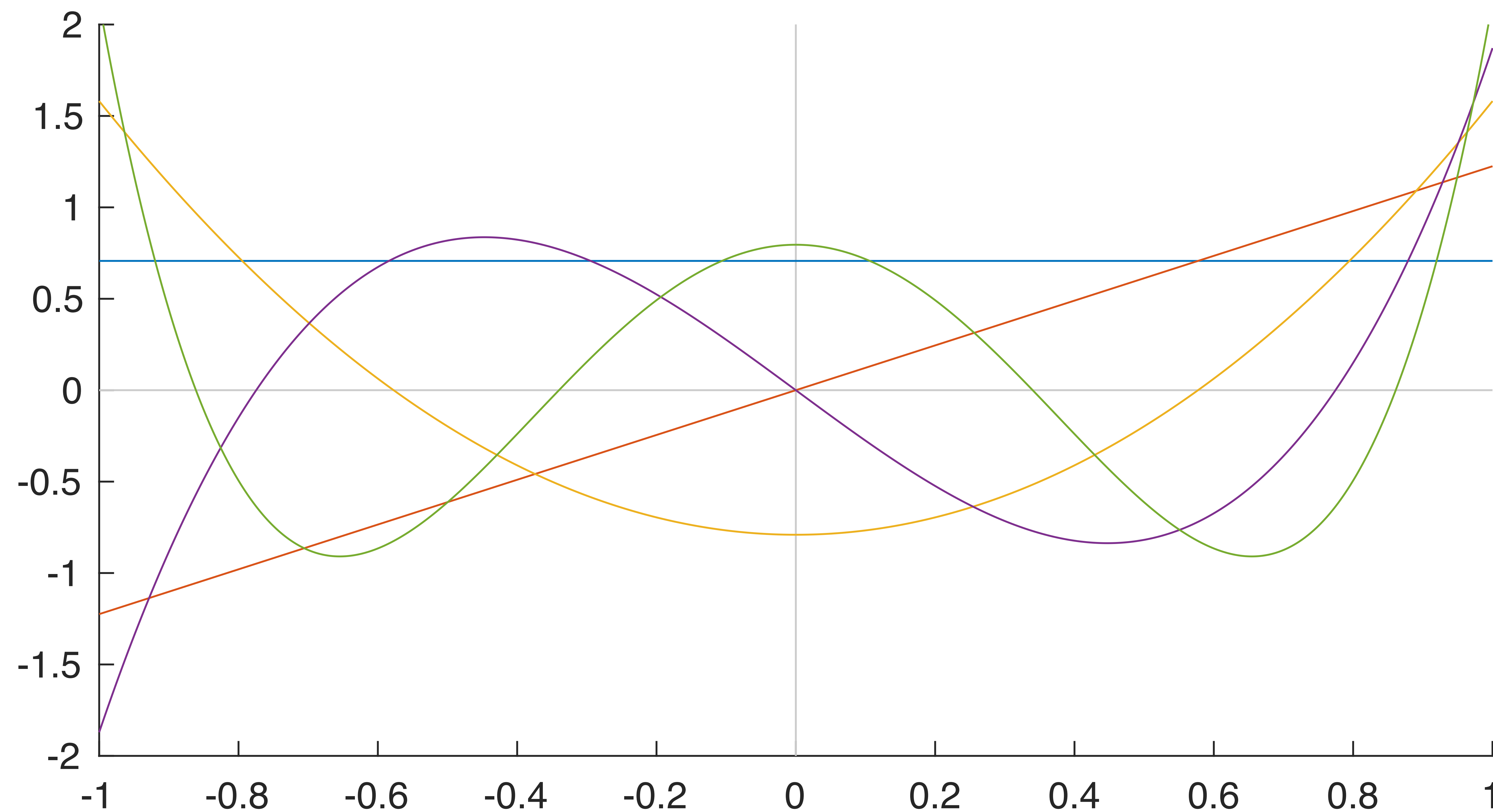
Reproducing kernel Hilbert spaces

$\mathcal{H}(X)$ is Hilbert and $\delta_{x_i}(f)$ continuous, then

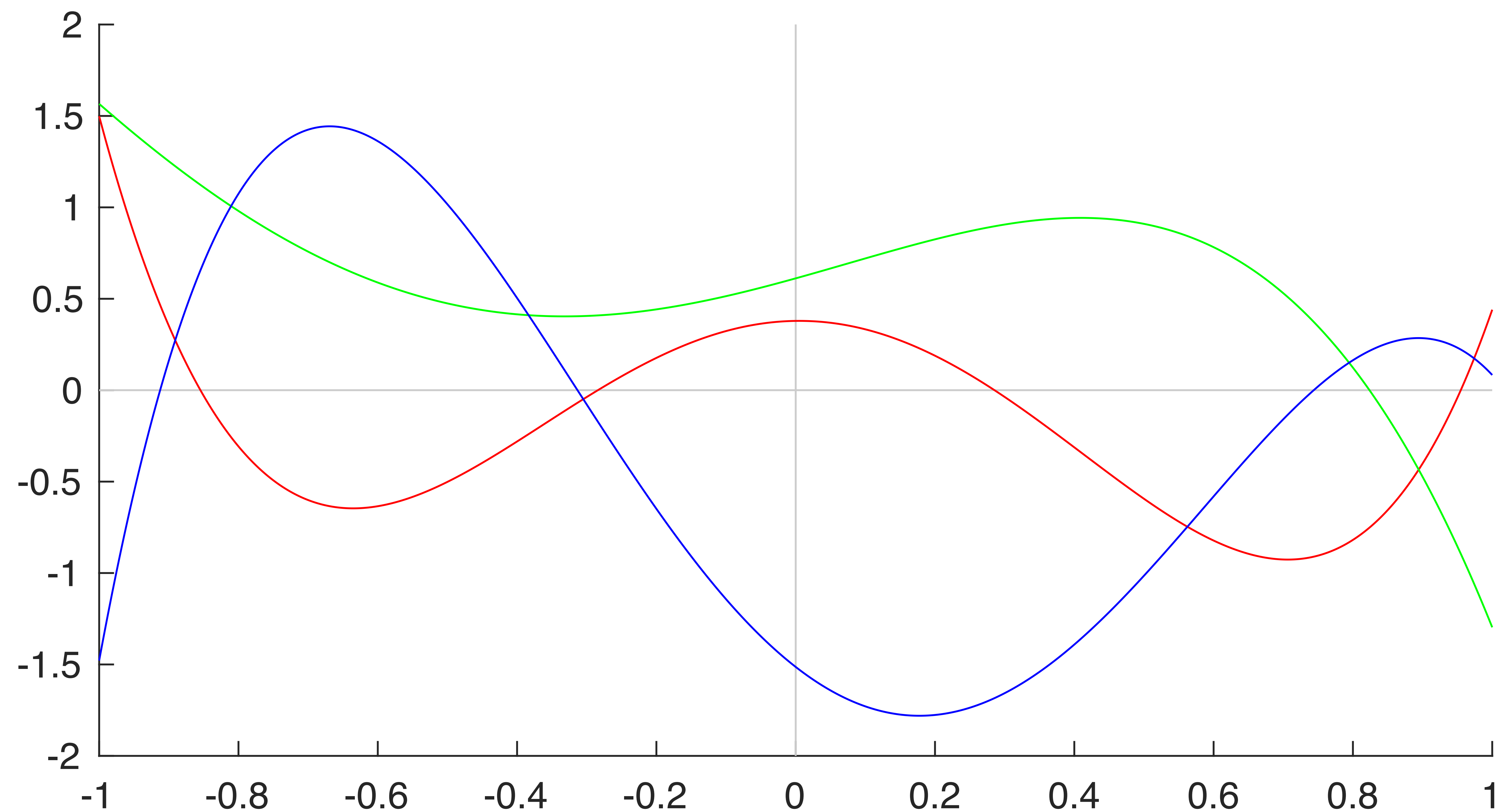
$$\forall \bar{x} \in X, \exists k_{\bar{x}}(x) \in \mathcal{H}(X) : \left\langle \boxed{k_{\bar{x}}(x)}, f(x) \right\rangle = f(\bar{x})$$

reproducing kernel

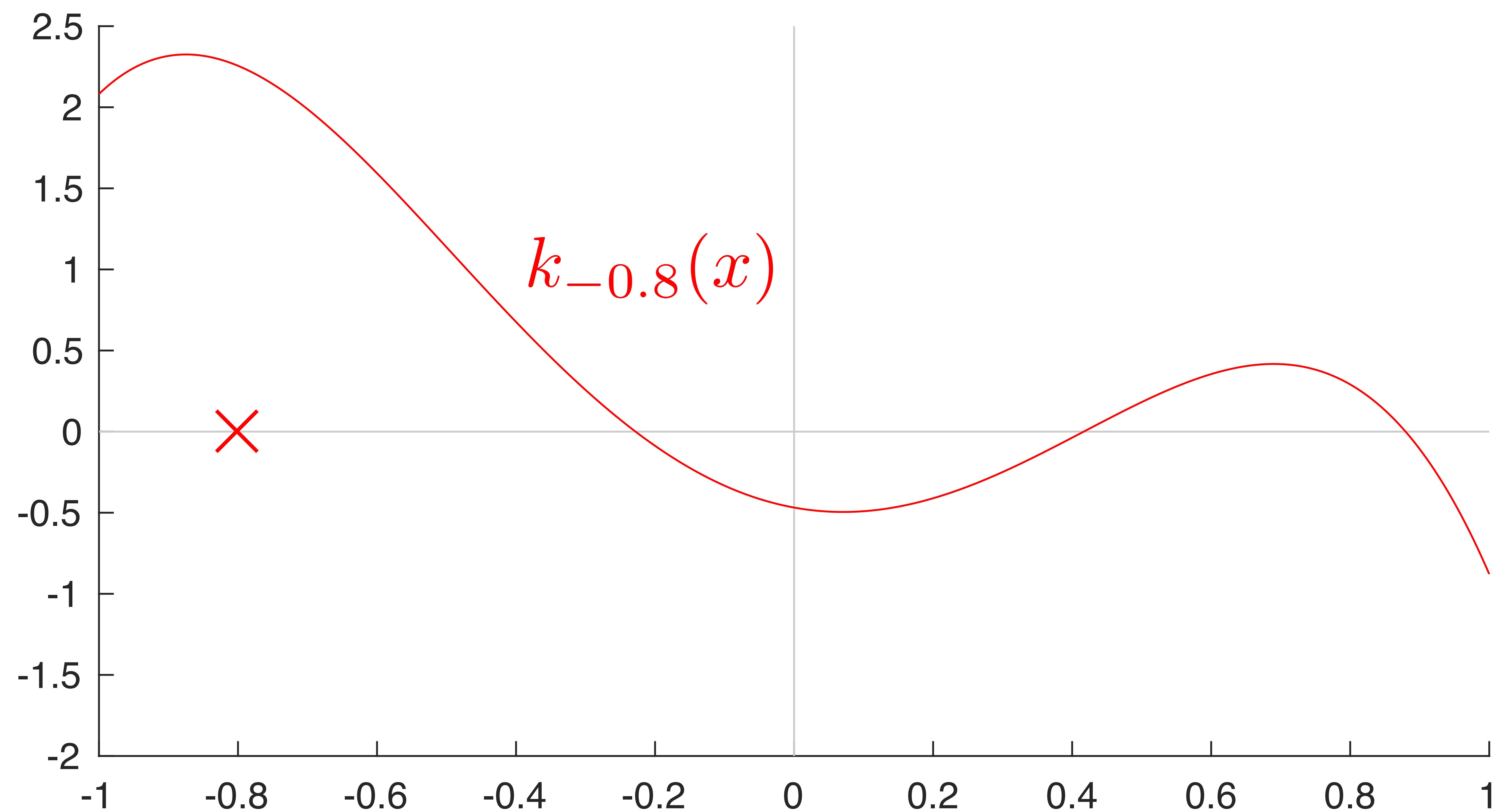
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



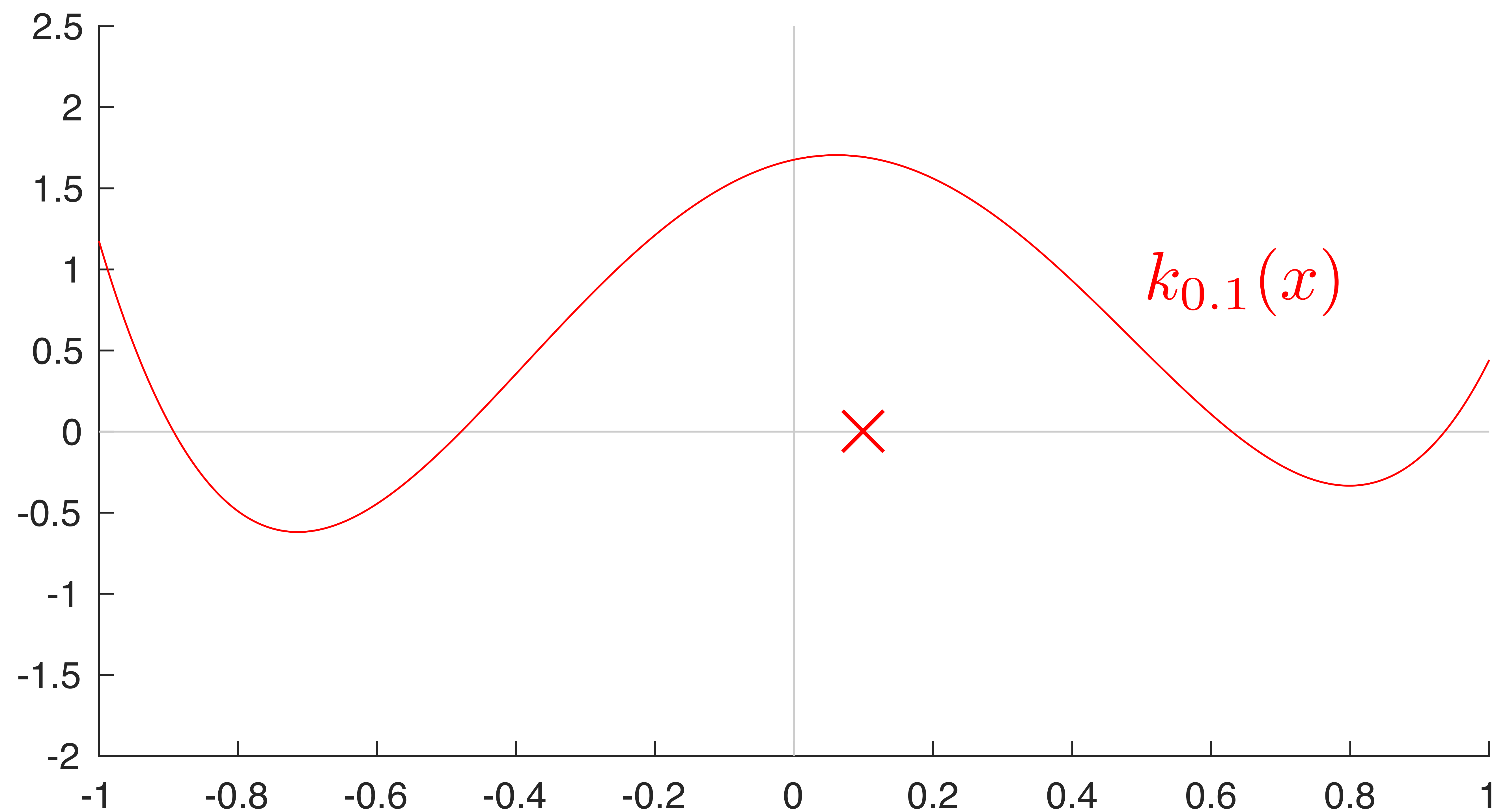
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



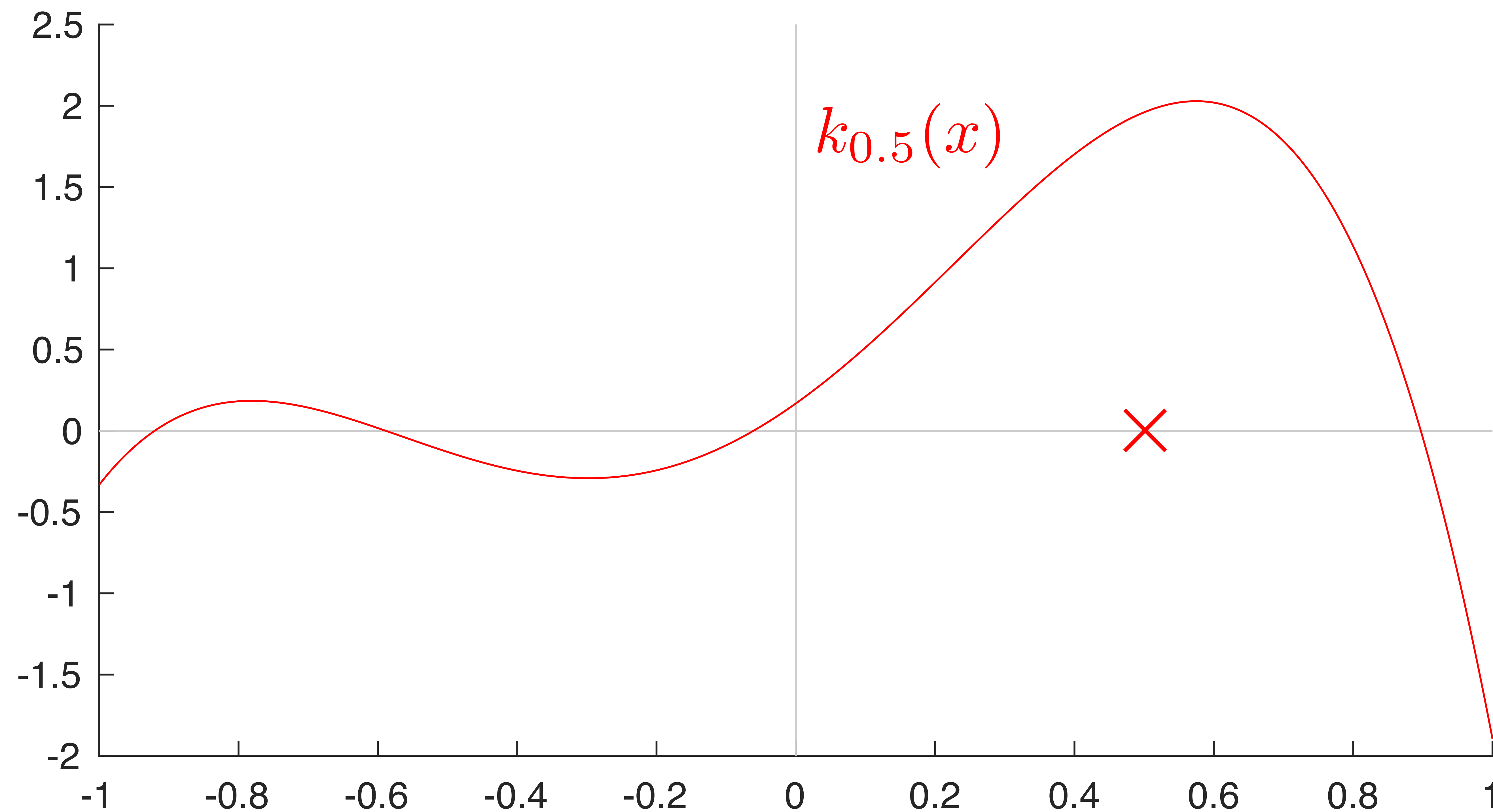
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



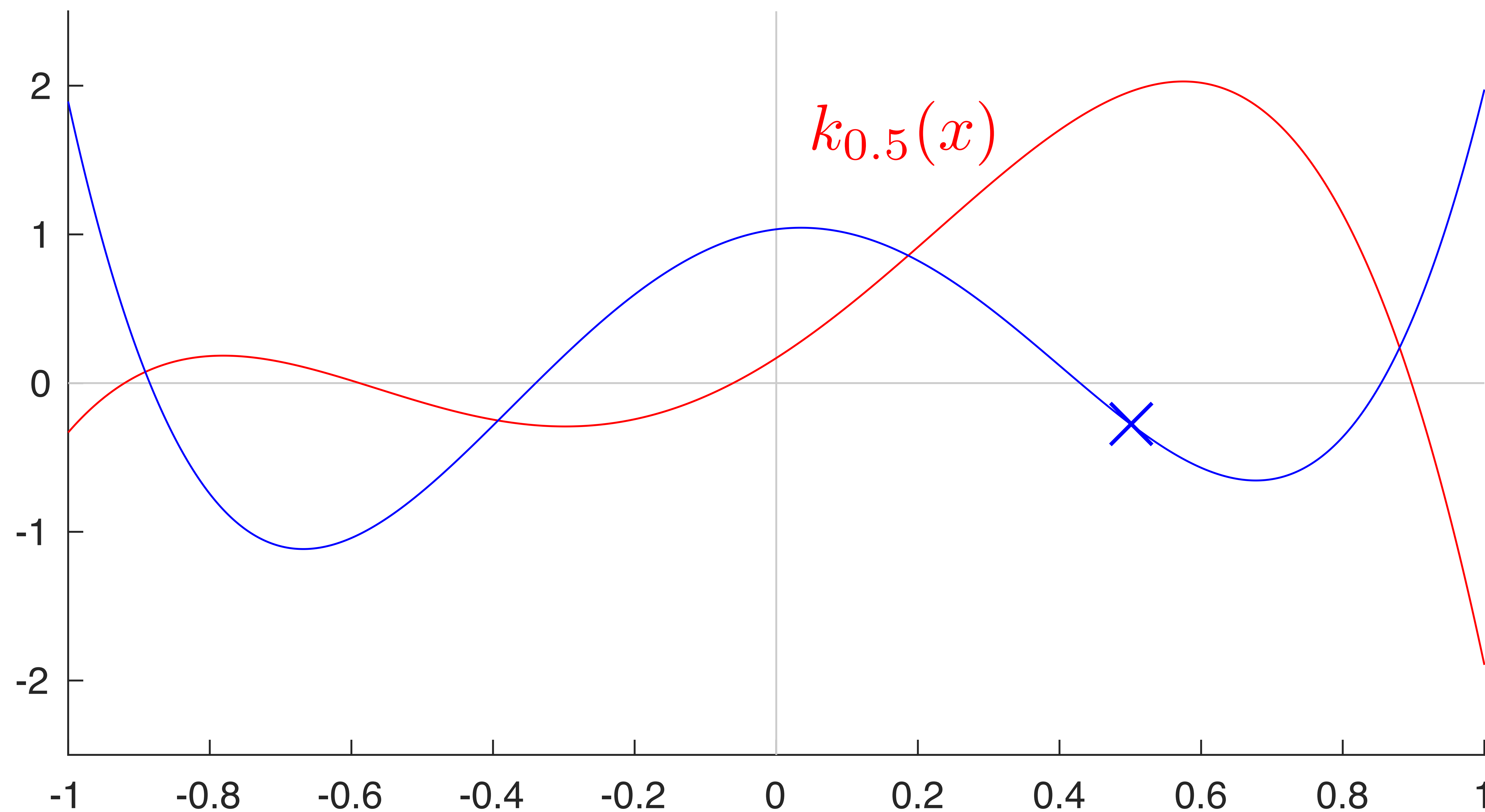
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



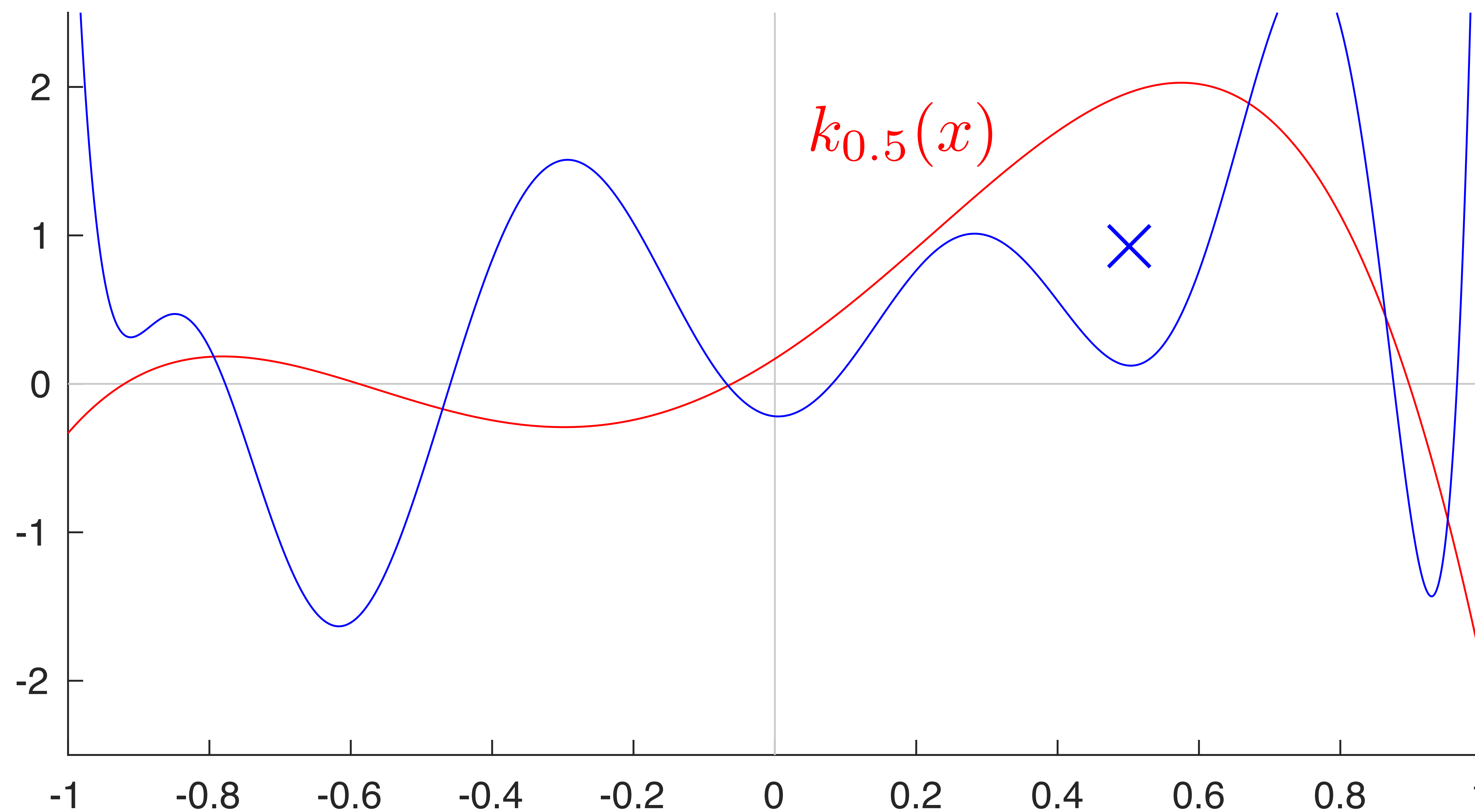
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



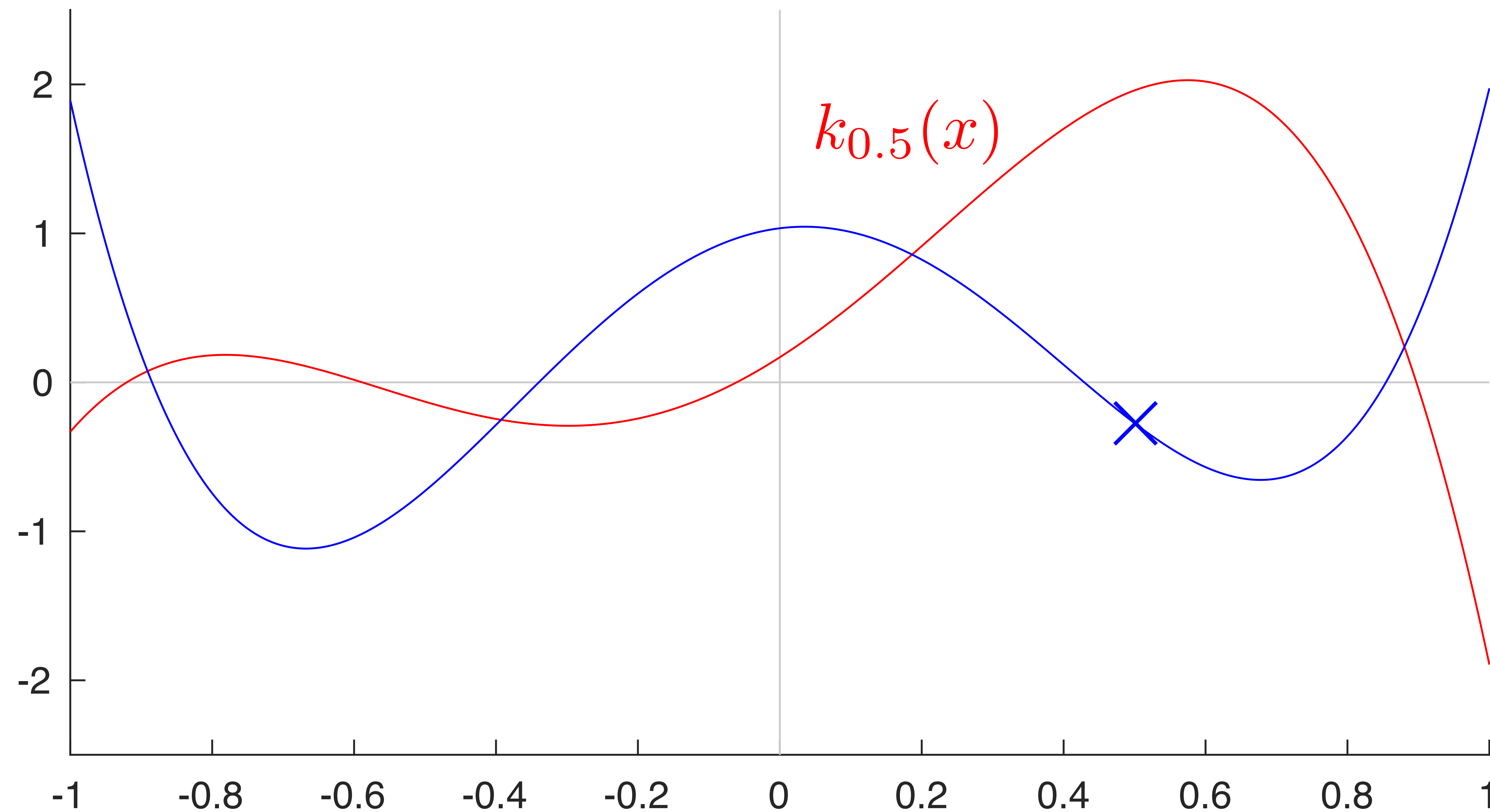
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



Reproducing kernel Hilbert spaces

$\mathcal{H}(X)$ is Hilbert and $\delta_{x_i}(f)$ continuous, then

$$\forall \bar{x} \in X, \exists k_{\bar{x}}(x) \in \mathcal{H}(X) : \left\langle k_{\bar{x}}(x), f(x) \right\rangle = f(\bar{x})$$

Reproducing kernel Hilbert spaces

$\mathcal{H}(X)$ is Hilbert and $\delta_{x_i}(f)$ continuous, then

$$\forall \bar{x} \in X, \boxed{\exists k_{\bar{x}}(x) \in \mathcal{H}(X)} : \left\langle k_{\bar{x}}(x), f(x) \right\rangle = f(\bar{x})$$

Reproducing kernel Hilbert spaces

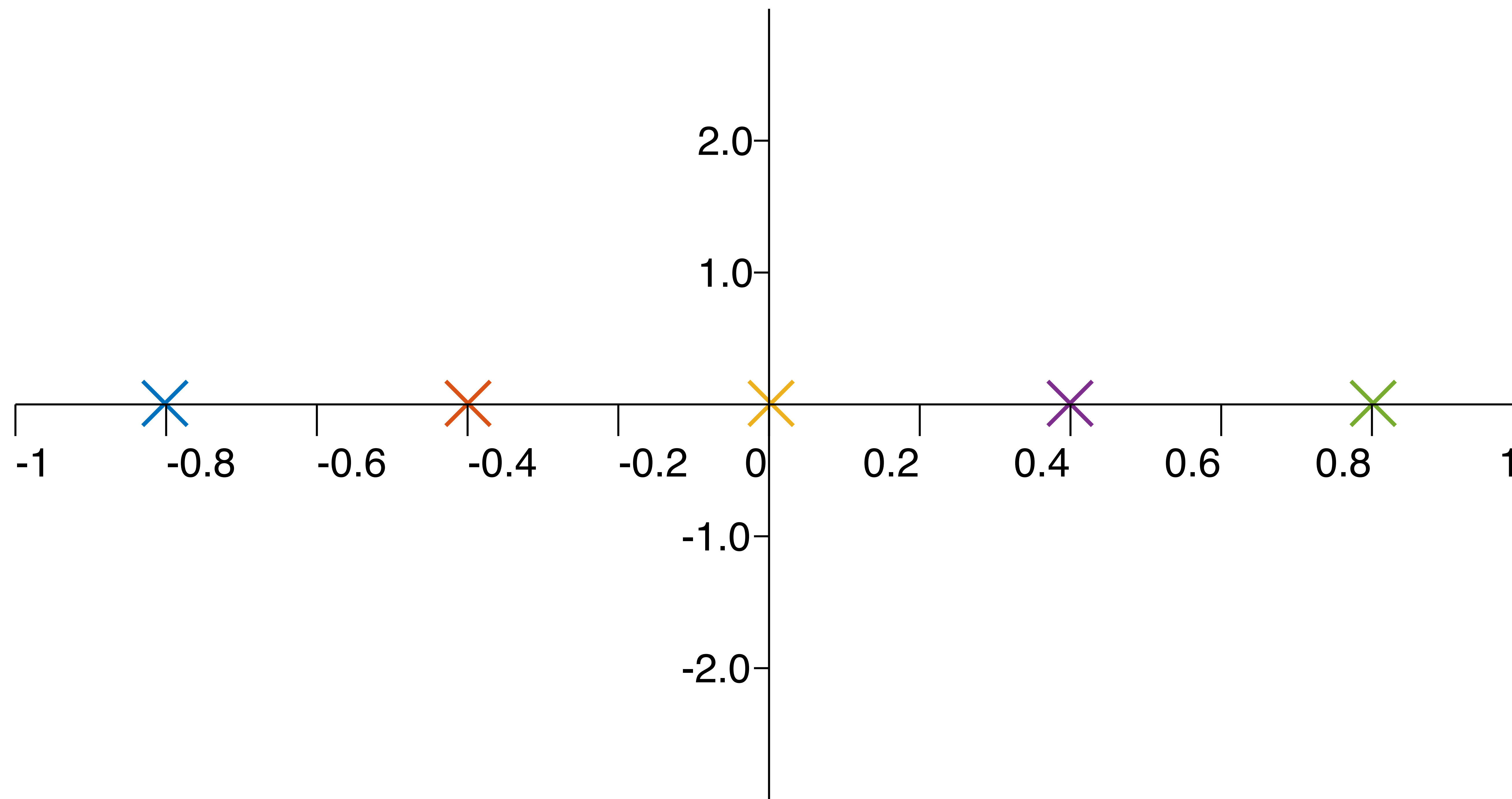
Let $\{\lambda_i\}$, $\lambda_i \in X$

Reproducing kernel Hilbert spaces

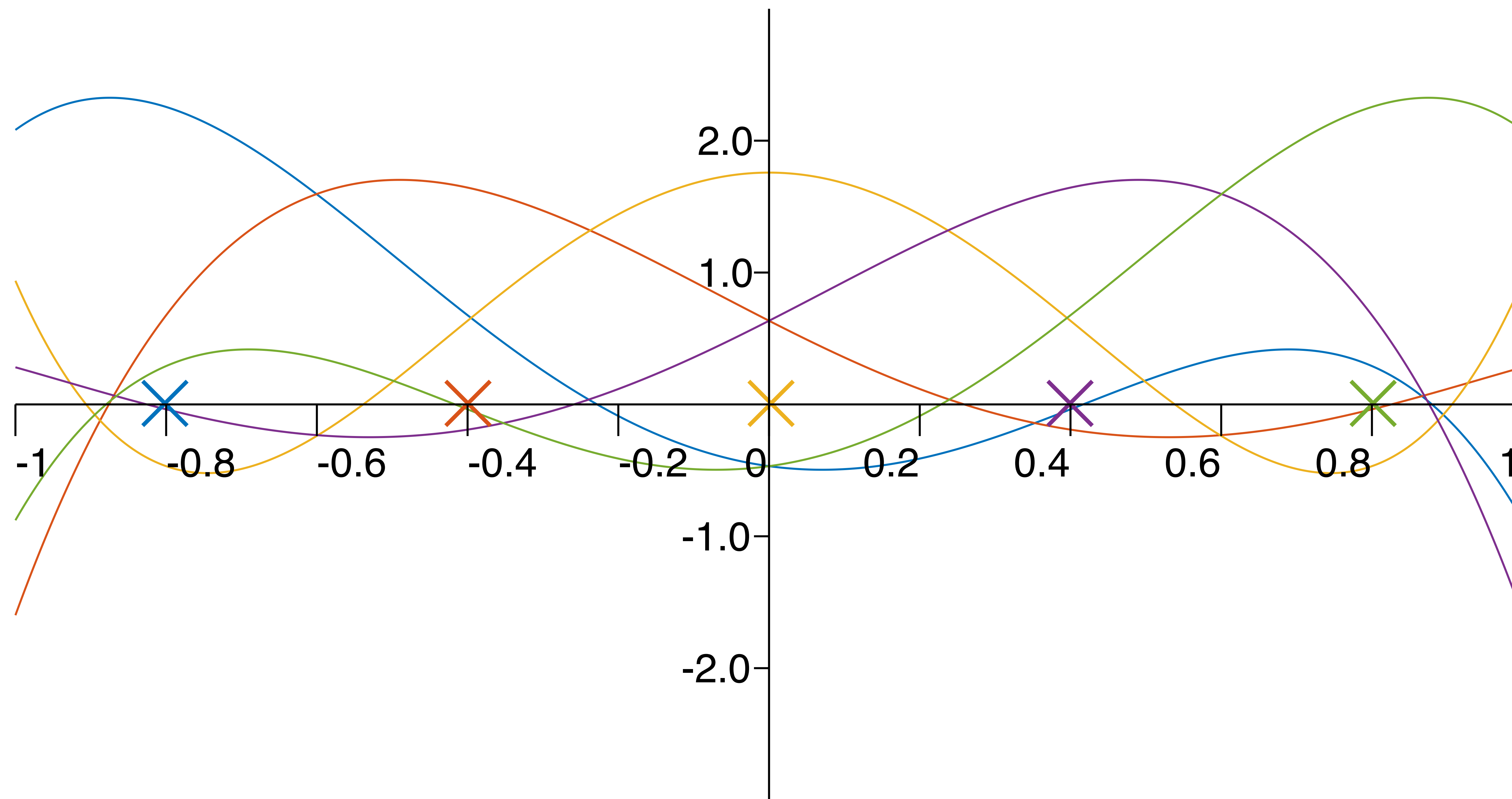
Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$

Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



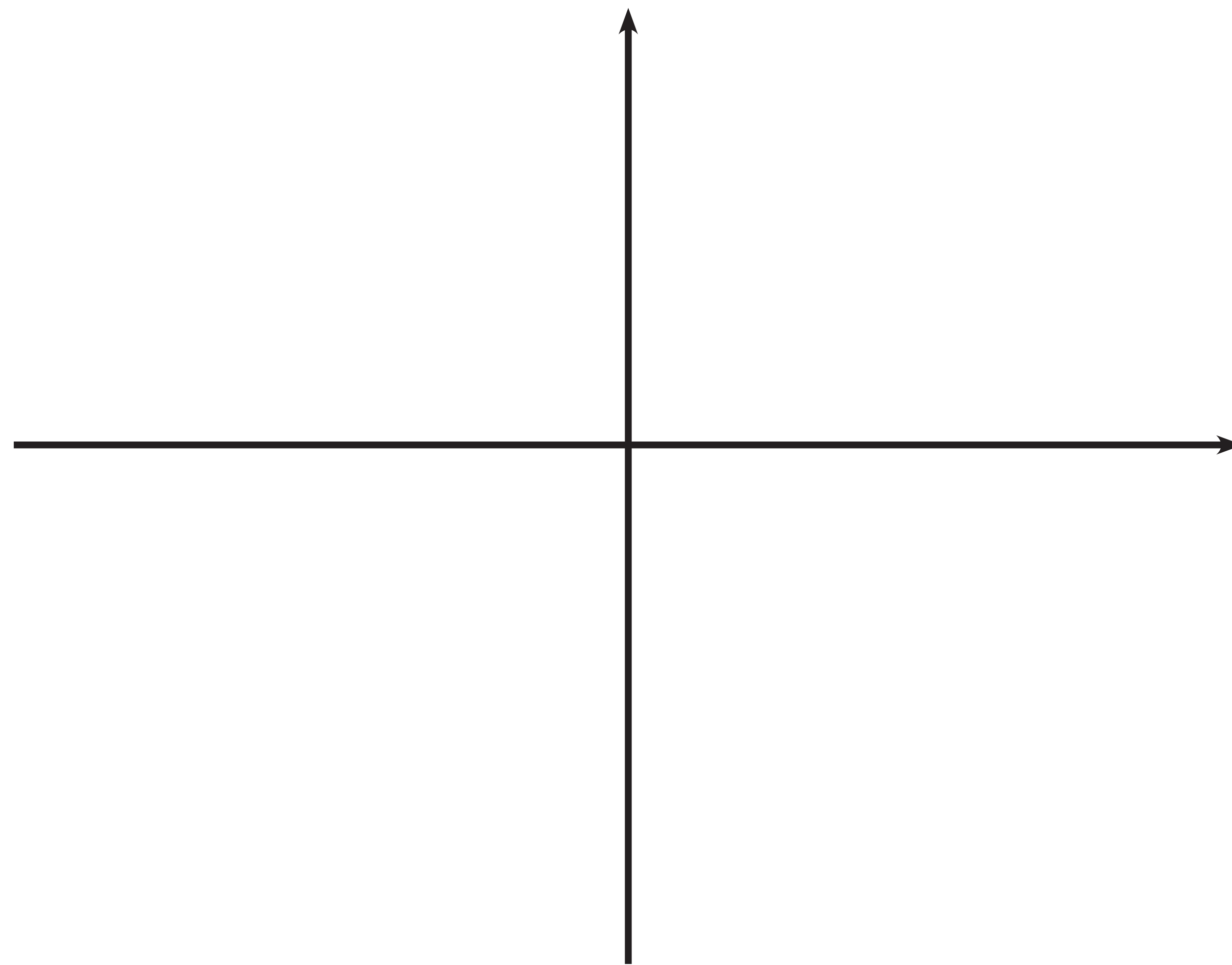
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

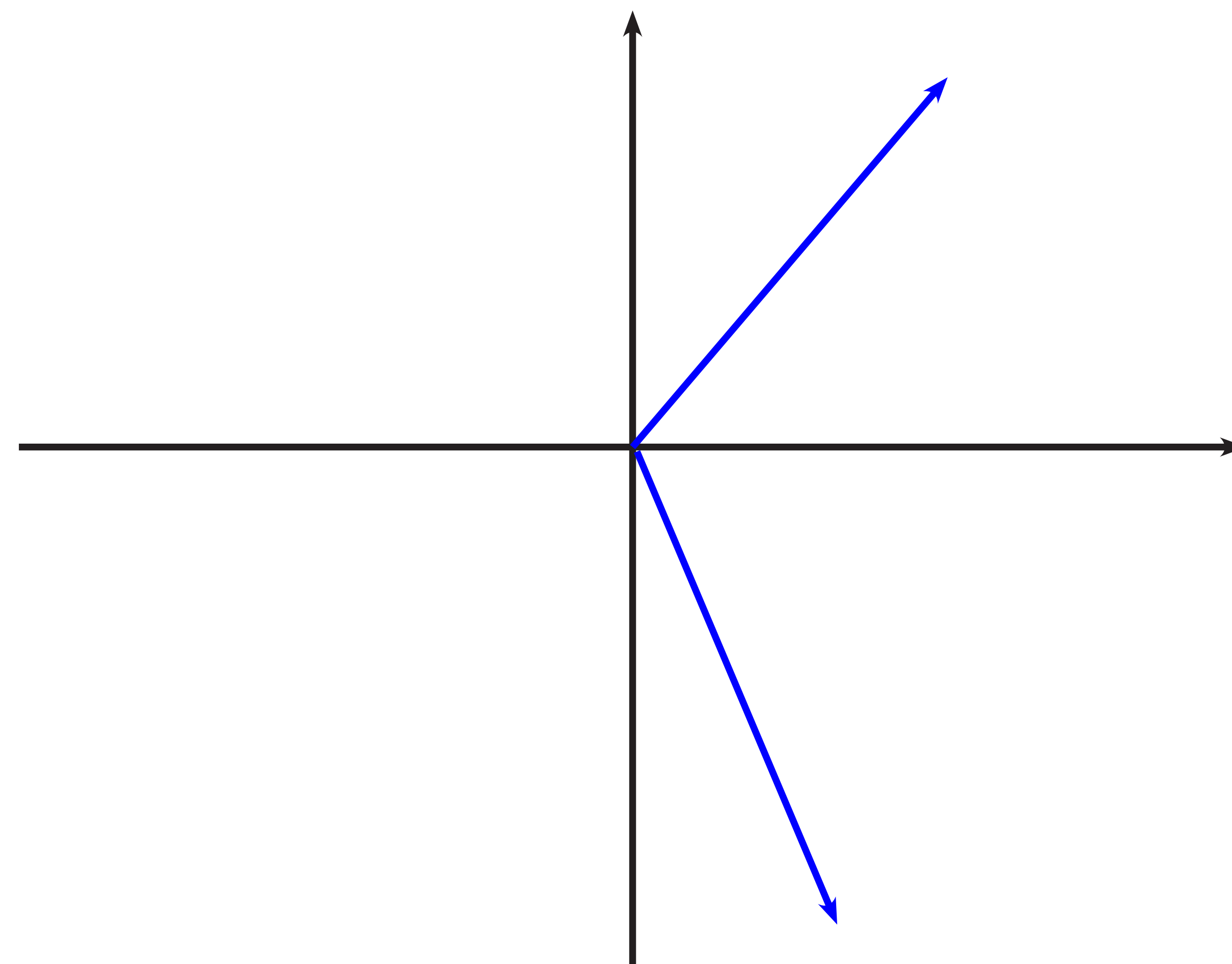
$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$



Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$

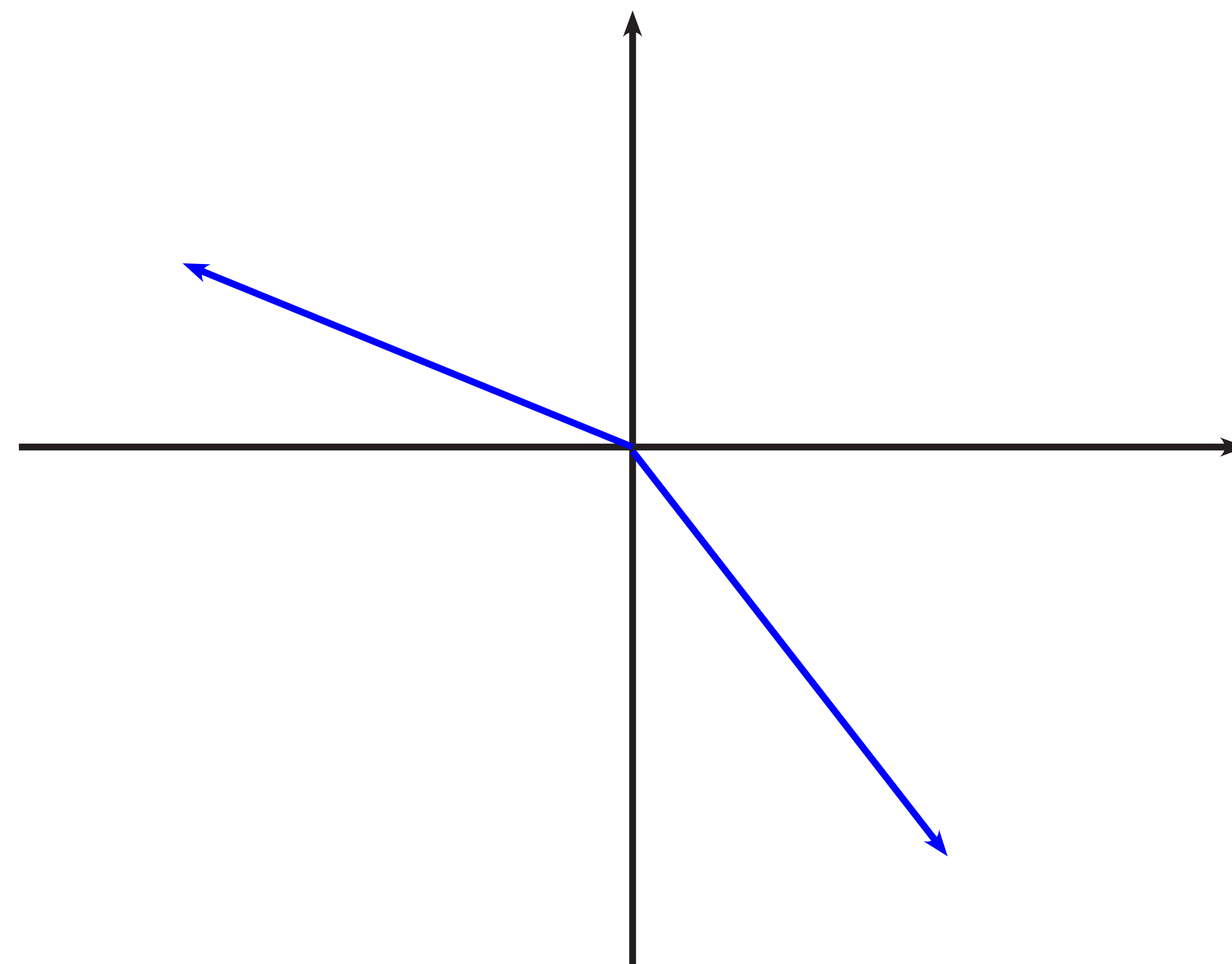


spanning set

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$

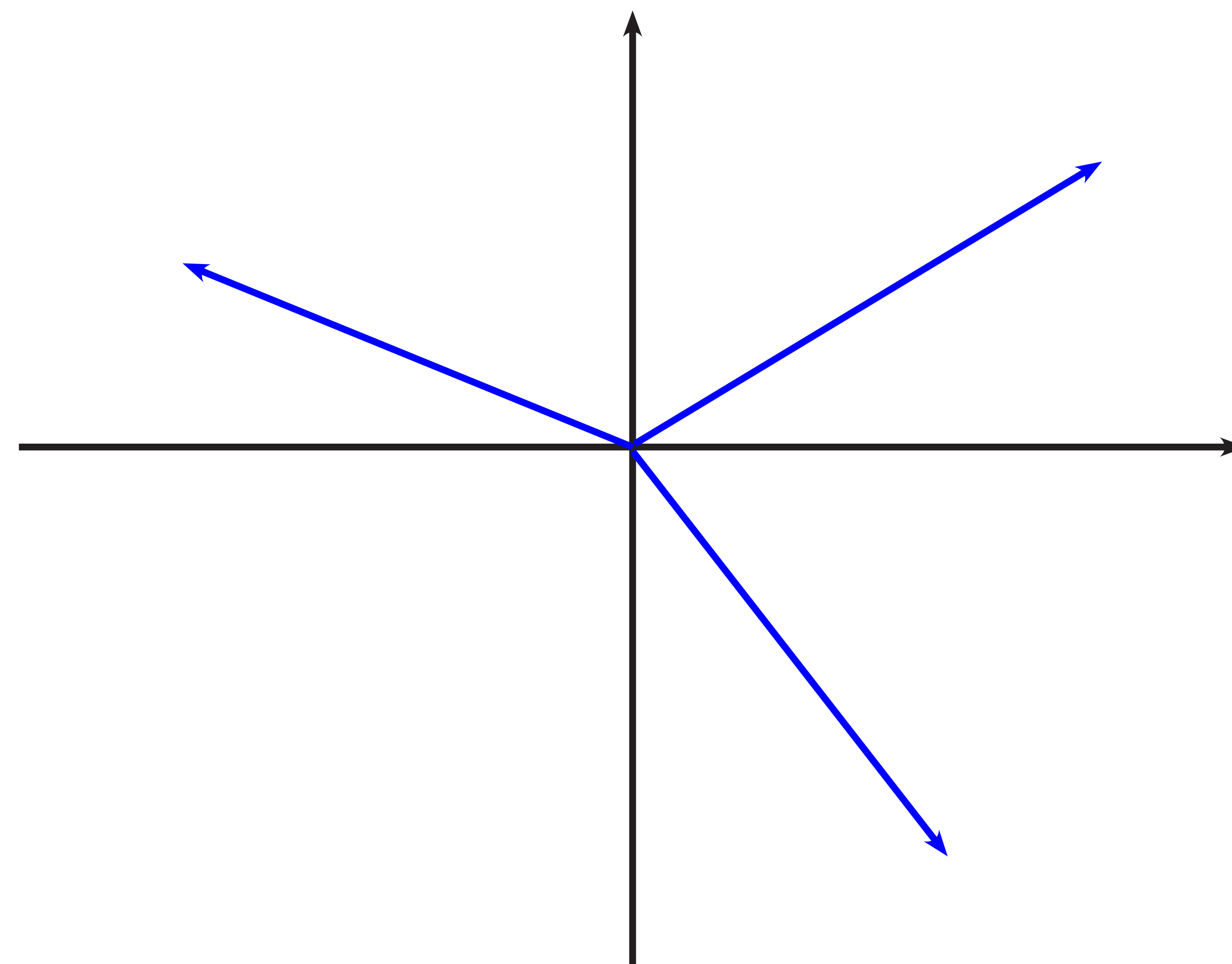


spanning set

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$

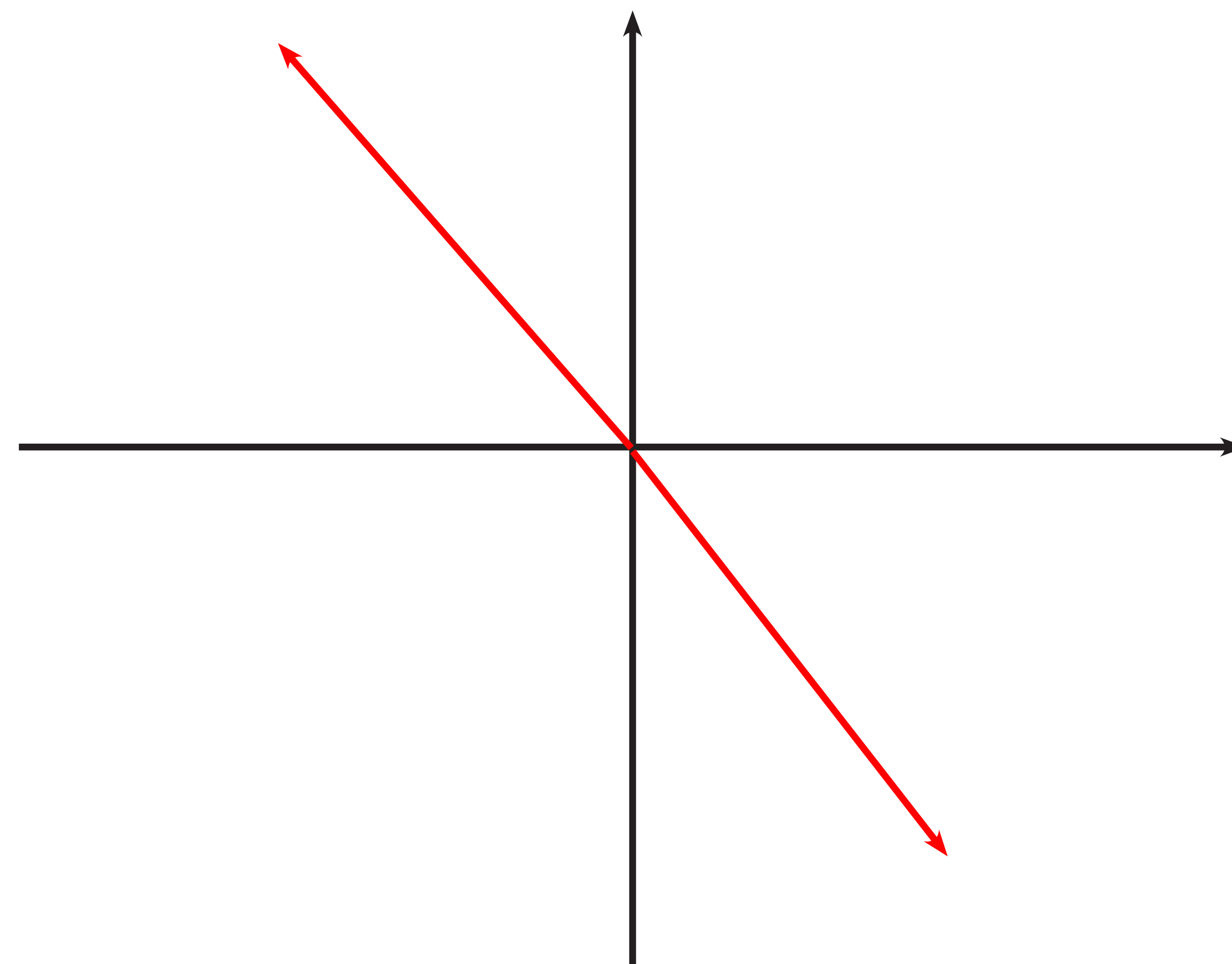


spanning set

Reproducing kernel Hilbert spaces

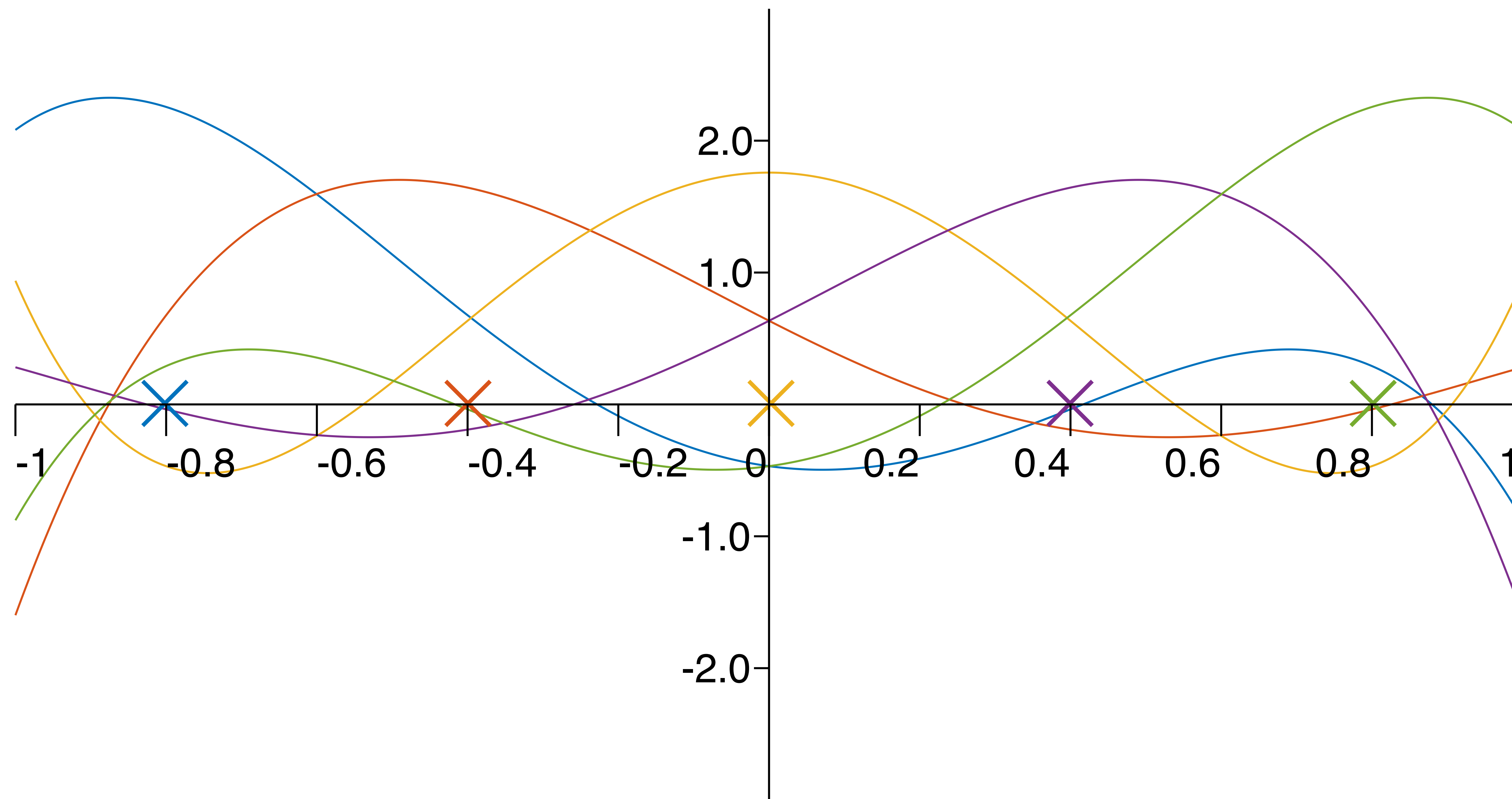
Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$

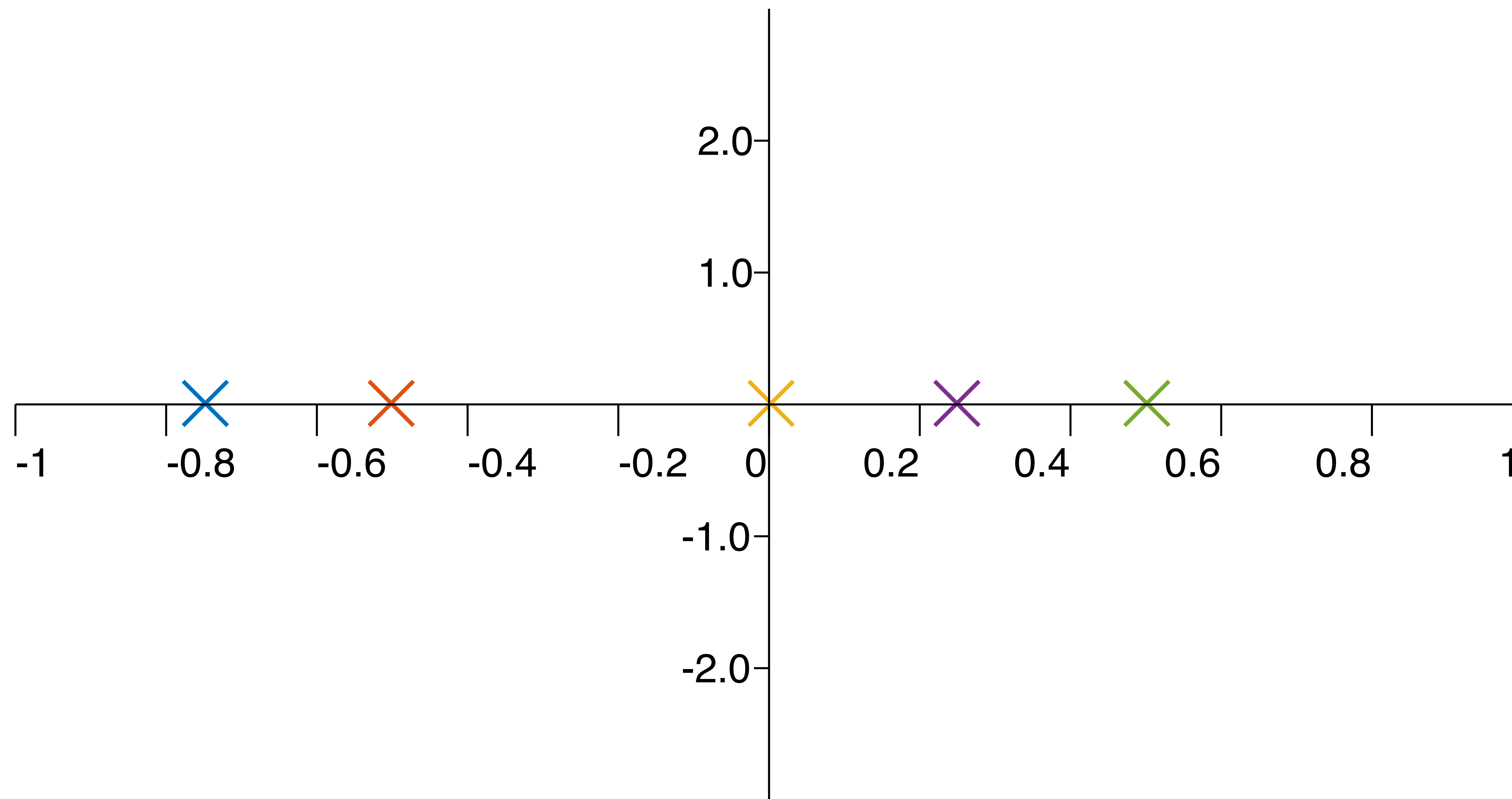


no spanning set

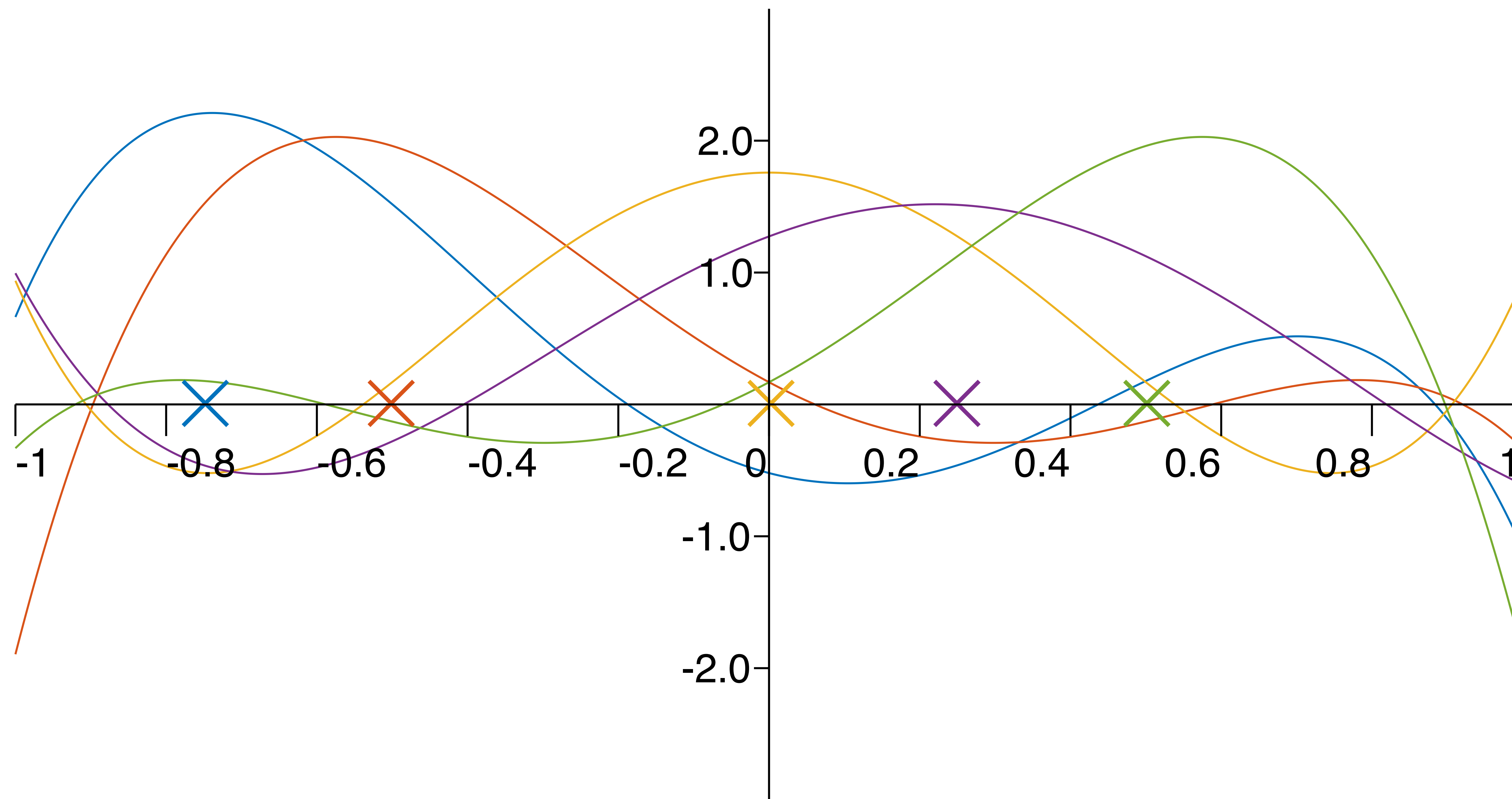
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$



$\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t.

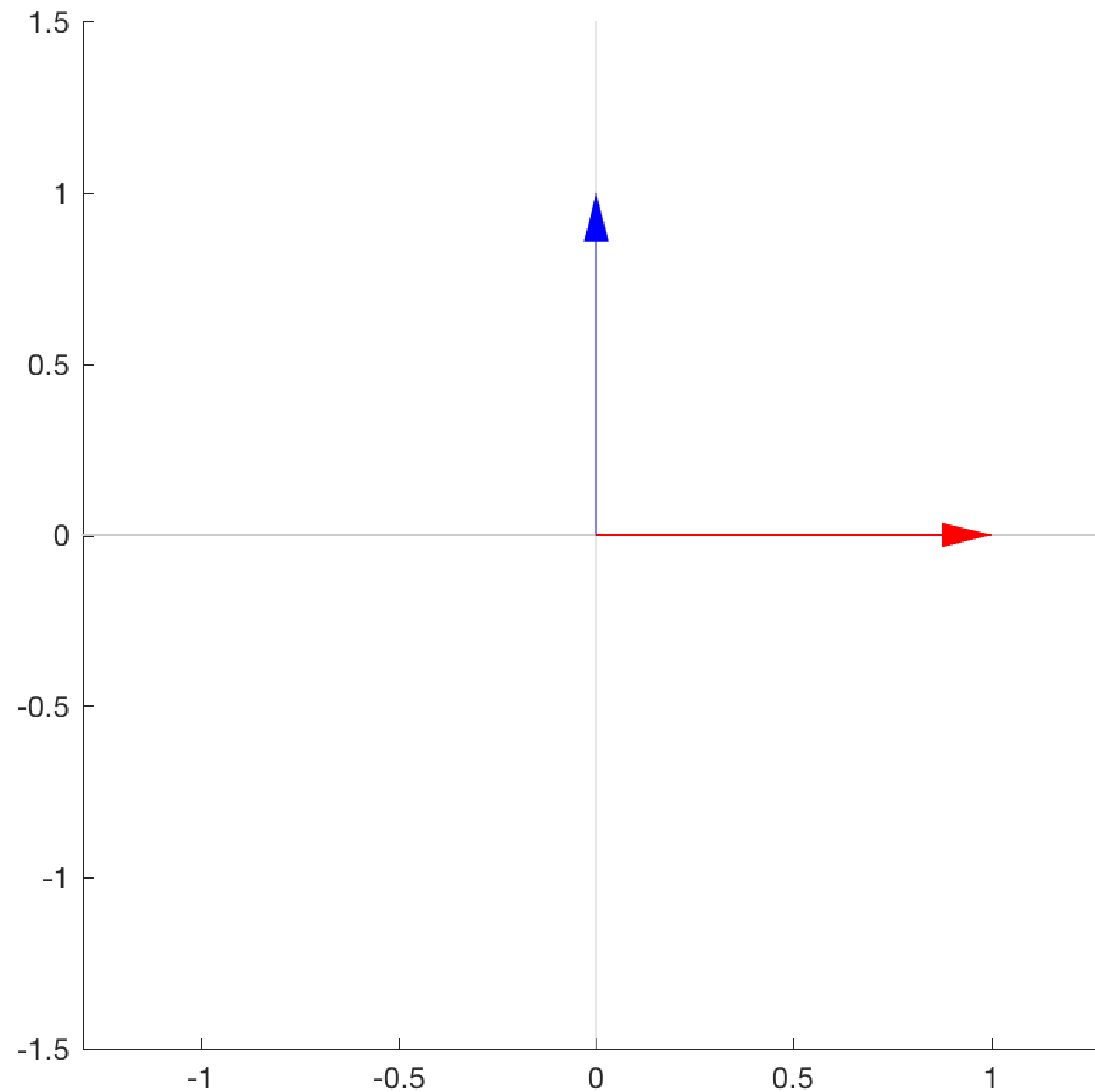
$$\text{span}(k_{\lambda_i}(x)) = \mathcal{H}(X)$$



$\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

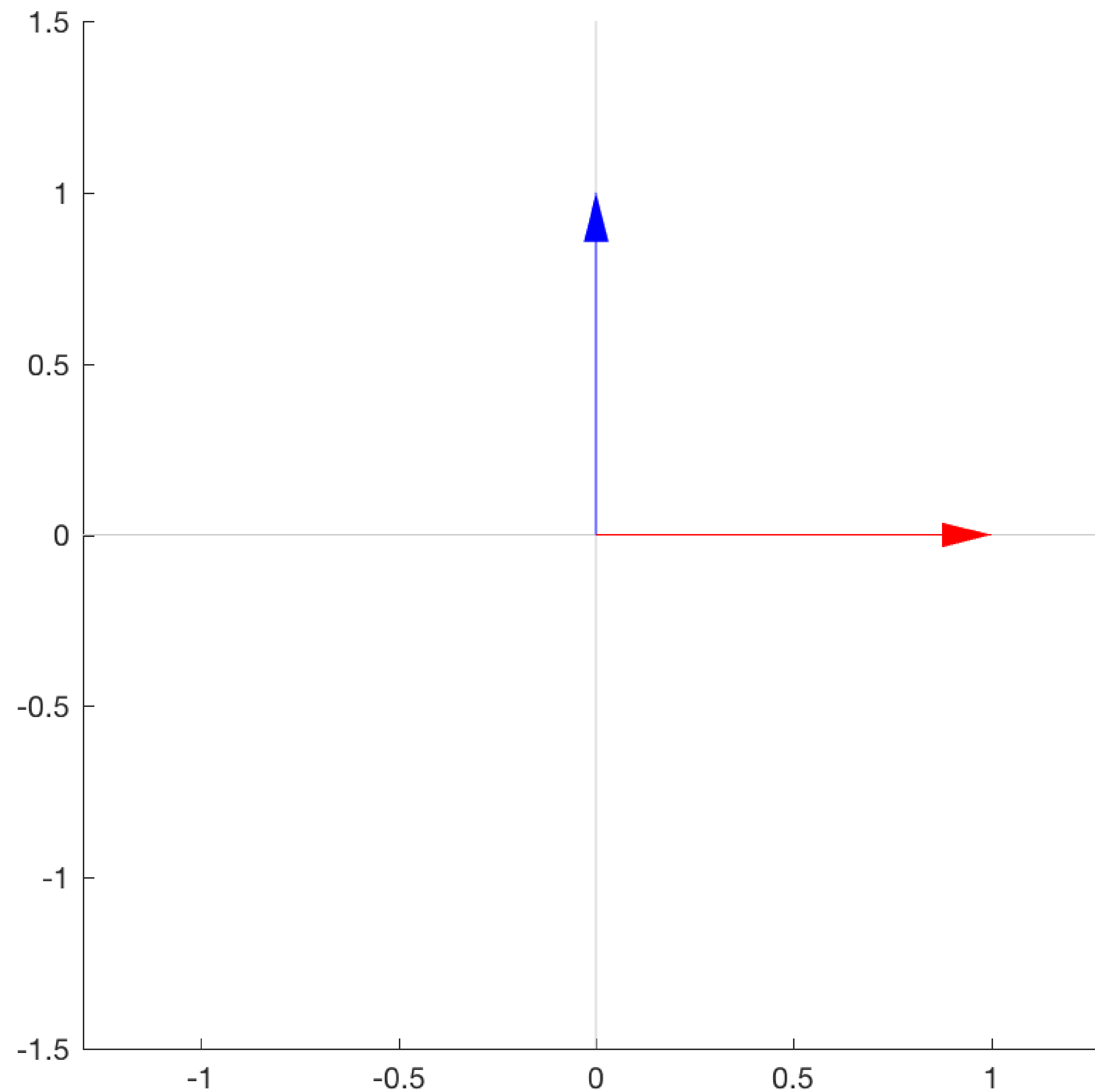
$$f(x) = \sum_{i=1}^N f_i k_{\lambda_i}(x)$$

Reproducing kernel Hilbert spaces



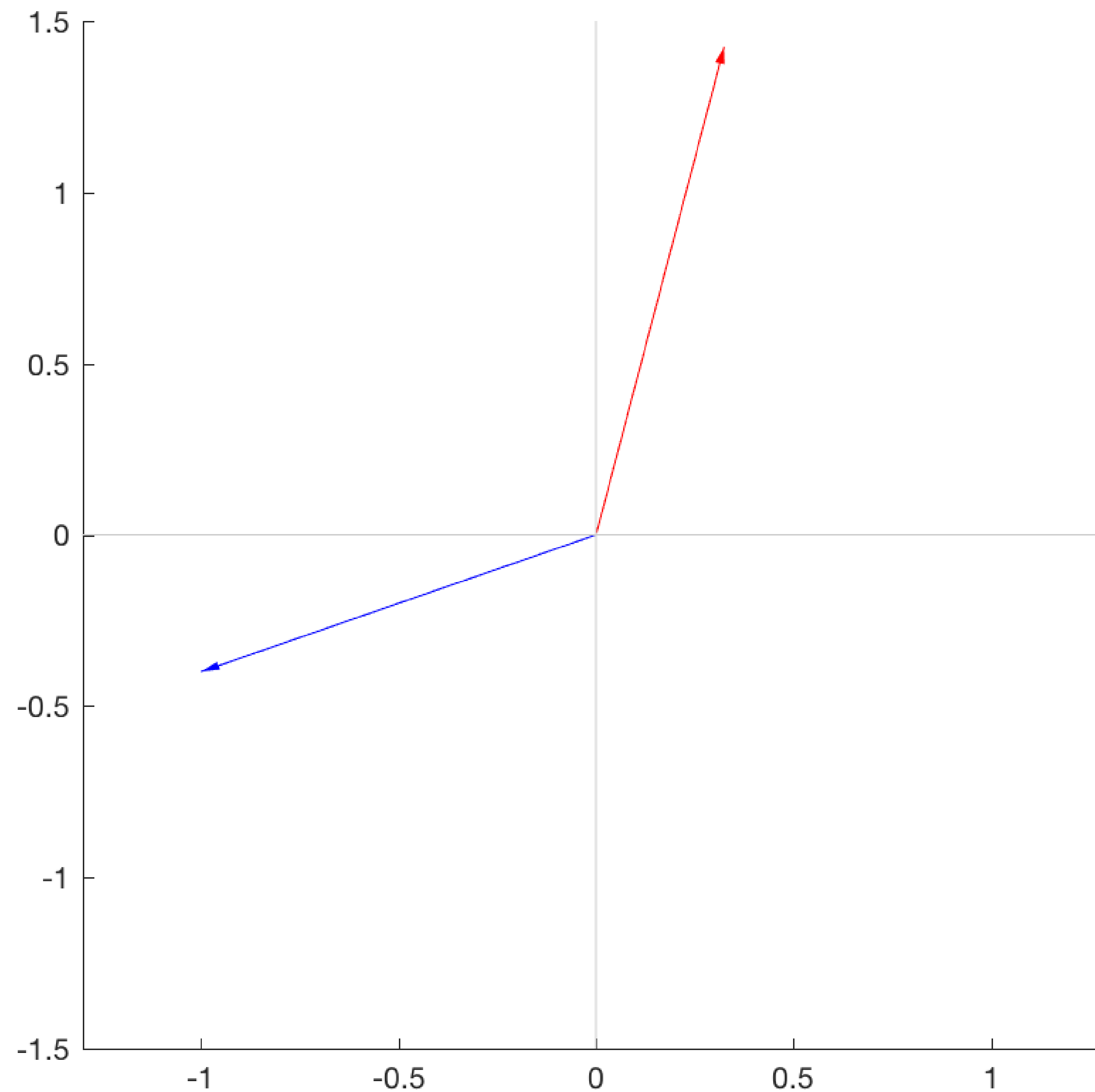
$$v = \sum_{i=1}^2 v_i e_i$$

Reproducing kernel Hilbert spaces



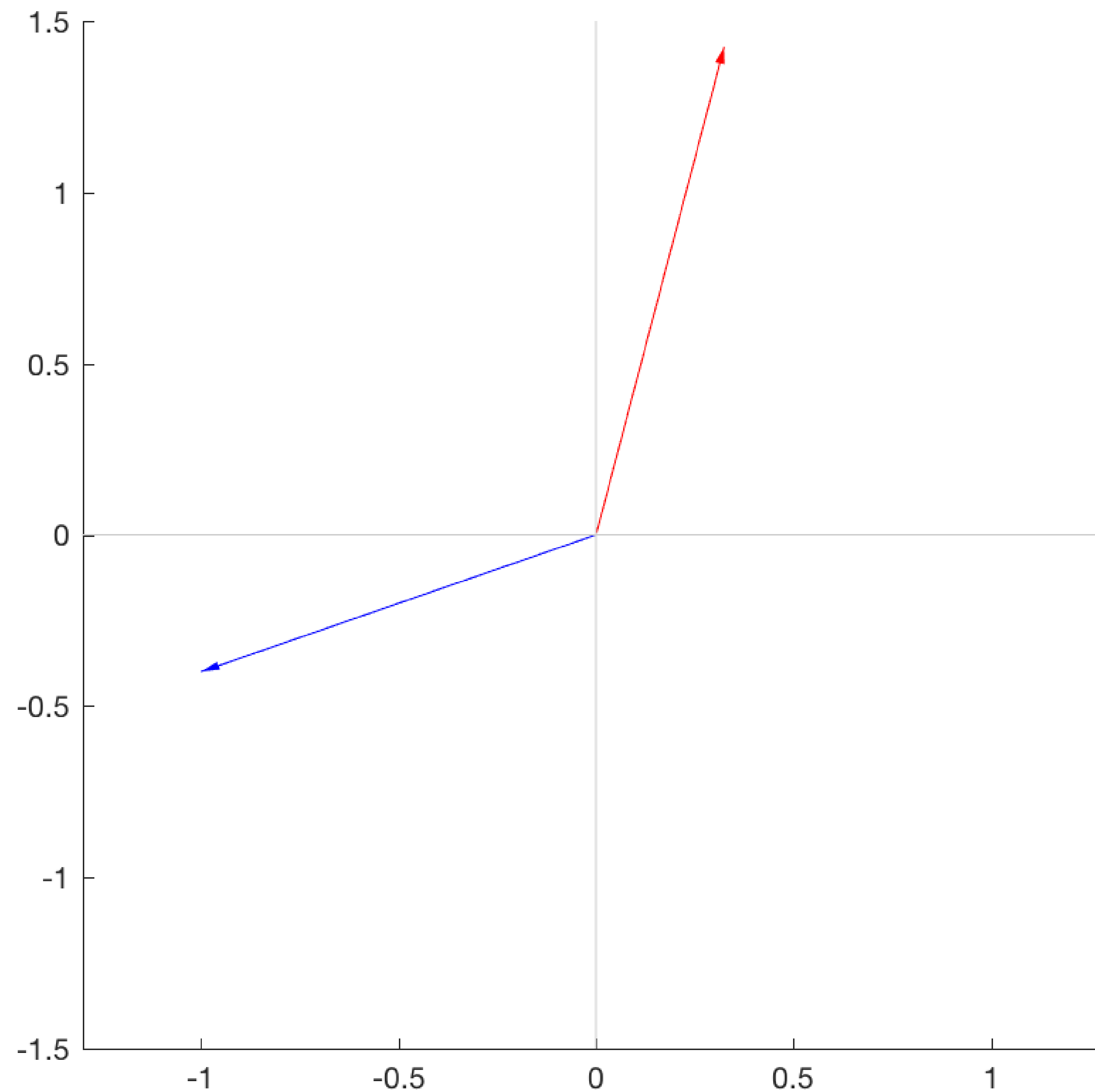
$$v = \sum_{i=1}^2 \langle v, e_i \rangle e_i$$

Reproducing kernel Hilbert spaces



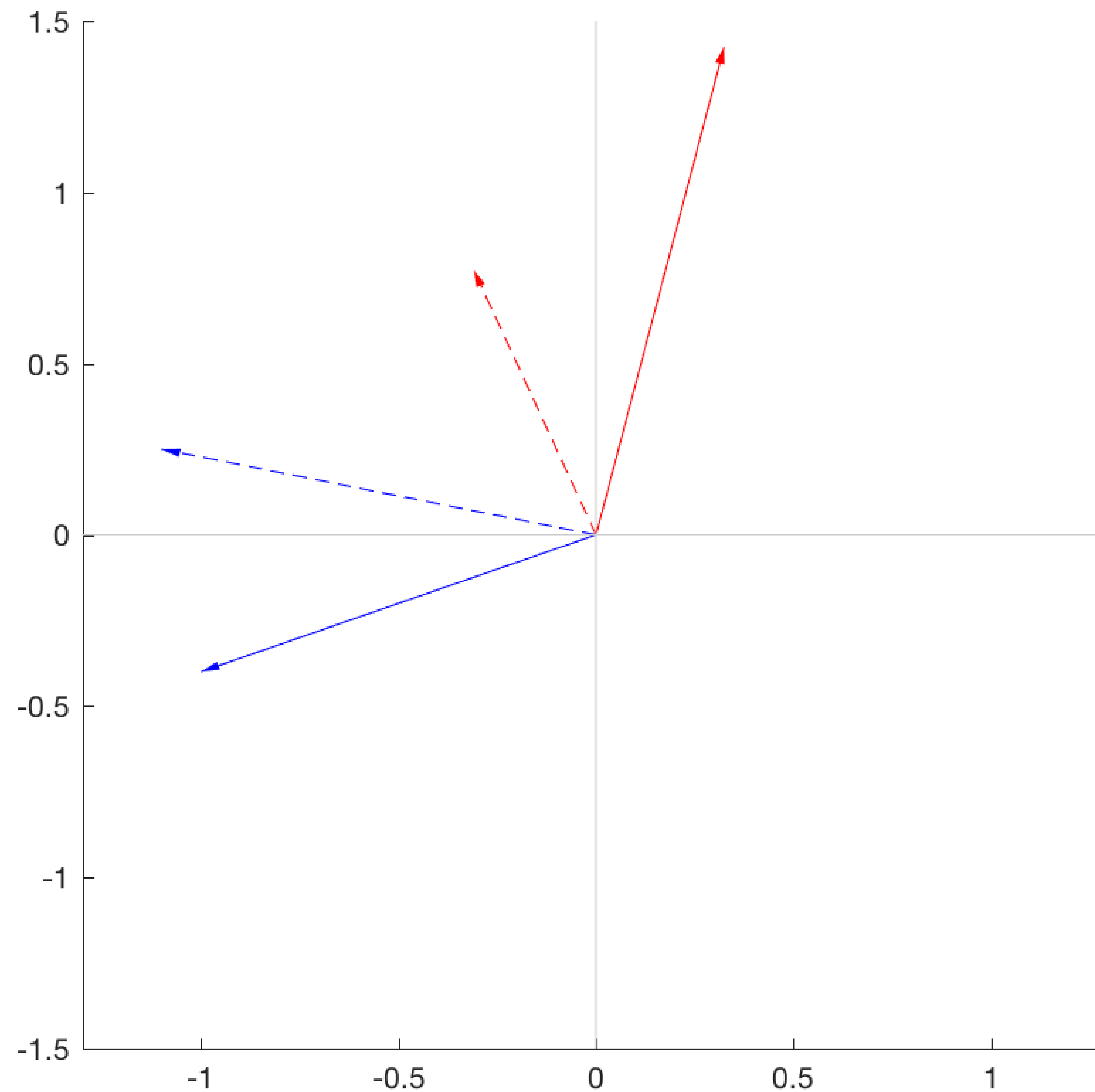
$$v = \sum_{i=1}^2 v_i u_i$$

Reproducing kernel Hilbert spaces



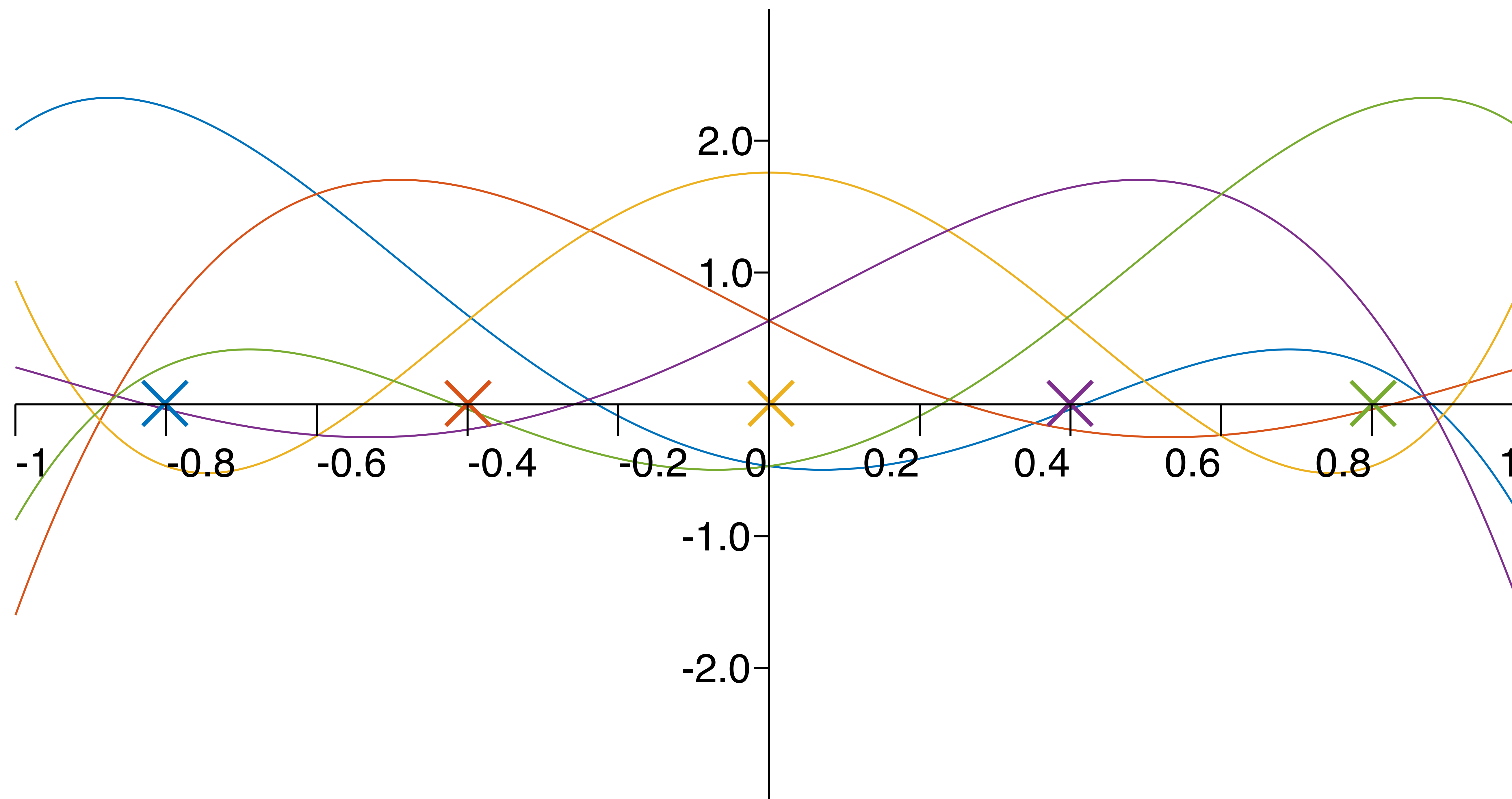
$$v = \sum_{i=1}^2 \langle v, ? \rangle u_i$$

Reproducing kernel Hilbert spaces

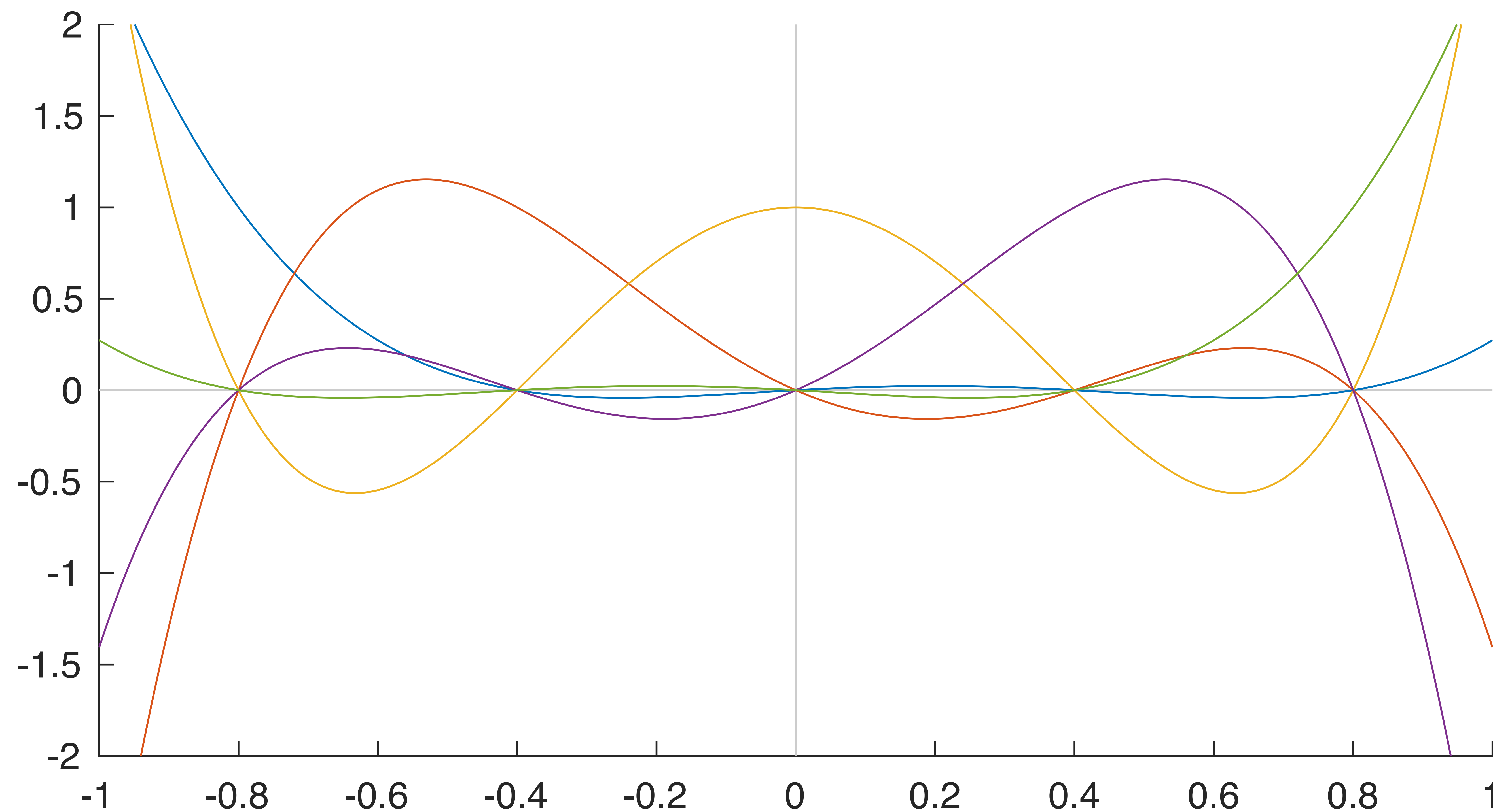


$$v = \sum_{i=1}^2 \langle v, \tilde{u}_i \rangle u_i$$

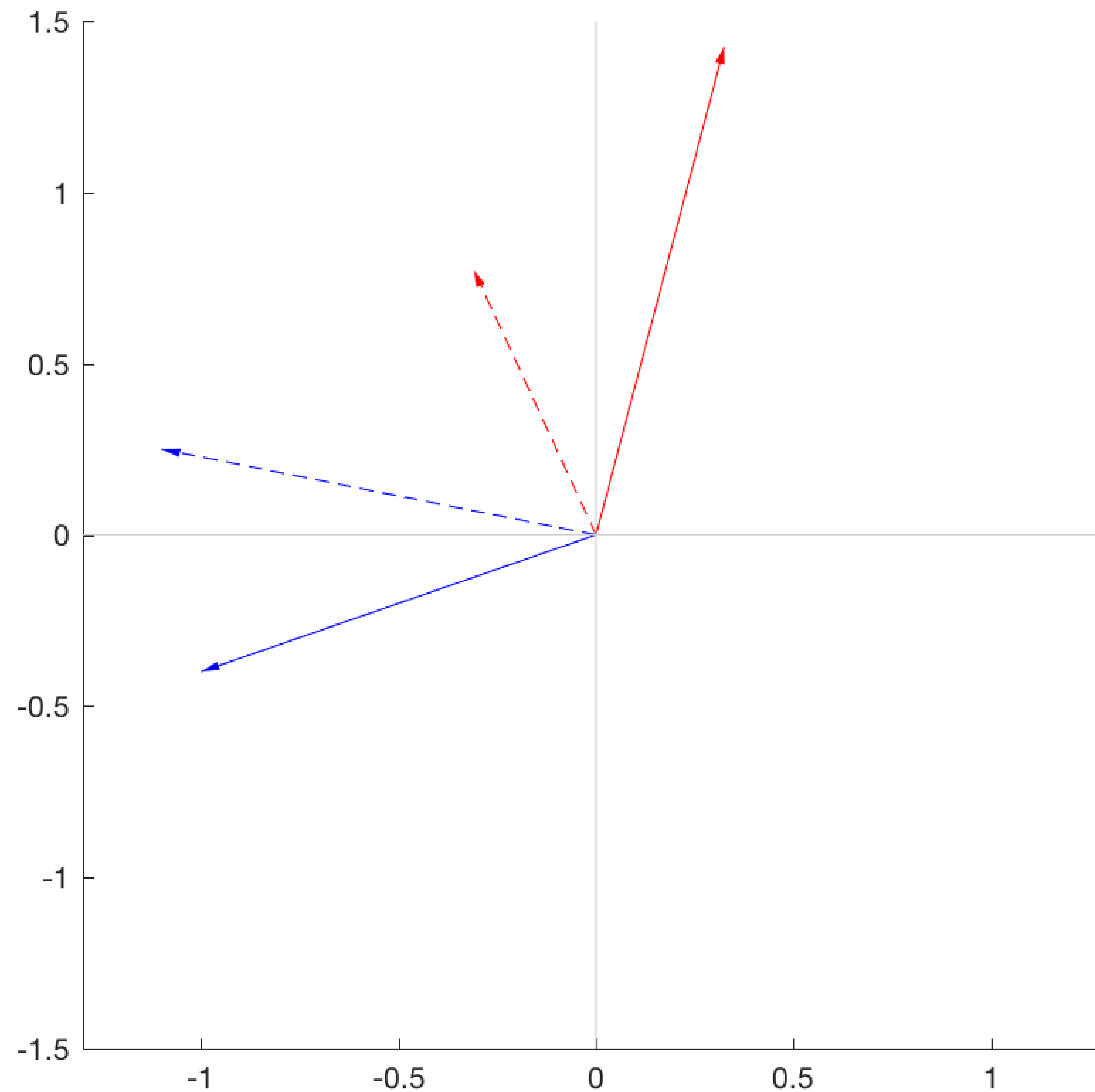
Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$



Example: $\mathcal{H}([-1, 1]) = \text{span}_{n=1 \dots N} \{P_n(x)\}$

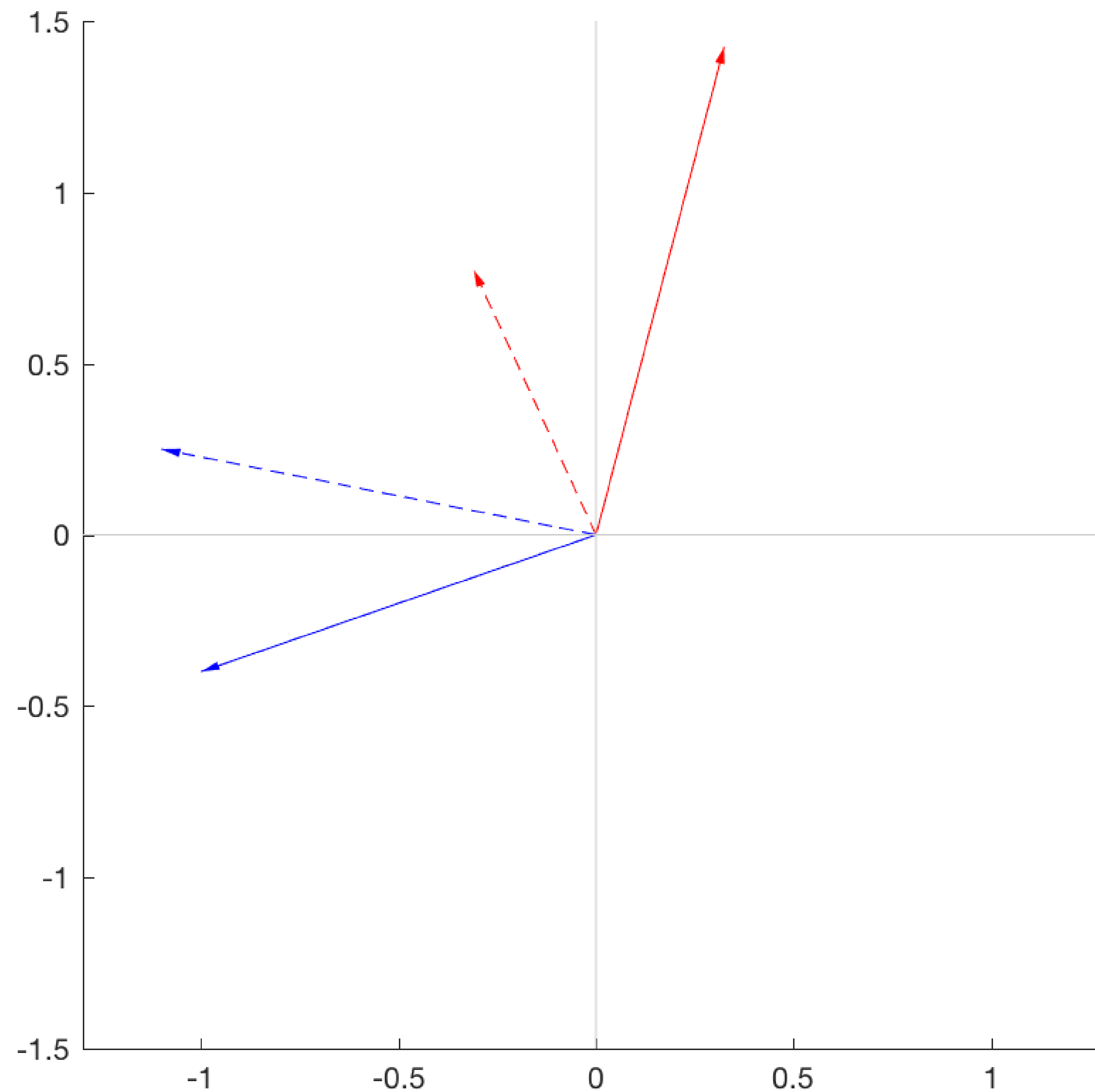


Reproducing kernel Hilbert spaces



$$v = \sum_{i=1}^2 \langle v, \tilde{u}_i \rangle u_i$$

Reproducing kernel Hilbert spaces



$$v = \sum_{i=1}^2 \langle v, u_i \rangle \tilde{u}_i$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f_i k_{\lambda_i}(x)$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f_i k_{\lambda_i}(x) = \sum_{i=1}^N \left\langle f(y), \tilde{k}_i(y) \right\rangle k_{\lambda_i}(x)$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$\begin{aligned} f(x) &= \sum_{i=1}^N f_i k_{\lambda_i}(x) = \sum_{i=1}^N \left\langle f(y), \tilde{k}_i(y) \right\rangle k_{\lambda_i}(x) \\ &= \sum_{i=1}^N \left\langle f(y), k_{\lambda_i}(y) \right\rangle \tilde{k}_i(x) \end{aligned}$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$\begin{aligned} f(x) &= \sum_{i=1}^N f_i k_{\lambda_i}(x) = \sum_{i=1}^N \left\langle f(y), \tilde{k}_i(y) \right\rangle k_{\lambda_i}(x) \\ &= \sum_{i=1}^N \left\langle f(y), k_{\lambda_i}(y) \right\rangle \tilde{k}_i(x) \end{aligned}$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$\begin{aligned} f(x) &= \sum_{i=1}^N f_i k_{\lambda_i}(x) = \sum_{i=1}^N \left\langle f(y), \tilde{k}_i(y) \right\rangle k_{\lambda_i}(x) \\ &= \sum_{i=1}^N \left\langle f(y), k_{\lambda_i}(y) \right\rangle \tilde{k}_i(x) \\ &\quad \quad \quad = f(\lambda_i) \end{aligned}$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$\begin{aligned} f(x) &= \sum_{i=1}^N f_i k_{\lambda_i}(x) = \sum_{i=1}^N \left\langle f(y), \tilde{k}_i(y) \right\rangle k_{\lambda_i}(x) \\ &= \sum_{i=1}^N \underbrace{\left\langle f(y), k_{\lambda_i}(y) \right\rangle}_{= f(\lambda_i)} \tilde{k}_i(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x) \end{aligned}$$

Reproducing kernel Hilbert spaces

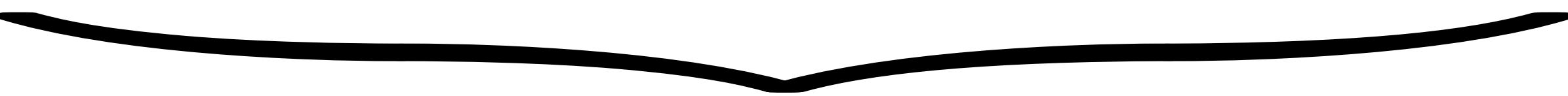
Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$


reproducing kernel frame

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

reproducing kernel frame $\iff$ sampling theorem

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

reproducing kernel frame $\iff$ sampling theorem

Shannon:
$$f(x) = \sum_{i=-\infty}^{\infty} f(i) \operatorname{sinc}(x - i)$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

reproducing kernel frame $\iff$ sampling theorem

$$\begin{aligned} \text{Shannon: } f(x) &= \sum_{i=-\infty}^{\infty} f(i) \operatorname{sinc}(x - i) \\ &= \sum_{i=-\infty}^{\infty} \langle f(y), \operatorname{sinc}(y - i) \rangle \operatorname{sinc}(x - i) \end{aligned}$$

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

reproducing kernel frame $\iff$ sampling theorem

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

reproducing kernel frame $\iff$ sampling theorem

- arbitrary locations

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

reproducing kernel frame $\iff$ sampling theorem

- arbitrary locations
- non-bandlimited functions

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

reproducing kernel frame $\iff$ sampling theorem

- arbitrary locations
- non-bandlimited functions
- over-sampling

Reproducing kernel Hilbert spaces

Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

reproducing kernel frame $\iff$ sampling theorem

- arbitrary locations
- non-bandlimited functions
- over-sampling
- arbitrary domains

Reproducing kernel Hilbert spaces

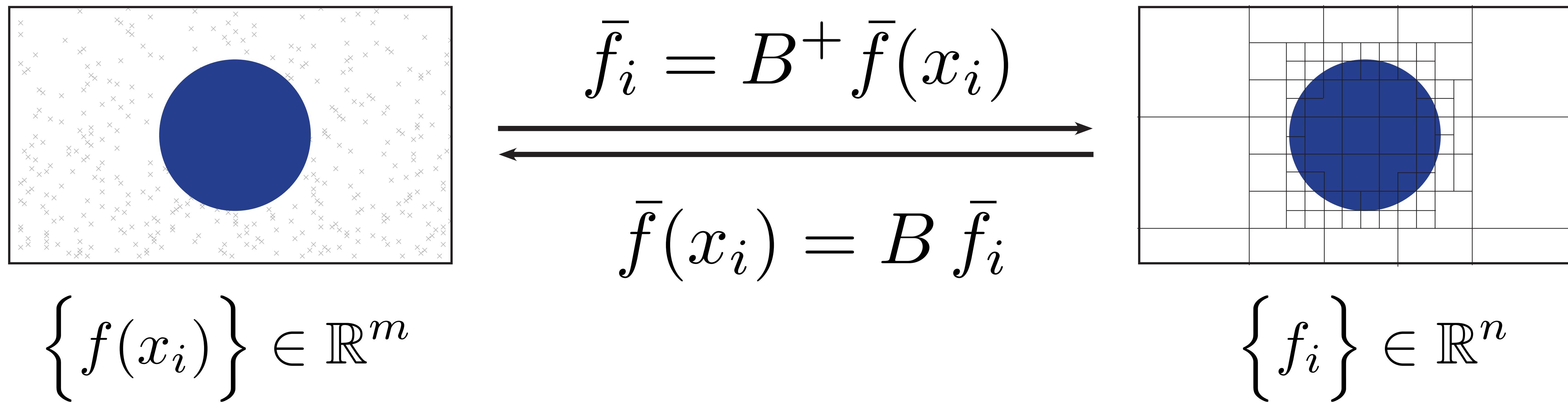
Let $\{\lambda_i\}$, $\lambda_i \in X$ s. t. $\{k_{\lambda_i}(x)\}$ forms a basis / frame for $\mathcal{H}(X)$

$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

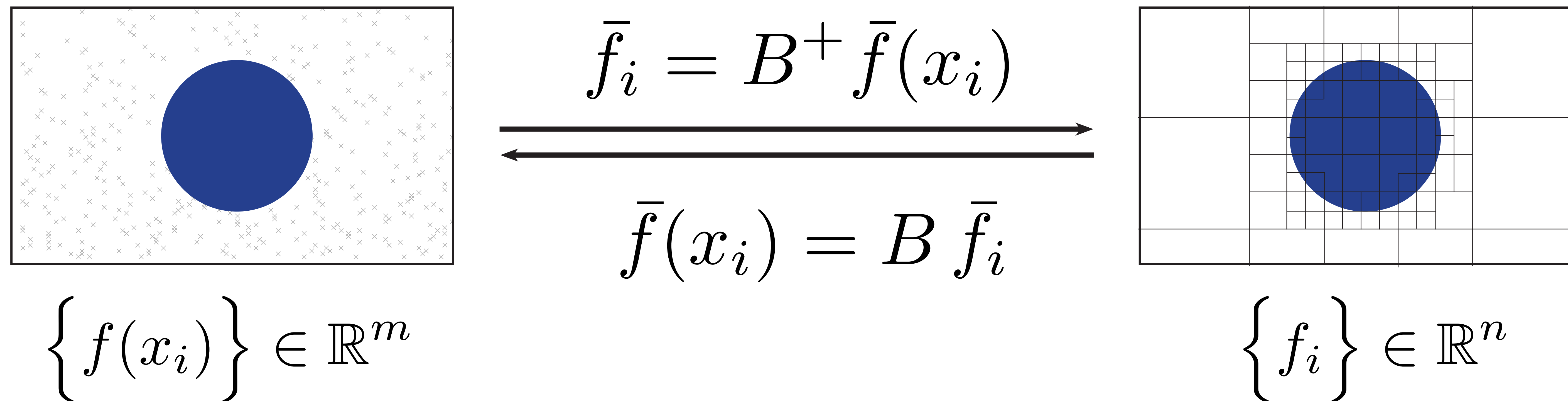
reproducing kernel frame $\iff$ sampling theorem

- arbitrary locations
- non-bandlimited functions
- over-sampling
- arbitrary domains
- optimal locations for setting

Reproducing kernel Hilbert spaces

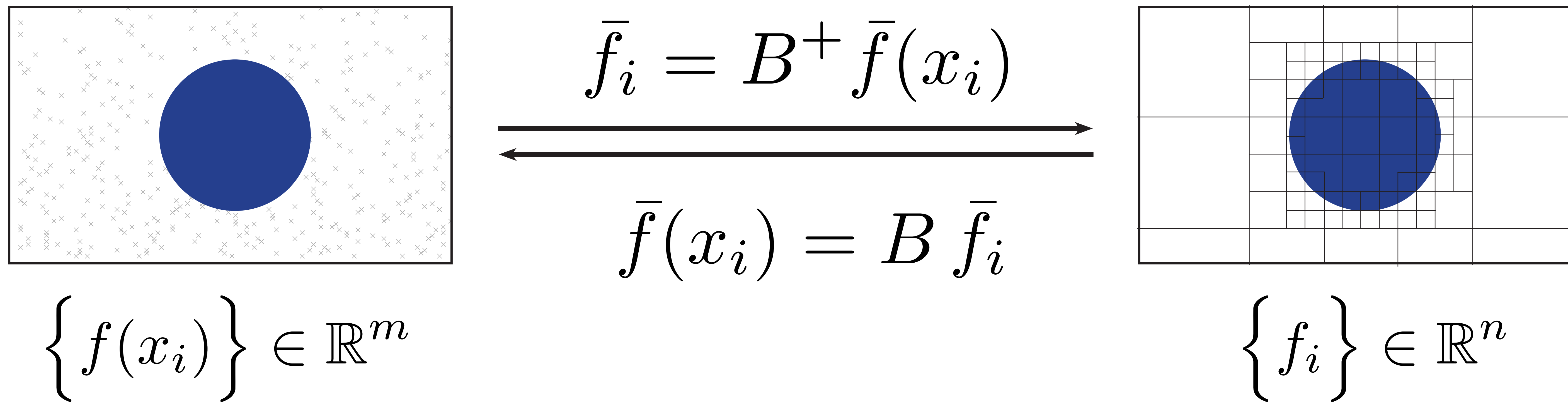


Reproducing kernel Hilbert spaces



$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

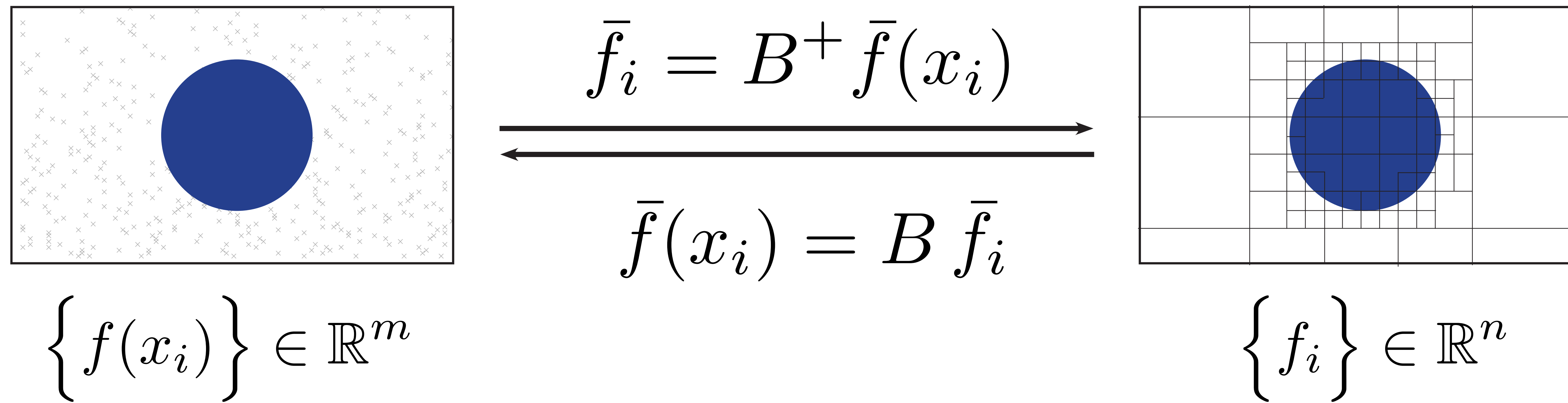
Reproducing kernel Hilbert spaces



$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x)$$

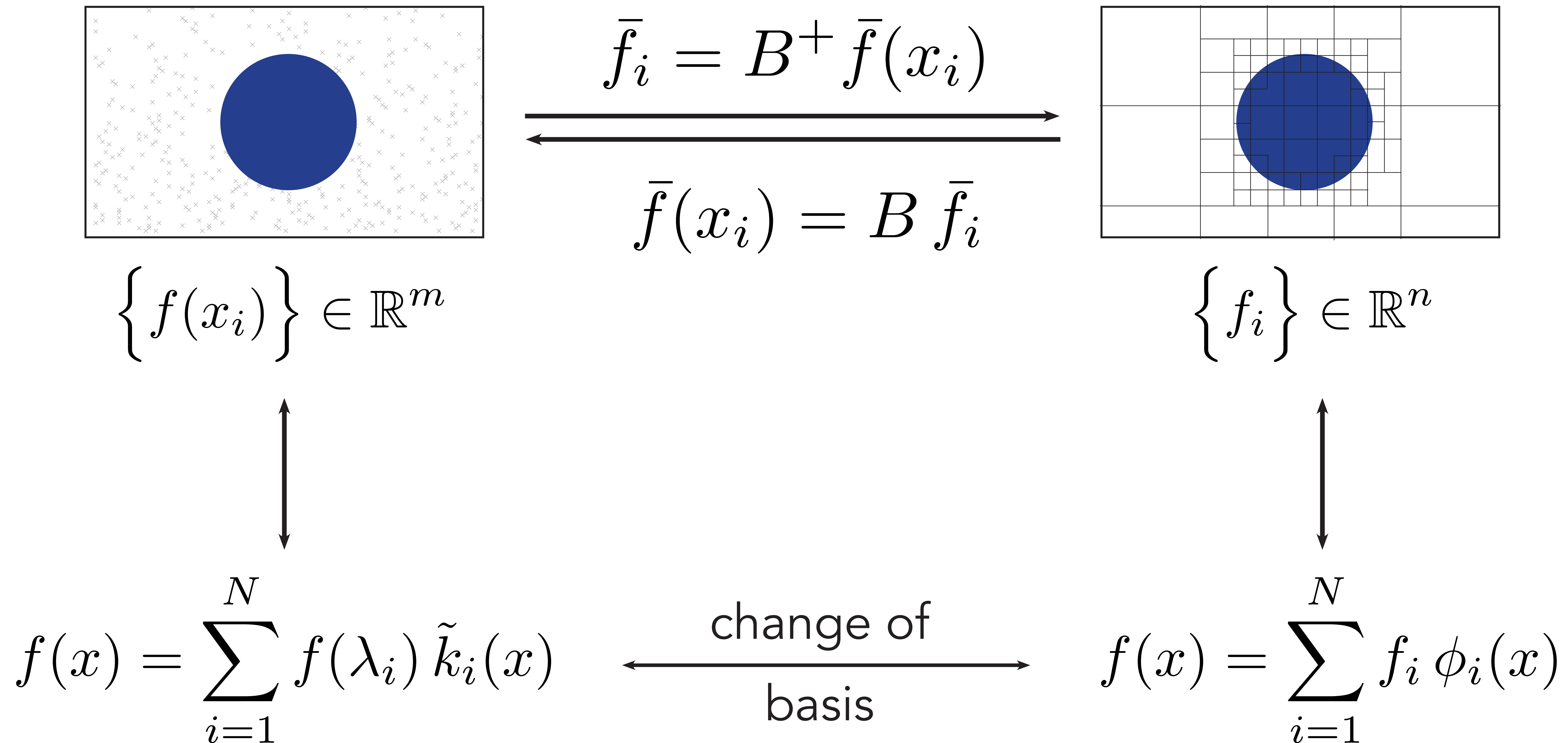
$$f(x) = \sum_{i=1}^N f_i \phi_i(x)$$

Reproducing kernel Hilbert spaces

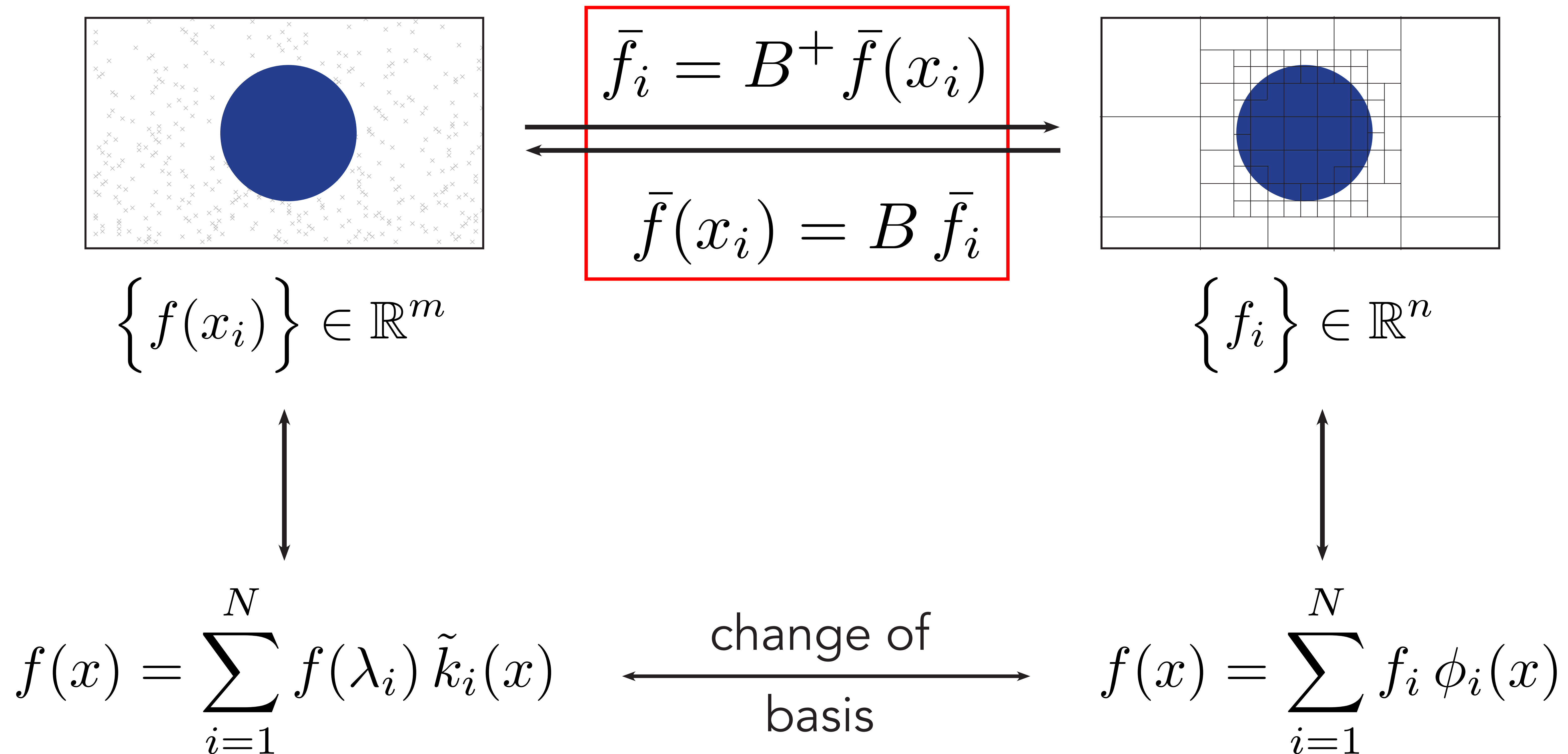


$$f(x) = \sum_{i=1}^N f(\lambda_i) \tilde{k}_i(x) \xrightarrow[\text{basis}]{\text{change of}} f(x) = \sum_{i=1}^N f_i \phi_i(x)$$

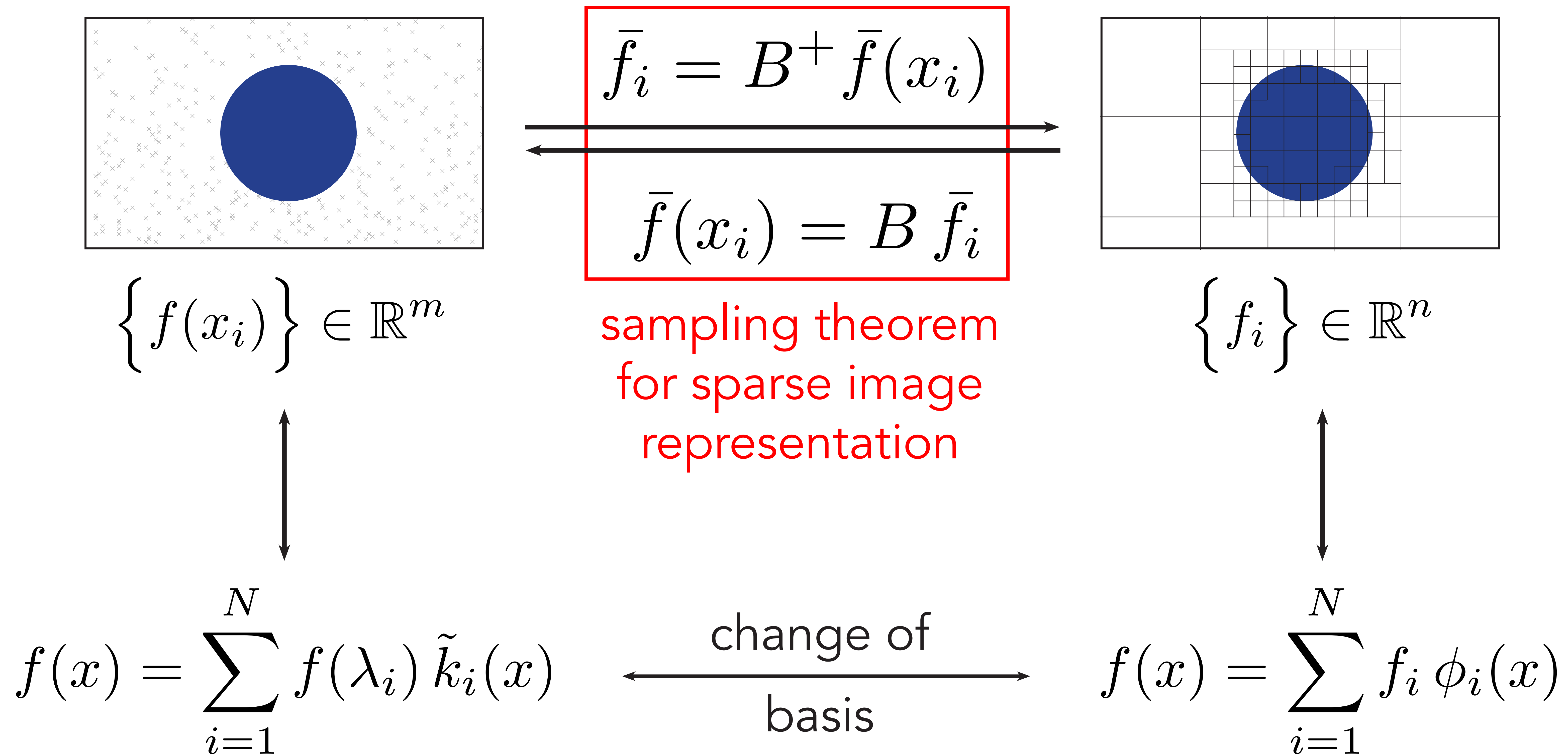
Reproducing kernel Hilbert spaces



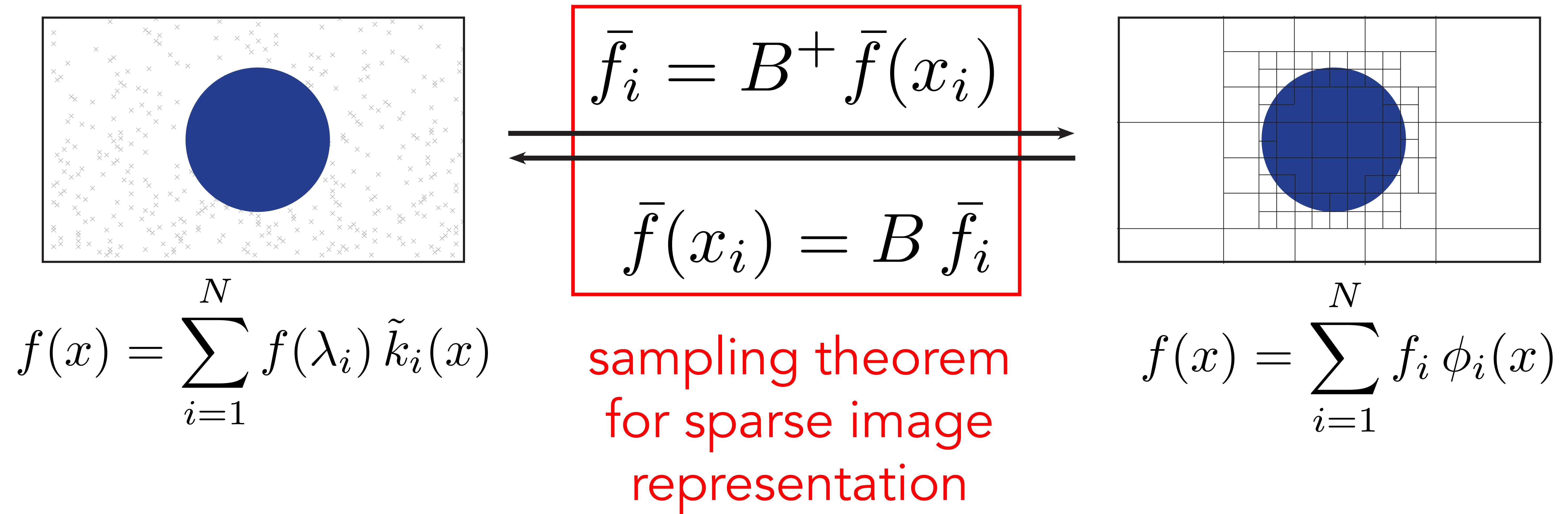
Reproducing kernel Hilbert spaces



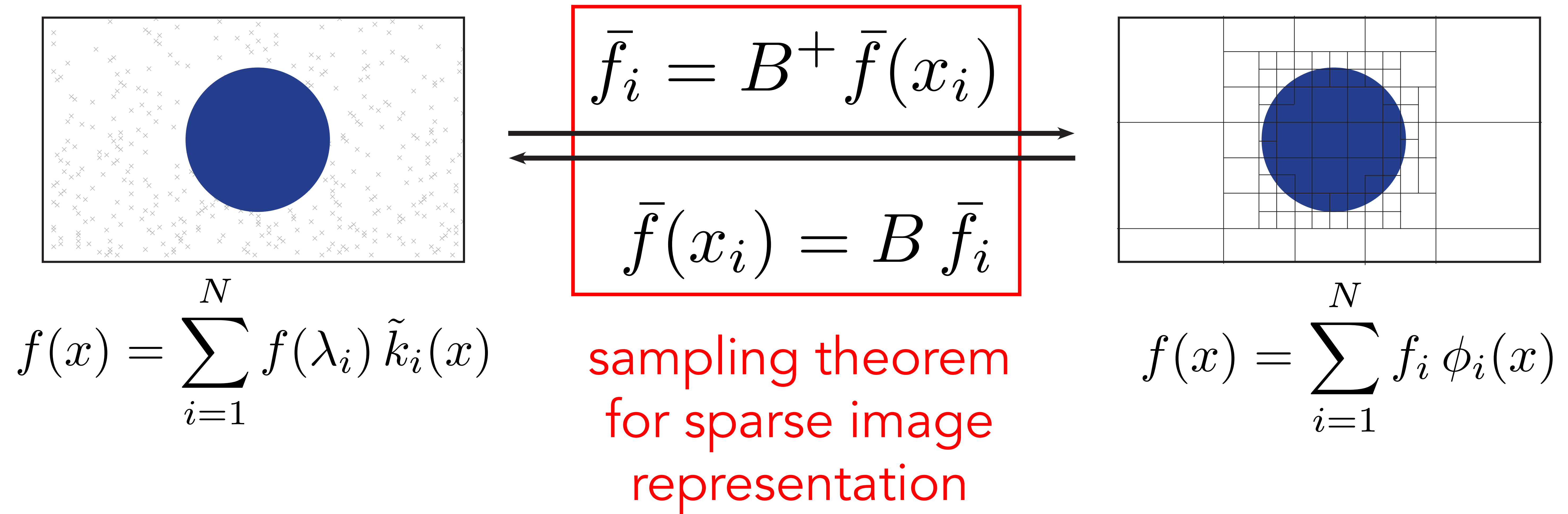
Reproducing kernel Hilbert spaces



Reproducing kernel Hilbert spaces

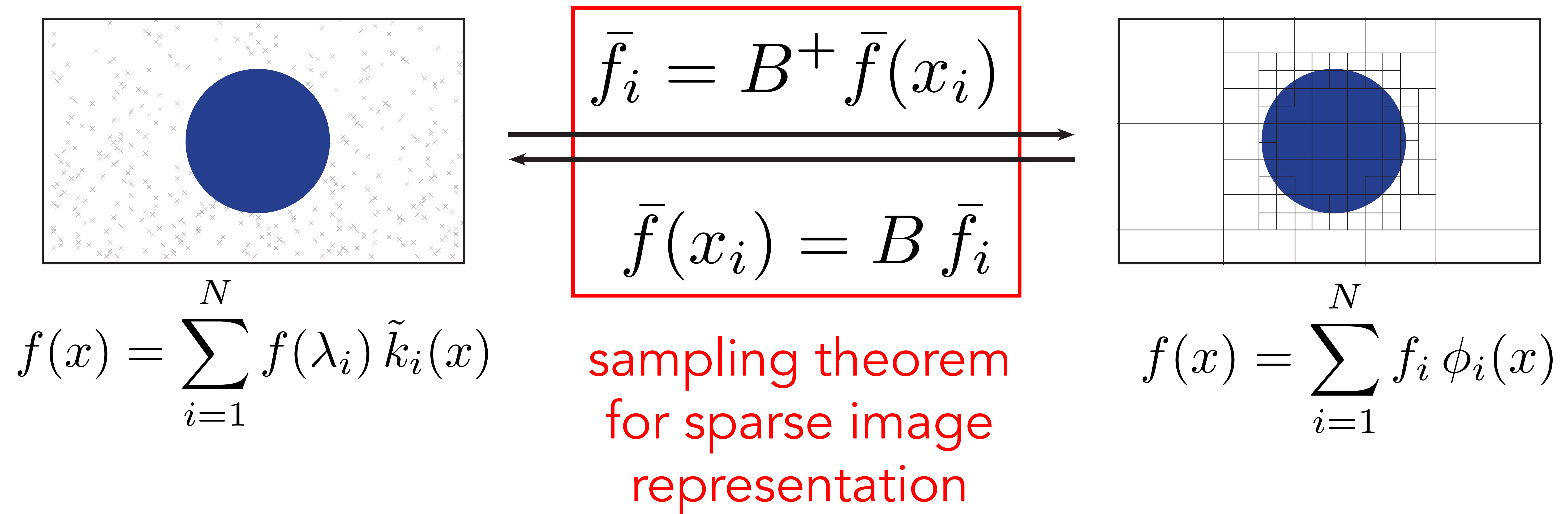


Reproducing kernel Hilbert spaces



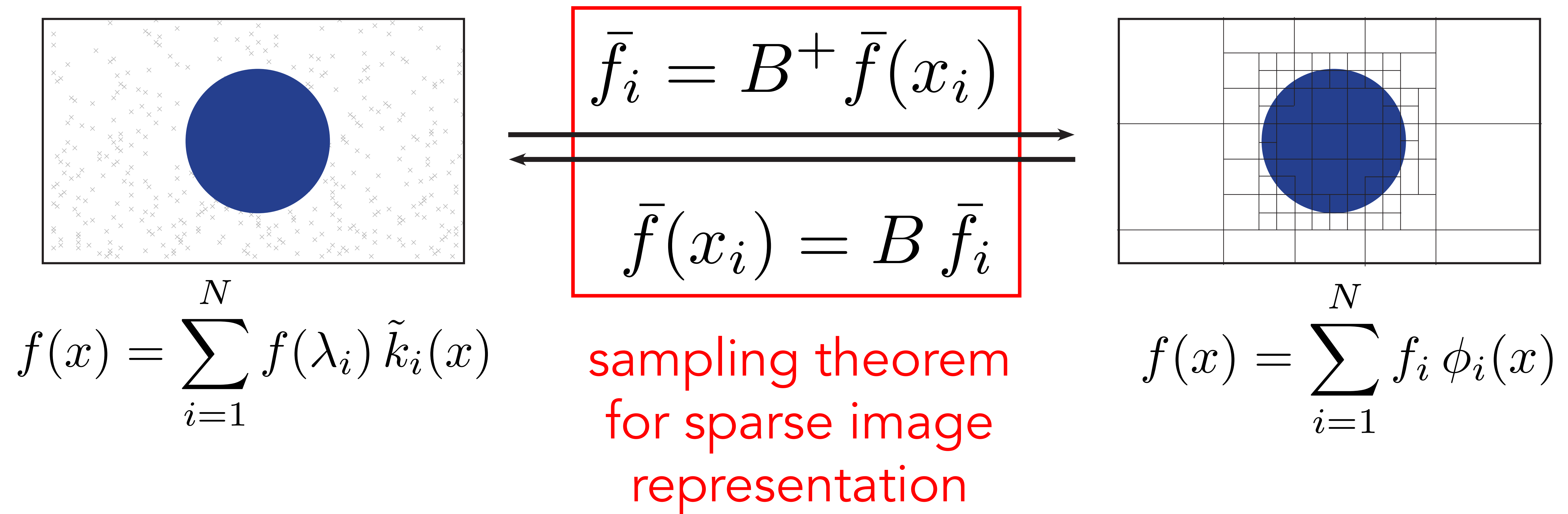
- optimal reconstruction kernels

Reproducing kernel Hilbert spaces



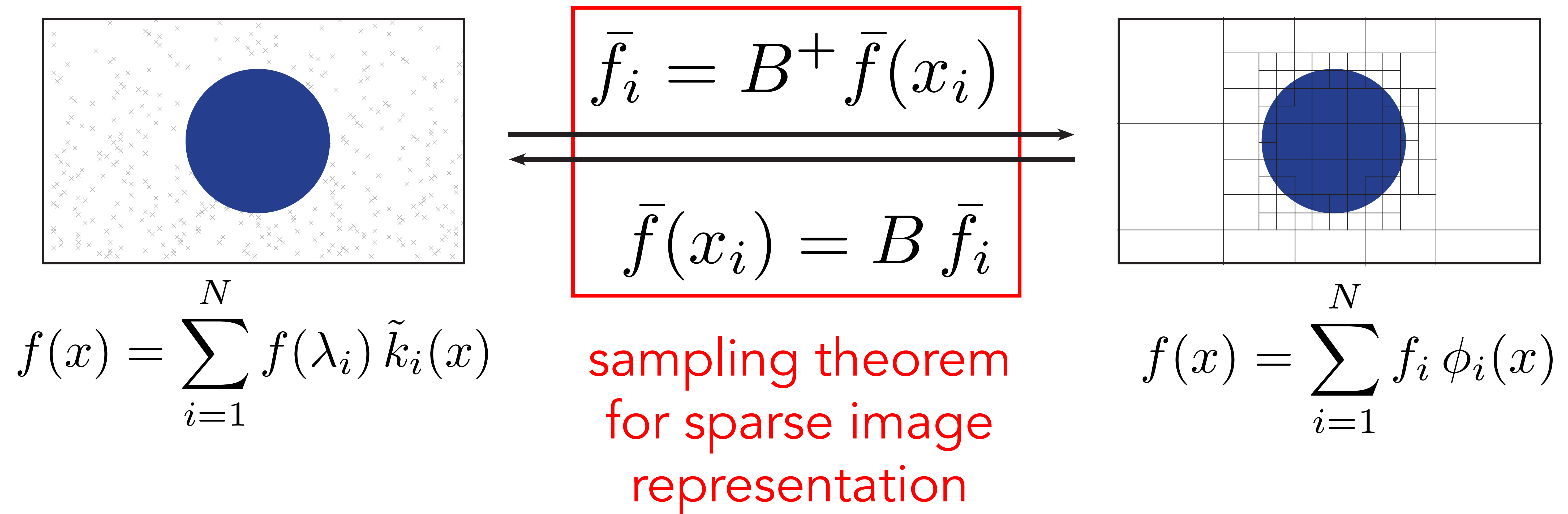
- optimal reconstruction kernels
- designed for non-bandlimited signals

Reproducing kernel Hilbert spaces



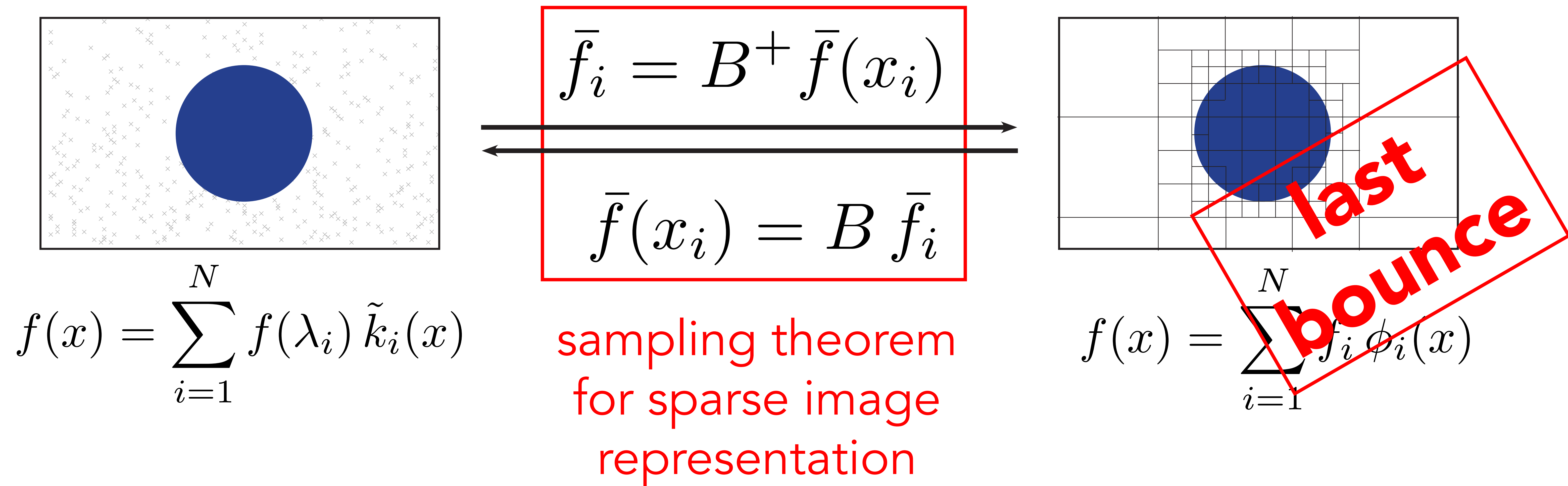
- optimal reconstruction kernels
- designed for non-bandlimited signals
- designed for non-uniform locations

Reproducing kernel Hilbert spaces



- optimal reconstruction kernels
- designed for non-bandlimited signals
- designed for non-uniform locations
- allows to optimize sampling locations

Reproducing kernel Hilbert spaces

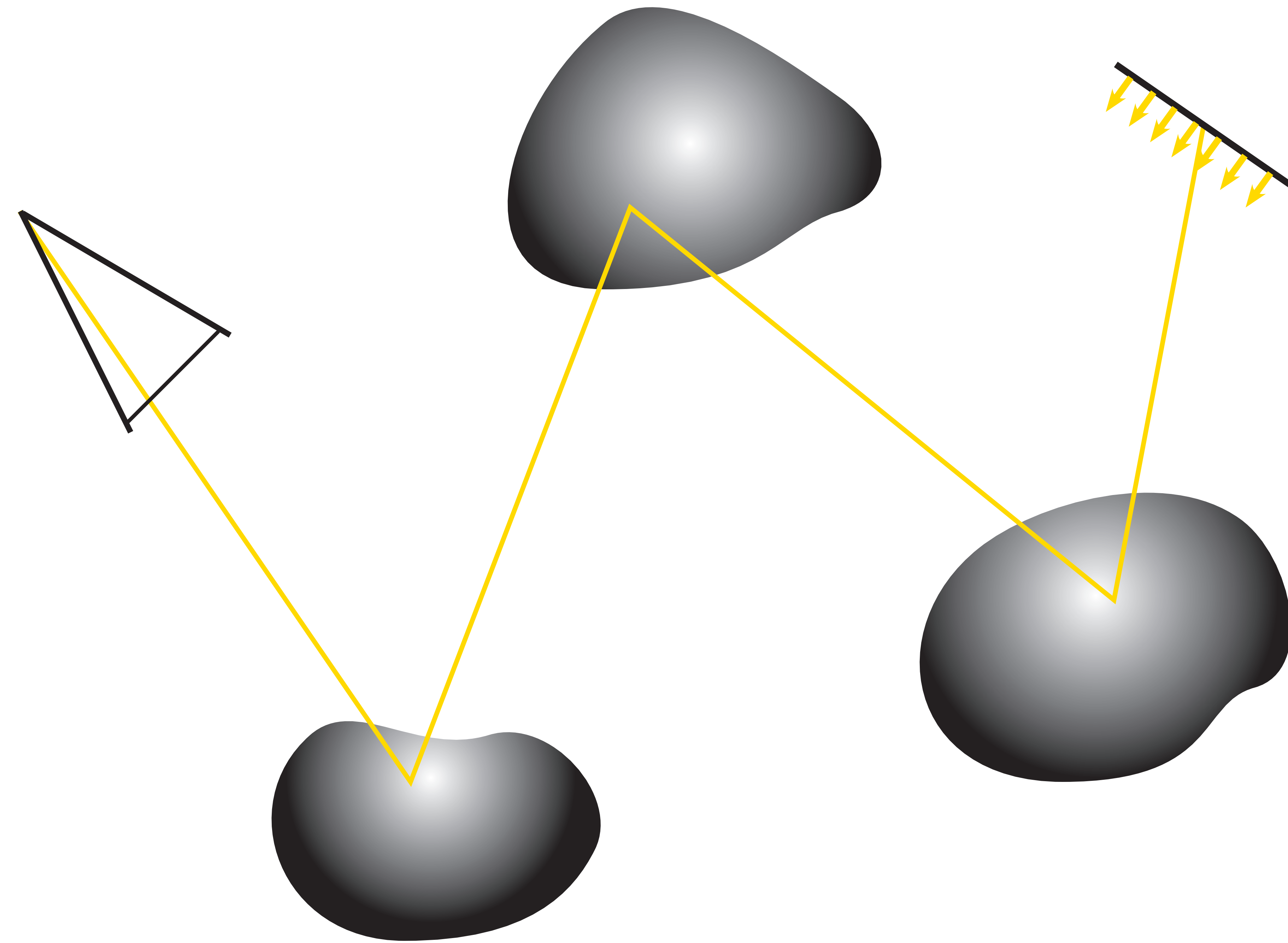


- optimal reconstruction kernels
- designed for non-bandlimited signals
- designed for non-uniform locations
- allows to optimize sampling locations

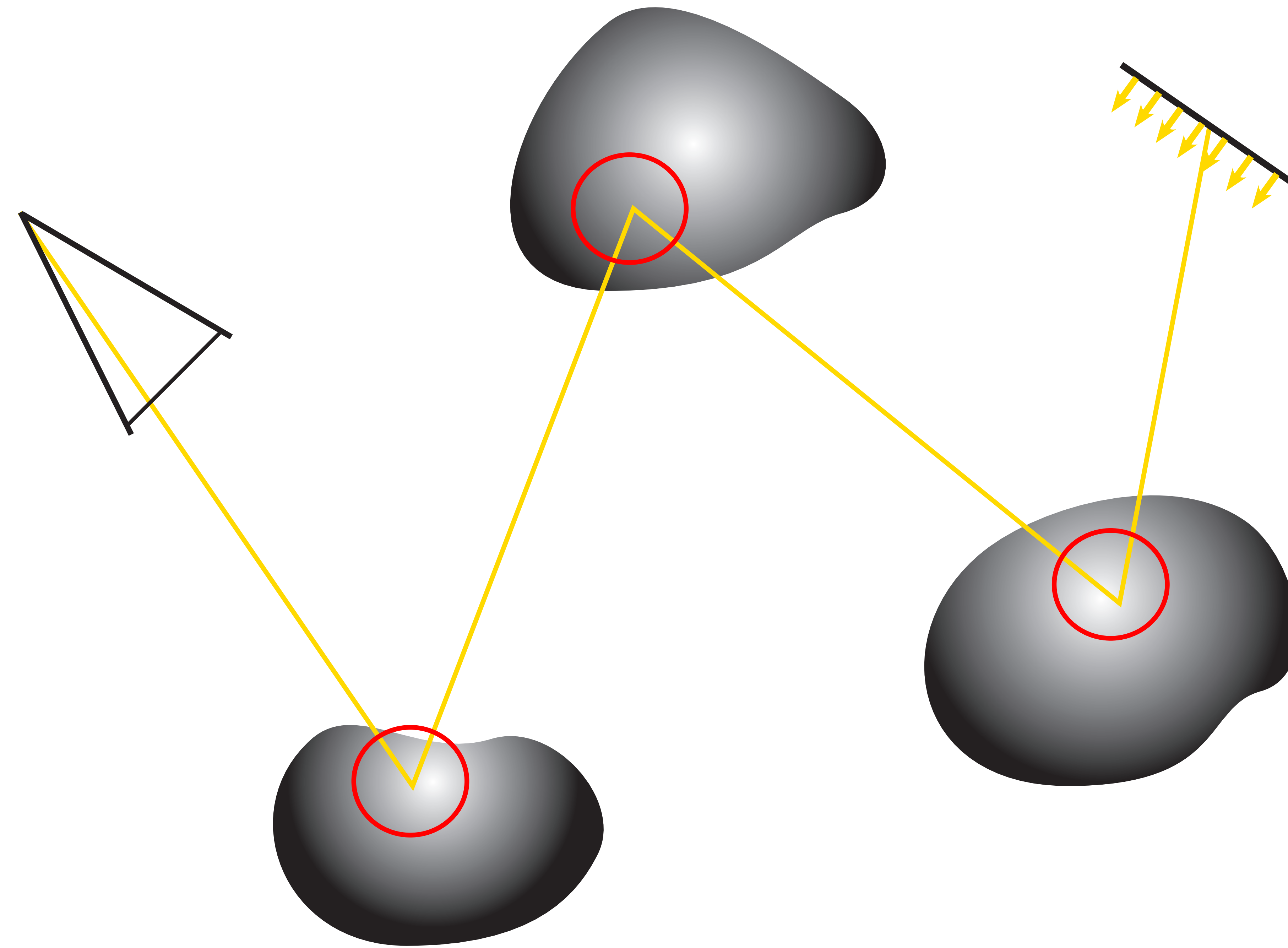
Recipe

1. Find (approximate) tight function space
2. Sample function
3. Construct reproducing kernel basis

Recipe



Recipe



Reproducing kernel Hilbert spaces

$$\int_X f(x) \, dx$$

Reproducing kernel Hilbert spaces

$$\int_X f(x) \, dx = \int_X \sum_{i=1}^N f(x_i) \tilde{k}_i(x) \, dx$$

Reproducing kernel Hilbert spaces

$$\begin{aligned}\int_X f(x) \, dx &= \int_X \sum_{i=1}^N f(x_i) \tilde{k}_i(x) \, dx \\ &= \sum_{i=1}^N f(x_i) \int_X \tilde{k}_i(x) \, dx\end{aligned}$$

Reproducing kernel Hilbert spaces

$$\begin{aligned}\int_X f(x) dx &= \int_X \sum_{i=1}^N f(x_i) \tilde{k}_i(x) dx \\ &= \sum_{i=1}^N f(x_i) \underbrace{\int_X \tilde{k}_i(x) dx}_{w_i}\end{aligned}$$

Reproducing kernel Hilbert spaces

$$\begin{aligned}\int_X f(x) dx &= \int_X \sum_{i=1}^N f(x_i) \tilde{k}_i(x) dx \\ &= \sum_{i=1}^N f(x_i) \underbrace{\int_X \tilde{k}_i(x) dx}_{w_i} \\ &= \sum_{i=1}^N w_i f(x_i)\end{aligned}$$

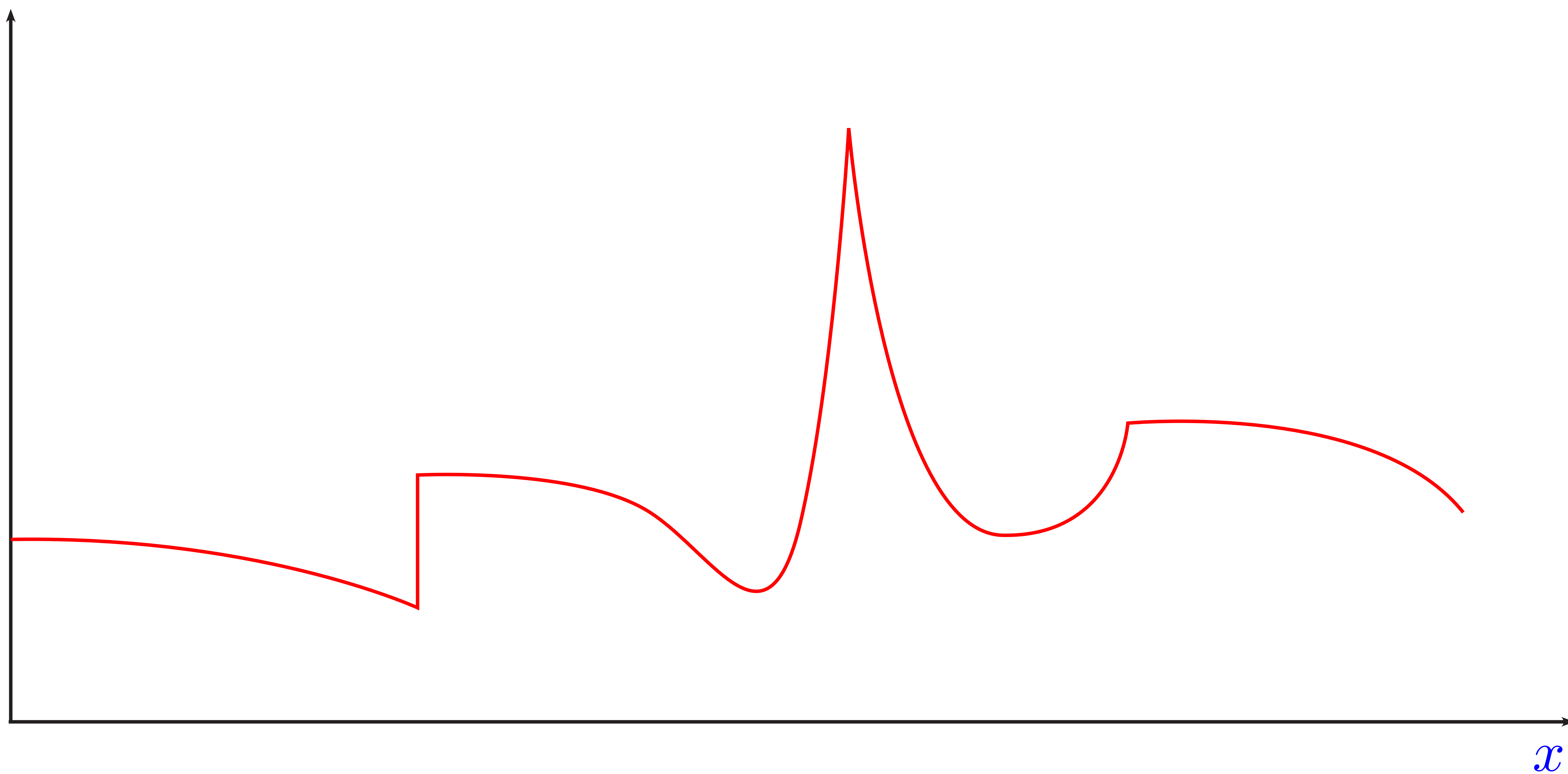
Recipe

1. Find (approximate) tight function space
2. Sample function
3. Construct reproducing kernel basis

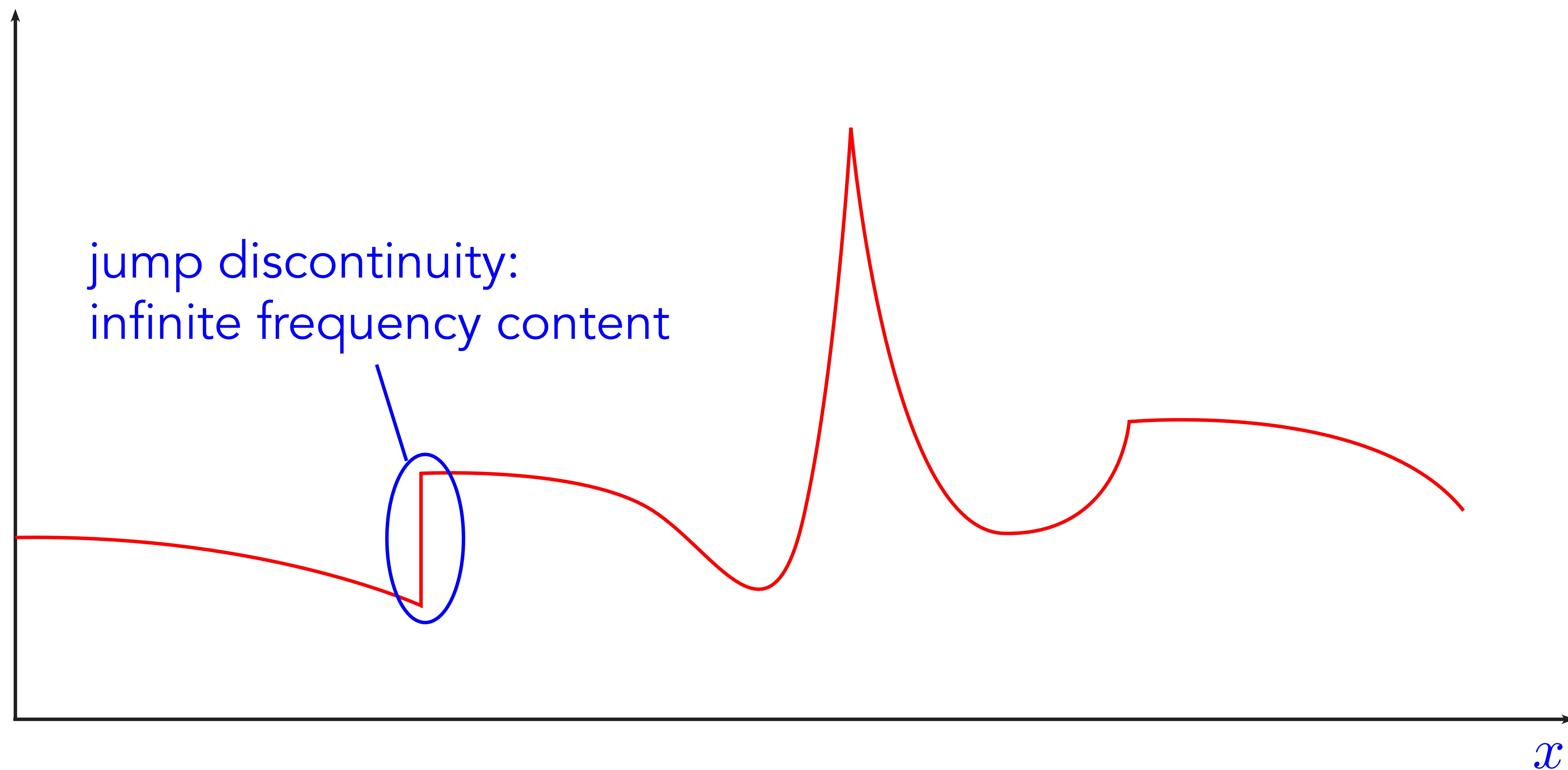
Recipe

1. Find (approximate) tight function space
2. Sample function
3. Construct reproducing kernel basis

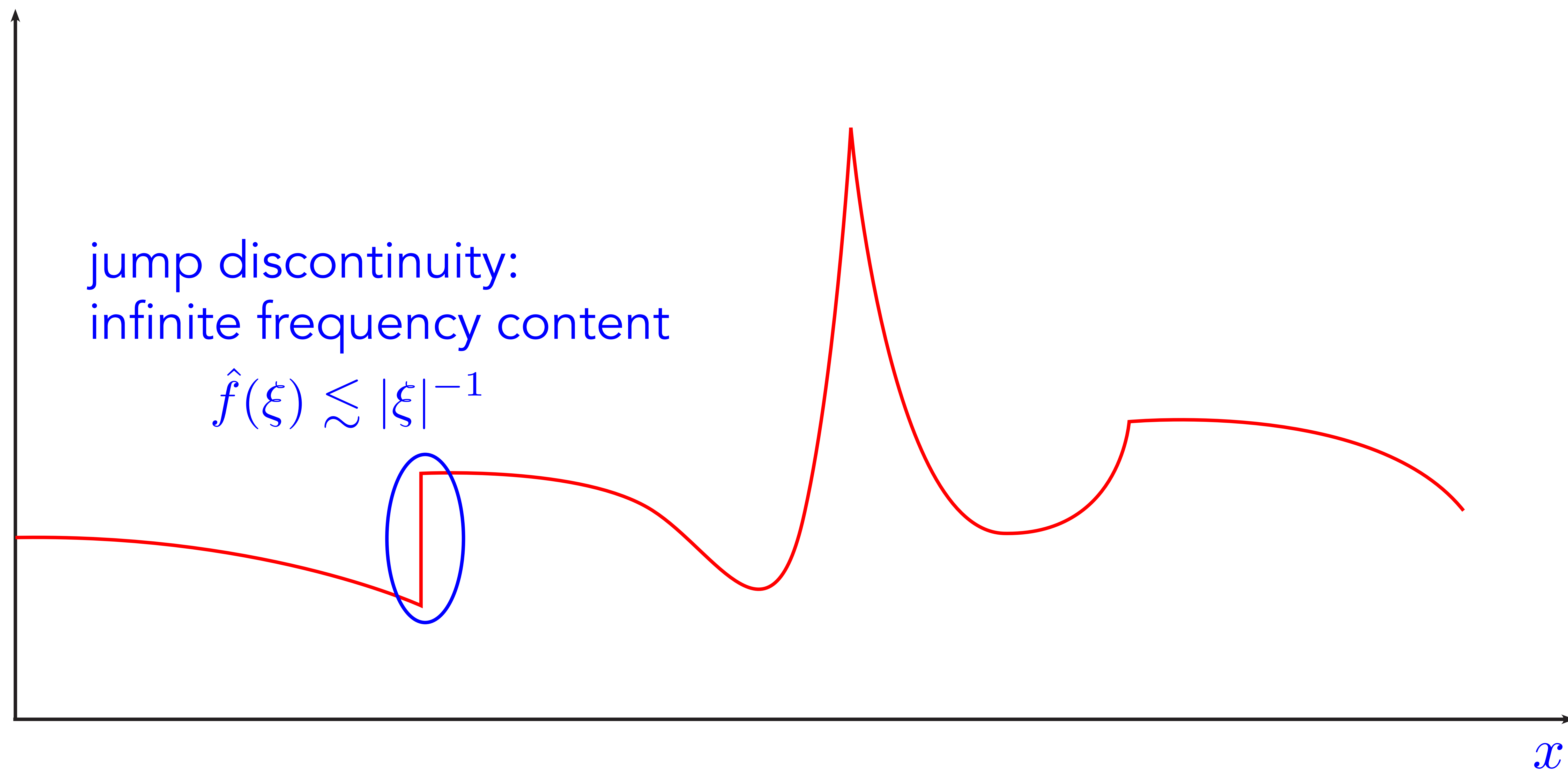
Tight function spaces



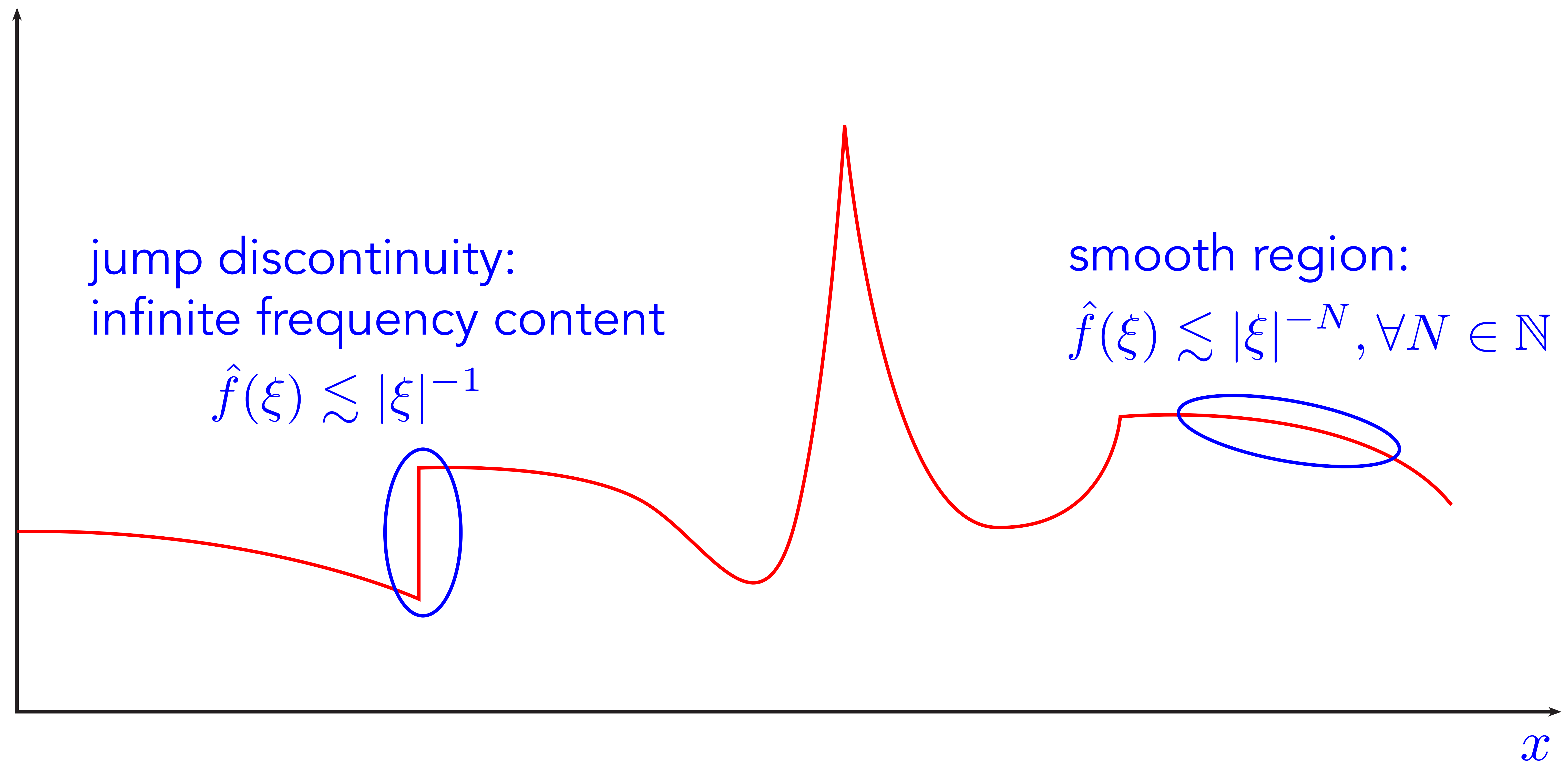
Tight function spaces



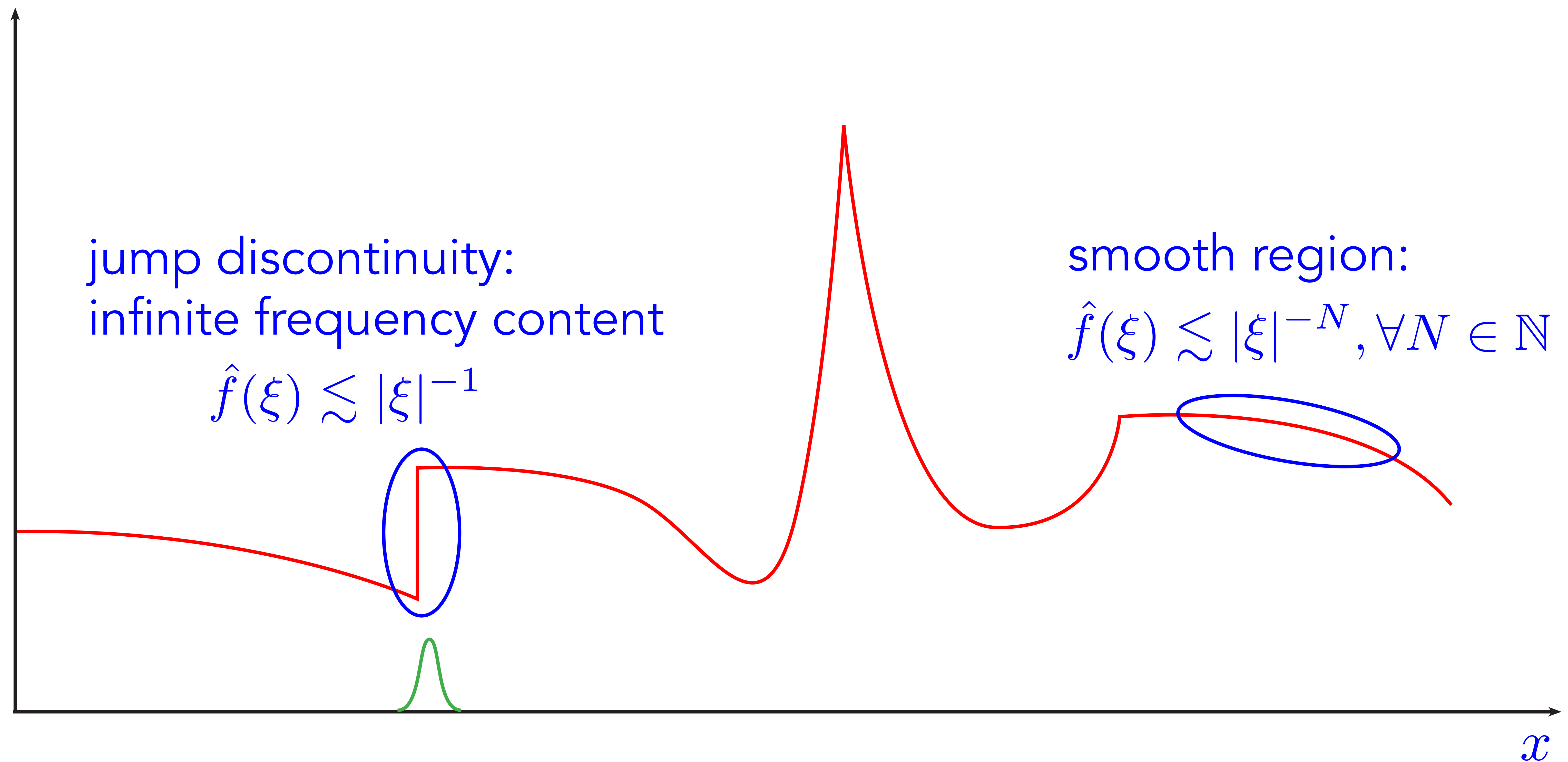
Tight function spaces



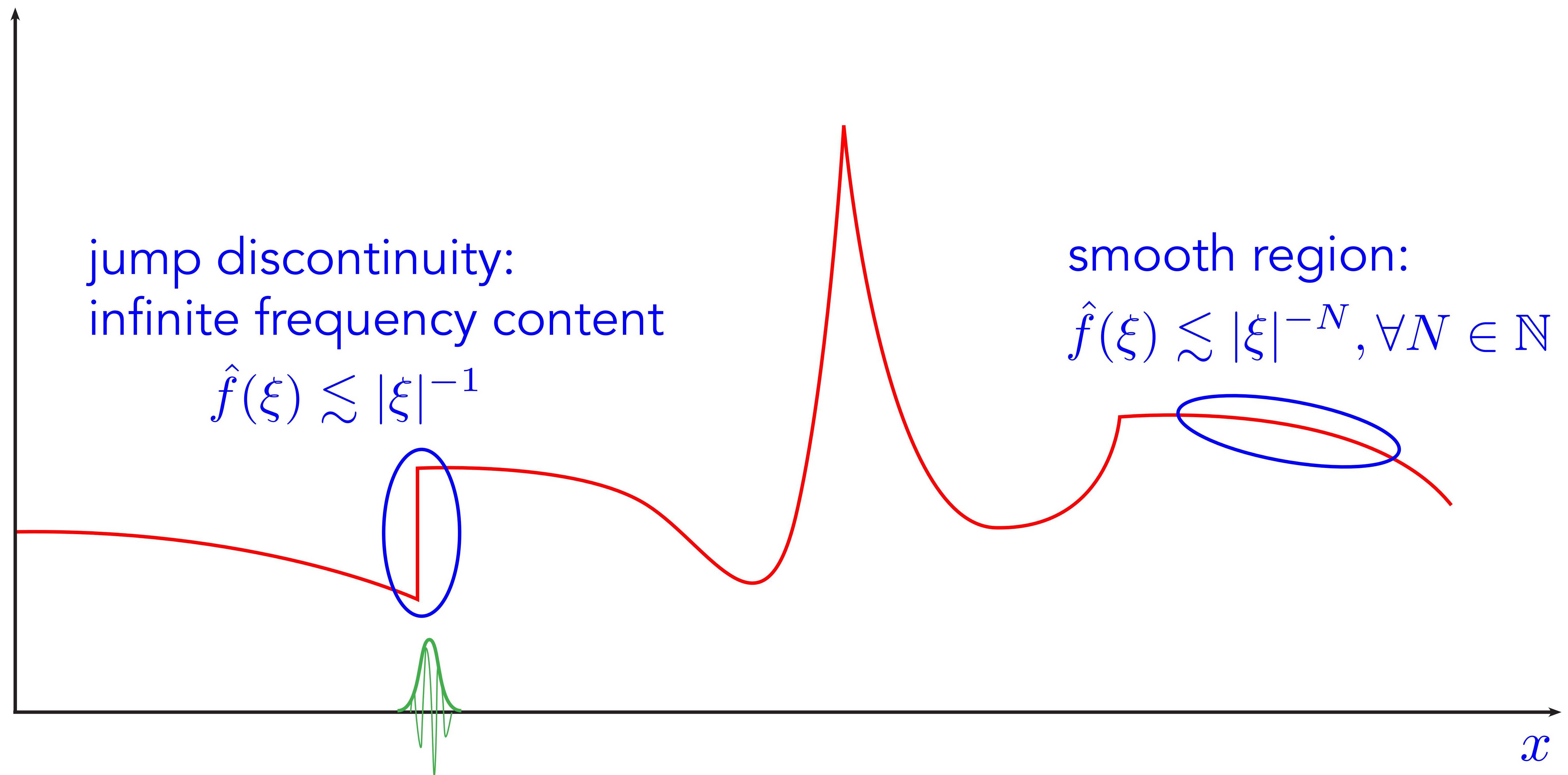
Tight function spaces



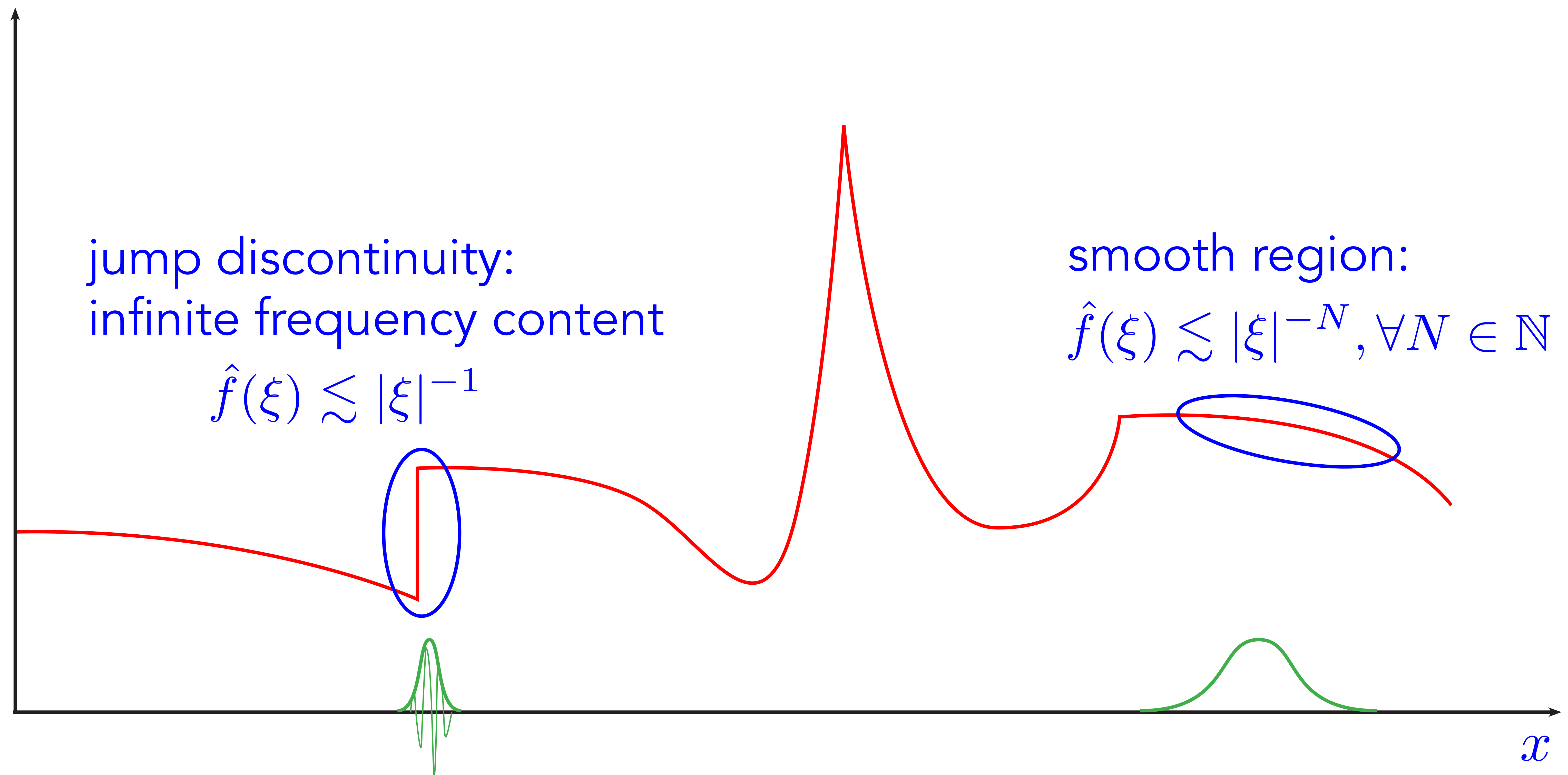
Tight function spaces



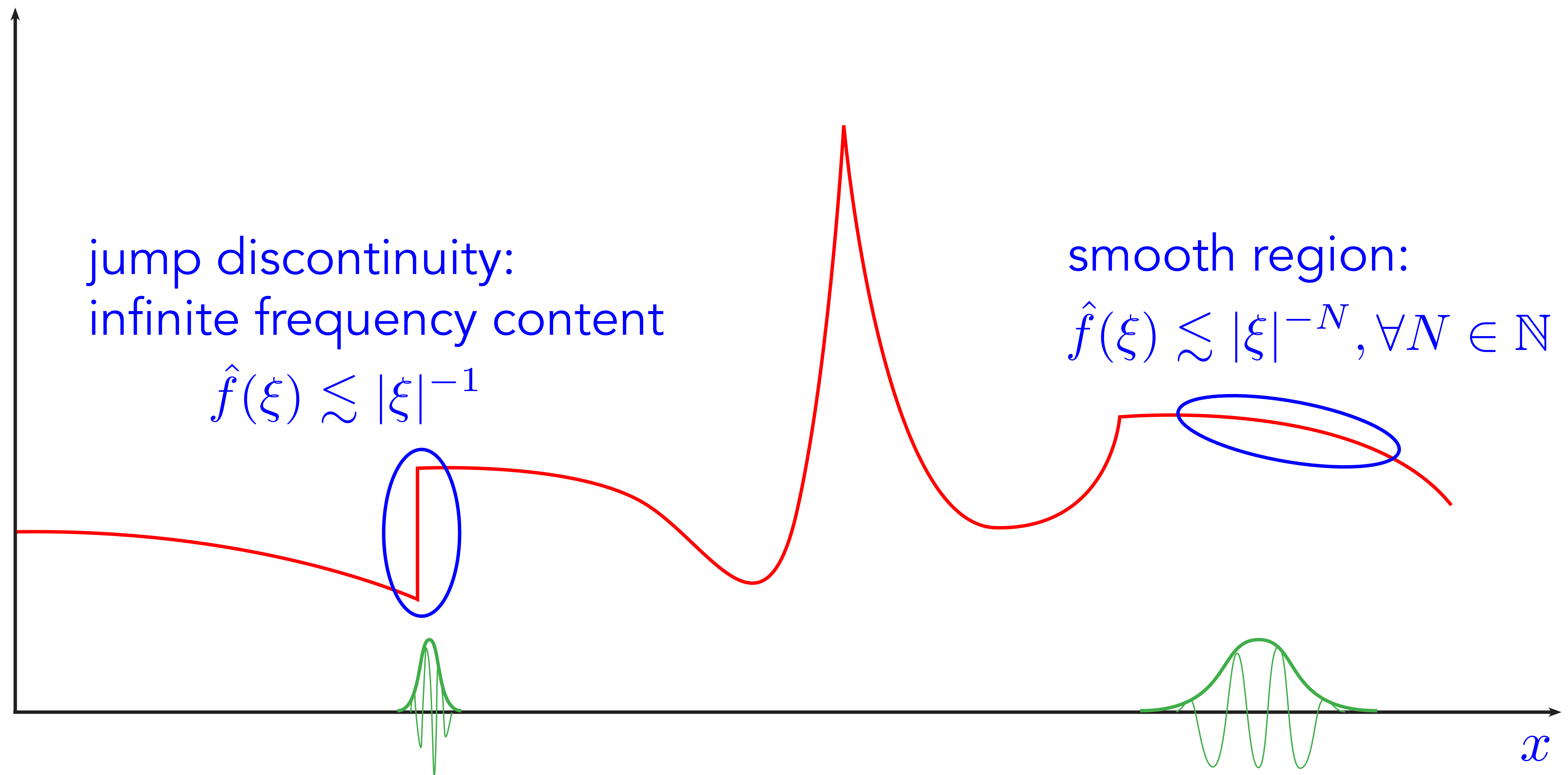
Tight function spaces



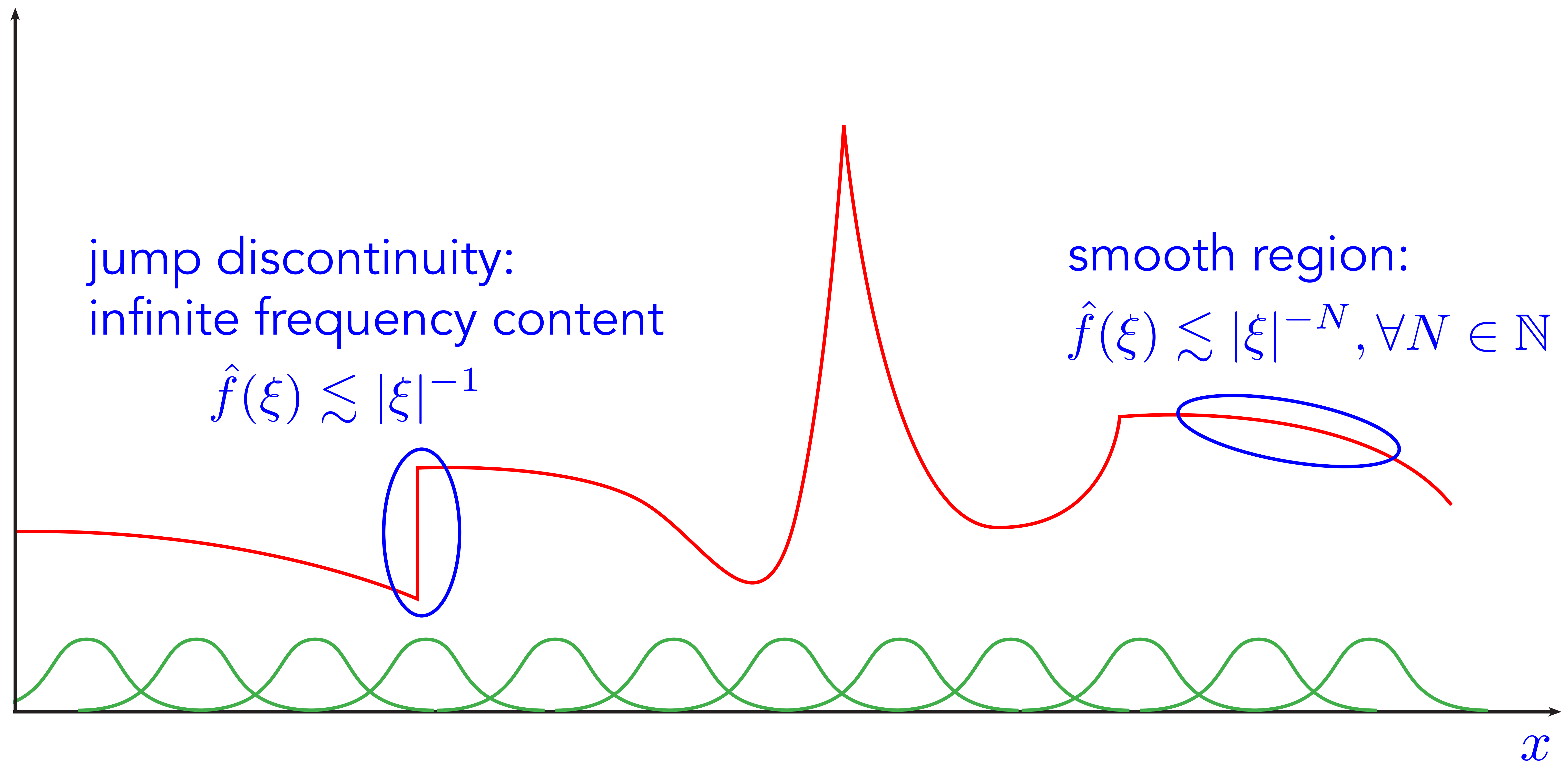
Tight function spaces



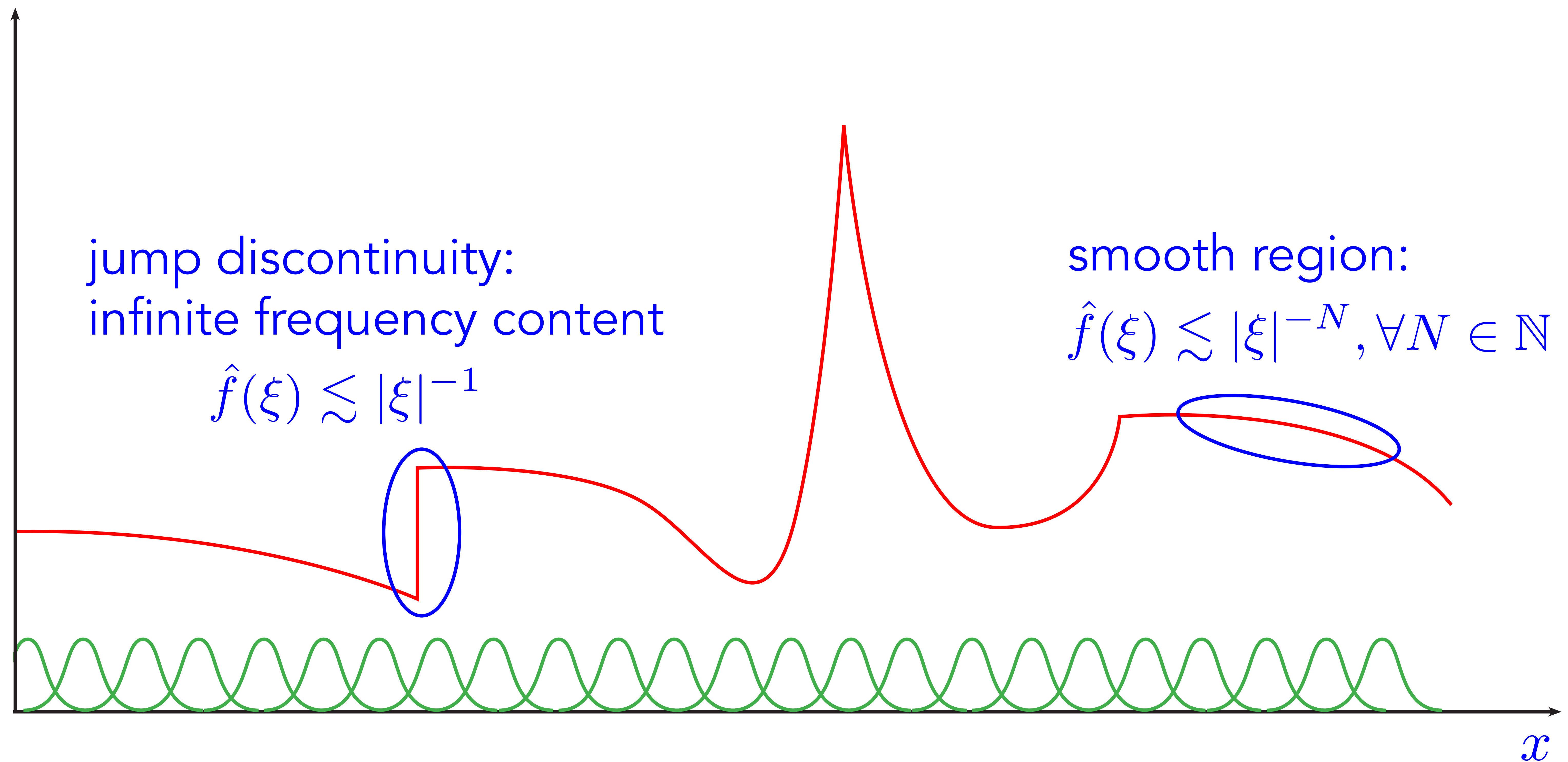
Tight function spaces



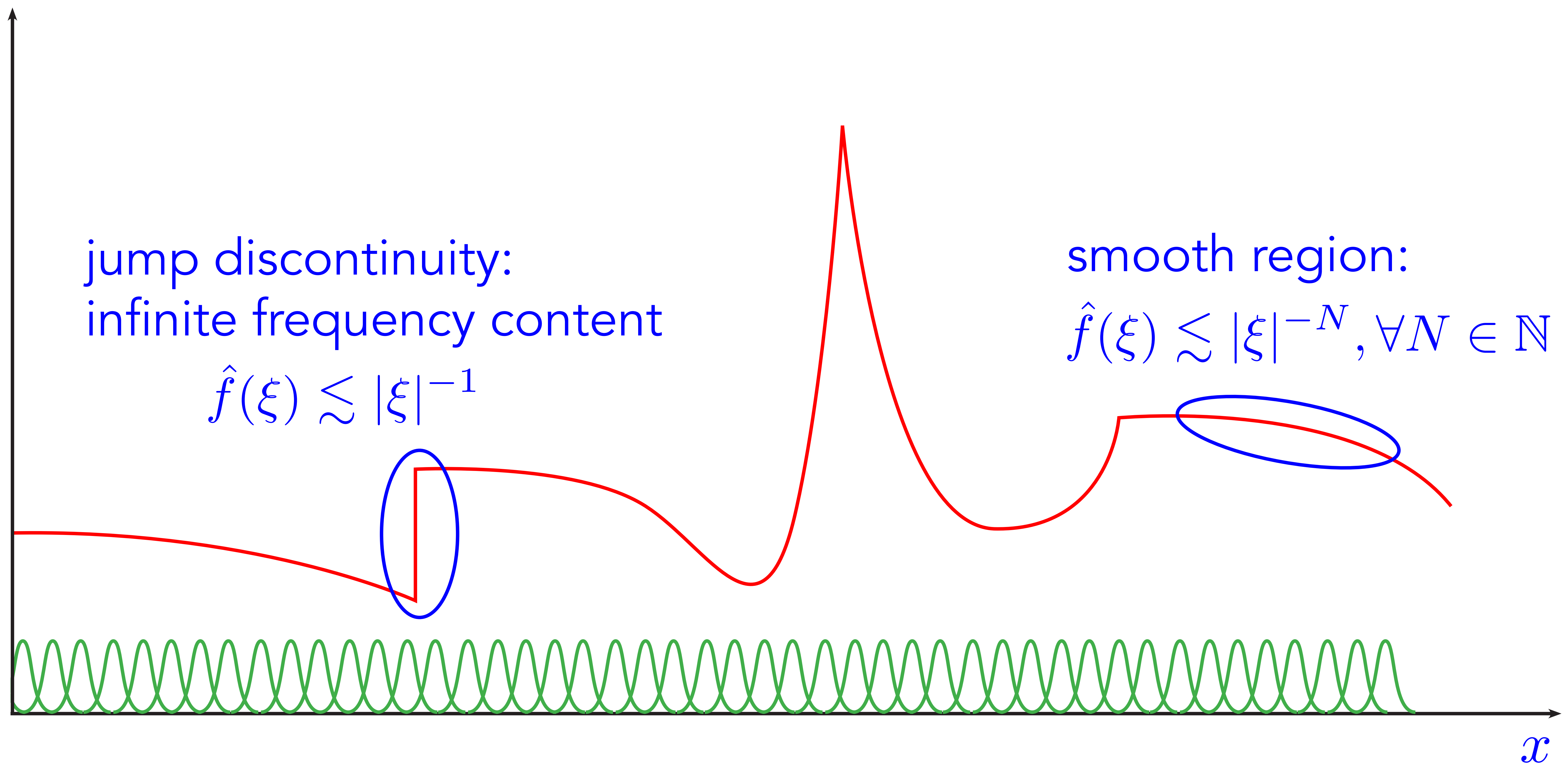
Tight function spaces



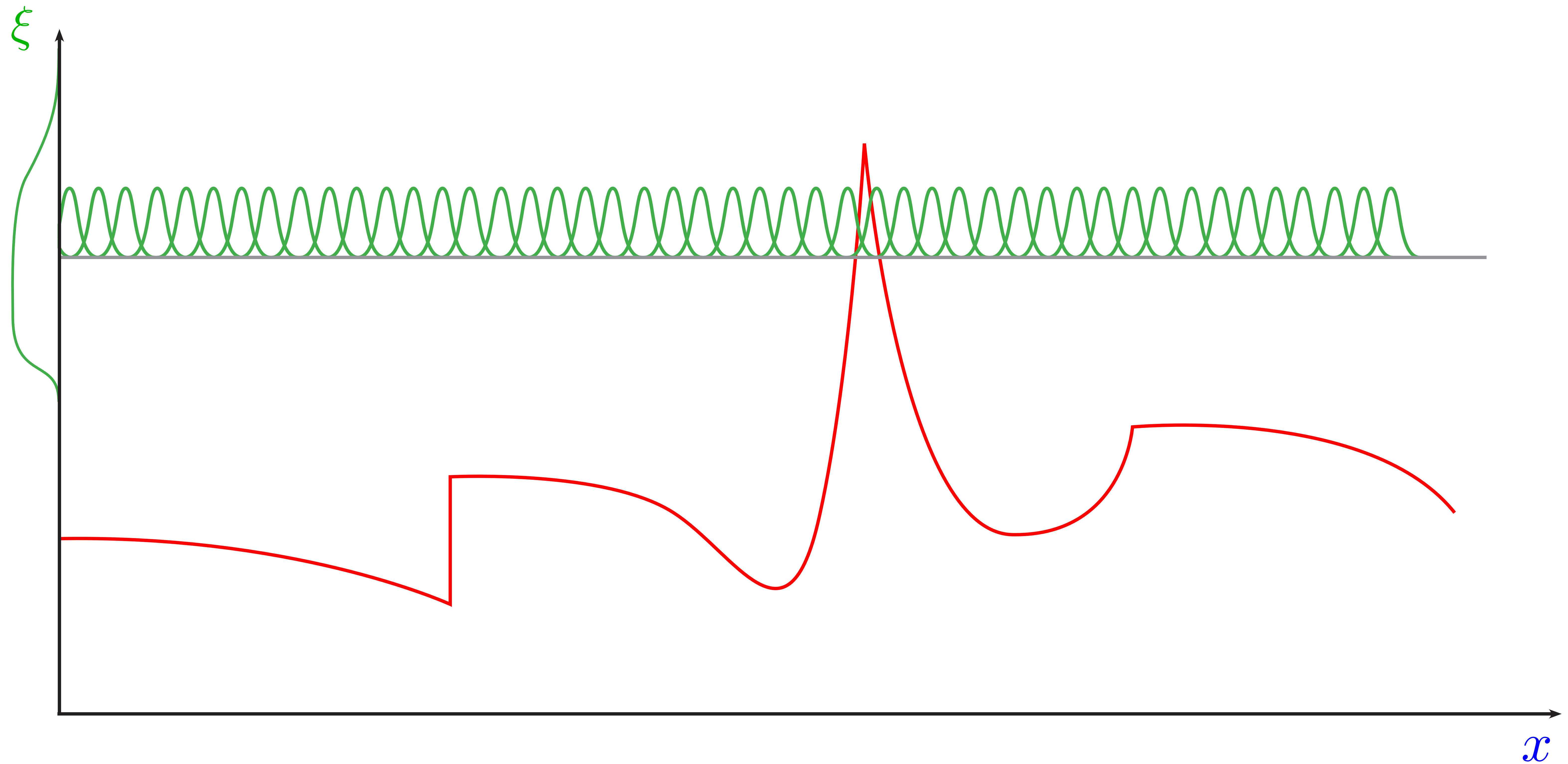
Tight function spaces



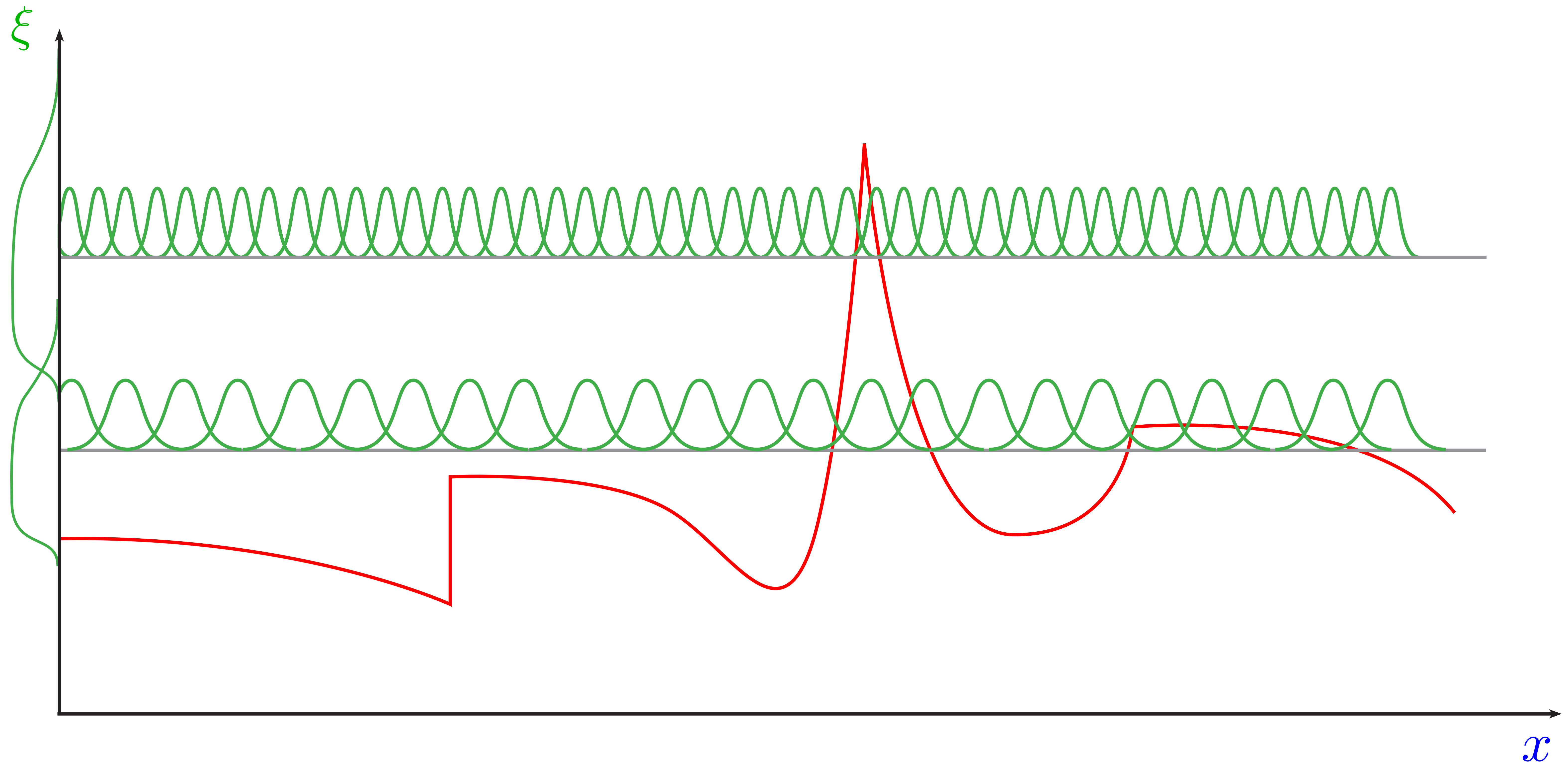
Tight function spaces



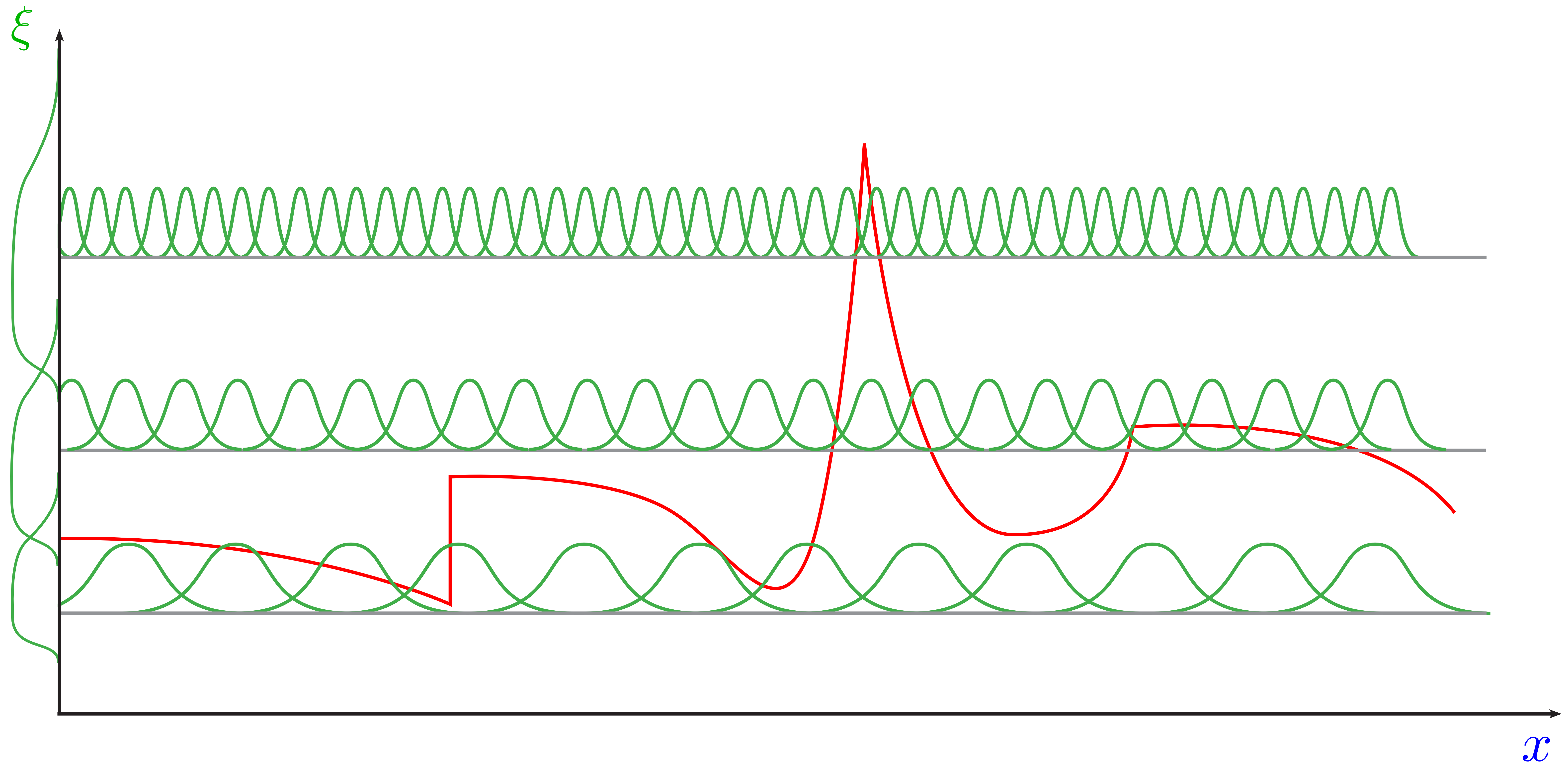
Tight function spaces



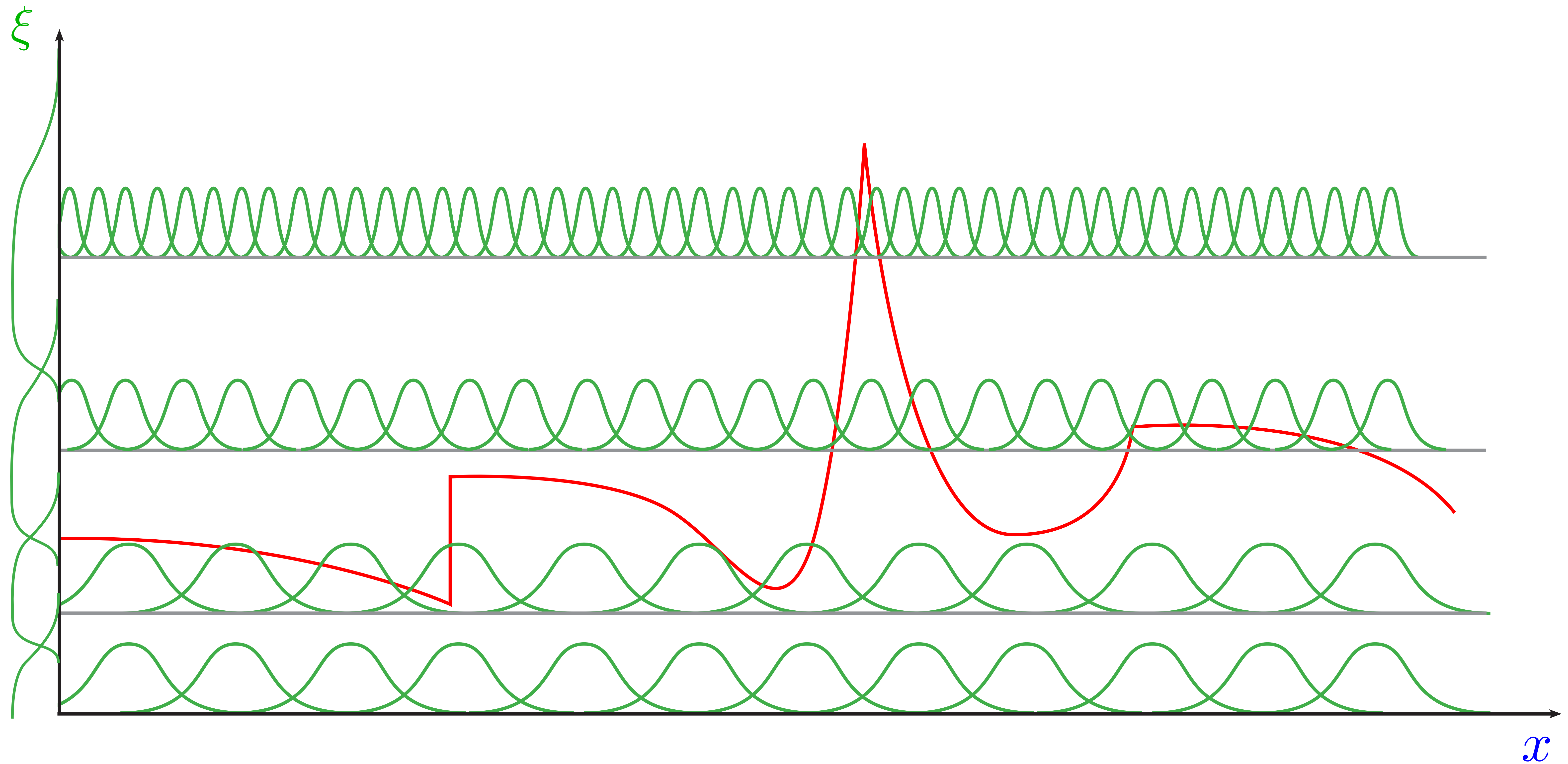
Tight function spaces



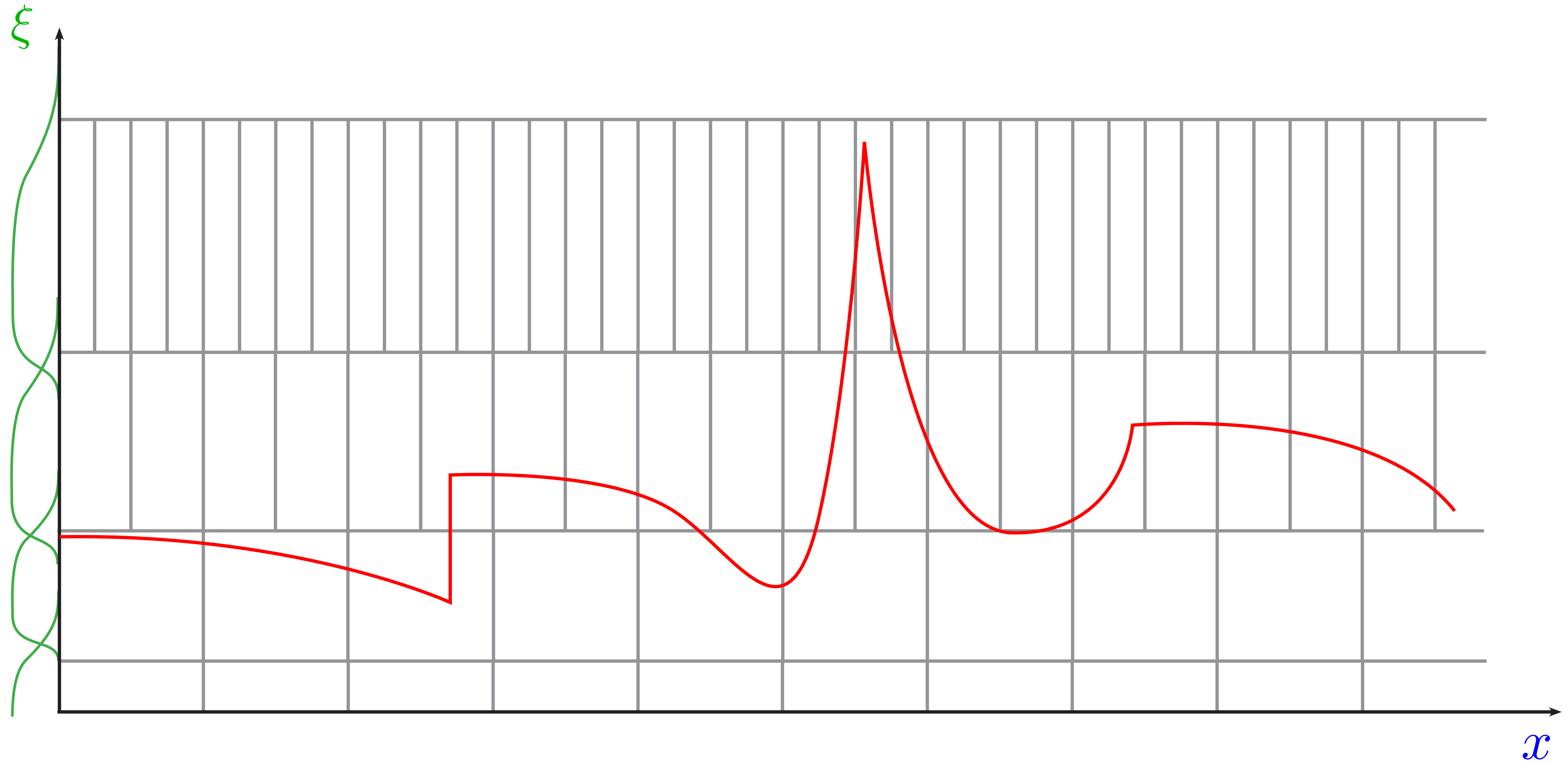
Tight function spaces



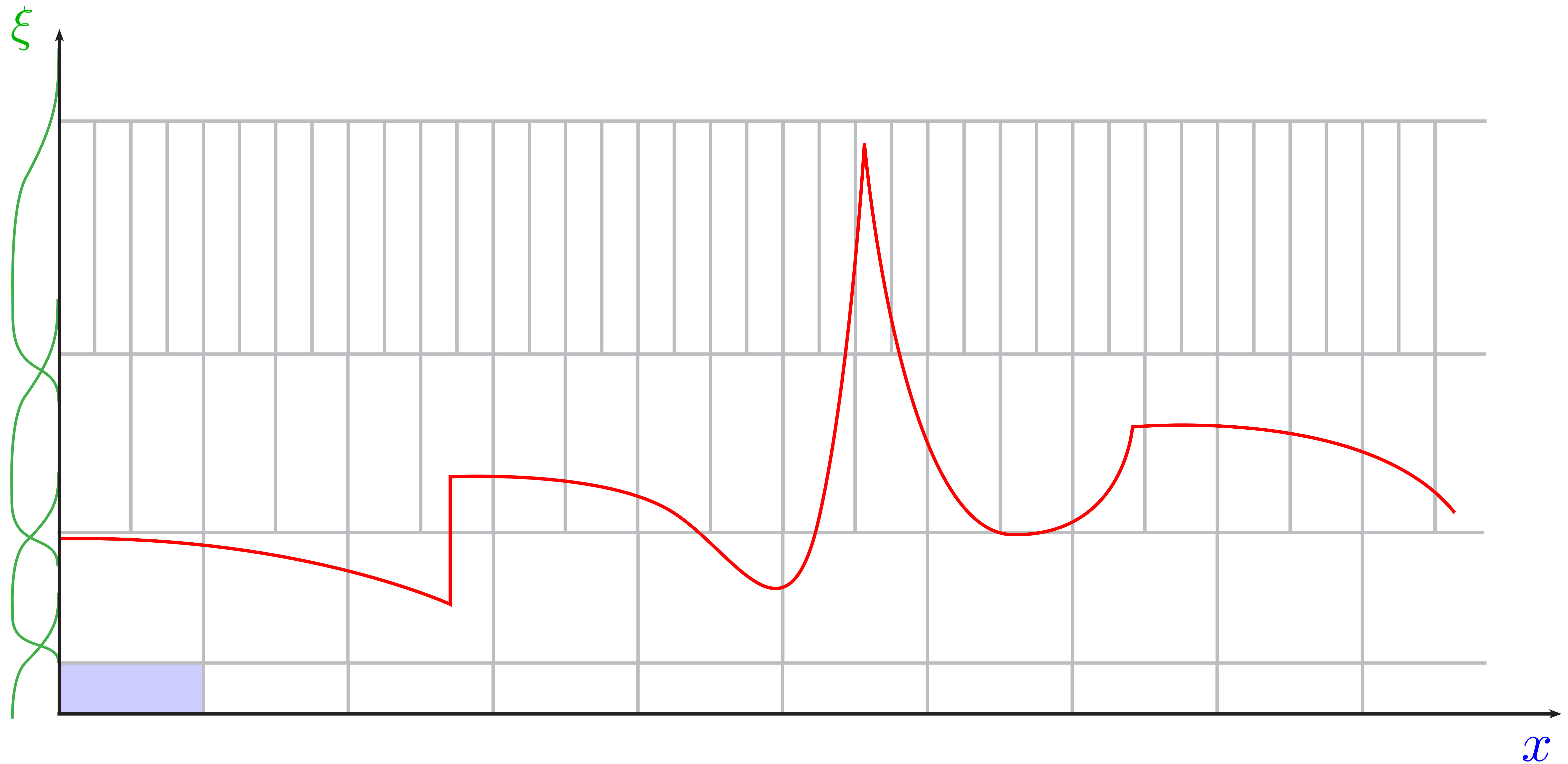
Tight function spaces



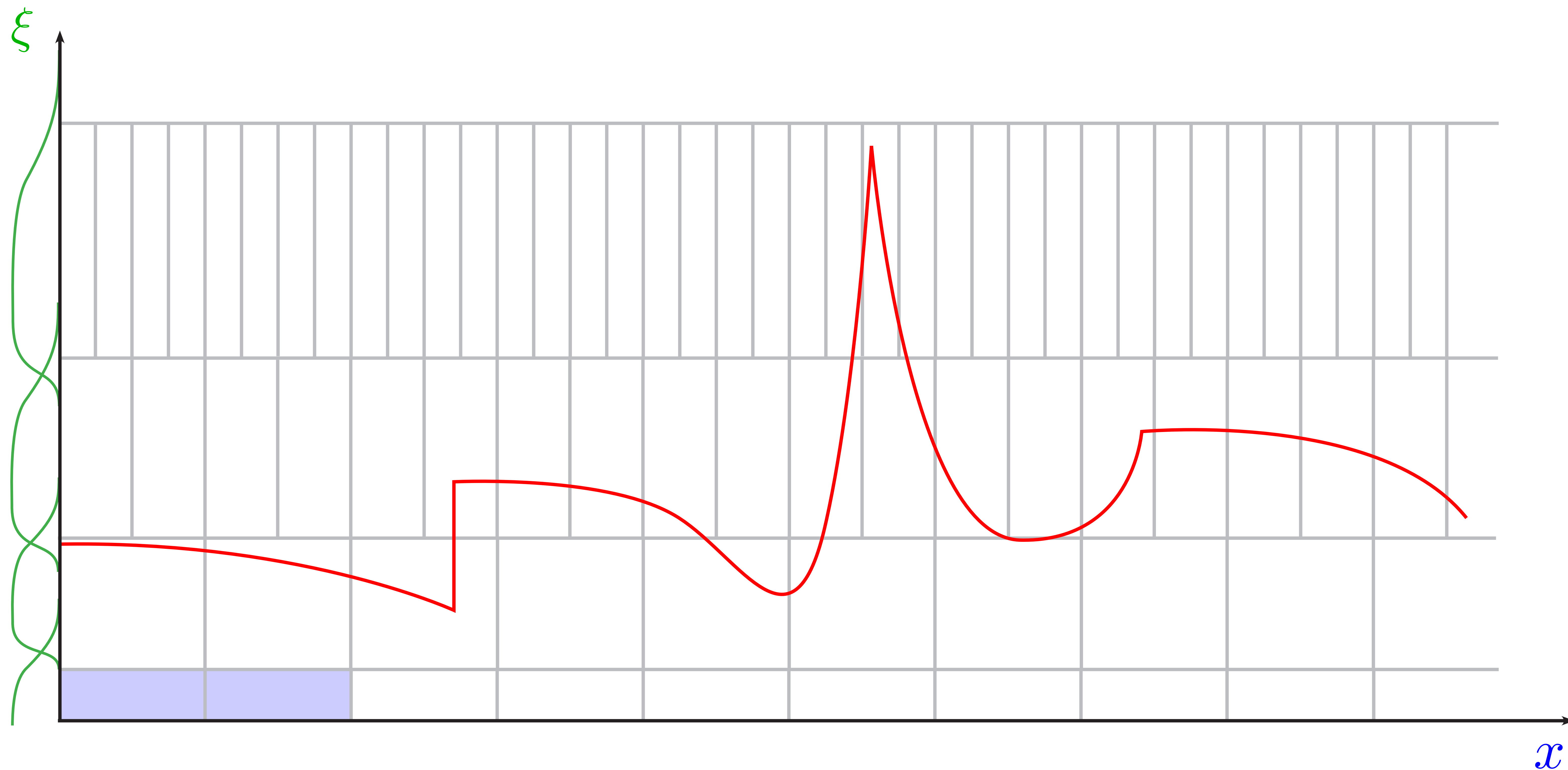
Tight function spaces



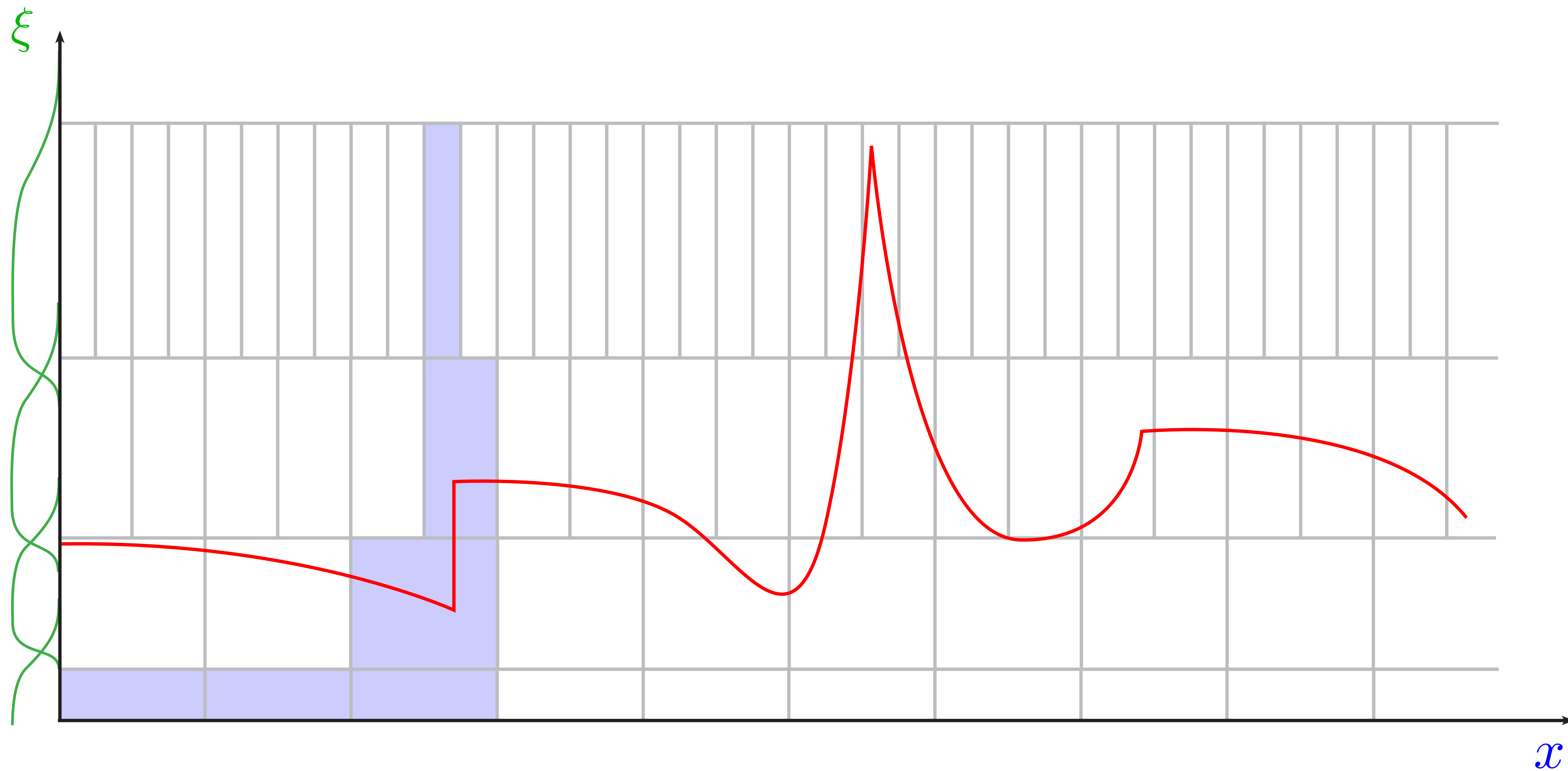
Tight function spaces



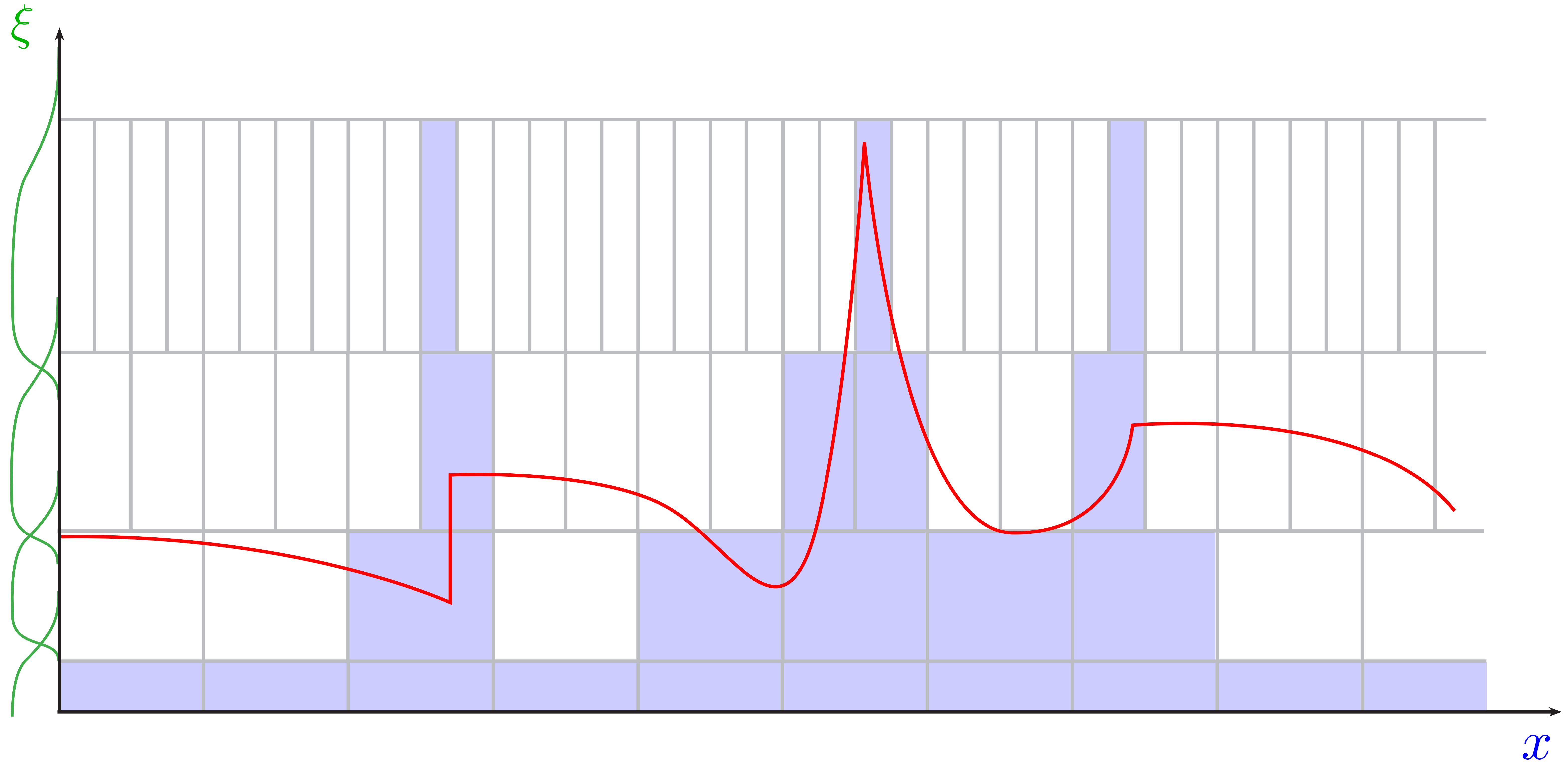
Tight function spaces



Tight function spaces



Tight function spaces

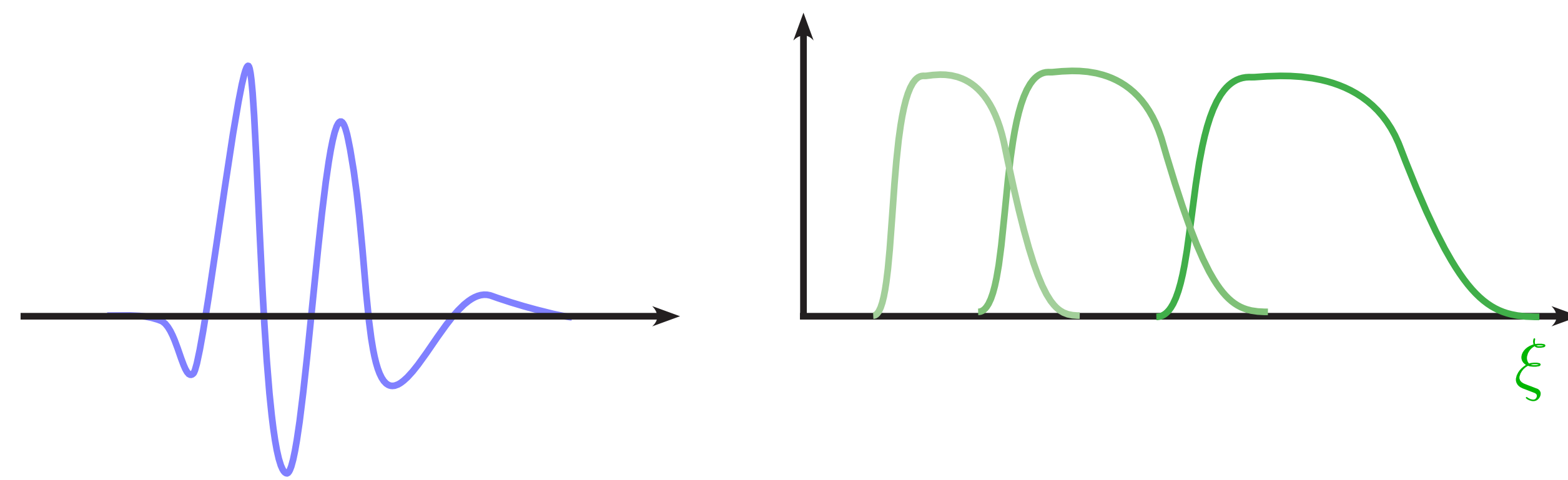


Tight function spaces



Tight function spaces

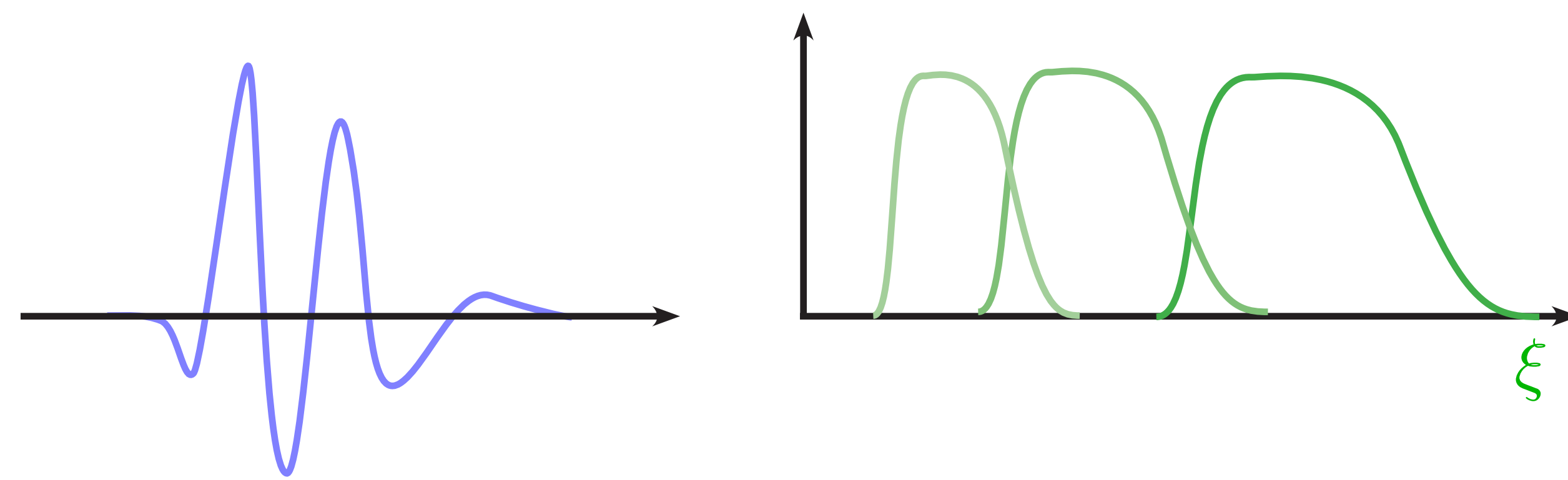
$$f(x) = \sum_{i \in \mathcal{I}} f_i \boxed{\psi_i(x)} \approx (2^{-l}k, 2^l\xi)$$



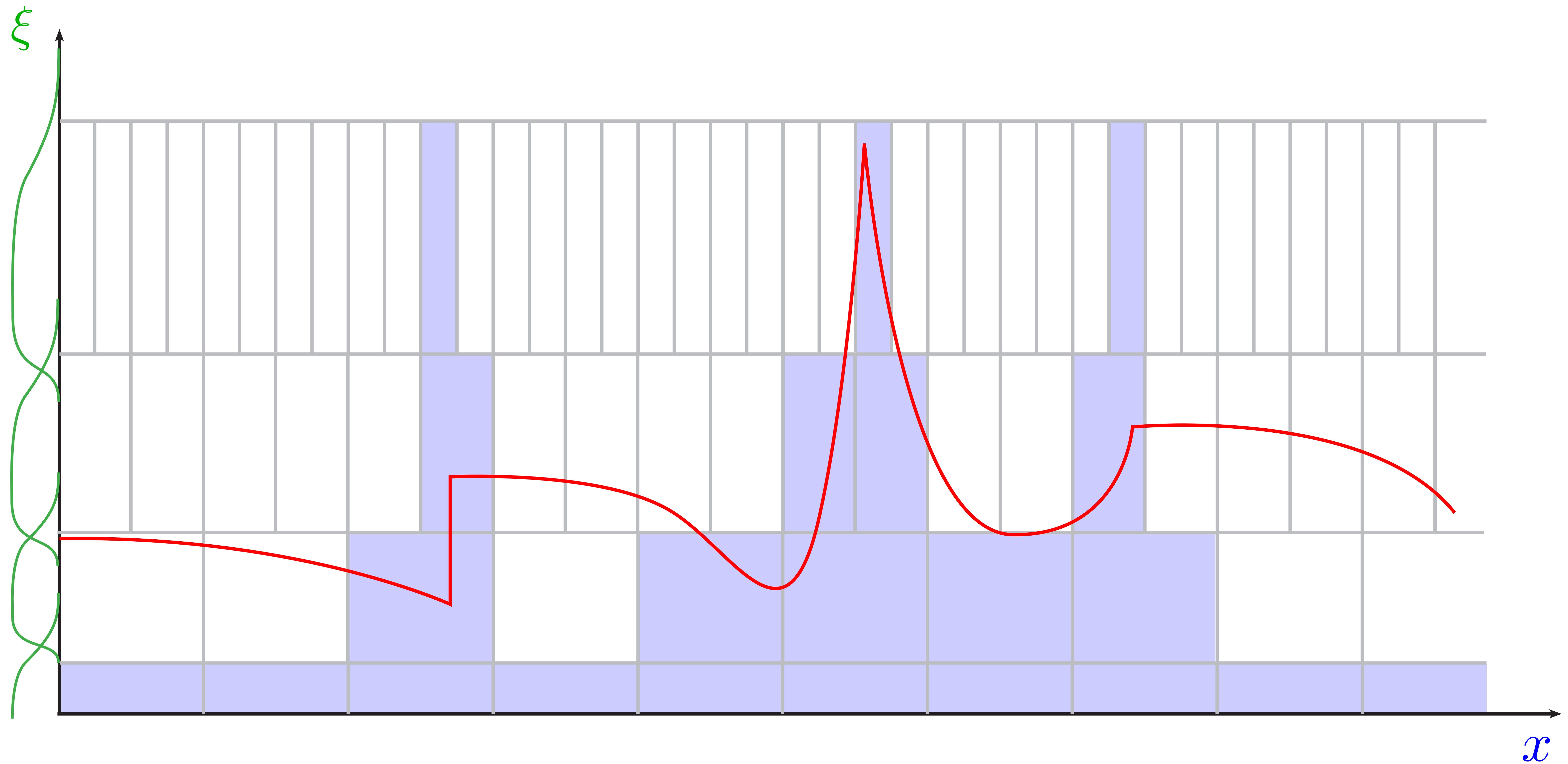
Tight function spaces

$$f(x) = \sum_{i \in \mathcal{I}} f_i \psi_i(x) \approx (2^{-l}k, 2^l\xi)$$

depends on
image function



Tight function spaces



Tight function spaces

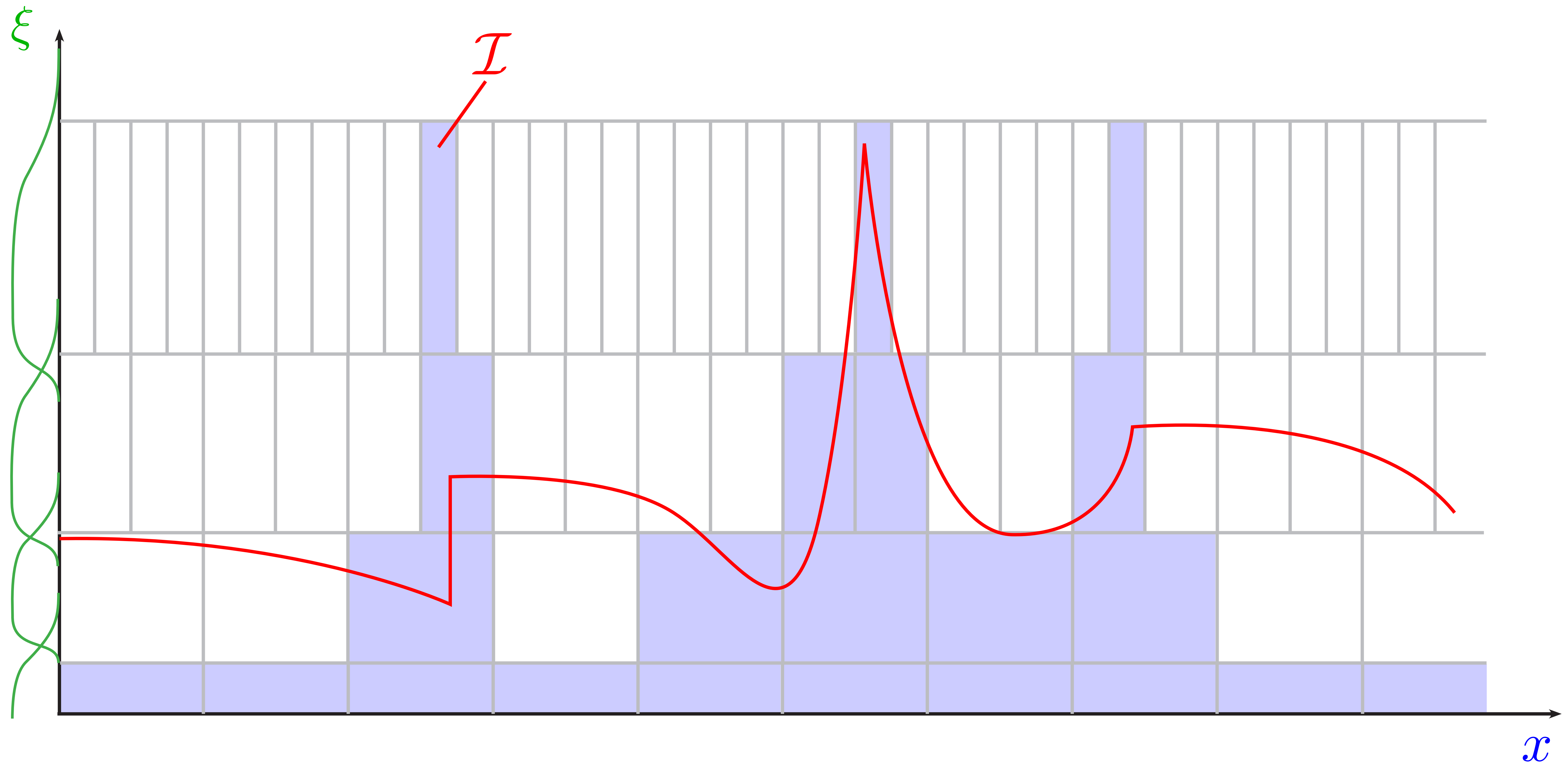


Image generation

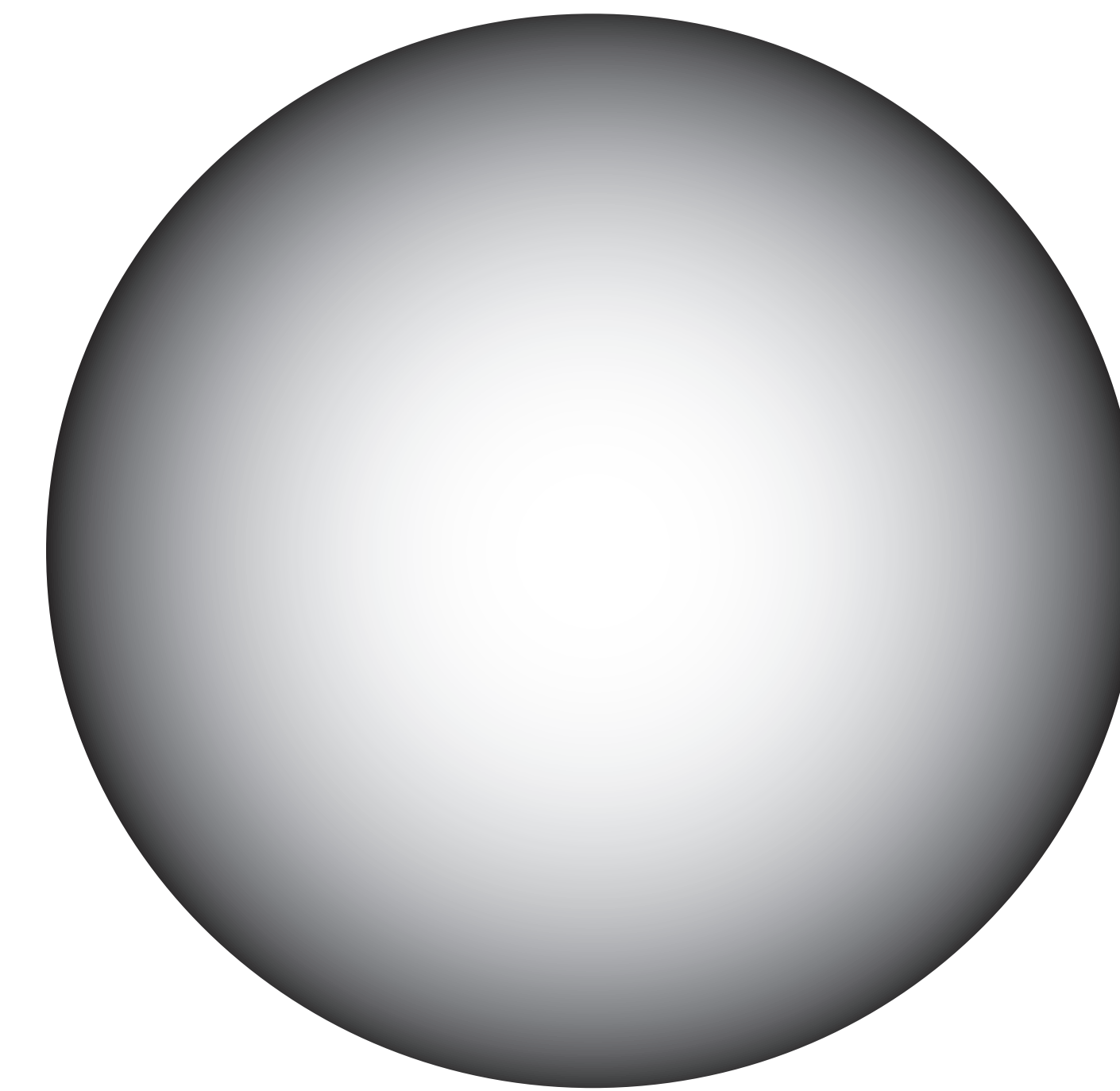
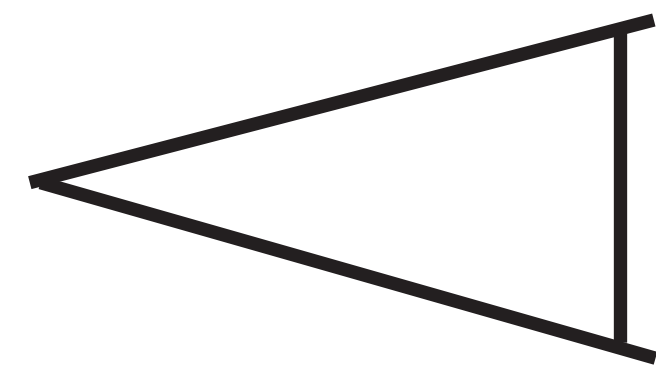


Image generation

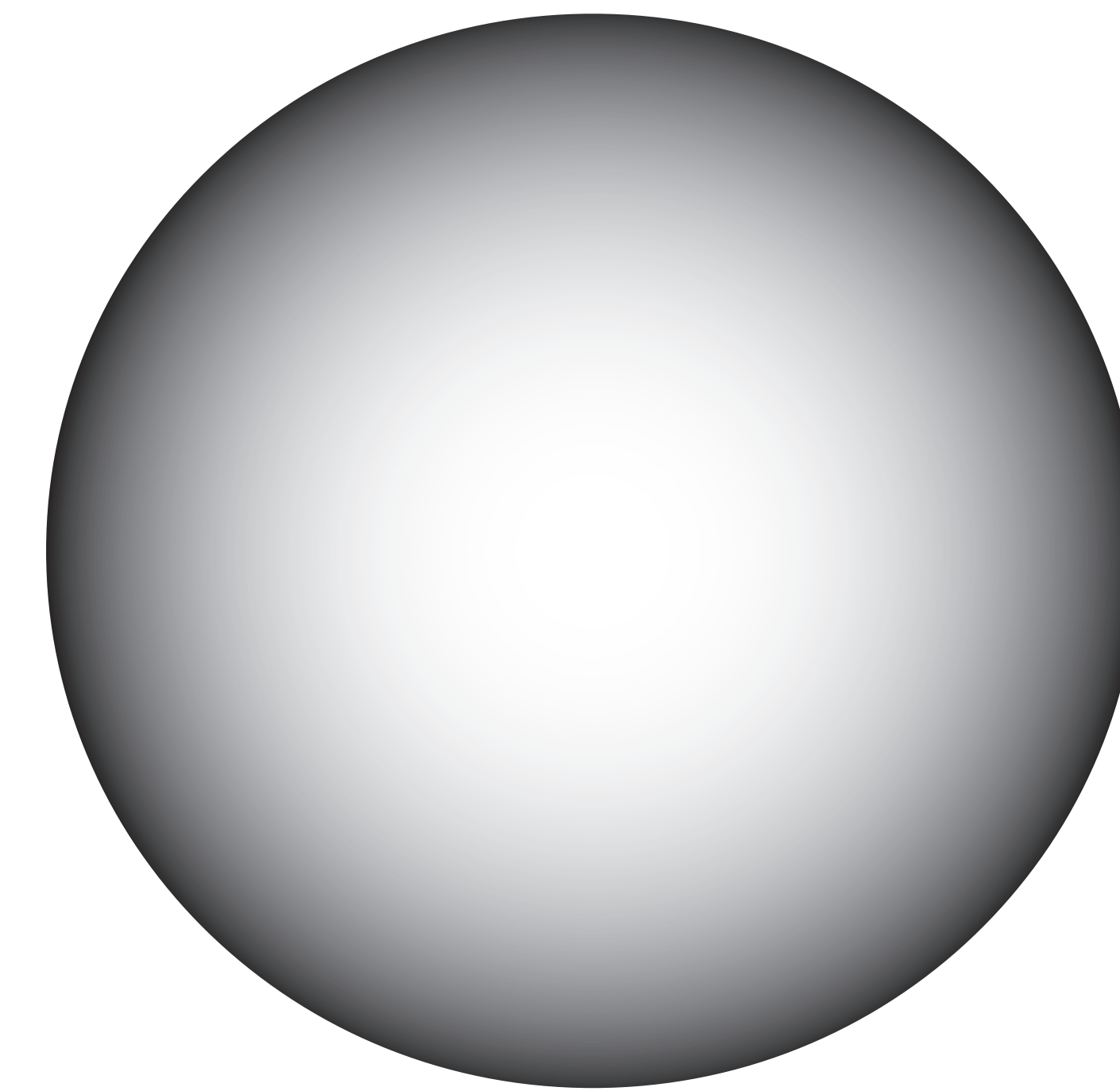
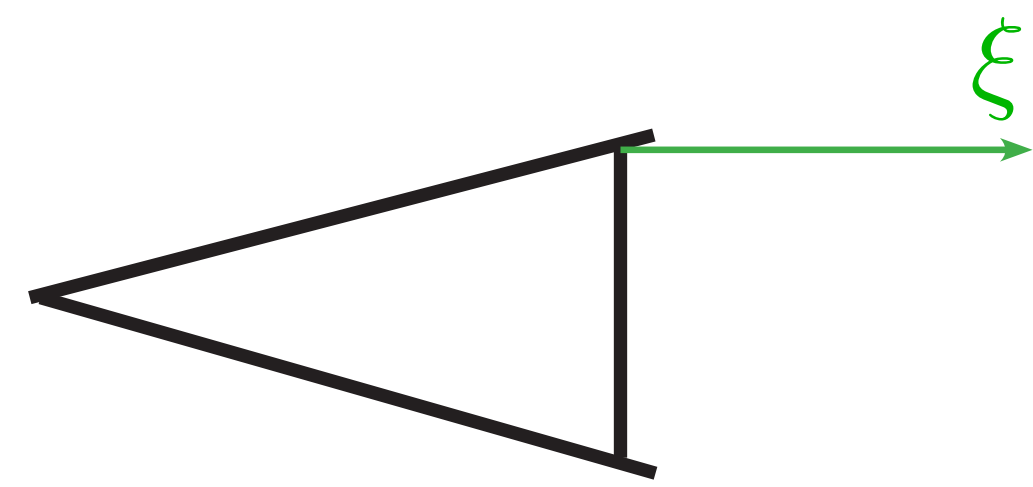


Image generation

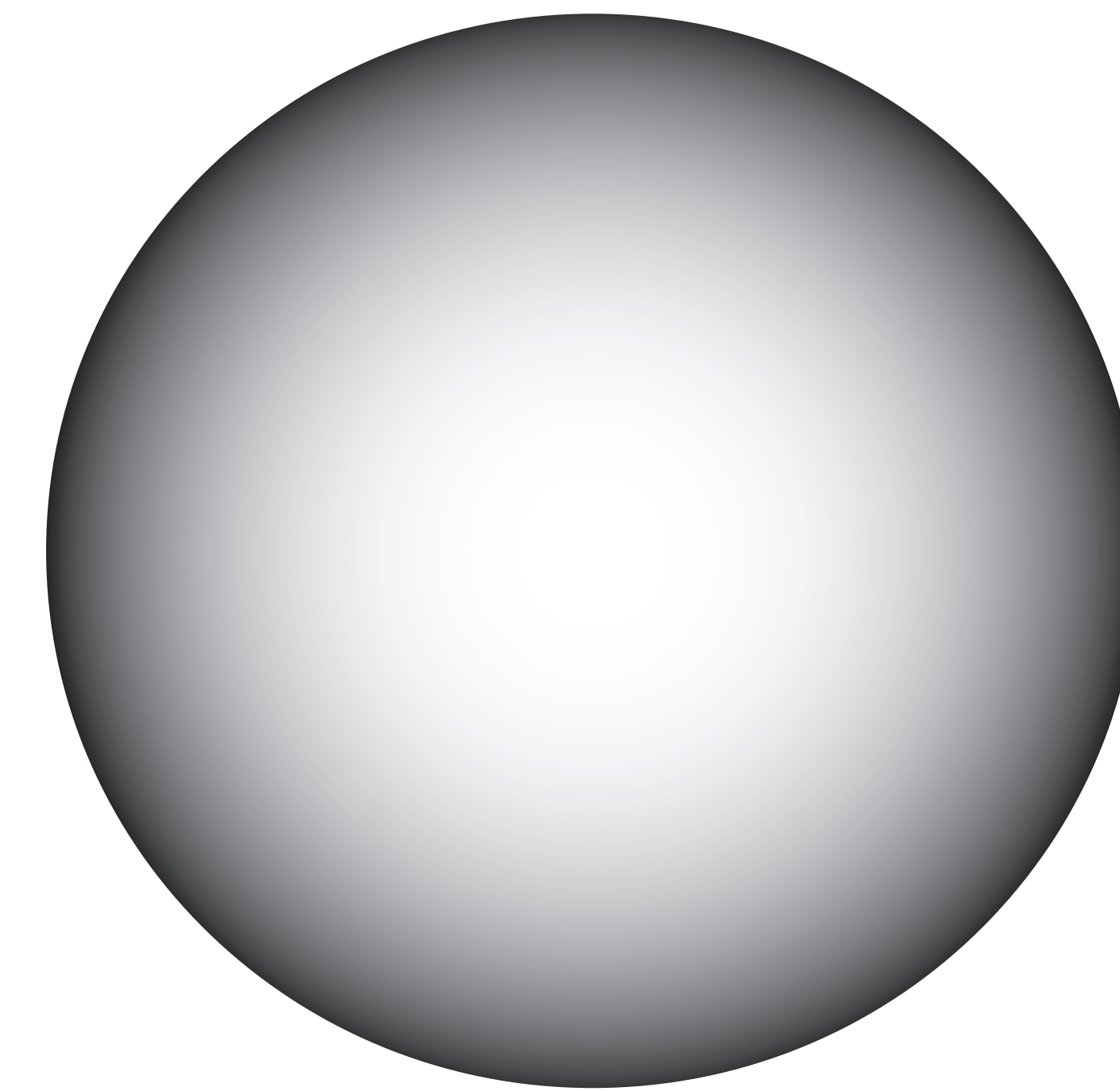
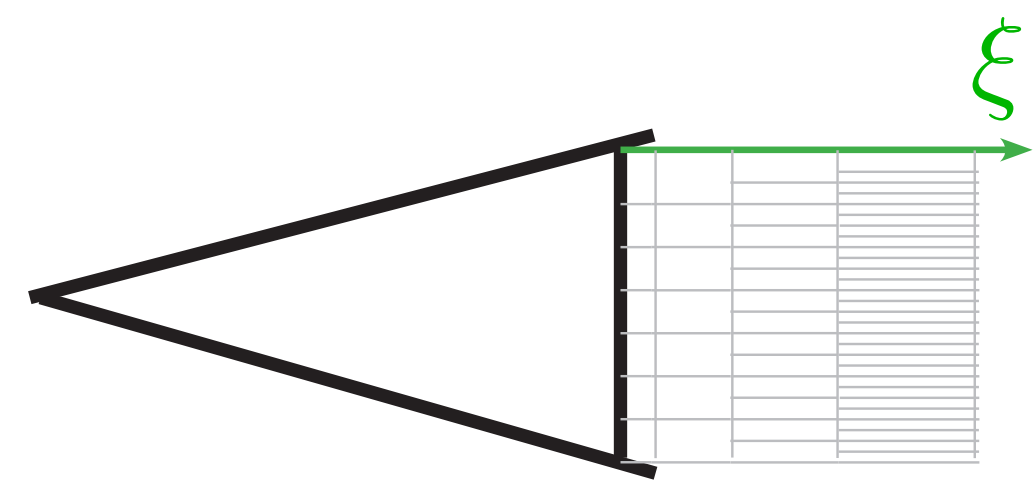


Image generation

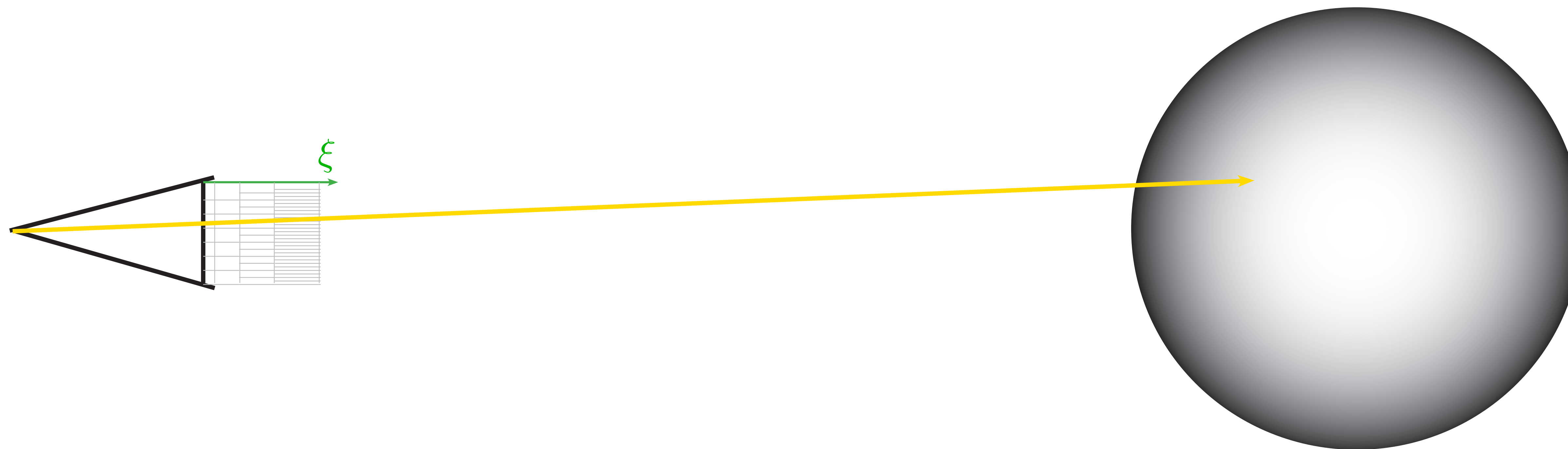


Image generation

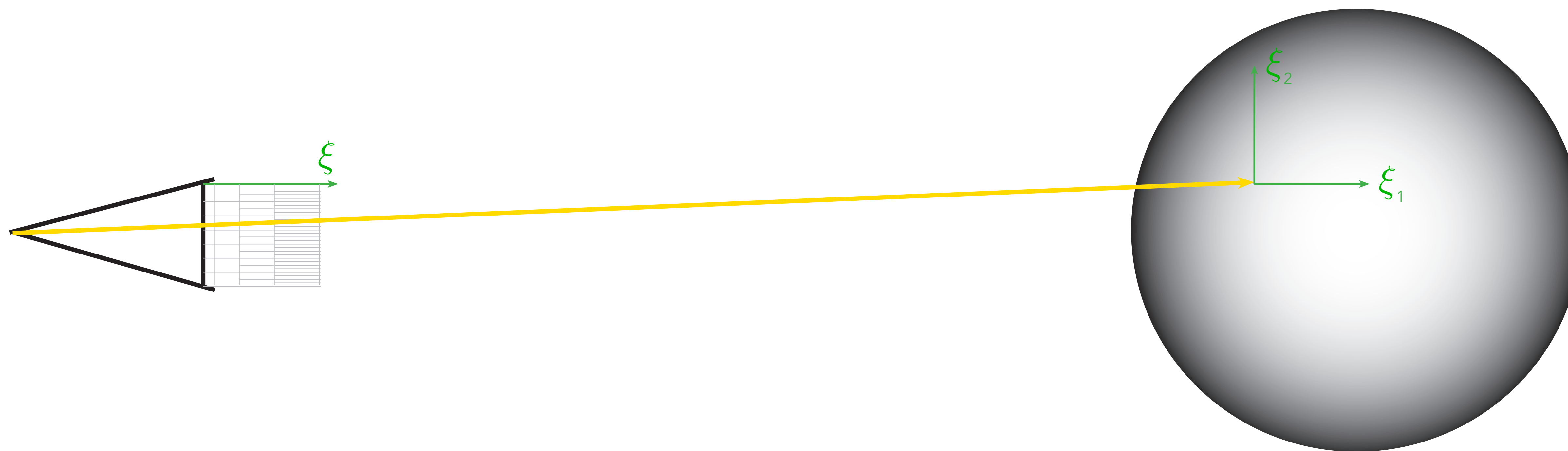


Image generation

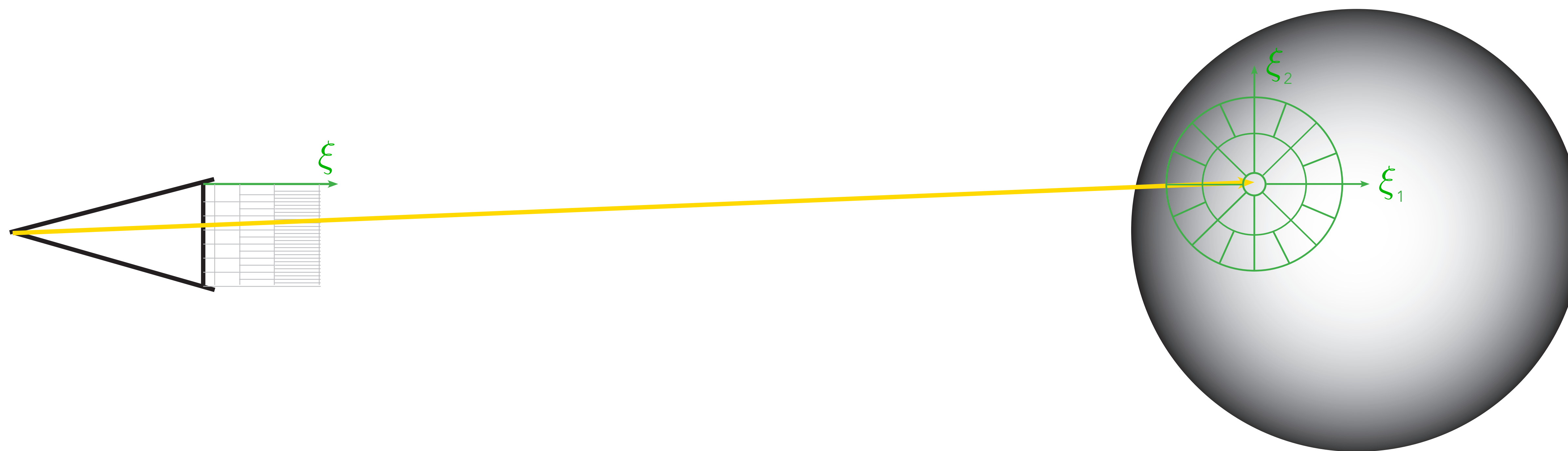


Image generation

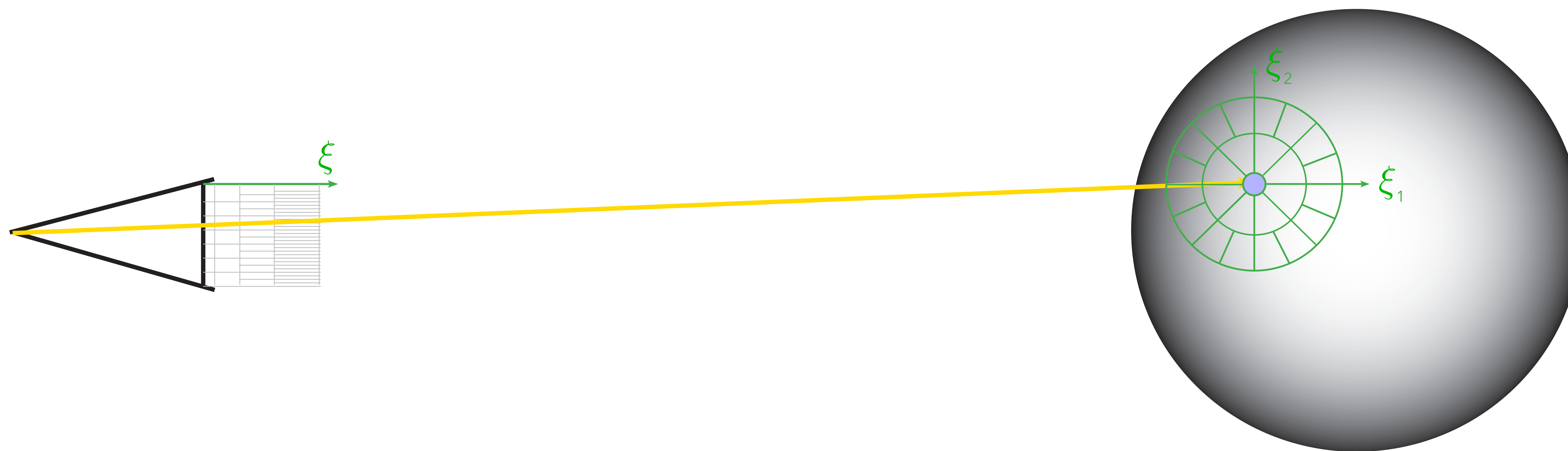


Image generation

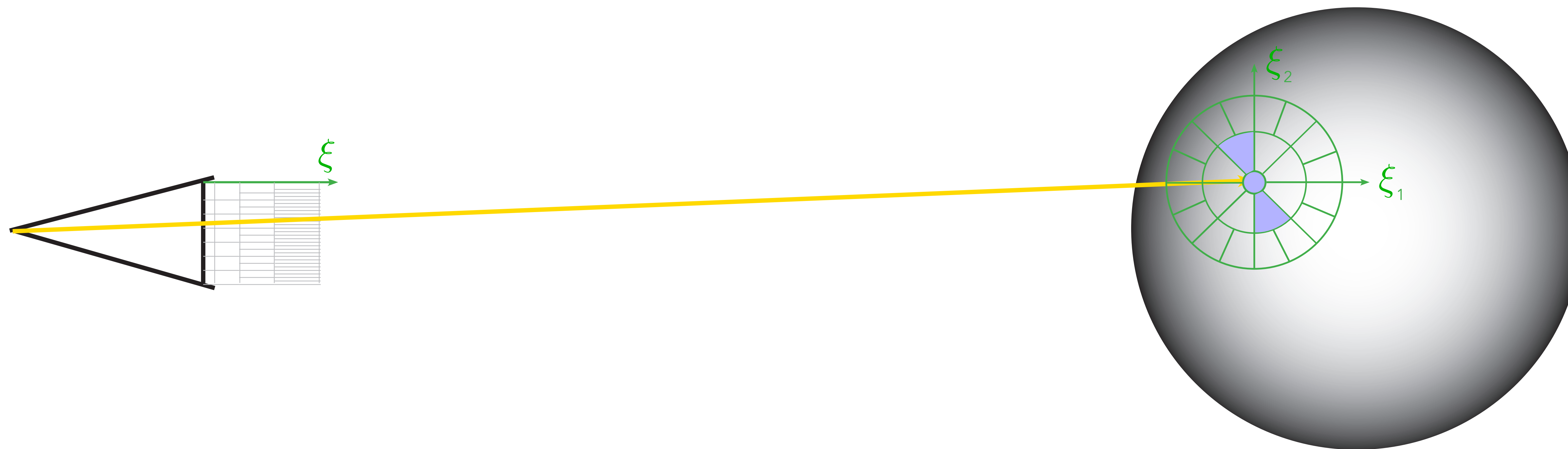


Image generation

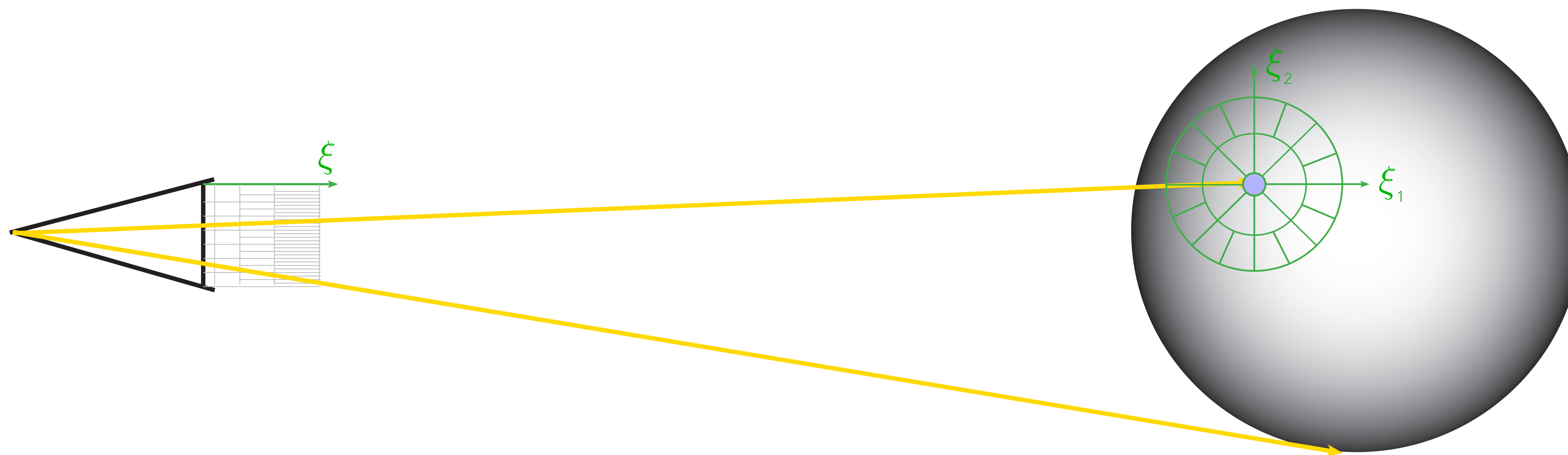


Image generation

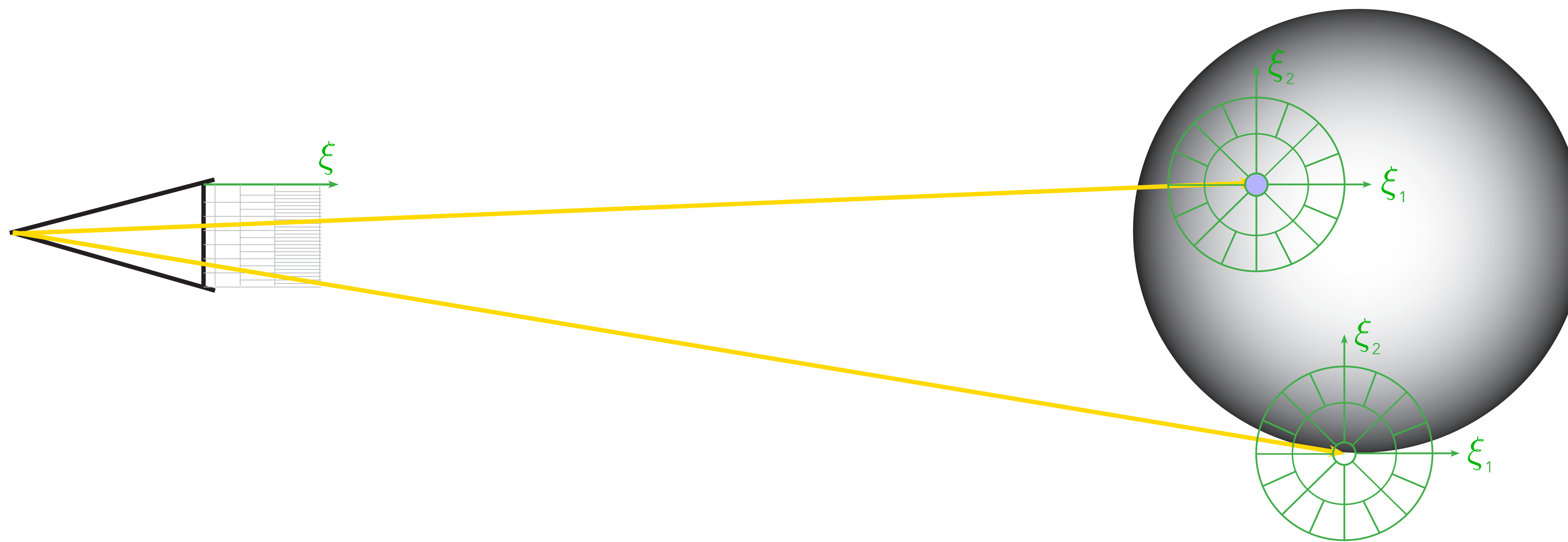


Image generation

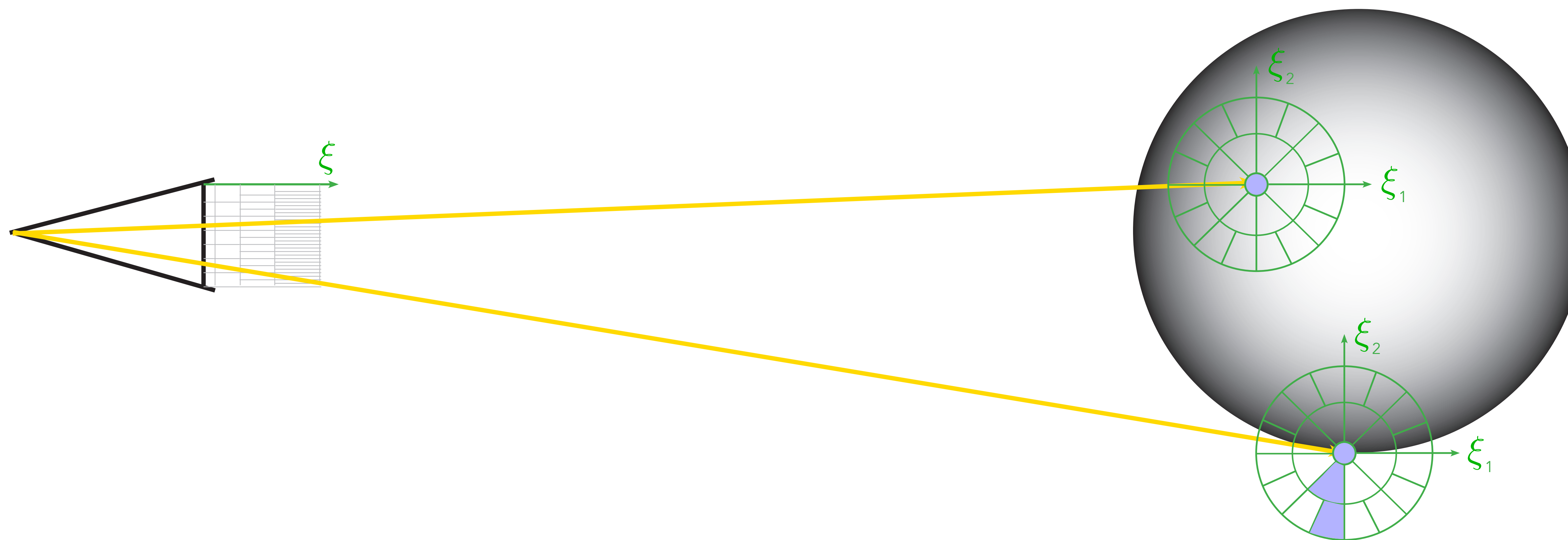


Image generation

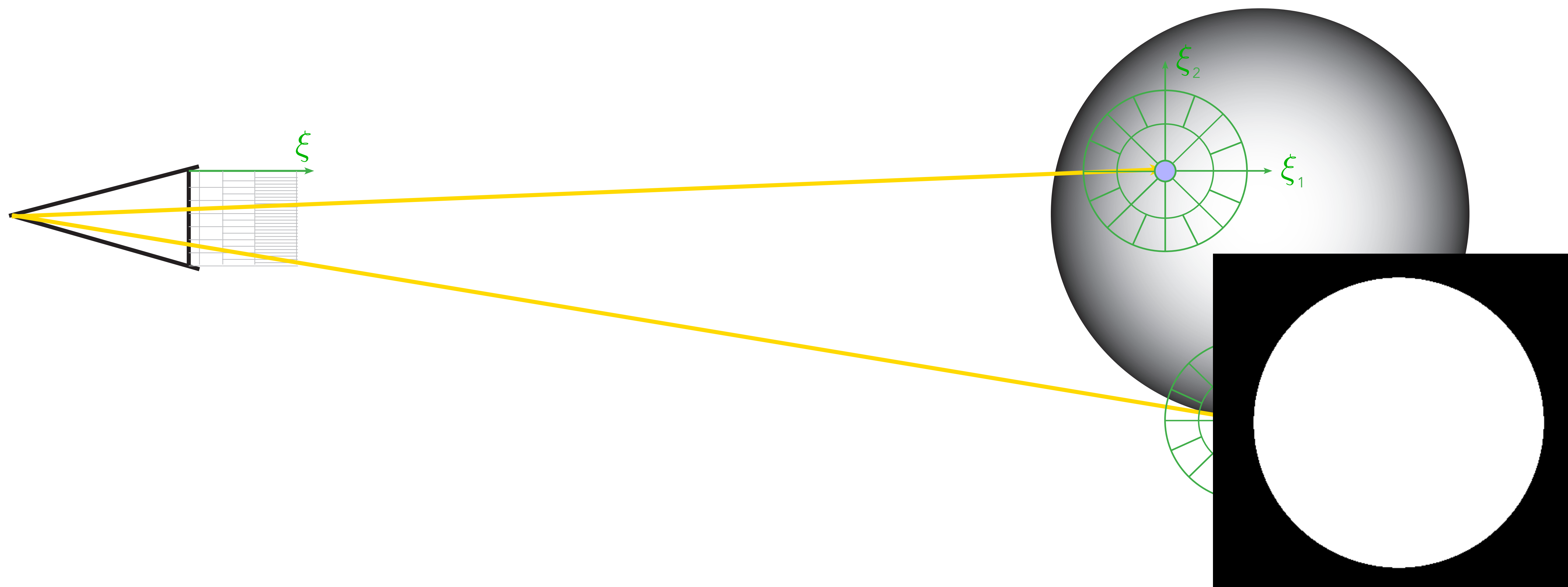


Image generation

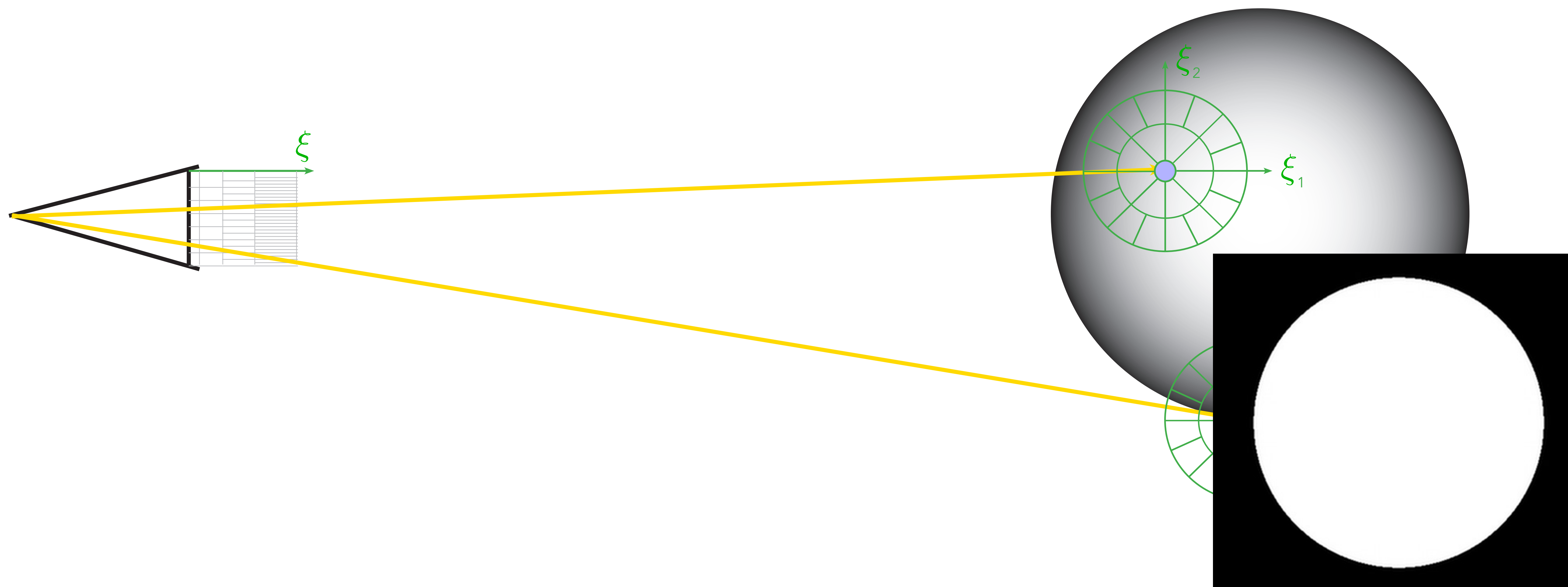


Image generation

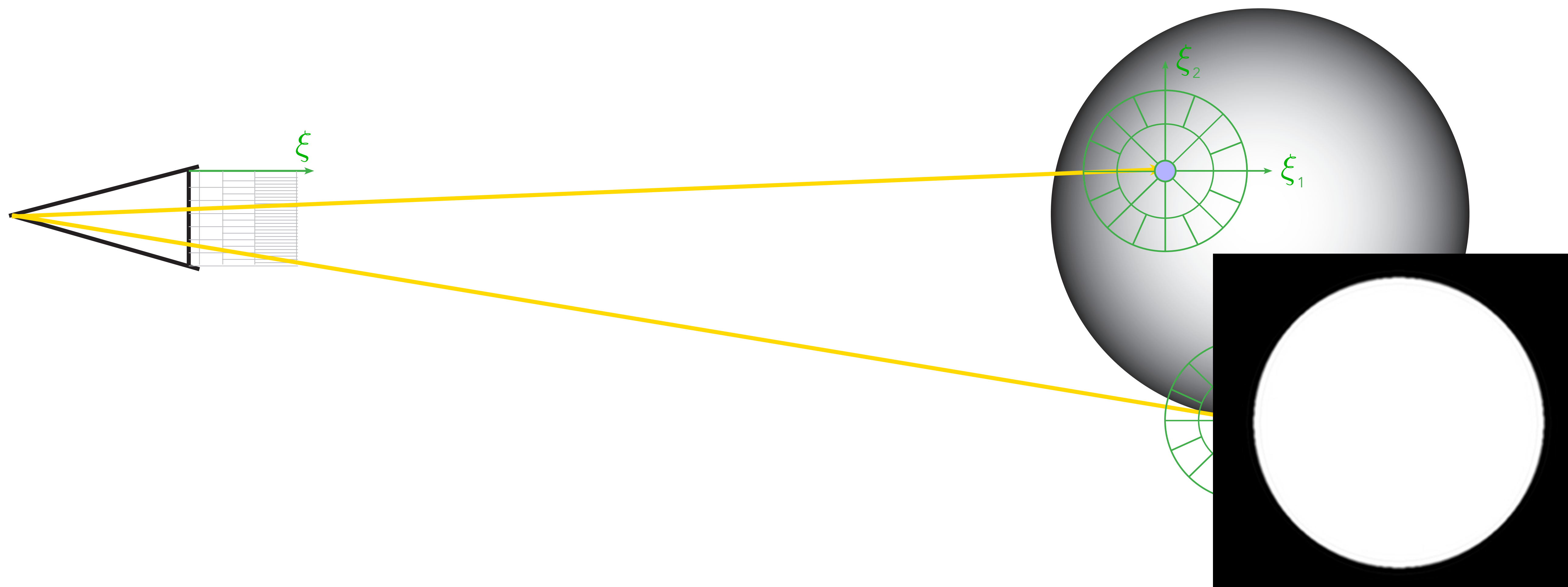


Image generation

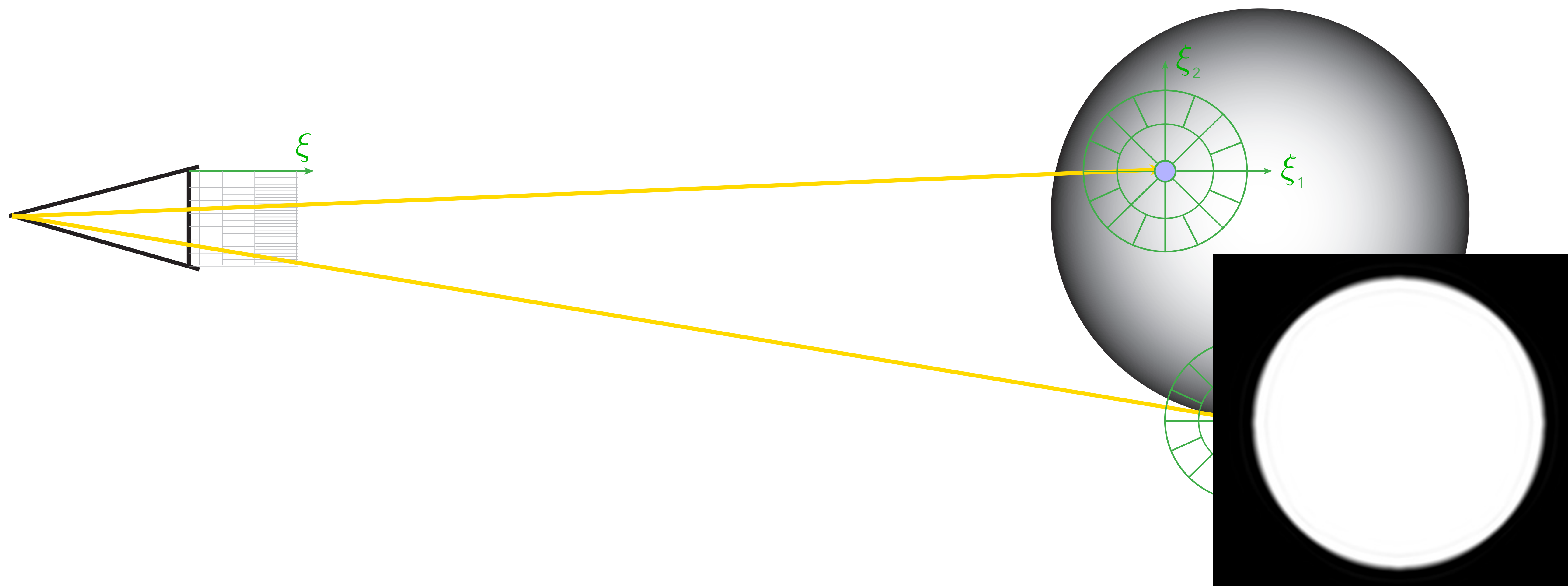


Image generation

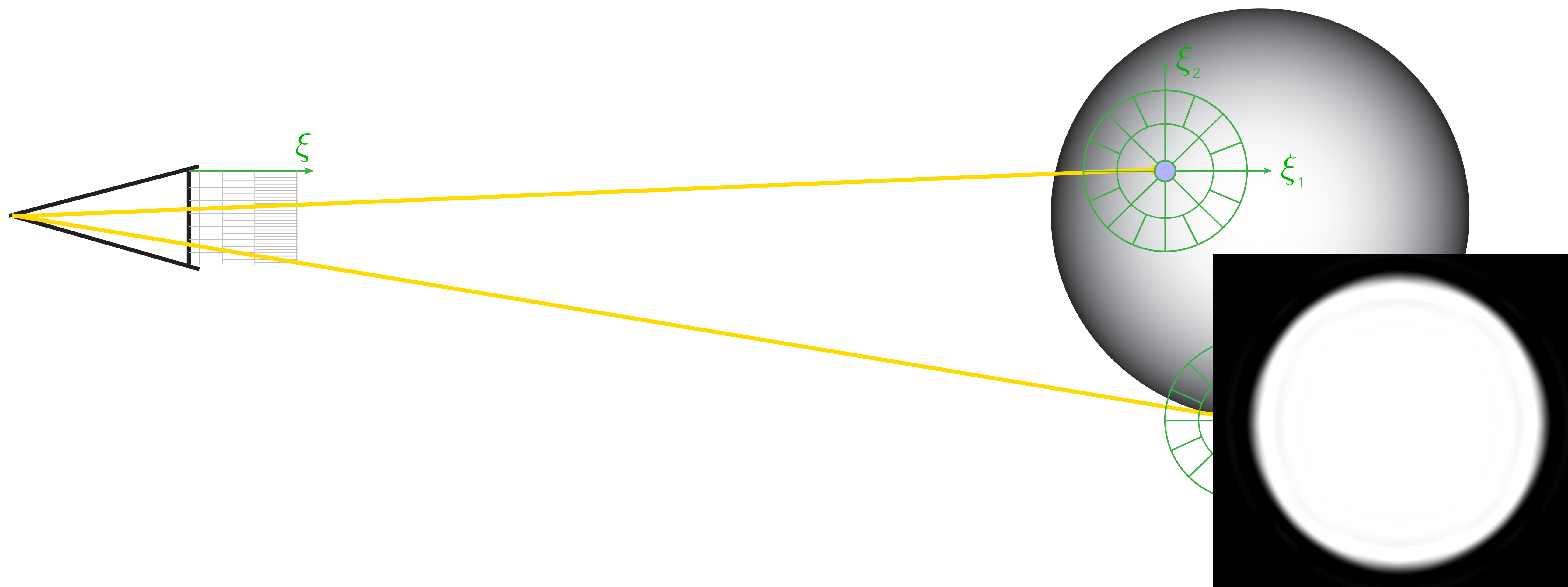


Image generation

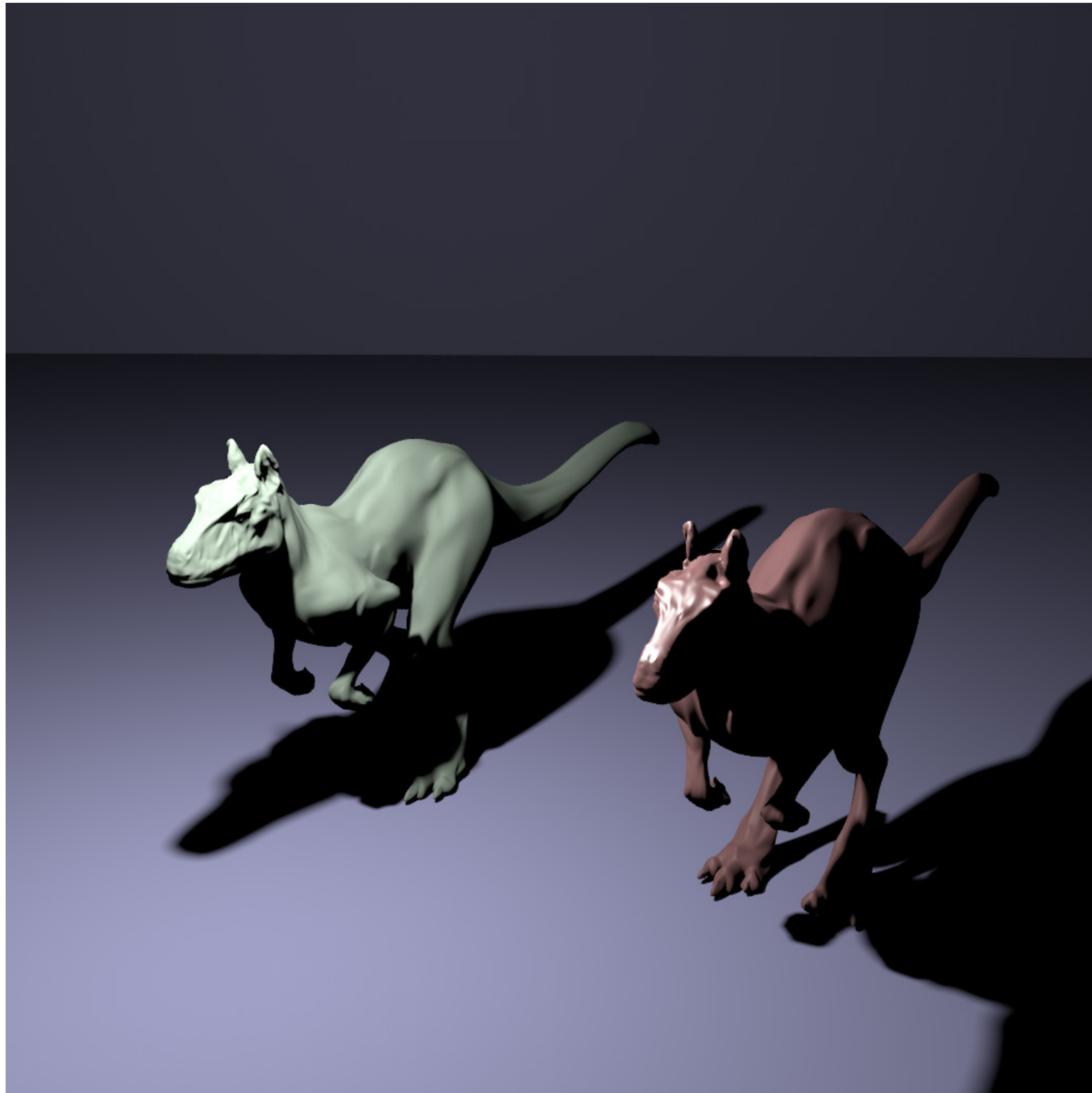


Image generation

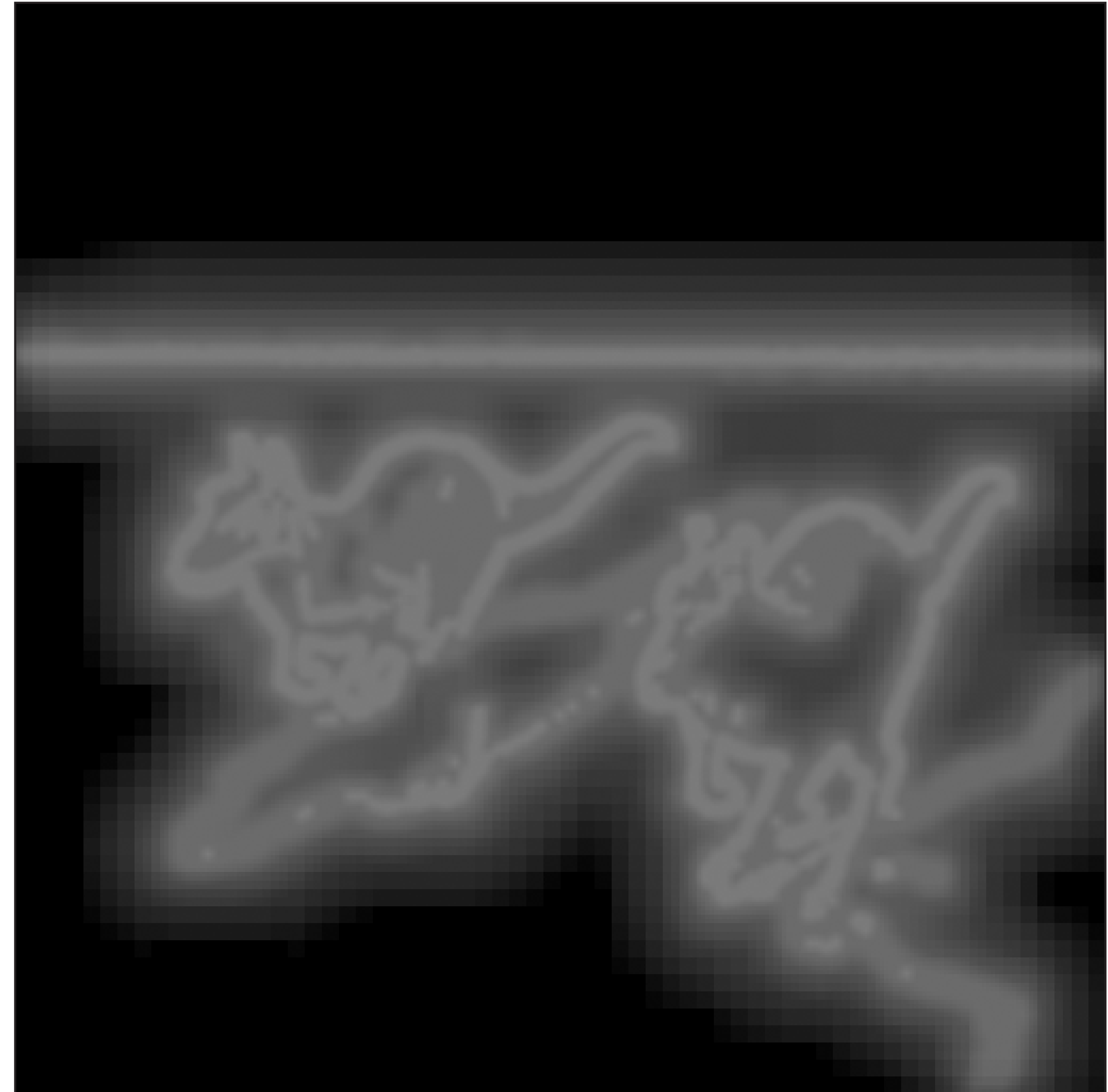
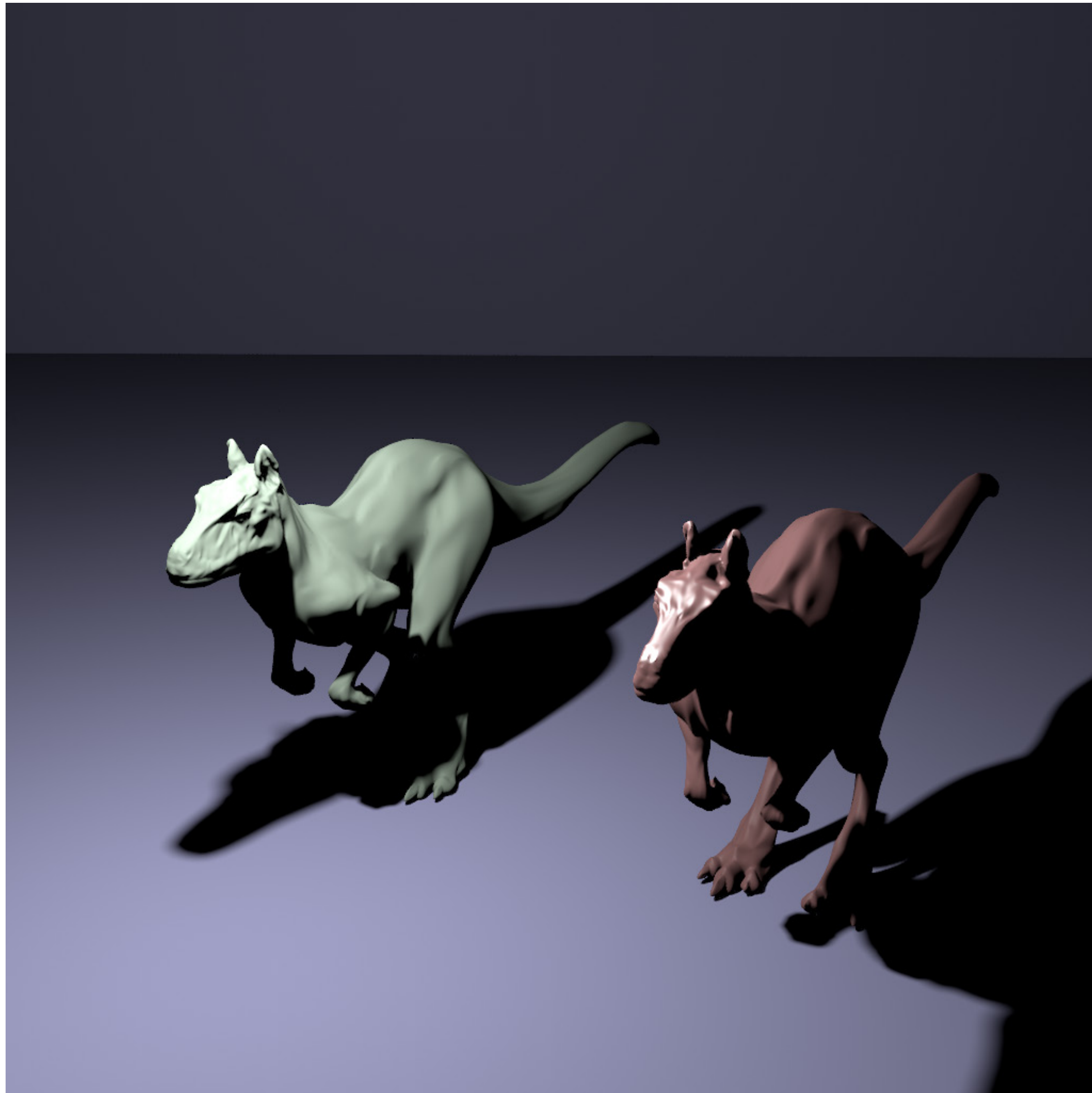


Image generation

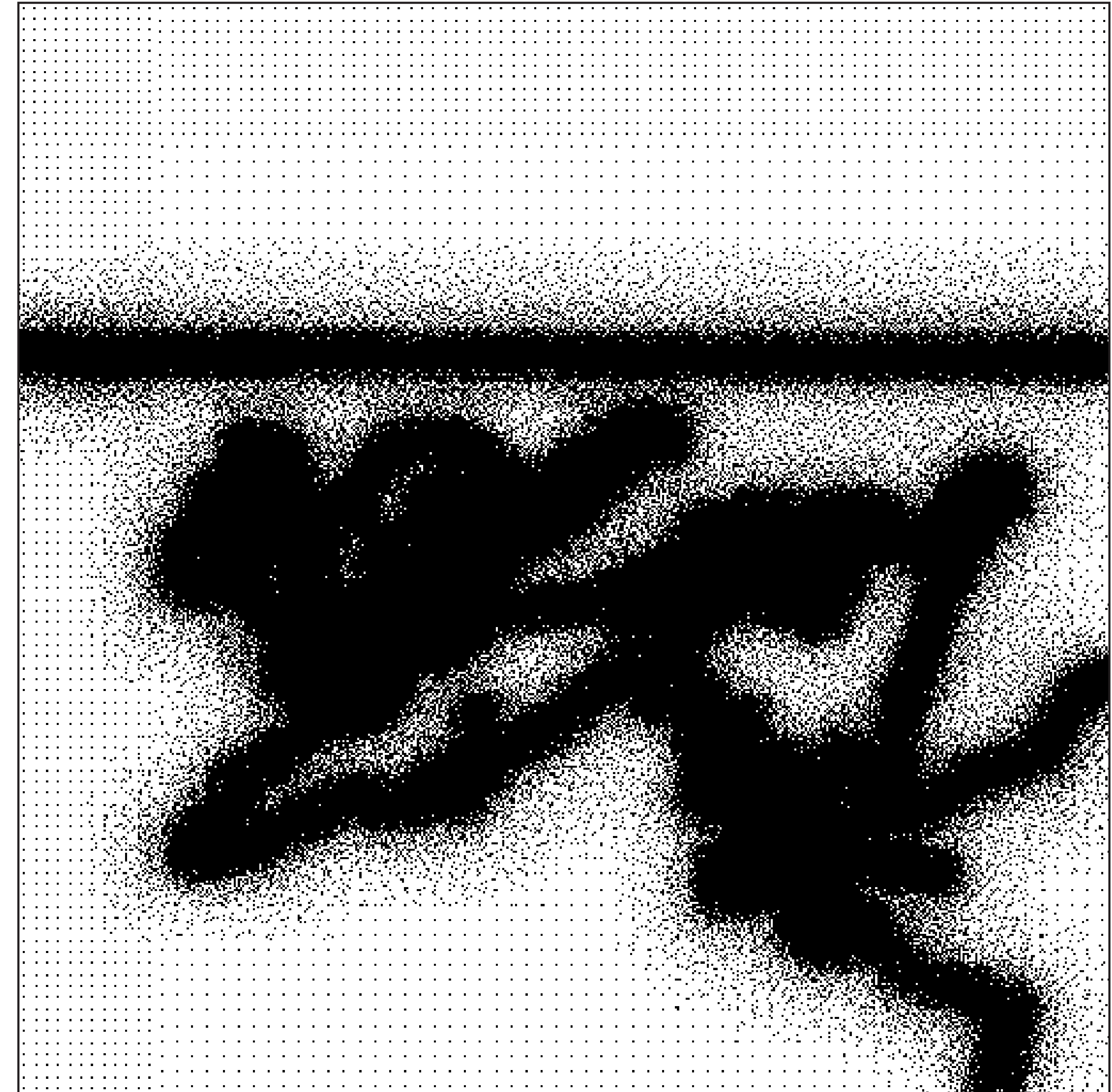
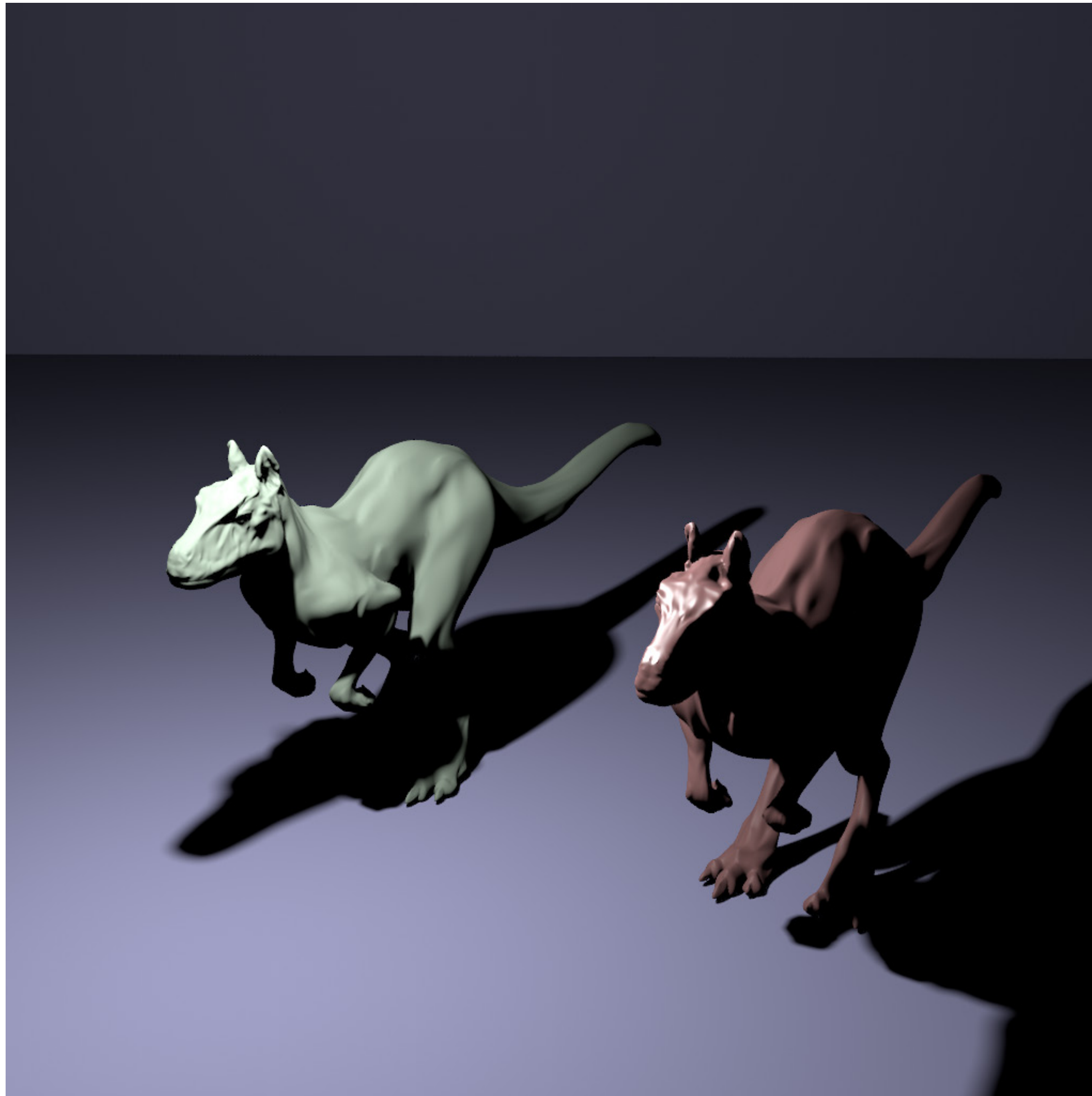


Image generation

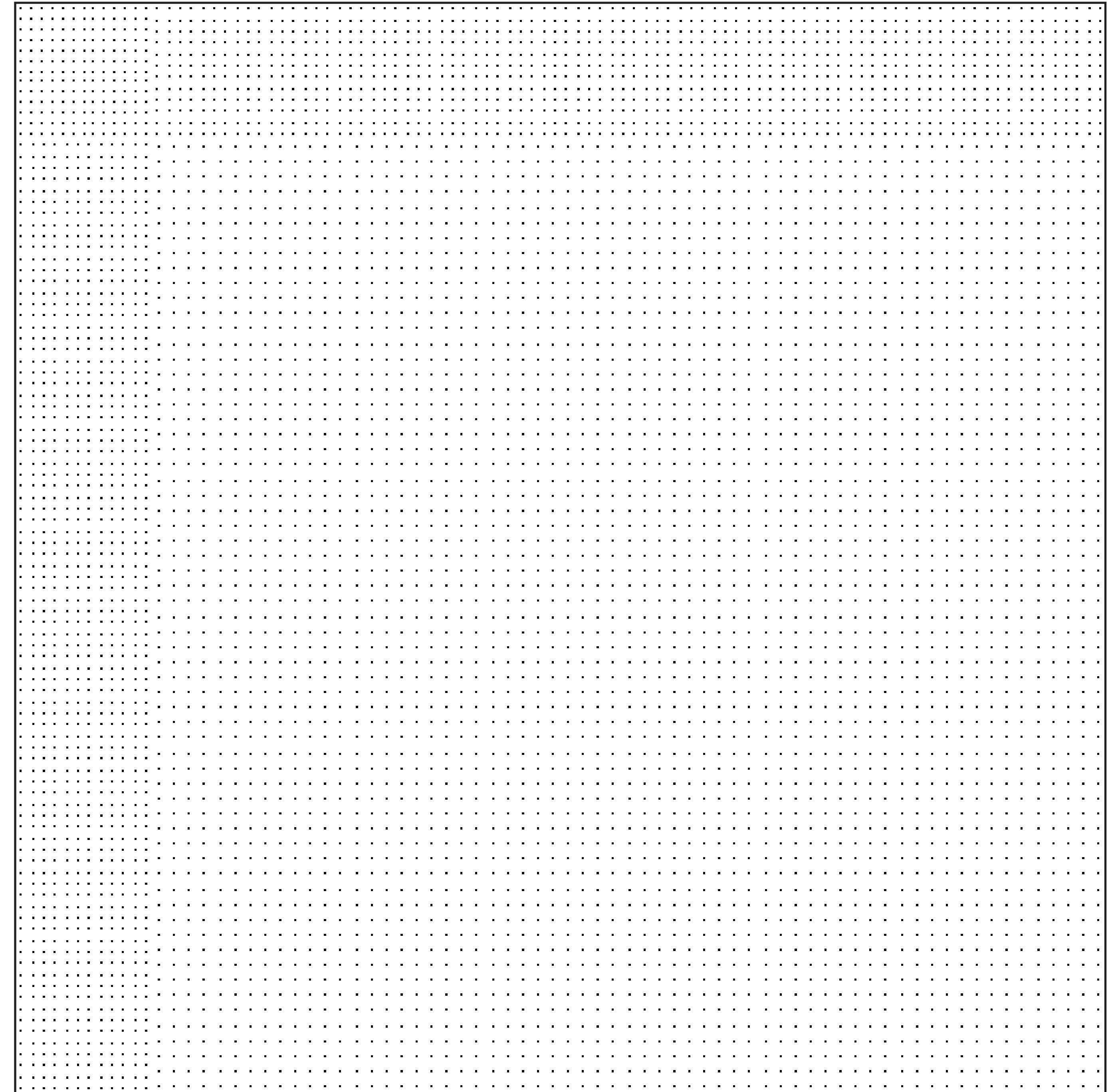
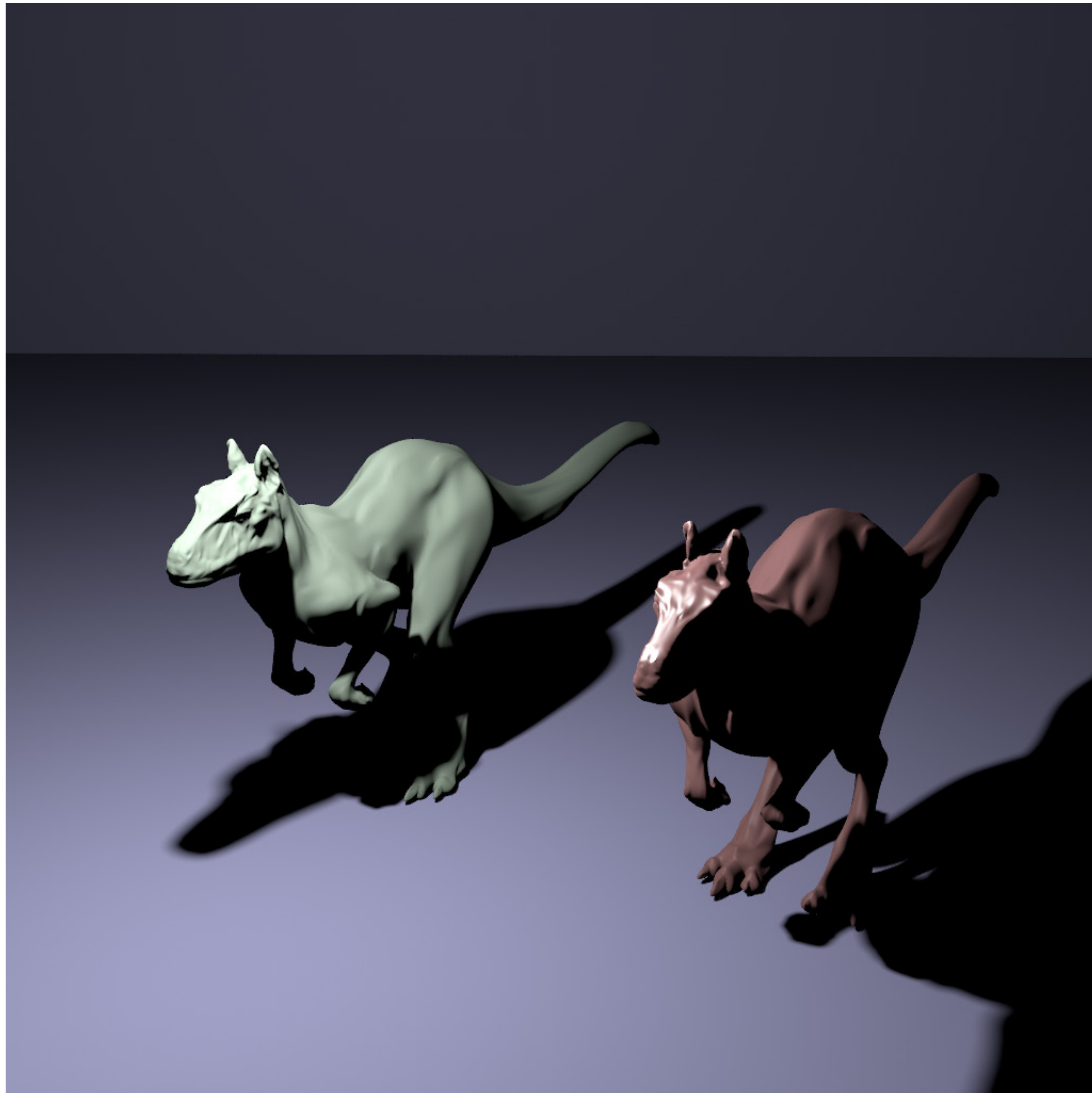


Image generation

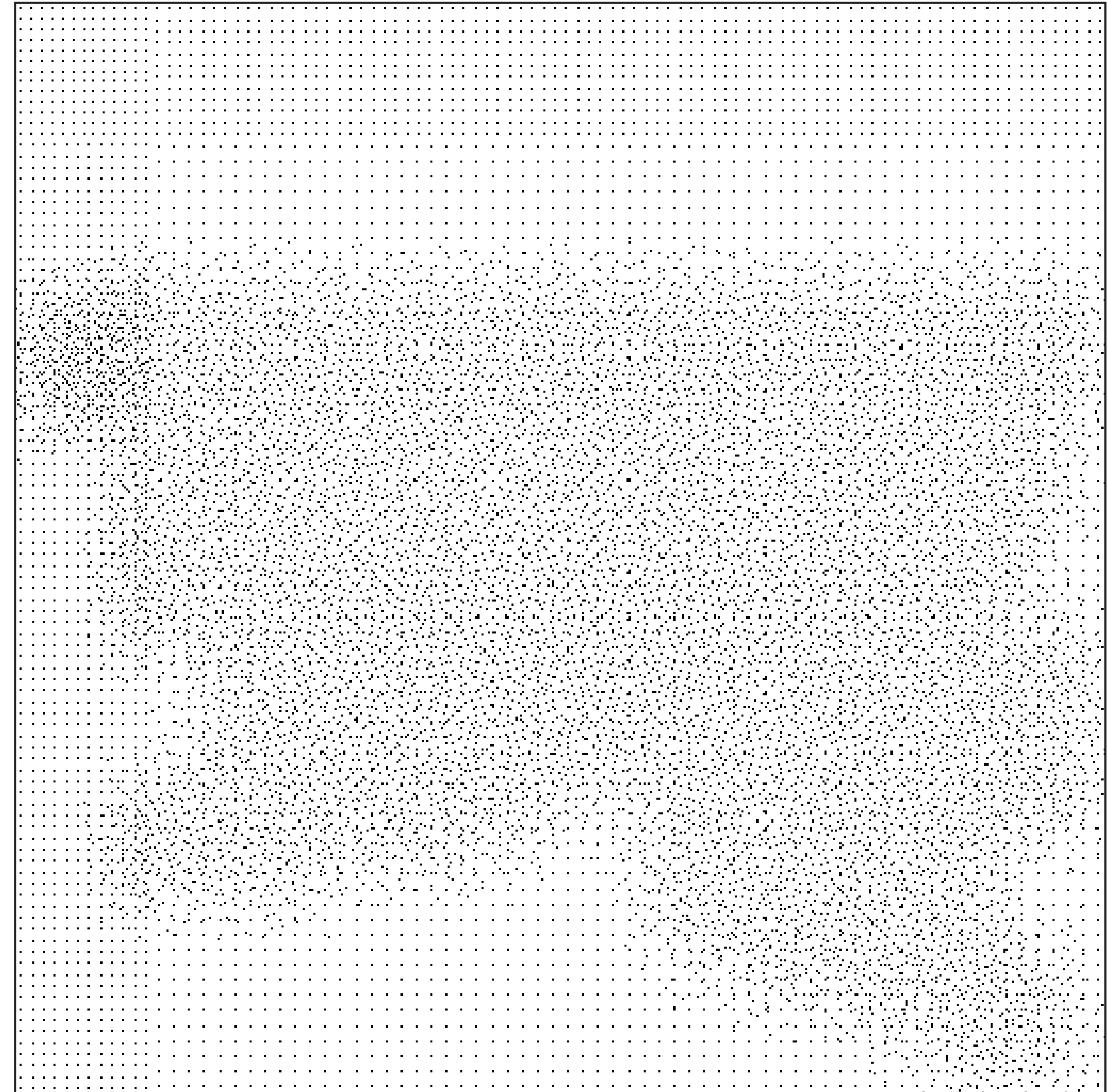
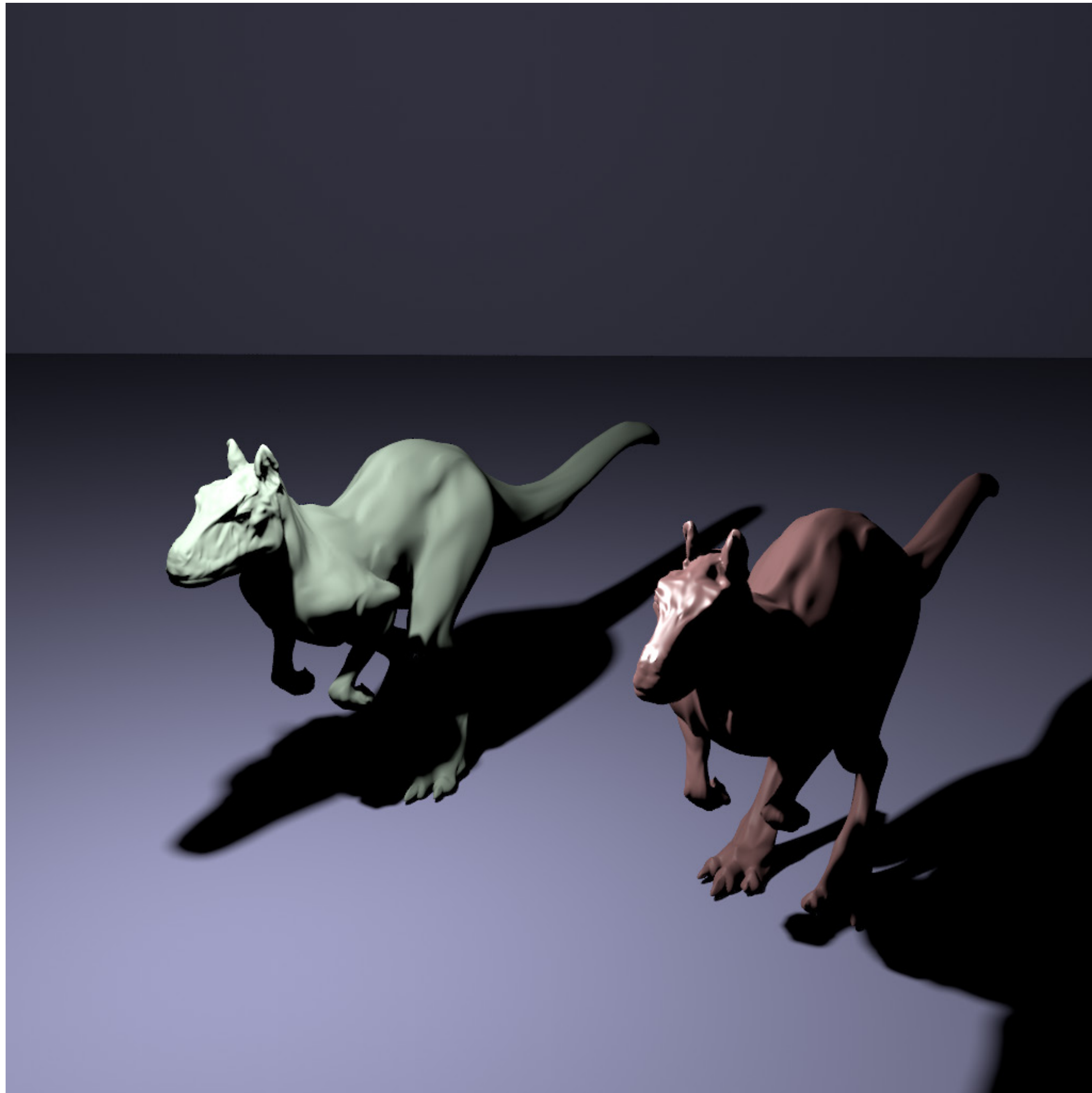


Image generation

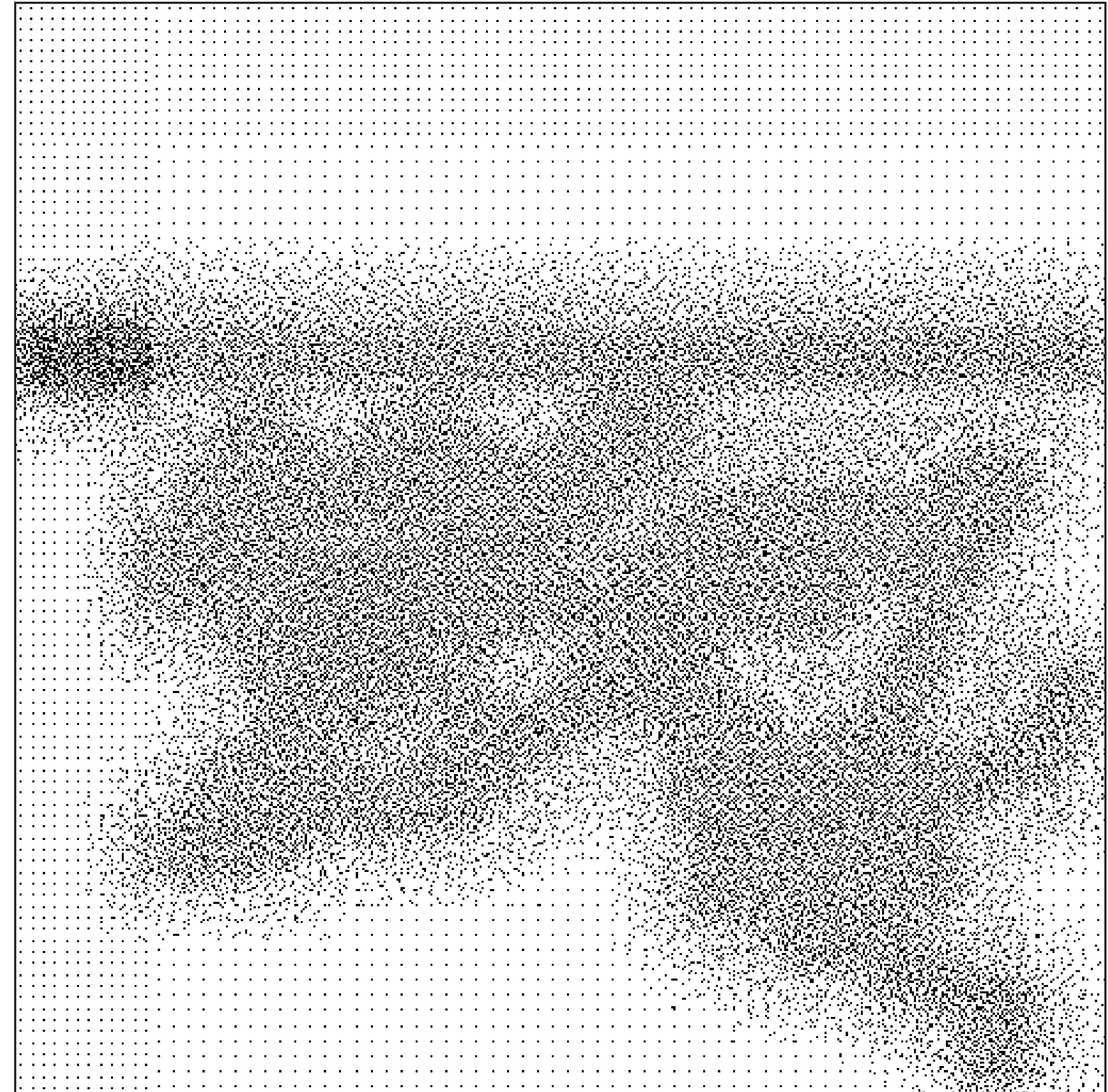
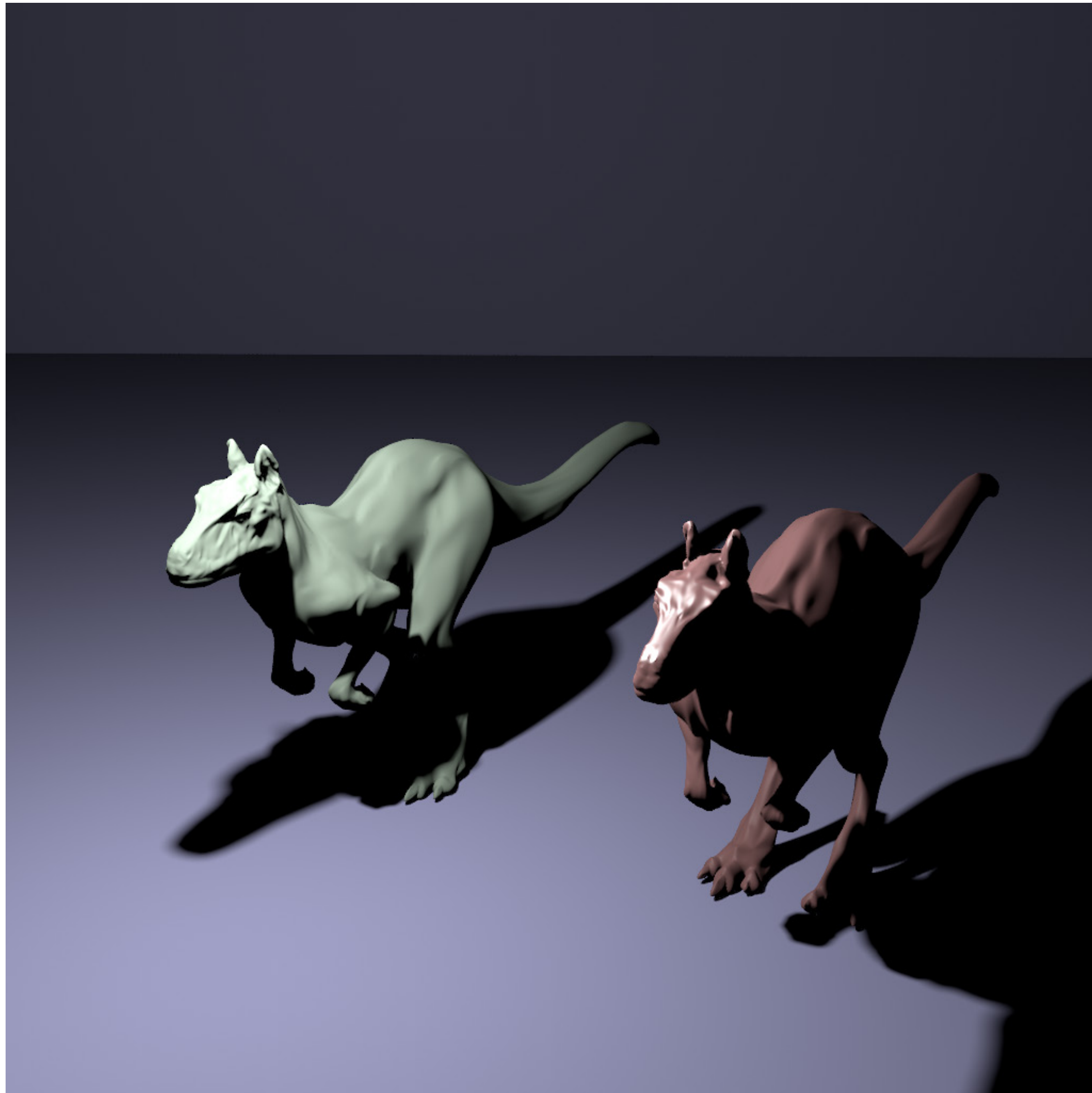


Image generation

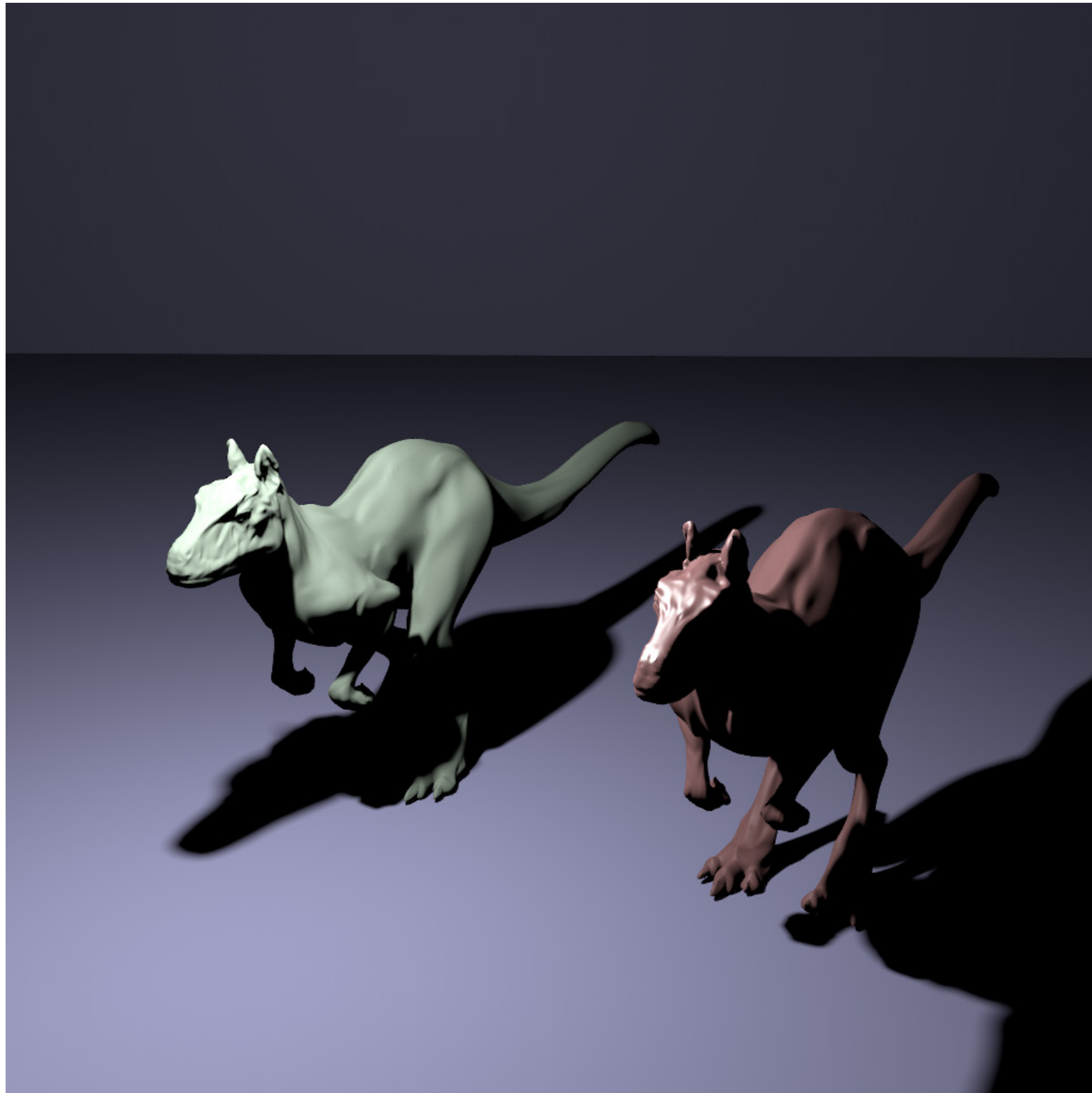


Image generation

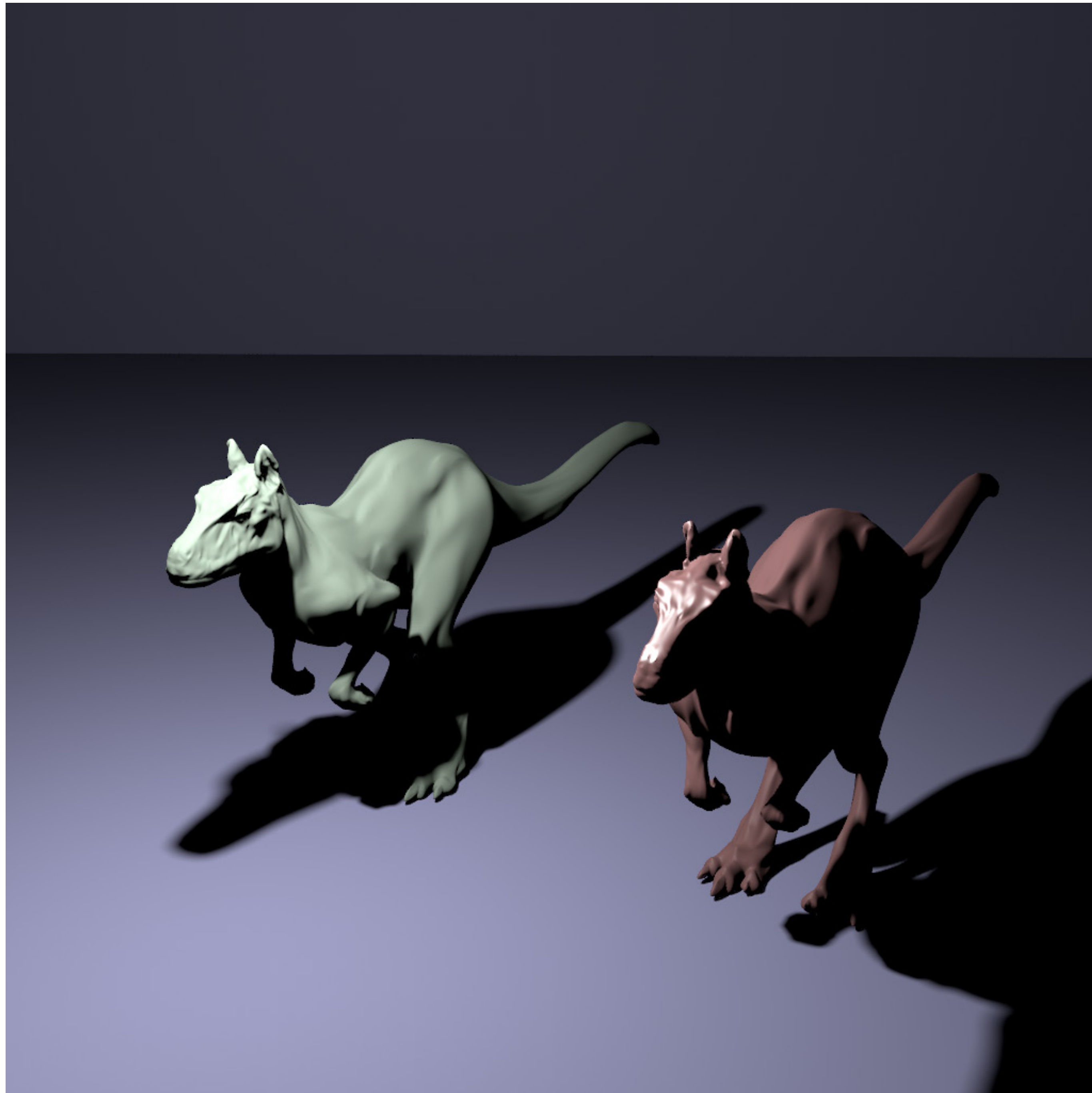


Image generation

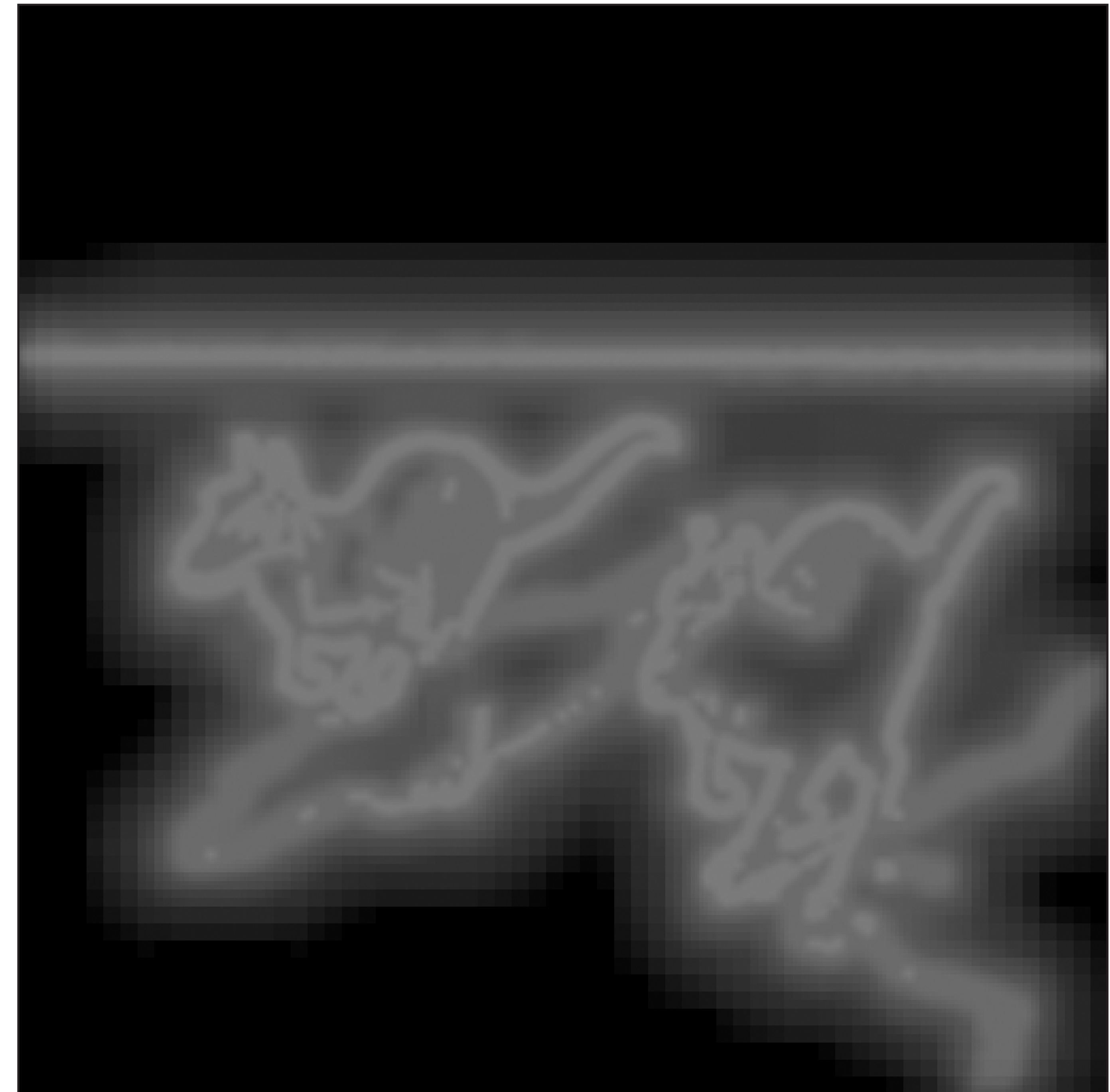
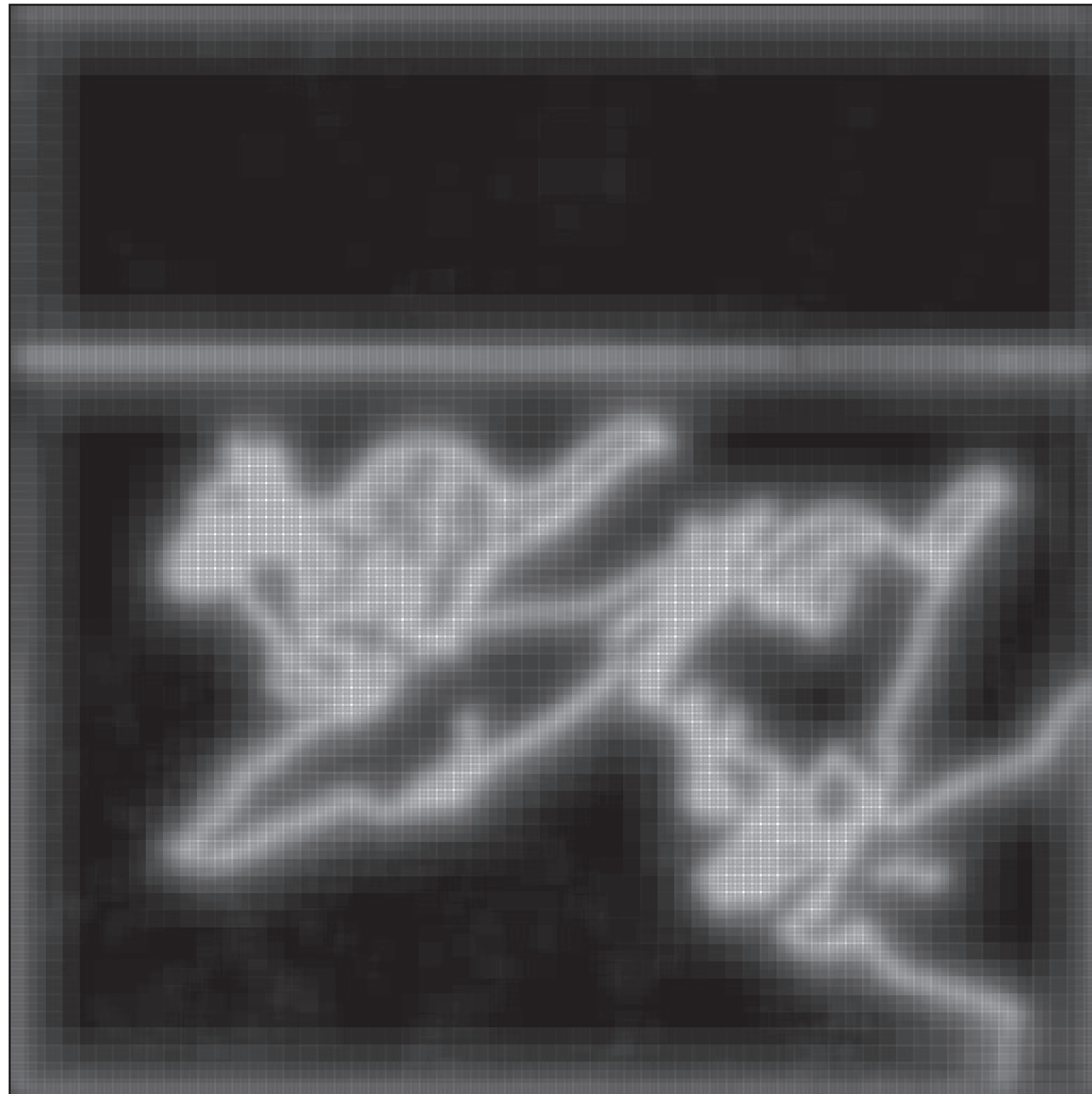
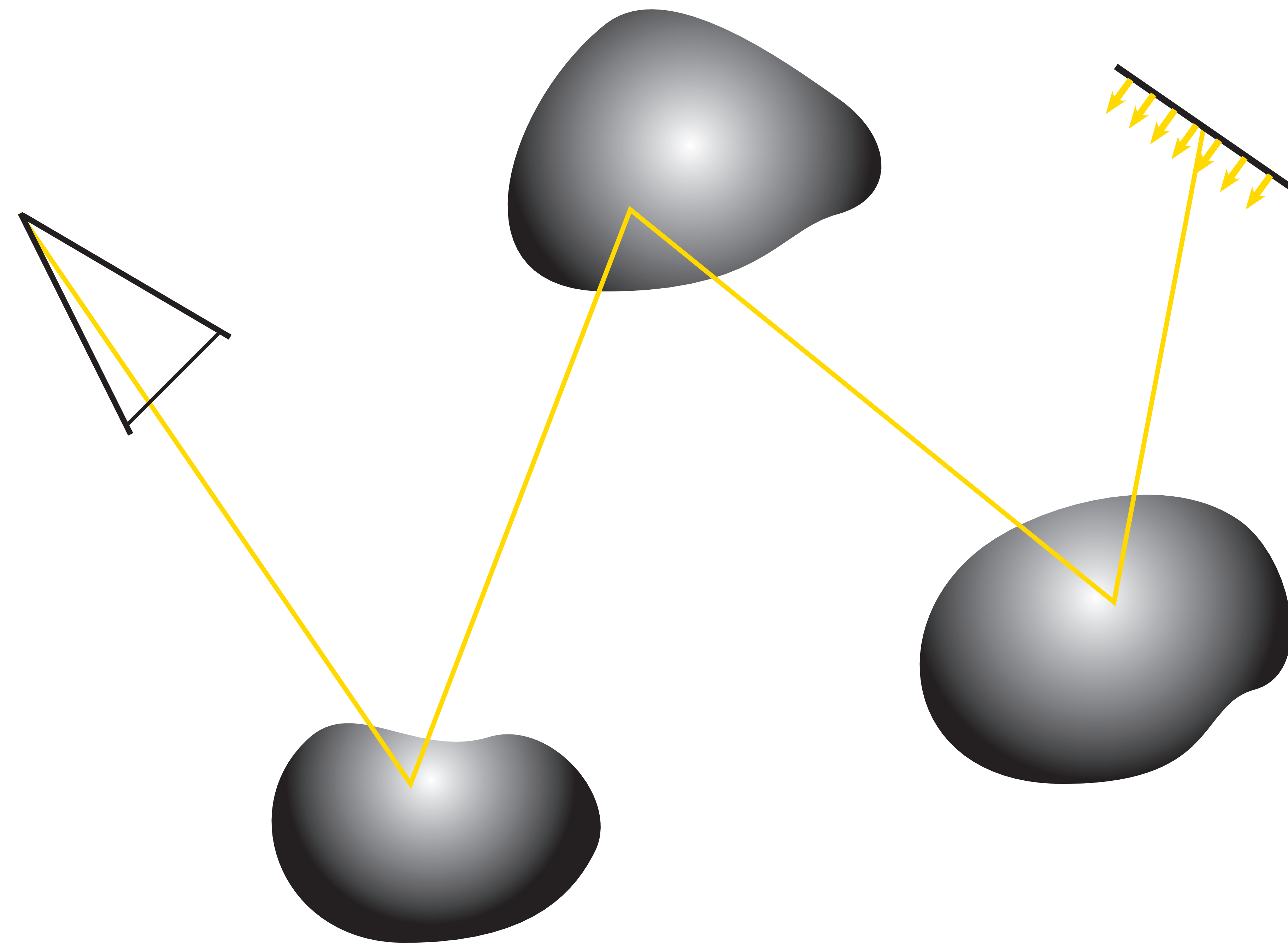


Image generation

1. Find (approximate) tight function space
2. Sample function
3. Construct reproducing kernel basis

Image generation



How many rays do we need?

How many rays do we need?

» As many as there are nonzero coefficients in the sparsest signal representation (times a small constant). «

How many rays do we need?

» As many as there are nonzero coefficients in the sparsest signal representation (times a small constant). «

Recipe:

1. Find (approximate) tight function space
2. Sample function
3. Construct reproducing kernel basis