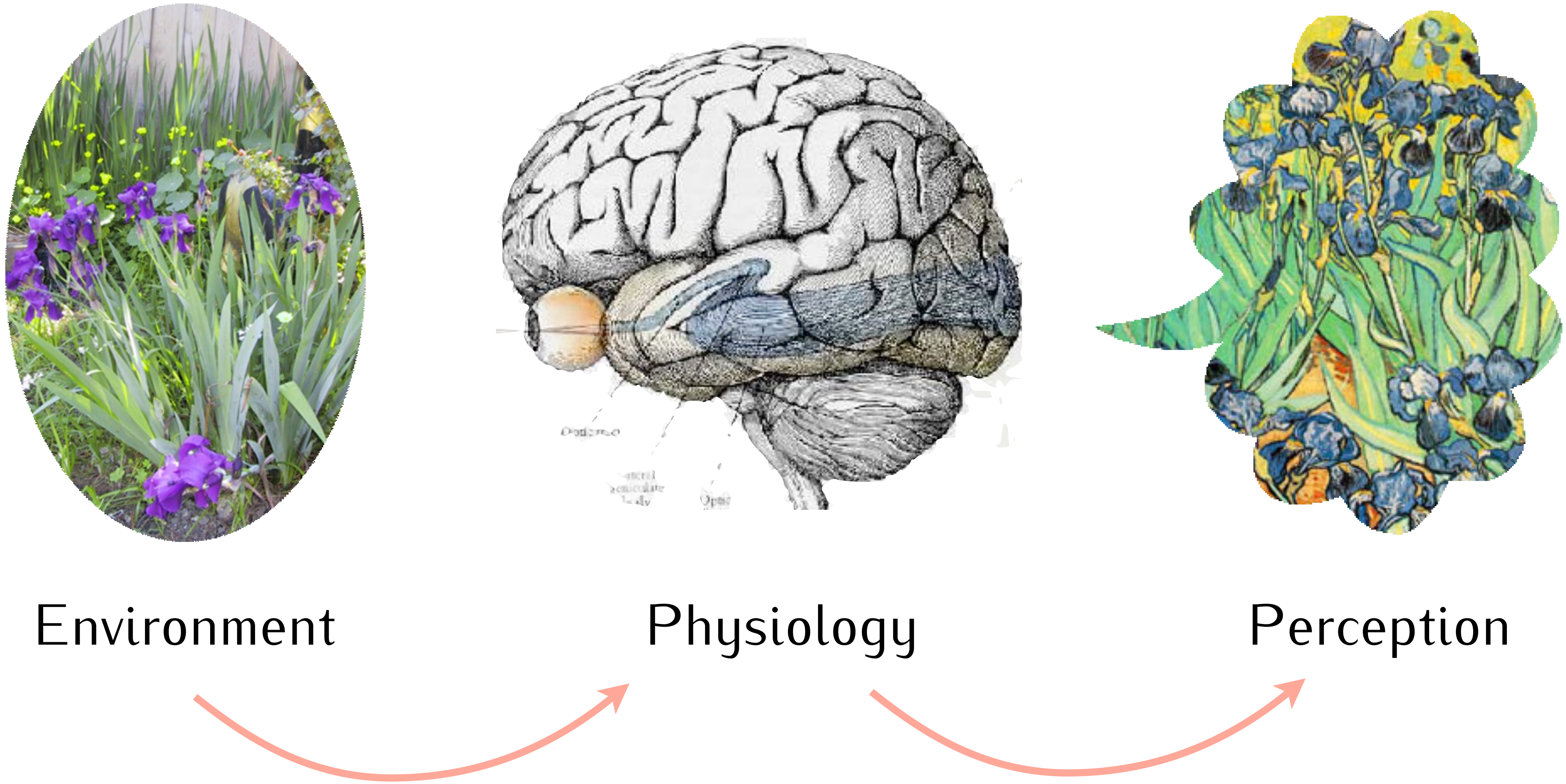
The importance of local gain control

Johannes Ballé
Center for Neural Science, NYU
Howard Hughes Medical Institute
(now with Google Inc.)

joint work with:
Valero Laparra, Universitat de València
Eero P. Simoncelli, CNS/Courant Institute/HHMI

Institut Henri Poincaré, Paris, 4 May 2017



- How do populations of neurons extract/represent visual information?
- In what ways is this matched to our visual environment?
- How do these representations enable/limit perception?
- What new principles may be gleaned from these representations, and applied to engineered imaging or vision systems?

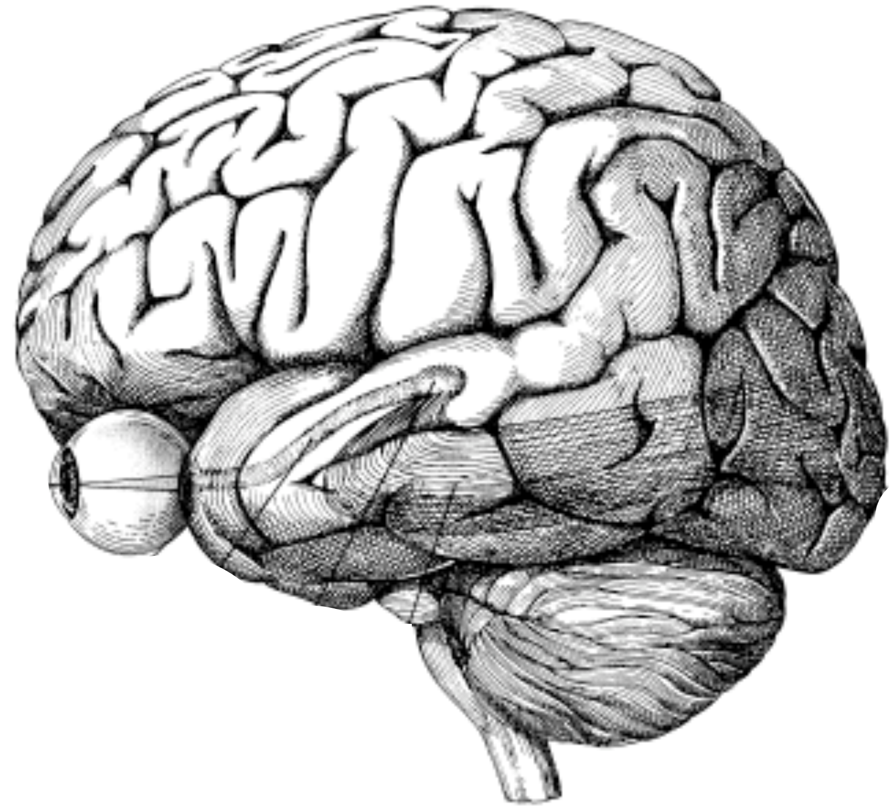
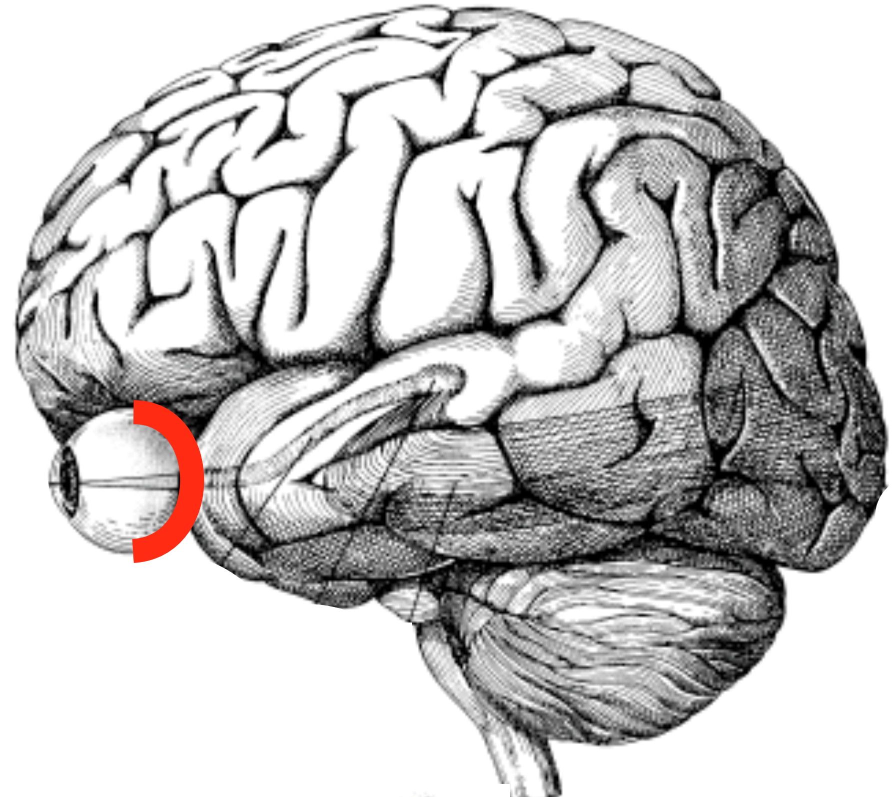


figure: Hubel 1995



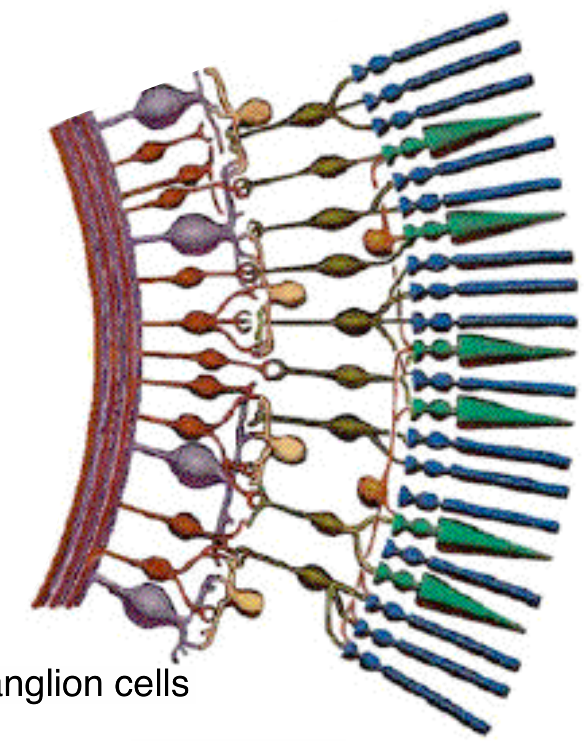


figure: Hubel 1995

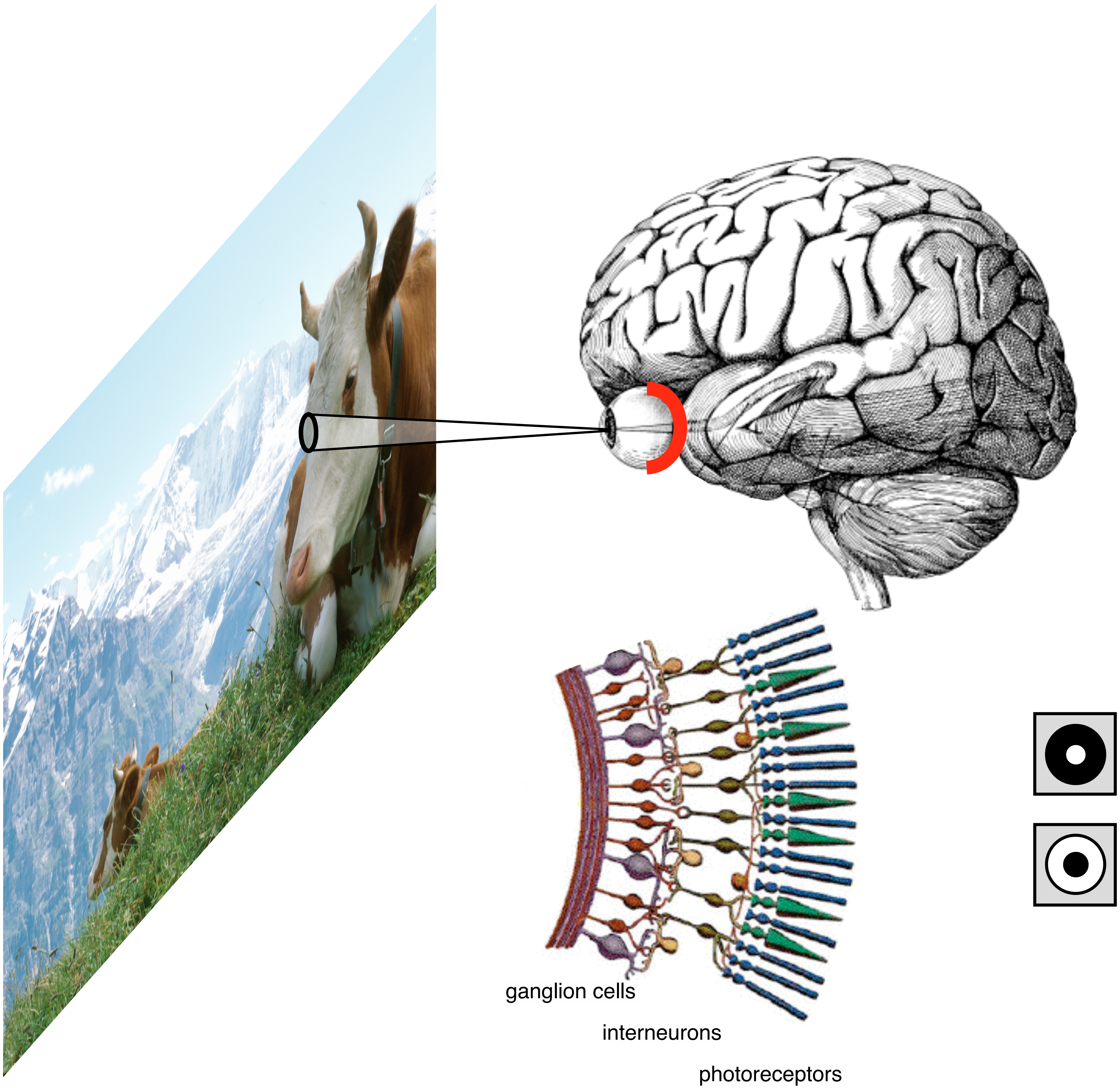
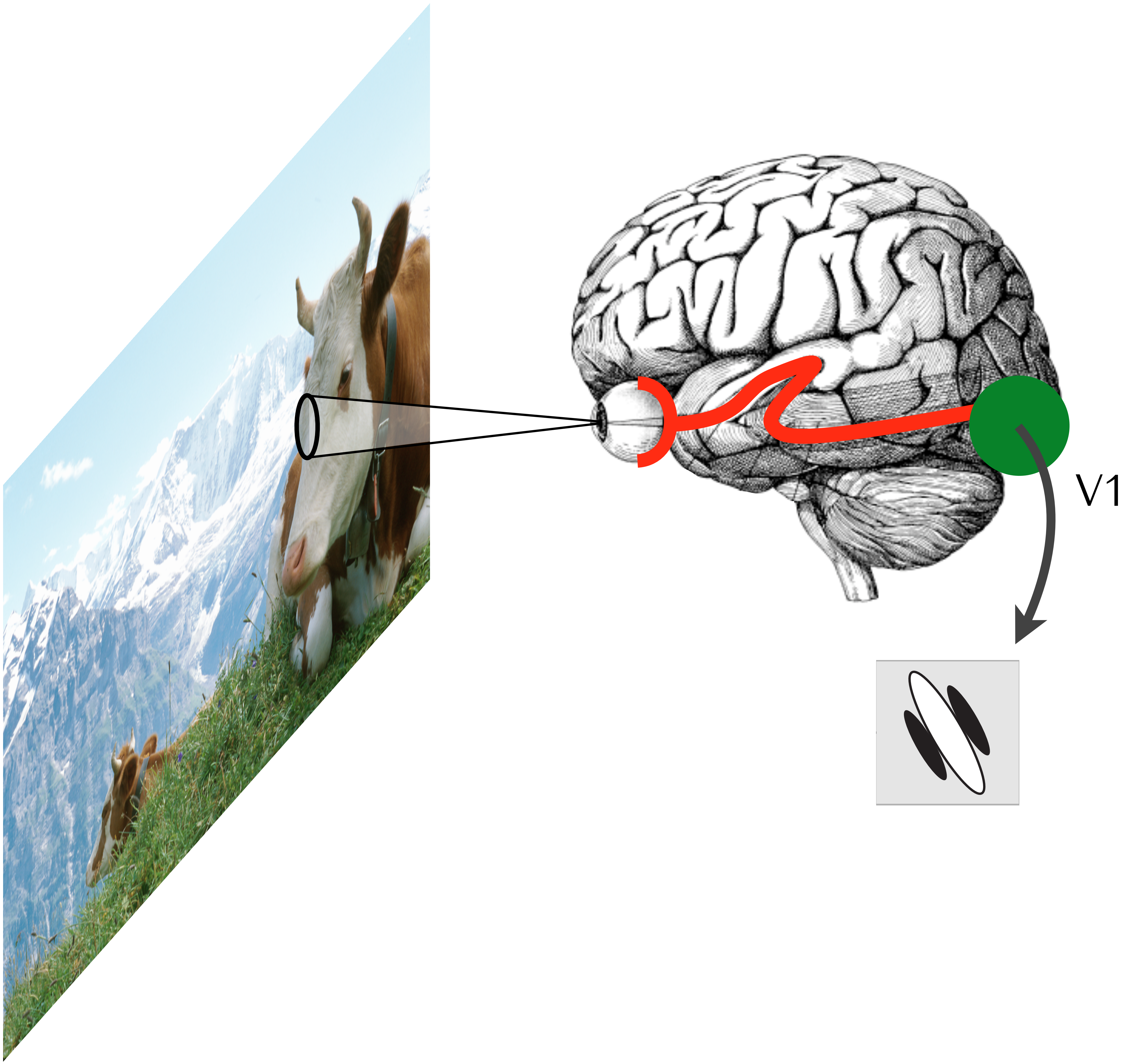
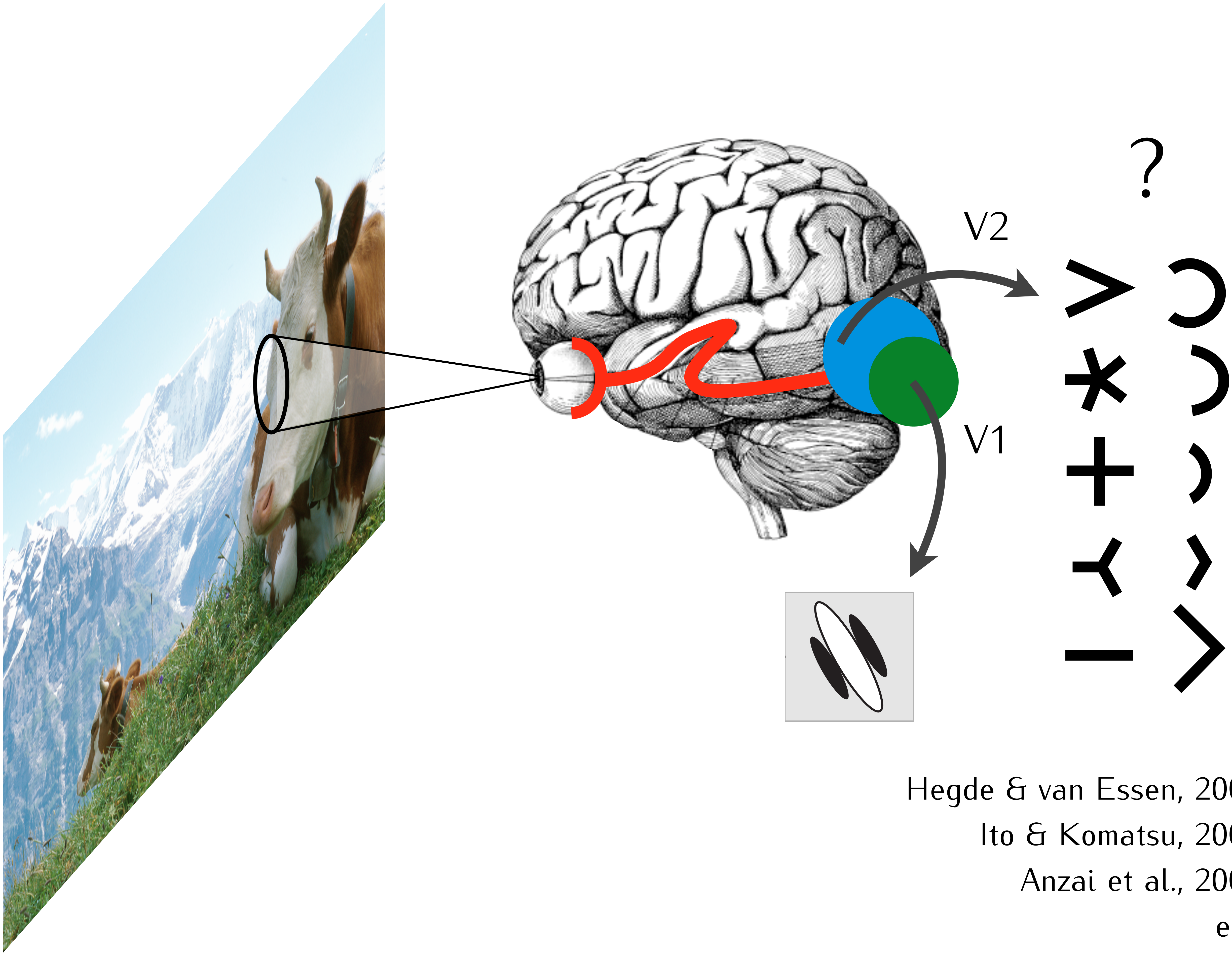


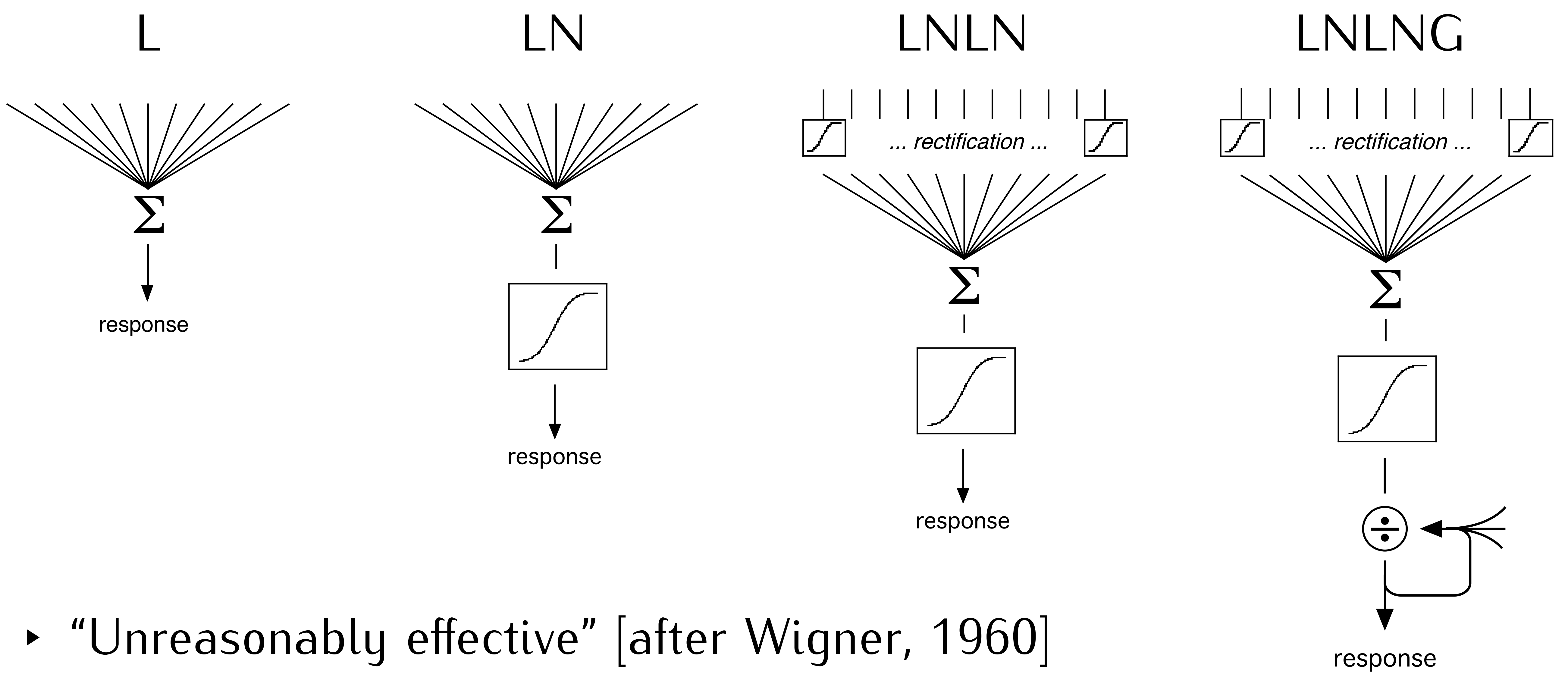
figure: Hubel 1995


V1

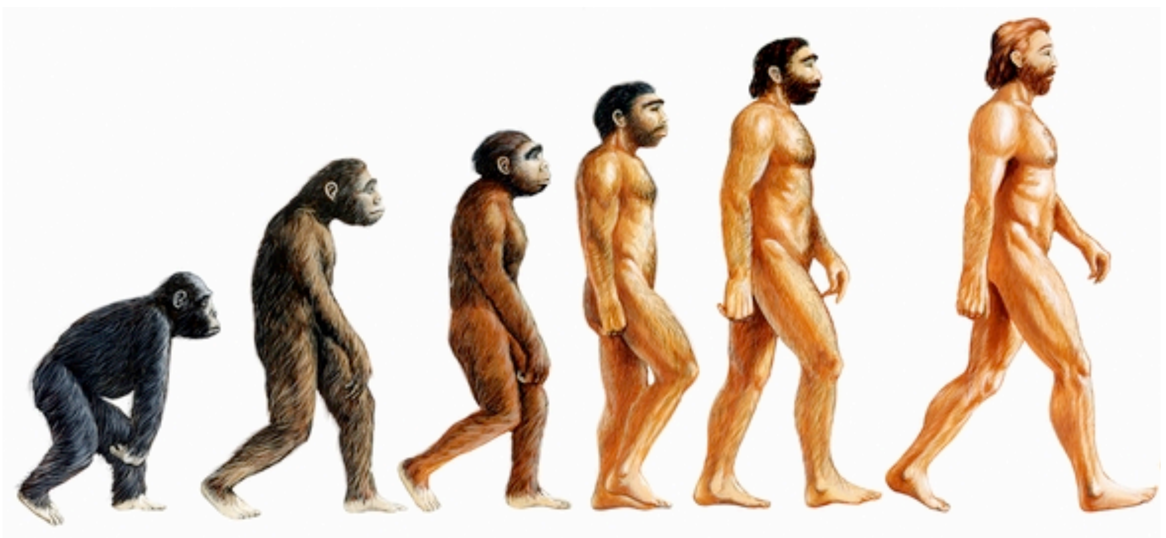


Hegde & van Essen, 2000
Ito & Komatsu, 2004
Anzai et al., 2007
etc.

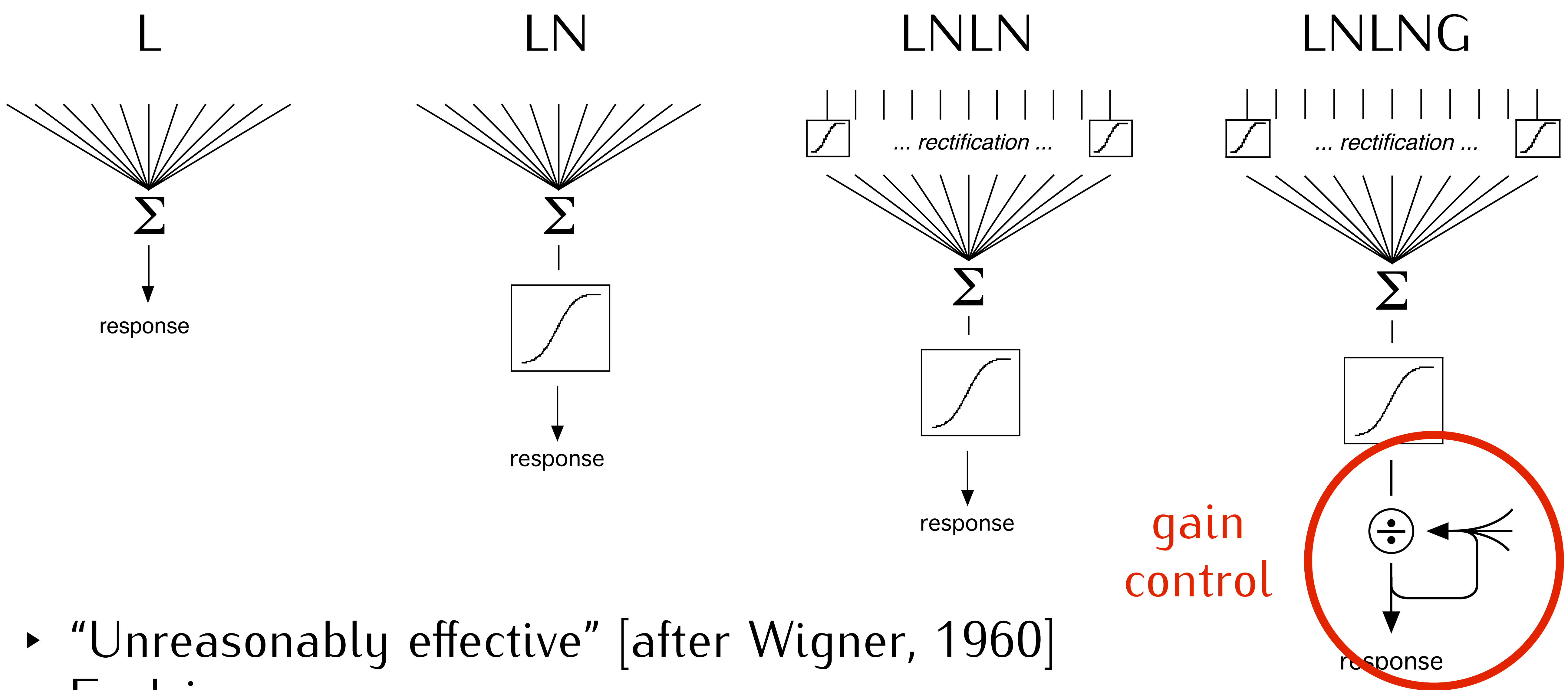
Canonical functional models for sensory neurons



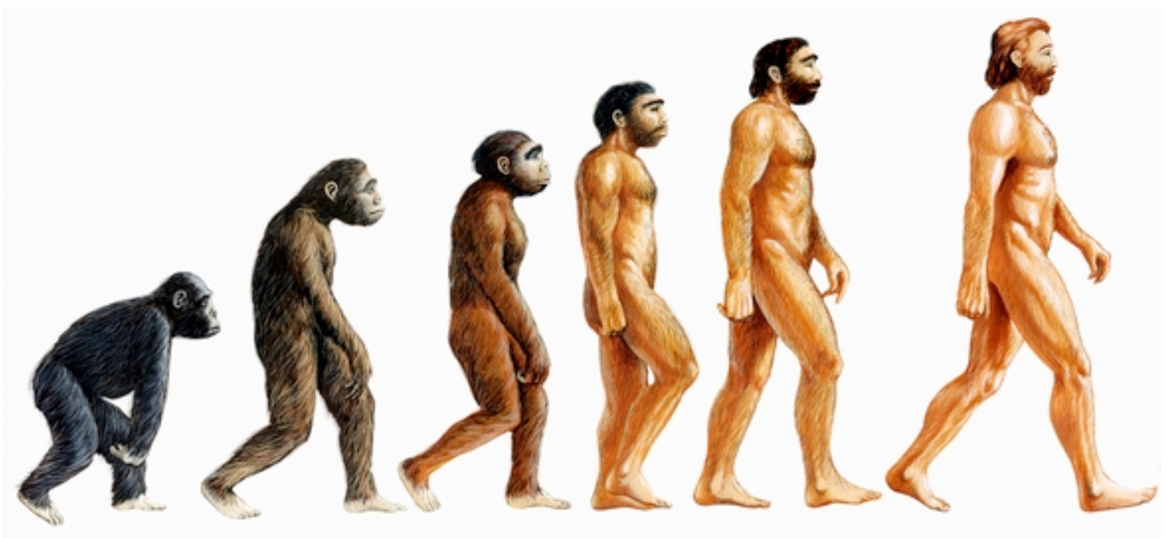
- "Unreasonably effective" [after Wigner, 1960]
- Evolving ...



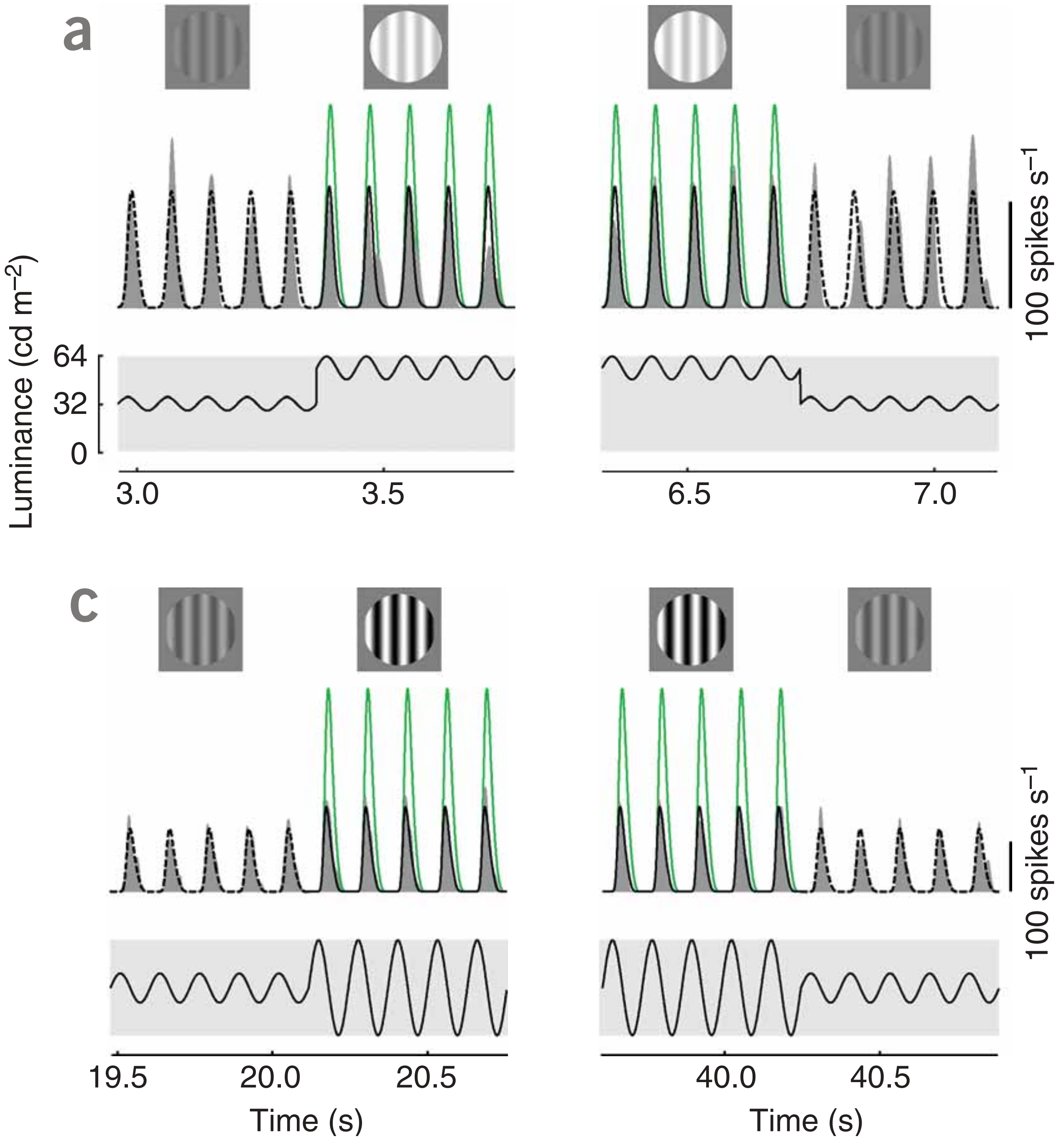
Canonical functional models for sensory neurons



- "Unreasonably effective" [after Wigner, 1960]
- Evolving ...

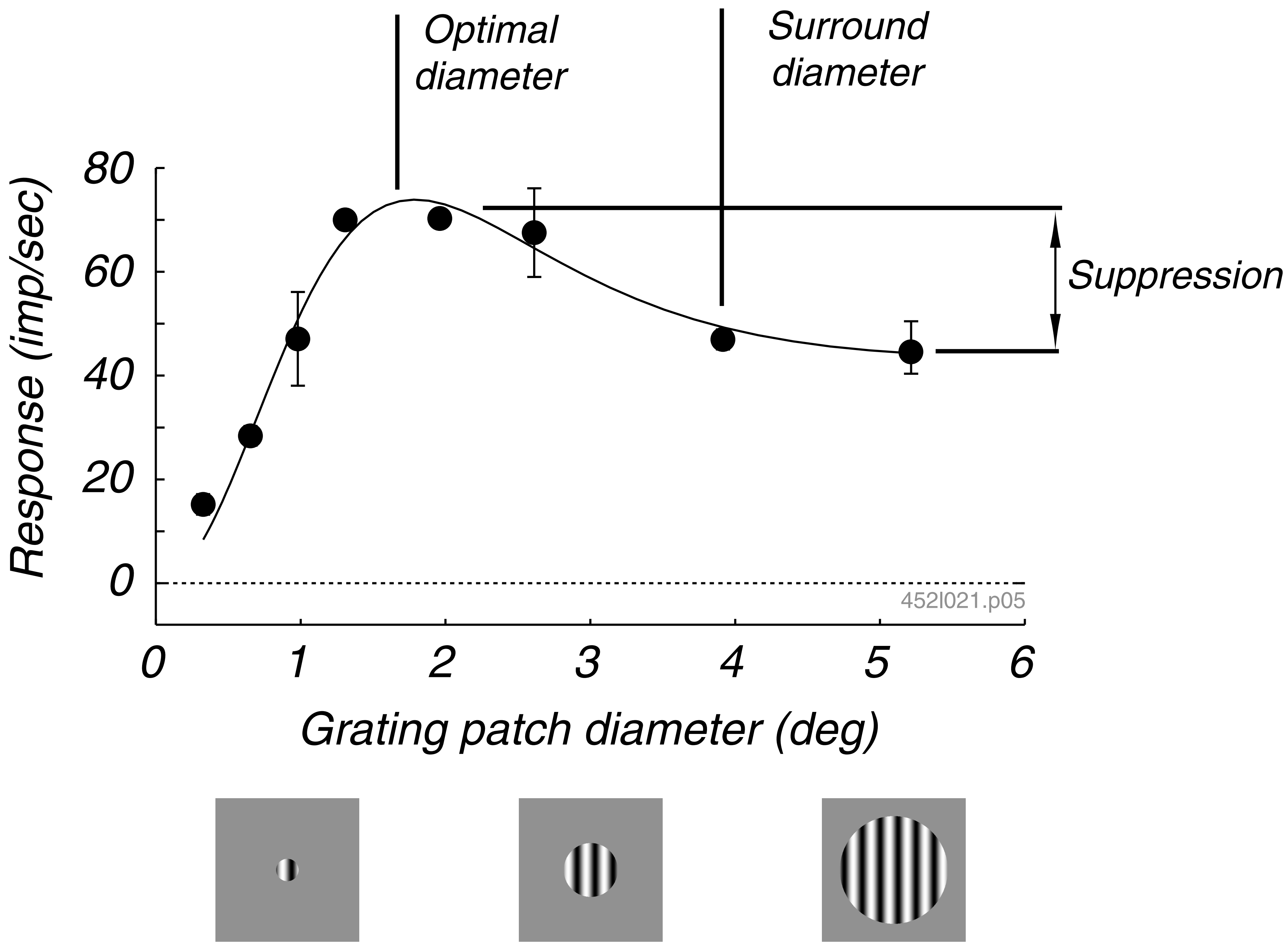


Temporally adaptive gain control



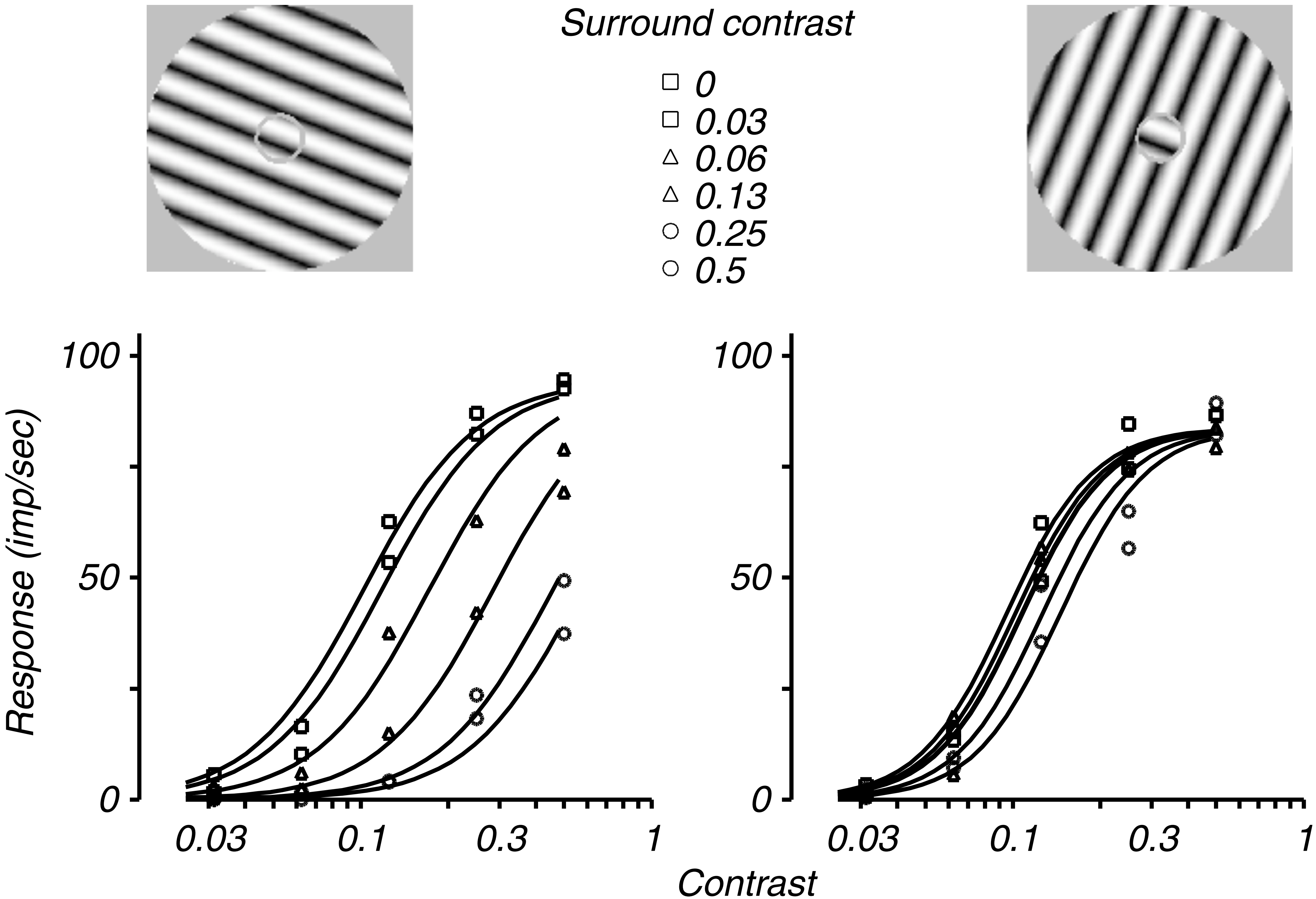
Mante et. al., 2005

V1: Surround suppression



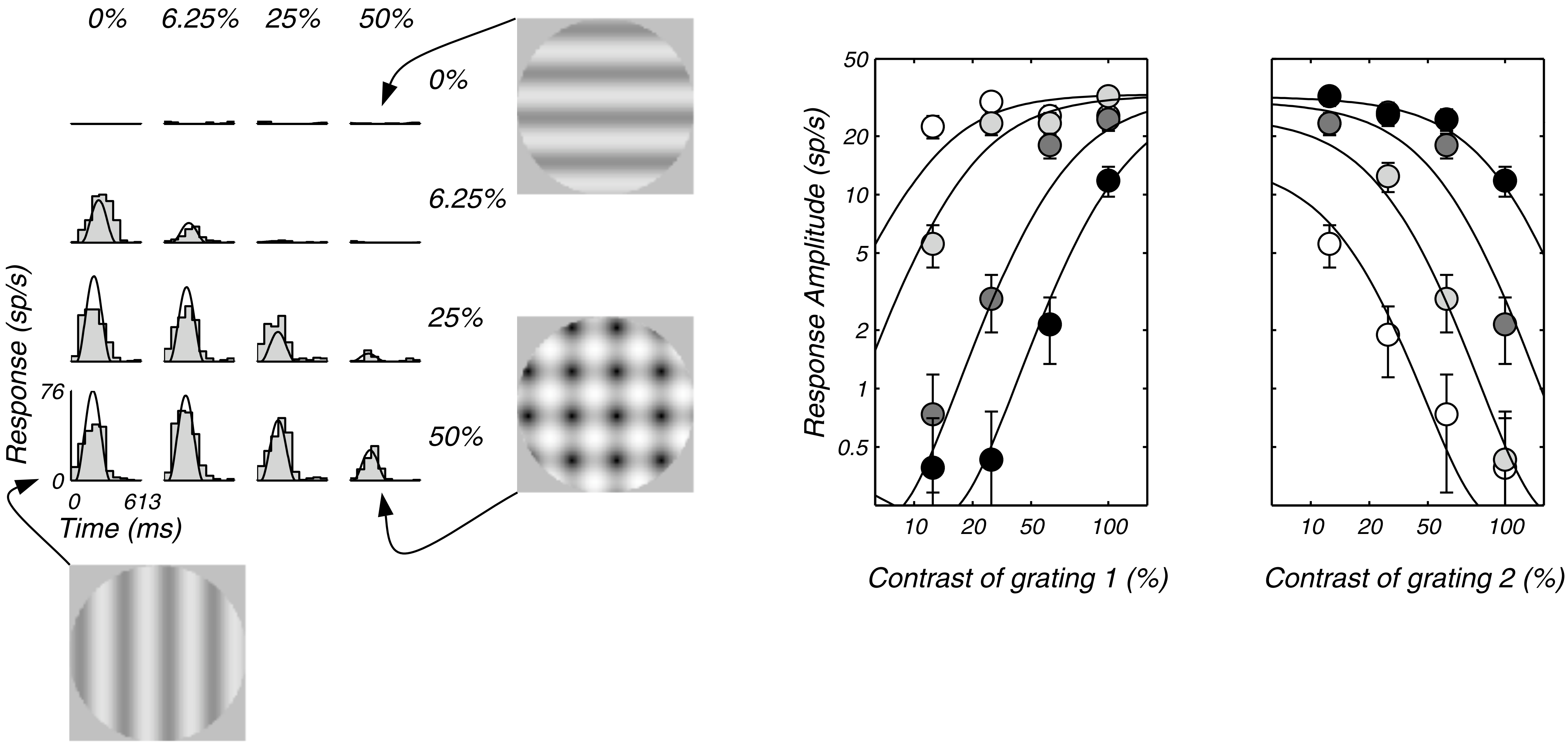
Cavanaugh et. al., 2002

V1: Cross-orientation suppression



Cavanaugh et. al., 2002

V1: Cross-orientation suppression

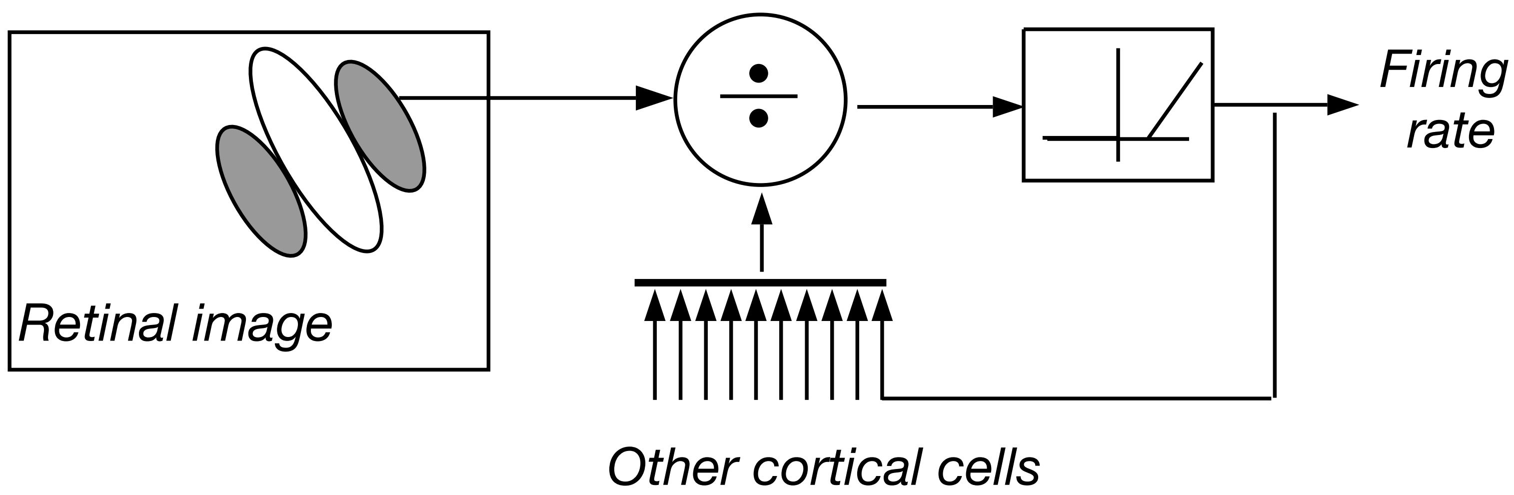


Carandini et. al., 1997

V1 normalization model

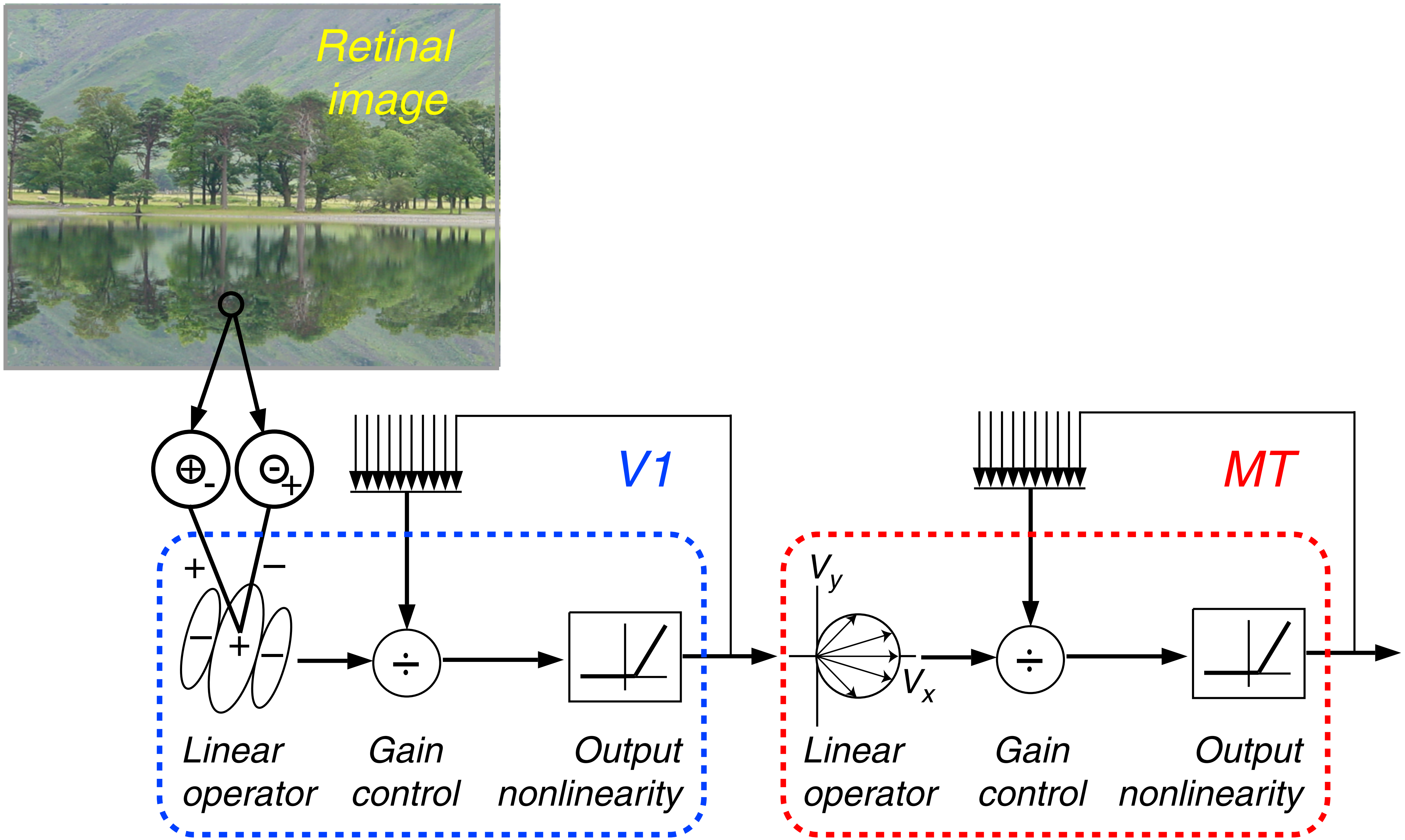
The linear model of simple cells

The normalization model of simple cells



Carandini, Heeger, and Movshon, 1996
Carandini & Heeger, 2012

Example: area MT



Simoncelli & Heeger, 1998

Generalized divisive normalization (GDN)

$\boldsymbol{x}$

$$\boldsymbol{y} = g_a(\boldsymbol{x}) \text{ such that:}$$

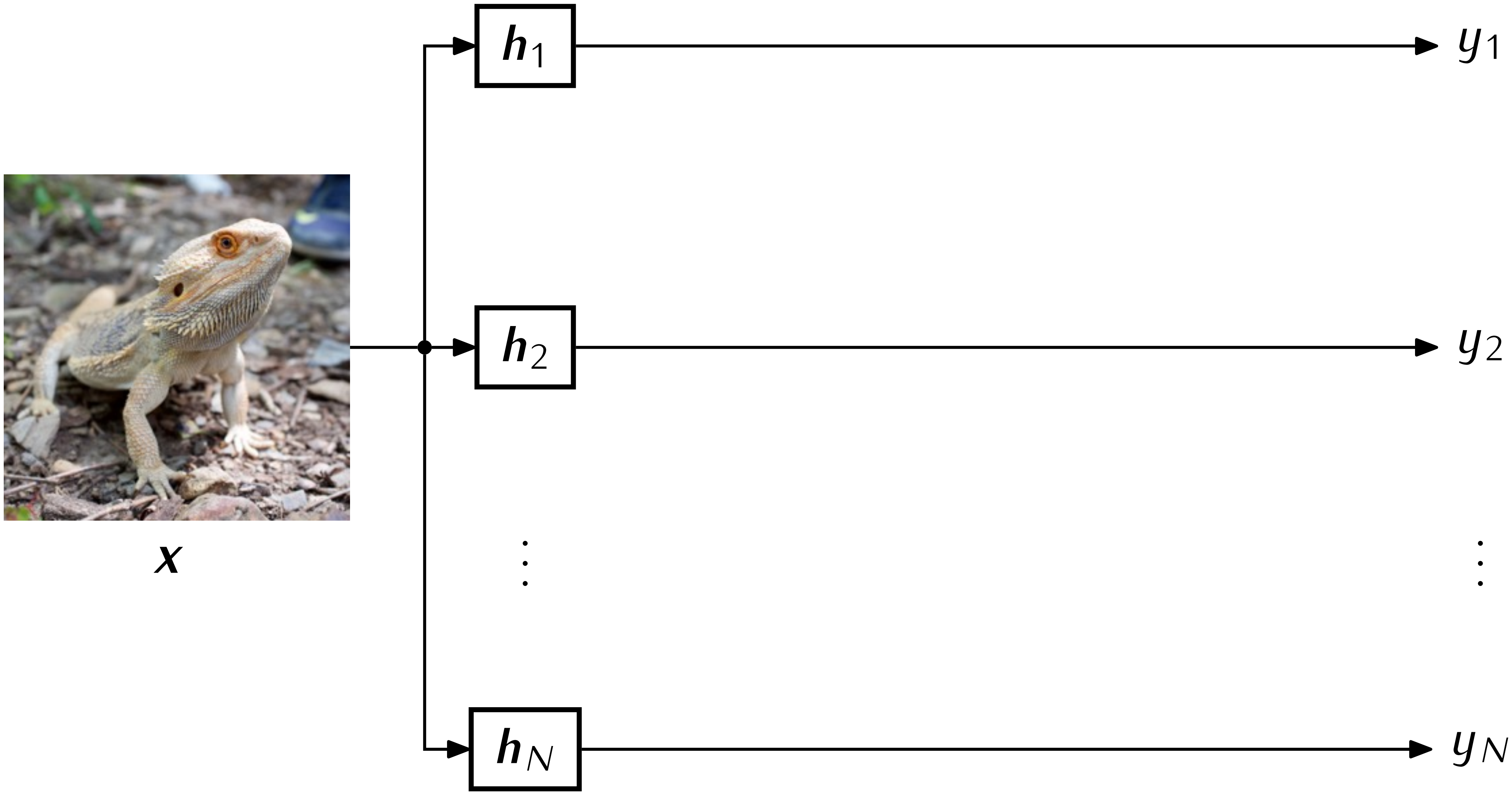
$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{x}$$

$$y_i = \frac{z_i}{(\beta_i + \sum_j \gamma_{ij}|z_j|^{\alpha_j})^{\varepsilon_i}}$$

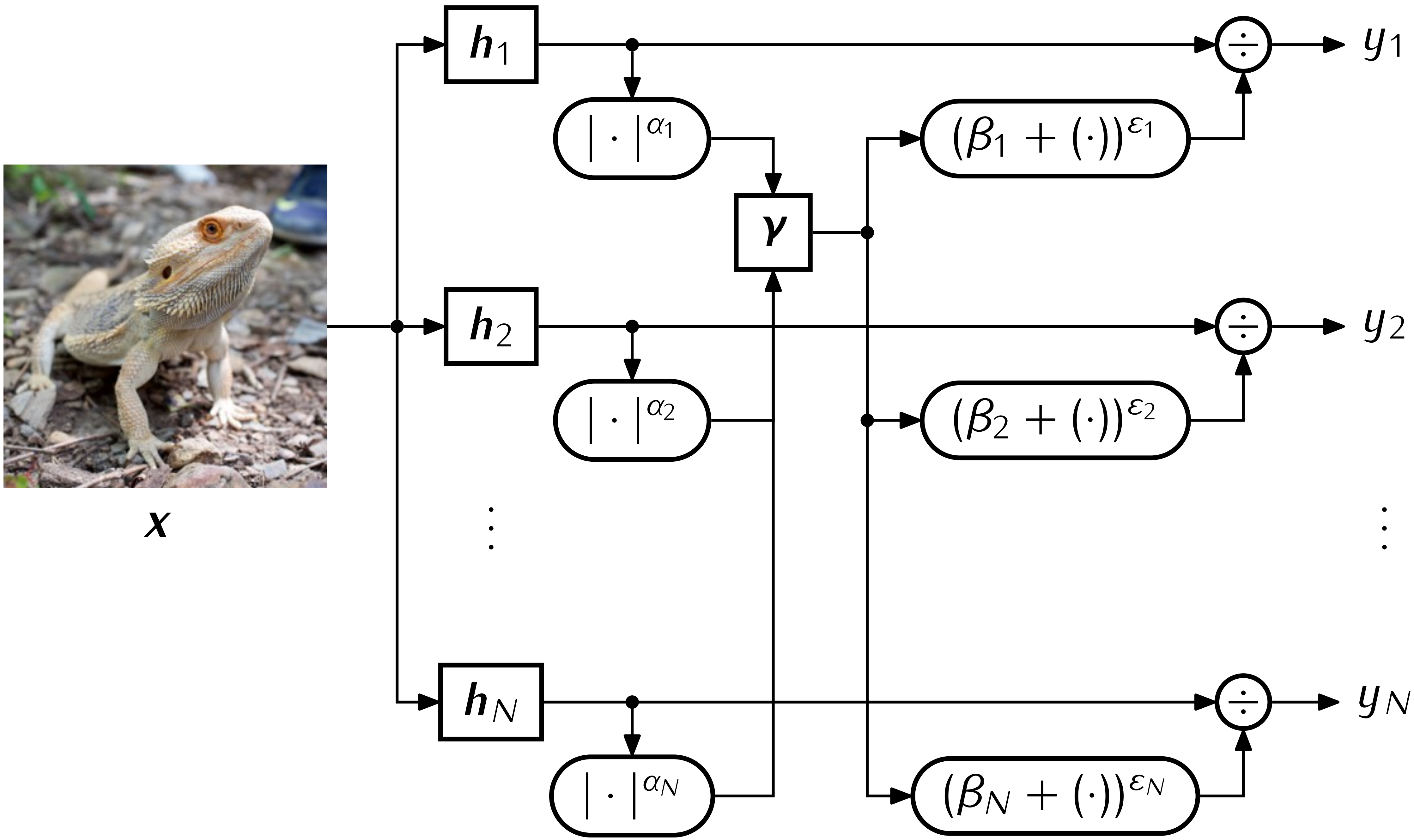
$\boldsymbol{H}$ linear front-end (set of linear filters, DCT, wavelets ...)

$\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\varepsilon}$ parameters of nonlinear transformation

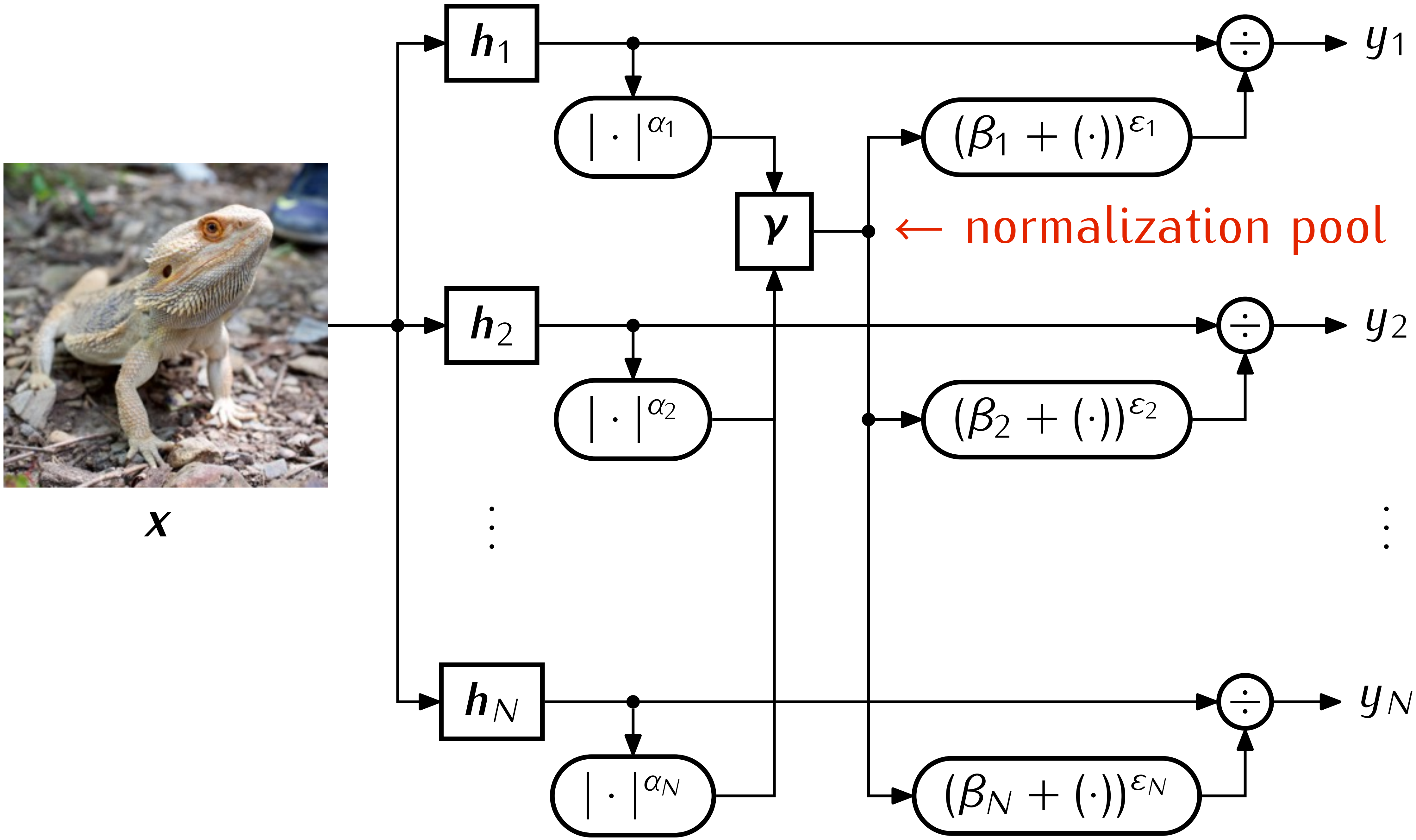
Generalized divisive normalization (GDN)



Generalized divisive normalization (GDN)



Generalized divisive normalization (GDN)



Approximate inverse (IGDN)

$\tilde{\boldsymbol{x}}$

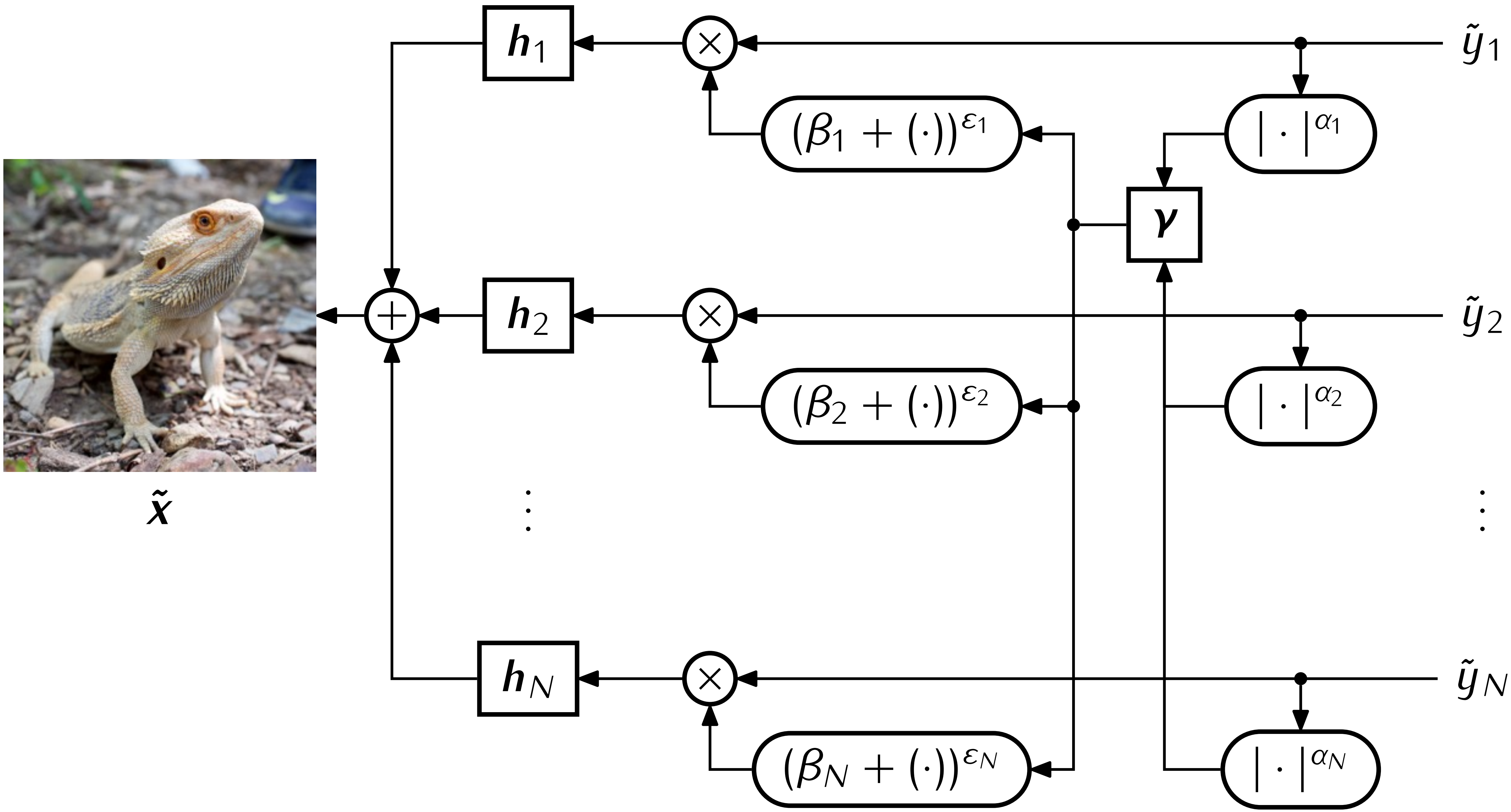
$\tilde{\boldsymbol{x}} = g_s(\tilde{\boldsymbol{y}})$ such that:

$$z_i = \tilde{y}_i \cdot (\beta'_i + \sum_j \gamma'_{ij} |\tilde{y}_j|^{\alpha'_j})^{\varepsilon'_i}$$

$$\tilde{\boldsymbol{x}} = \boldsymbol{H}' \boldsymbol{z}$$

one step of fixed point iteration
or approximate inverse with distinct parameters

Approximate inverse (IGDN)



one step of fixed point iteration
or approximate inverse with distinct parameters

How to choose the parameters?

We typically place some additional constraints, and then use optimization to find the best parameters for the application.

(not going to talk about details, please ask if you're interested)

Applications:

perceptual metrics
perceptually optimized rendering
density estimation
image compression
denoising
...

Applications:

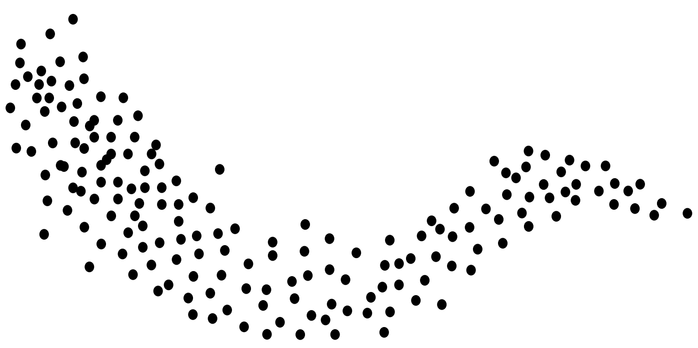
perceptual metrics
perceptually optimized rendering
density estimation
image compression
denoising

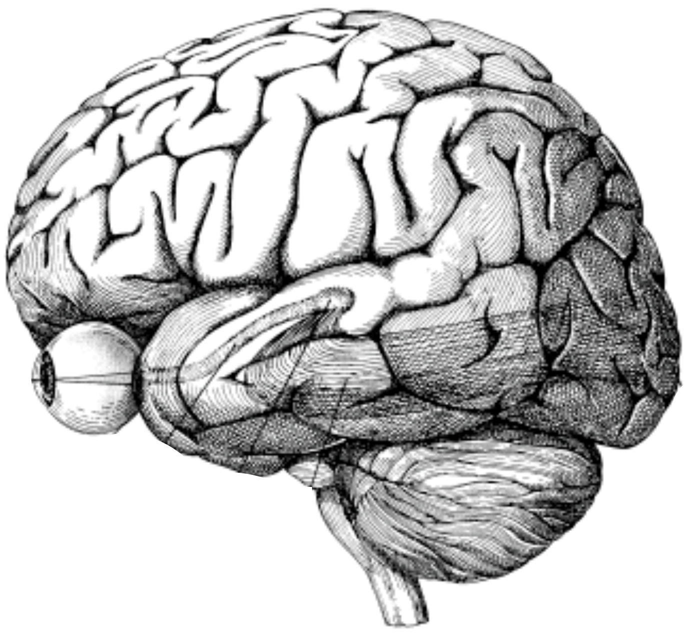
...

Measure distance in a perceptual representation

images
in signal domain

images
in perceptual domain





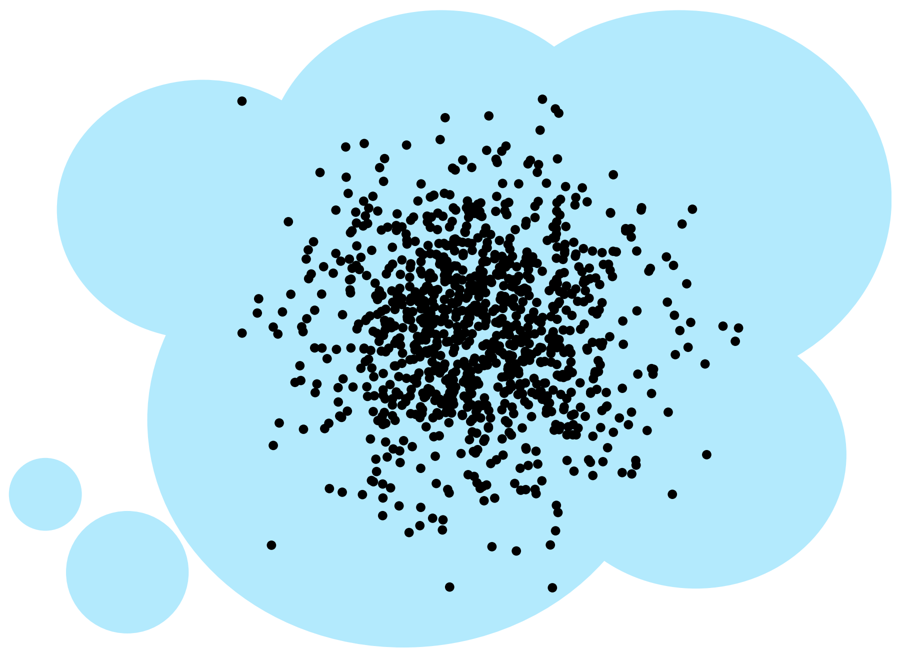


figure: Hubel, 1995

Measure distance in a perceptual representation

images
in signal domain

images
in perceptual domain

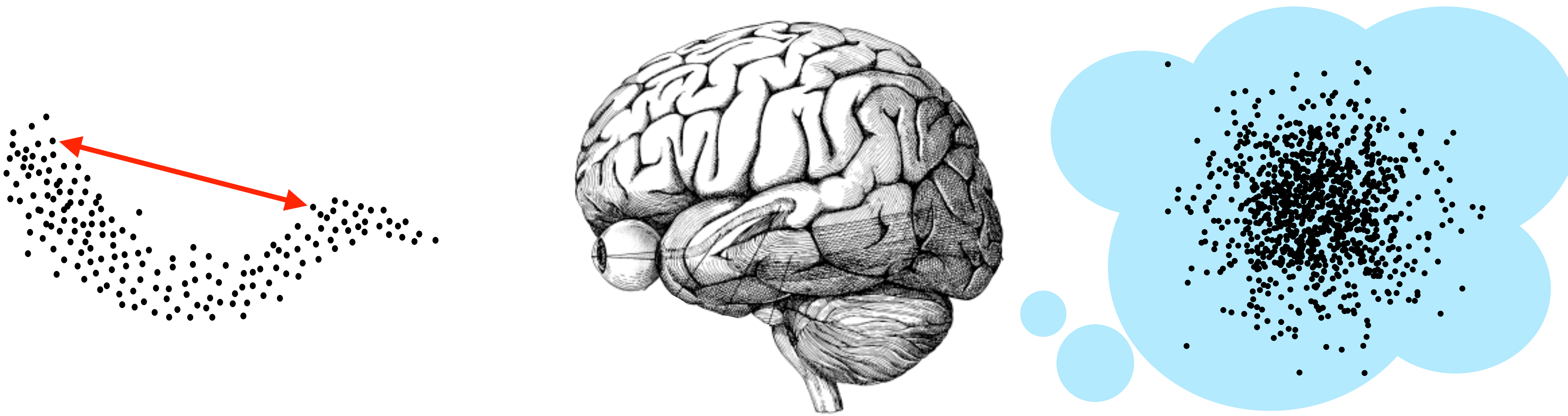


figure: Hubel, 1995

Measure distance in a perceptual representation

images
in signal domain

images
in perceptual domain

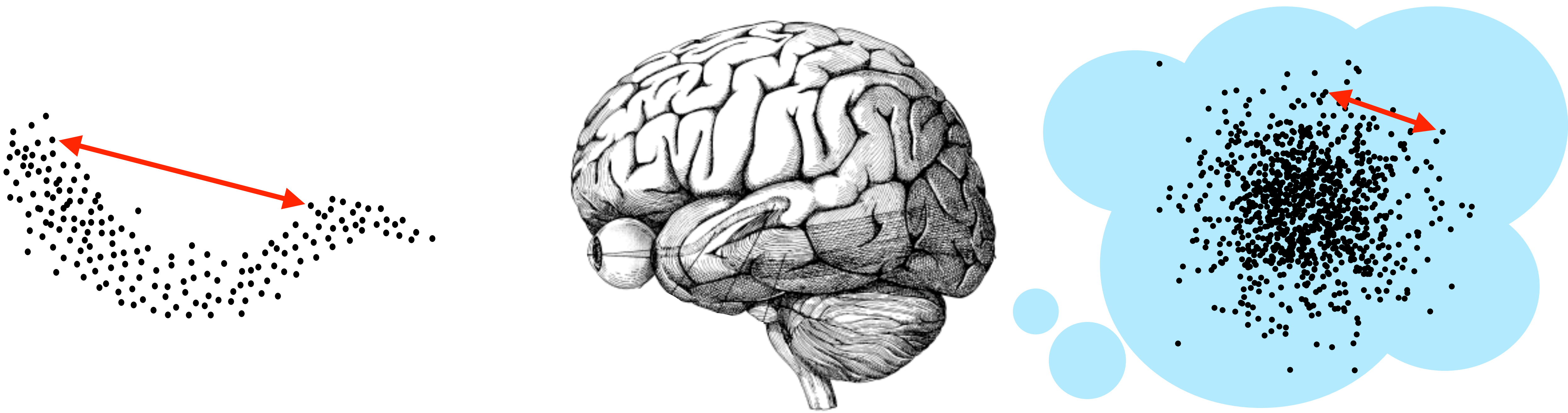
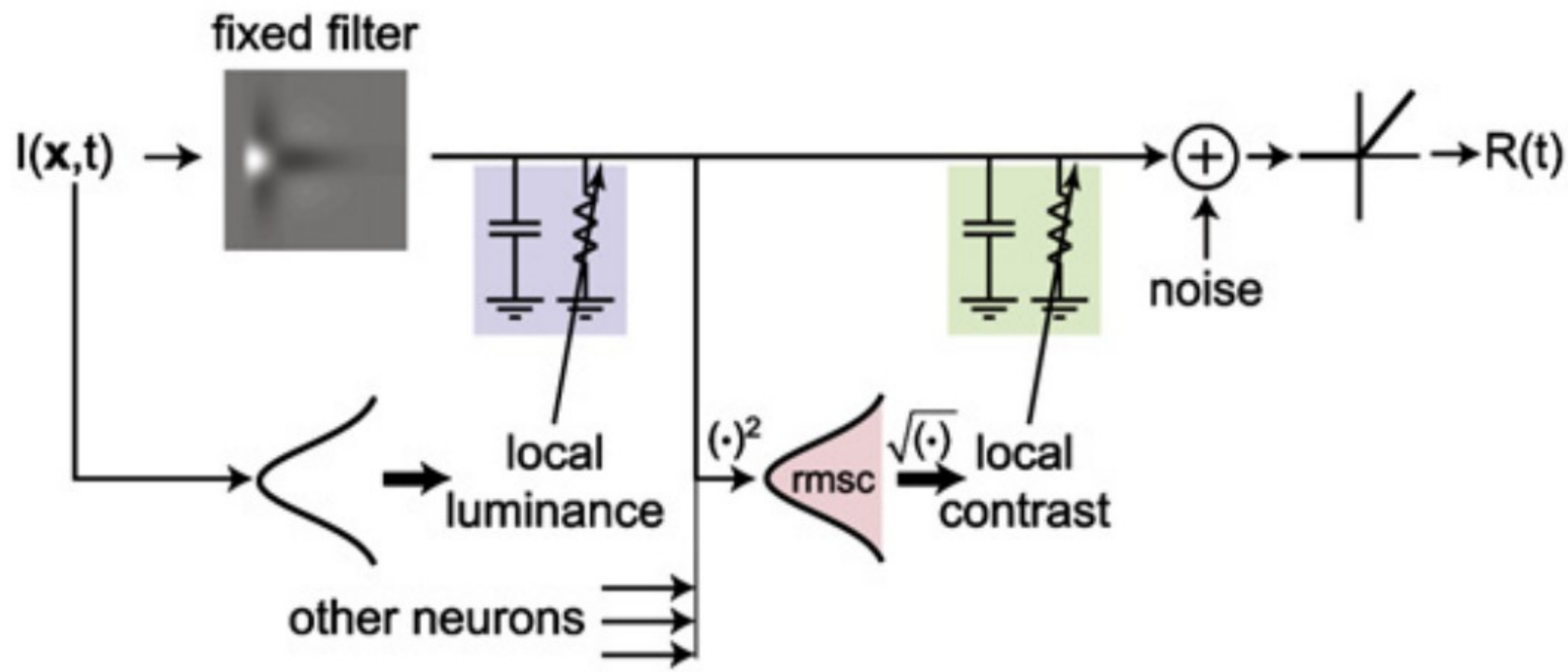


figure: Hubel, 1995

Aim: imitate processing in retina + LGN (pre-cortex)

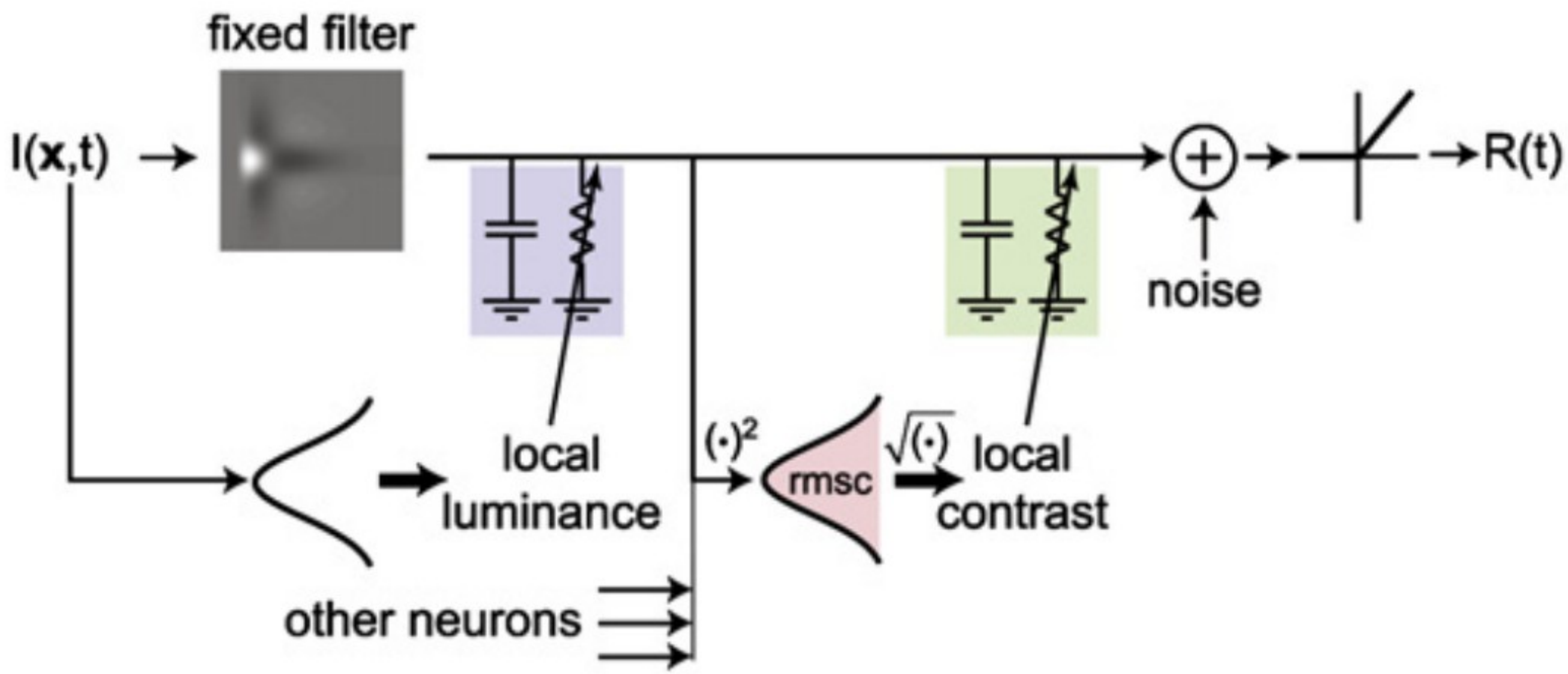
V. Mante, V. Bonin, and M. Carandini. Neuron 2008
“Functional mechanisms shaping lateral geniculate responses to artificial and natural stimuli”



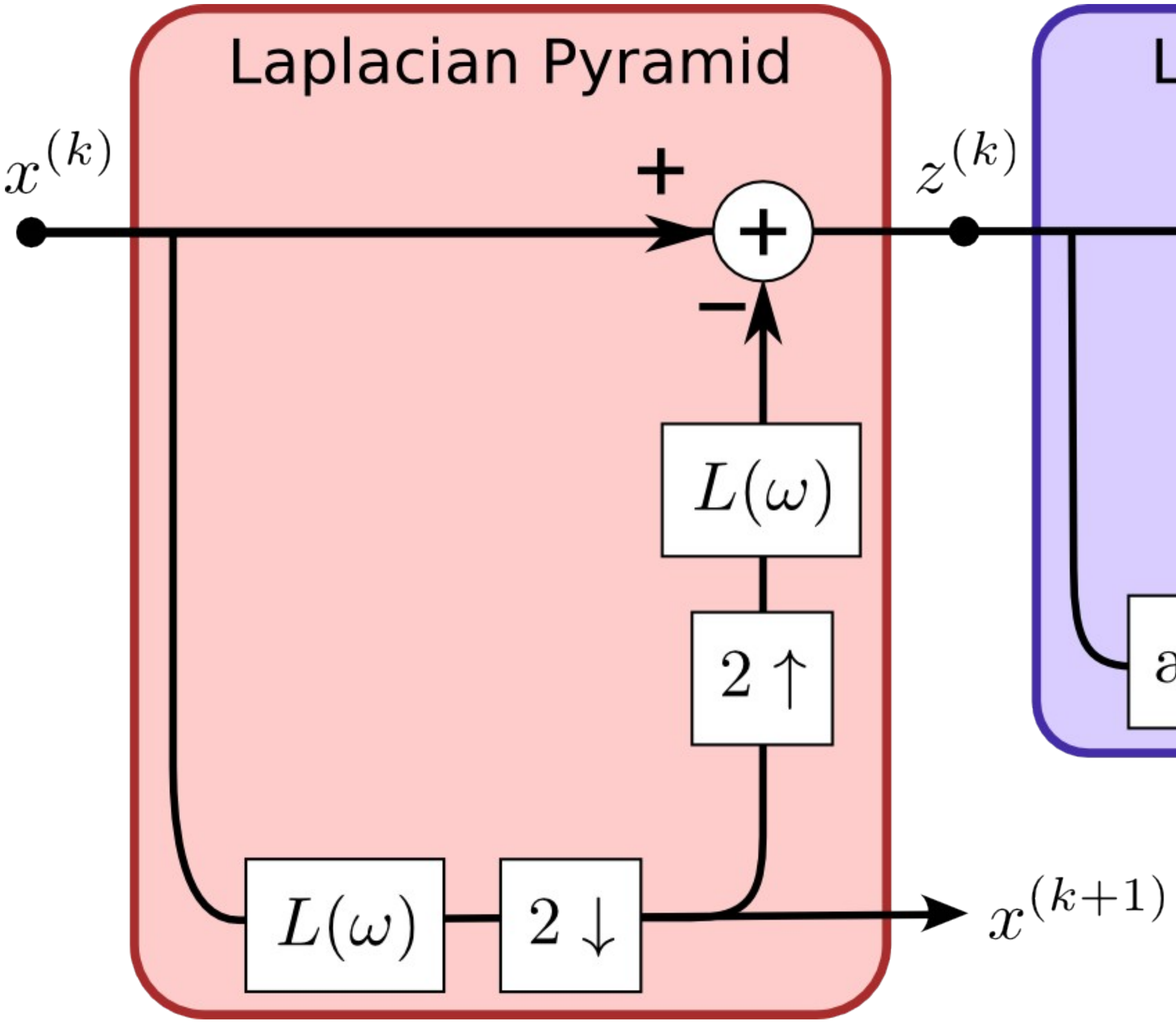
Aim: imitate processing in retina + LGN (pre-cortex)

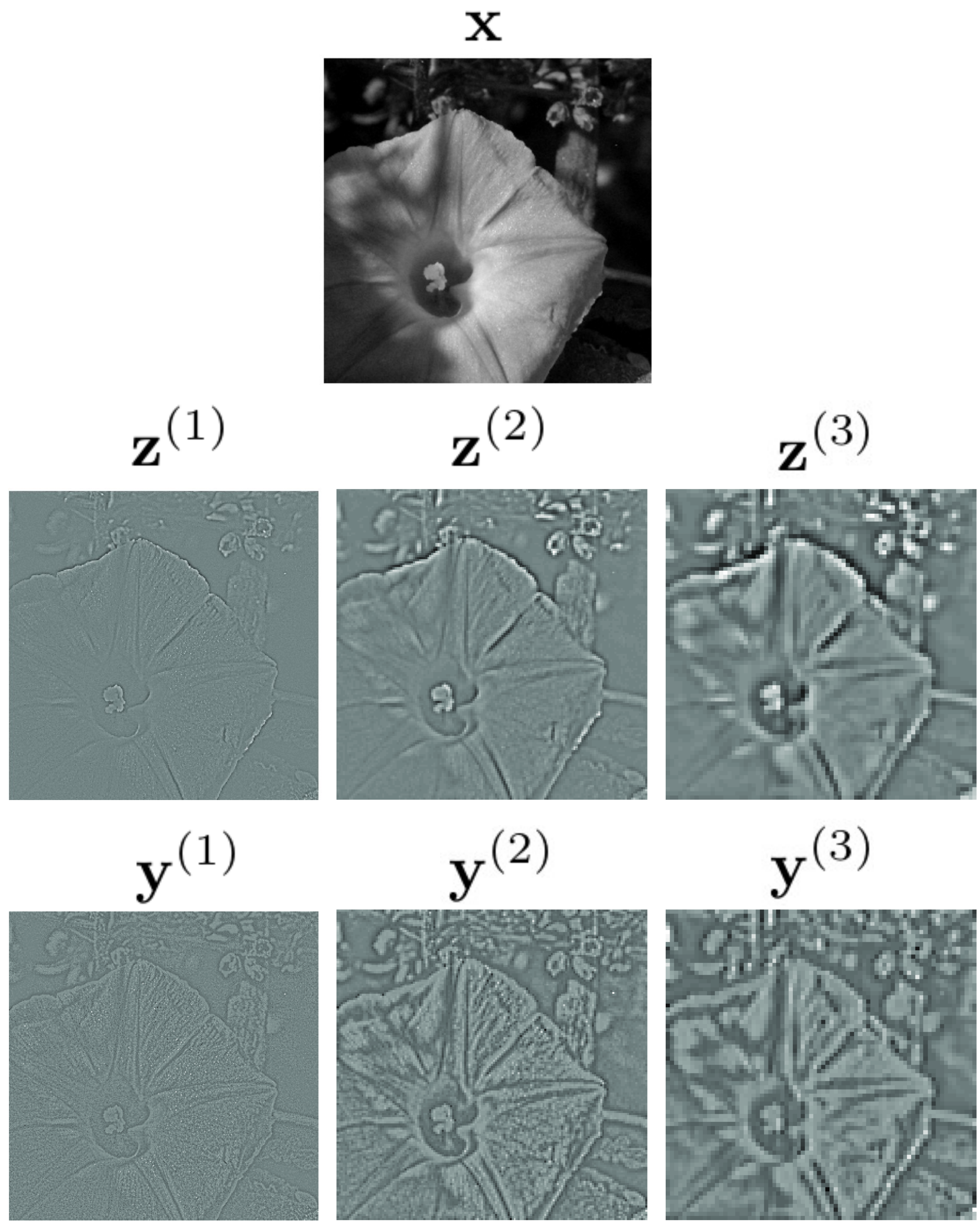
no oriented filters!

V. Mante, V. Bonin, and M. Carandini. Neuron 2008
"Functional mechanisms shaping lateral geniculate responses to artificial and natural stimuli"



Normalized Laplacian pyramid



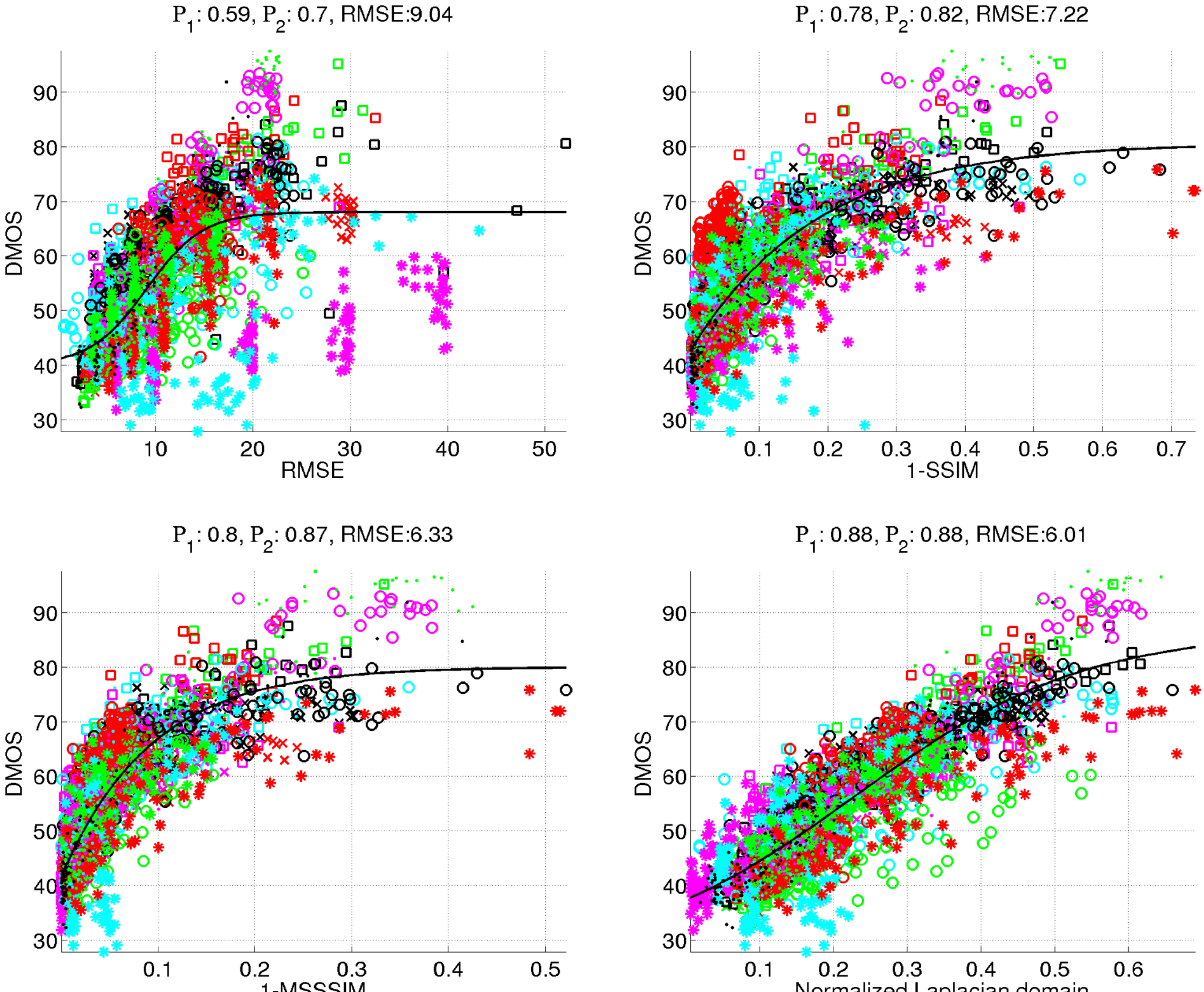


- Local mean subtraction

$$z = x - \hat{x}$$

- Local amplitude division (local Gain Control)

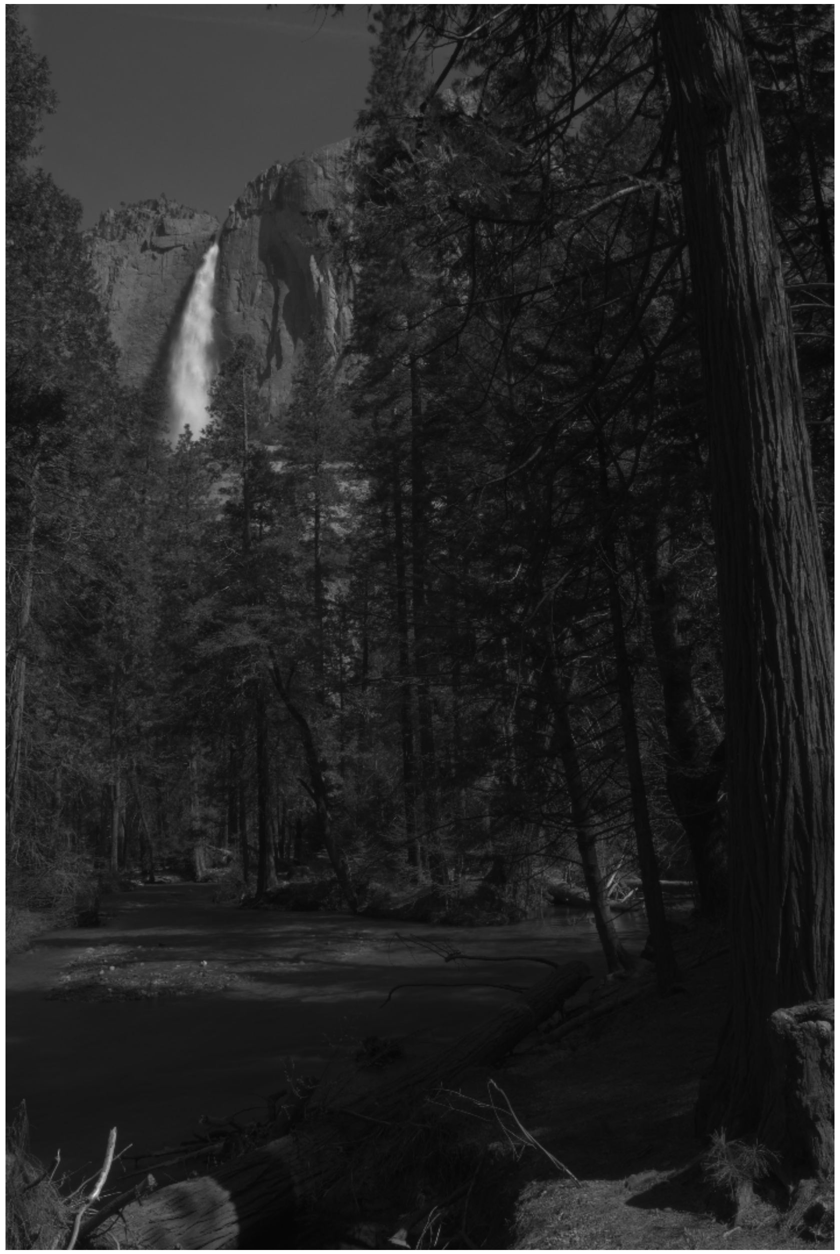
$$y = \frac{z}{\widehat{|z|}}$$



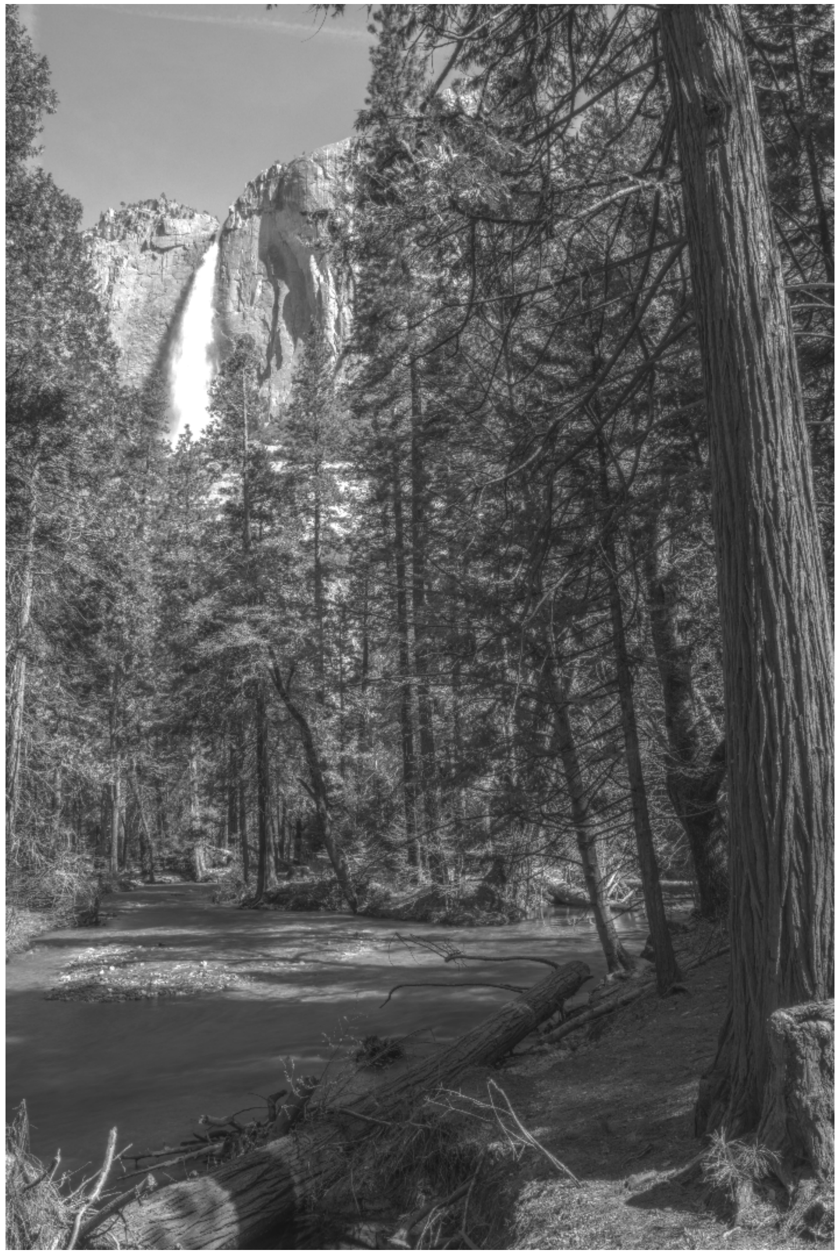
P1: 0.59, P2: 0.7, RMSE:9.04
DMOS
RMSE
P1: 0.78, P2: 0.82, RMSE:7.22
DMOS
1-SSIM
P1: 0.8, P2: 0.87, RMSE:6.33
DMOS
1-MSSSIM
P1: 0.88, P2: 0.88, RMSE:6.01
DMOS
Normalized Laplacian domain

Optimized HDR rendering without “tweaking” parameters

rescaled



Paris et al. [12]



NLP

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} d(\boldsymbol{x}, \boldsymbol{x}_{\text{HDR}}) \quad \text{s.t. } \forall i : x_{\min} \leq x_i \leq x_{\max}$$

Applications:

perceptual metrics
perceptually optimized rendering
density estimation
image compression
denoising
...

Applications:

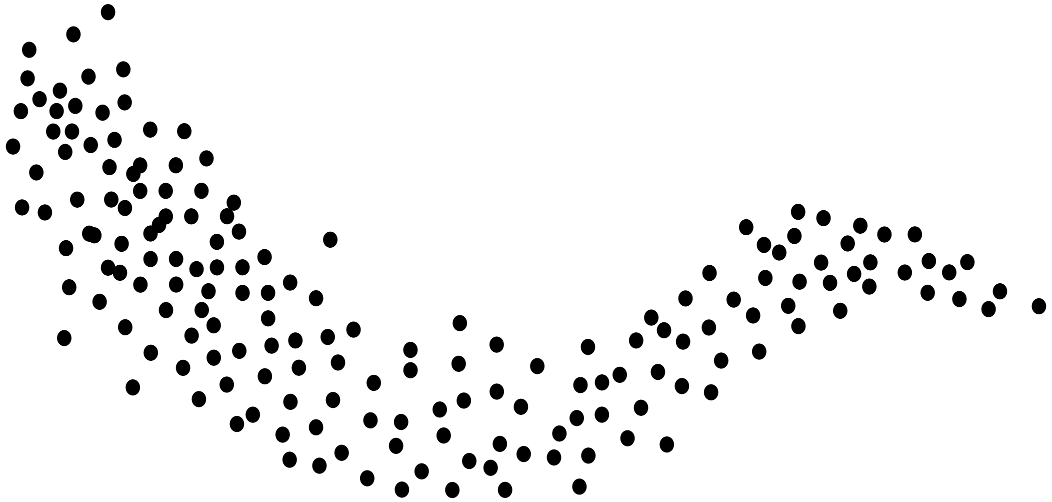
perceptual metrics
perceptually optimized rendering
density estimation
image compression
denoising

...

Has the human visual system
simply adapted to image statistics?

Density estimation (parametric density)

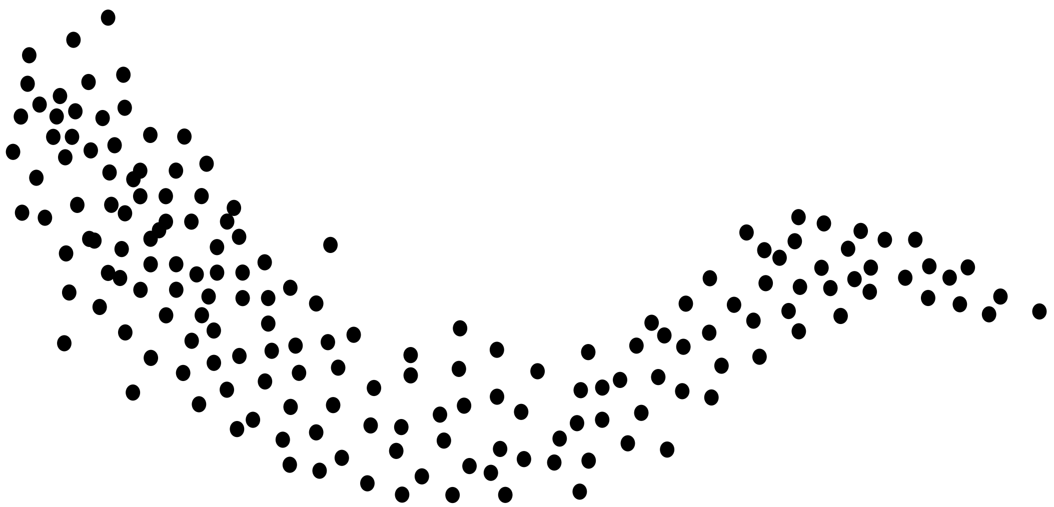
$$p_x(\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\Big(-f(\boldsymbol{x}; \boldsymbol{\theta})\Big)$$



Density estimation (parametric density)

$$p_x(\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\big(-f(\boldsymbol{x};\boldsymbol{\theta})\big)$$

$$Z(\boldsymbol{\theta}) = \int \exp\big(-f(\boldsymbol{x};\boldsymbol{\theta})\big)\, \mathrm{d}\boldsymbol{x}$$

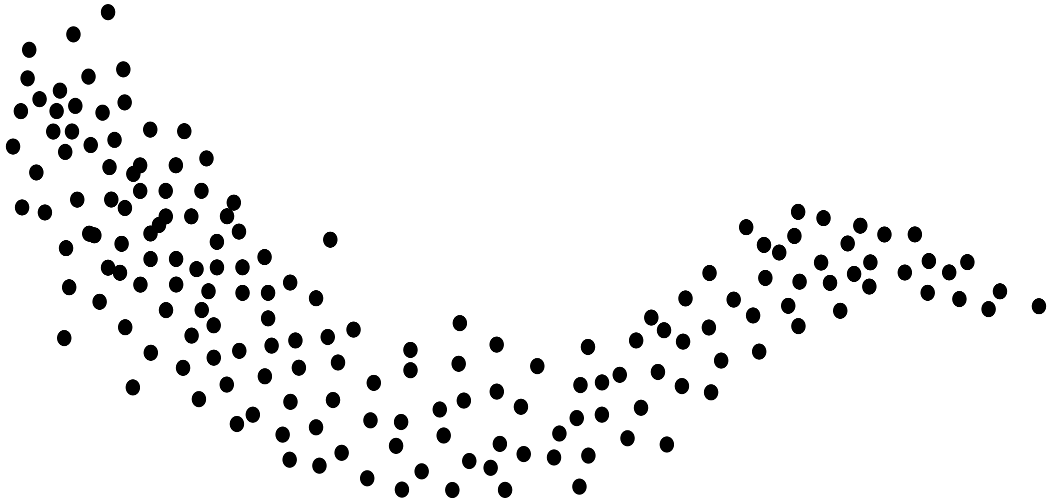


Density estimation (parametric density)

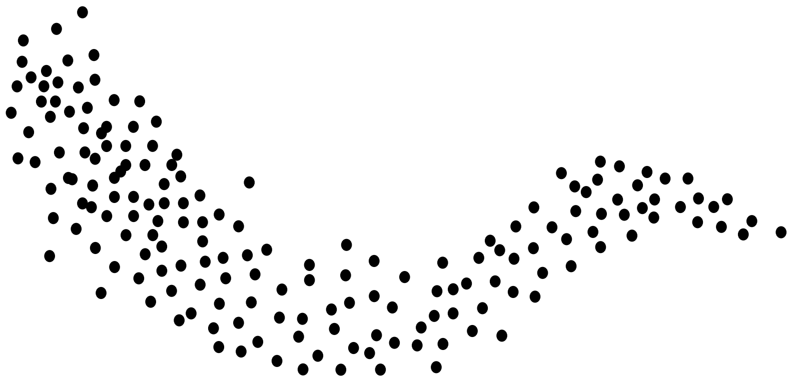
$$p_x(\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\big(-f(\boldsymbol{x};\boldsymbol{\theta})\big)$$

$$Z(\boldsymbol{\theta}) = \int \exp\big(-f(\boldsymbol{x};\boldsymbol{\theta})\big)\,\mathrm{d}\boldsymbol{x}$$

tractable?



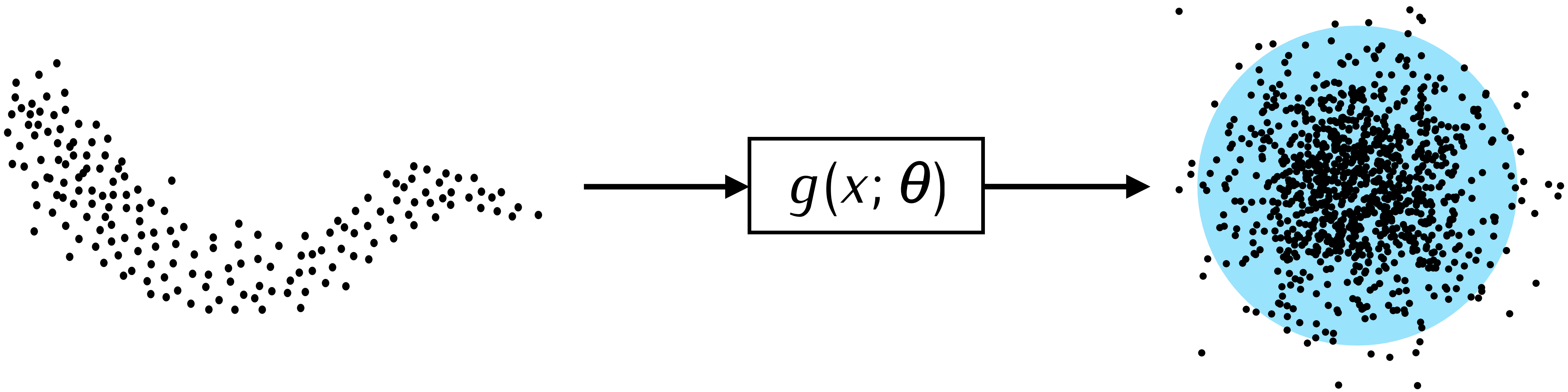
Density estimation (parametric transformation)



$\boldsymbol{x} \sim p_{\boldsymbol{x}}$

Friedman, 1984
Chen & Gopinath, 2001
Lyu & Simoncelli, 2009
Laparra et al., 2010
Dinh et al., 2015

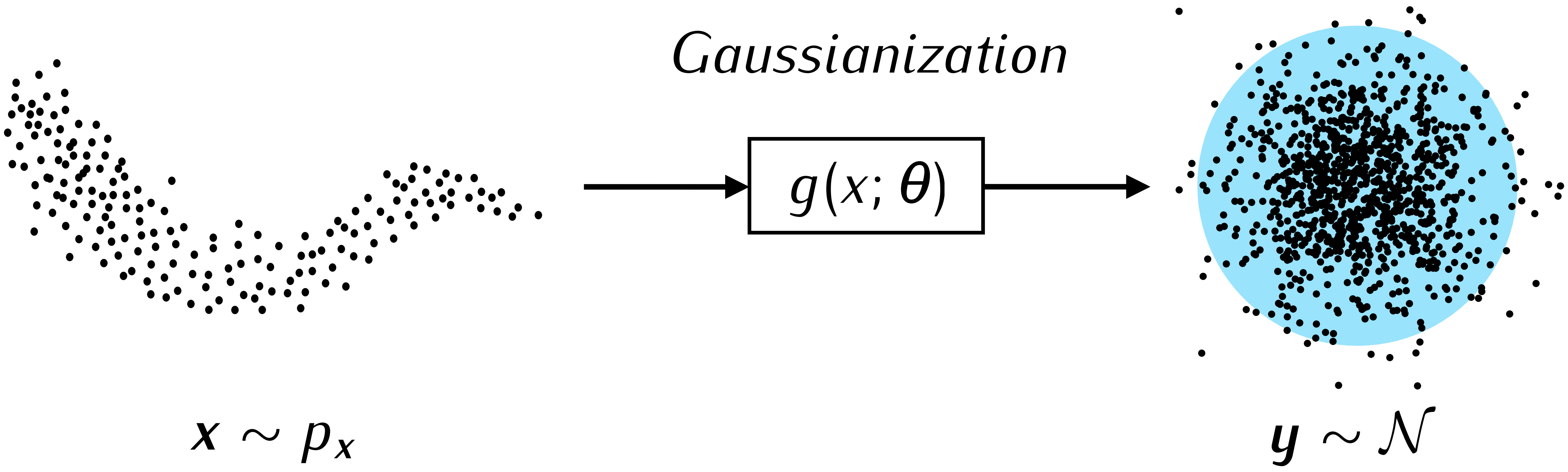
Density estimation (parametric transformation)



$\boldsymbol{x} \sim p_{\boldsymbol{x}}$ $\qquad$ $\boldsymbol{y} \sim \mathcal{N}$

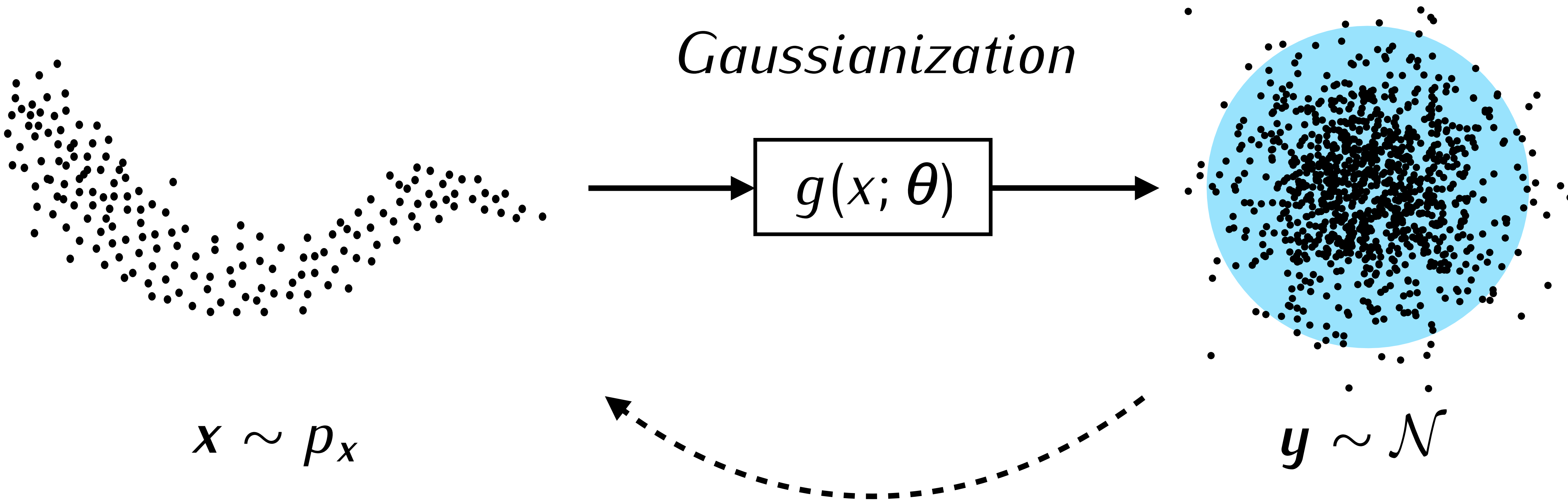
Friedman, 1984
Chen & Gopinath, 2001
Lyu & Simoncelli, 2009
Laparra et al., 2010
Dinh et al., 2015

Density estimation (parametric transformation)



Friedman, 1984
Chen & Gopinath, 2001
Lyu & Simoncelli, 2009
Laparra et al., 2010
Dinh et al., 2015

Density estimation (parametric transformation)

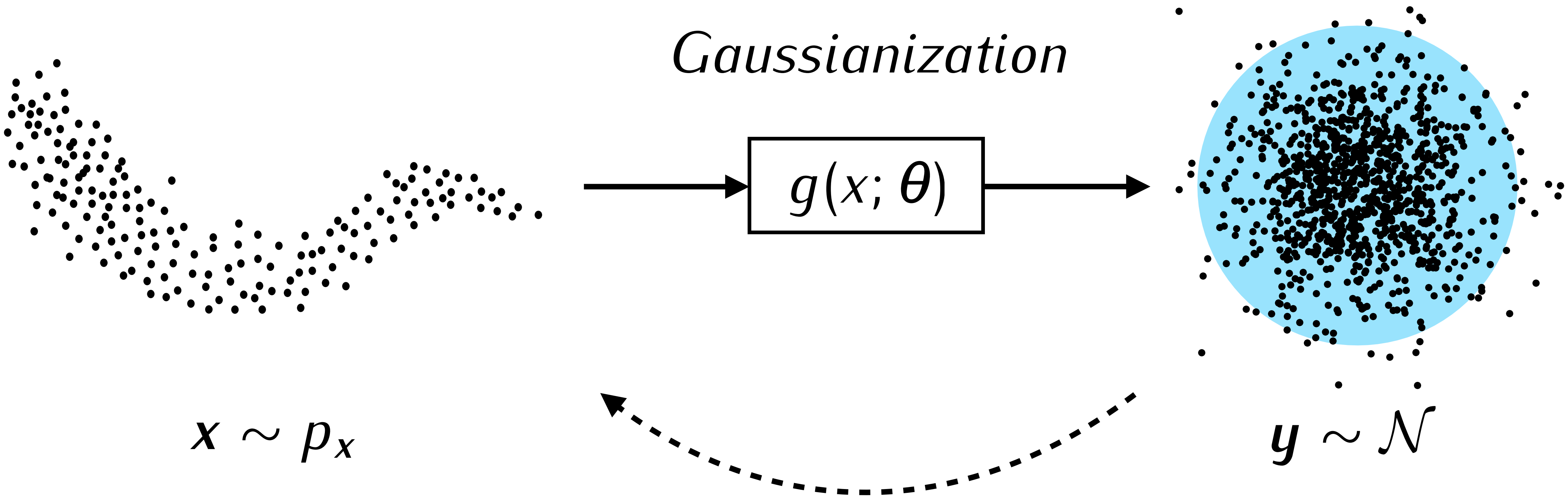


"inferred" density:

$$p_x(\boldsymbol{x}) = \left| \frac{\partial g(\boldsymbol{x}; \boldsymbol{\theta})}{\partial \boldsymbol{x}} \right| \mathcal{N}\big(g(\boldsymbol{x}; \boldsymbol{\theta})\big)$$

Friedman, 1984
Chen & Gopinath, 2001
Lyu & Simoncelli, 2009
Laparra et al., 2010
Dinh et al., 2015

Density estimation (parametric transformation)



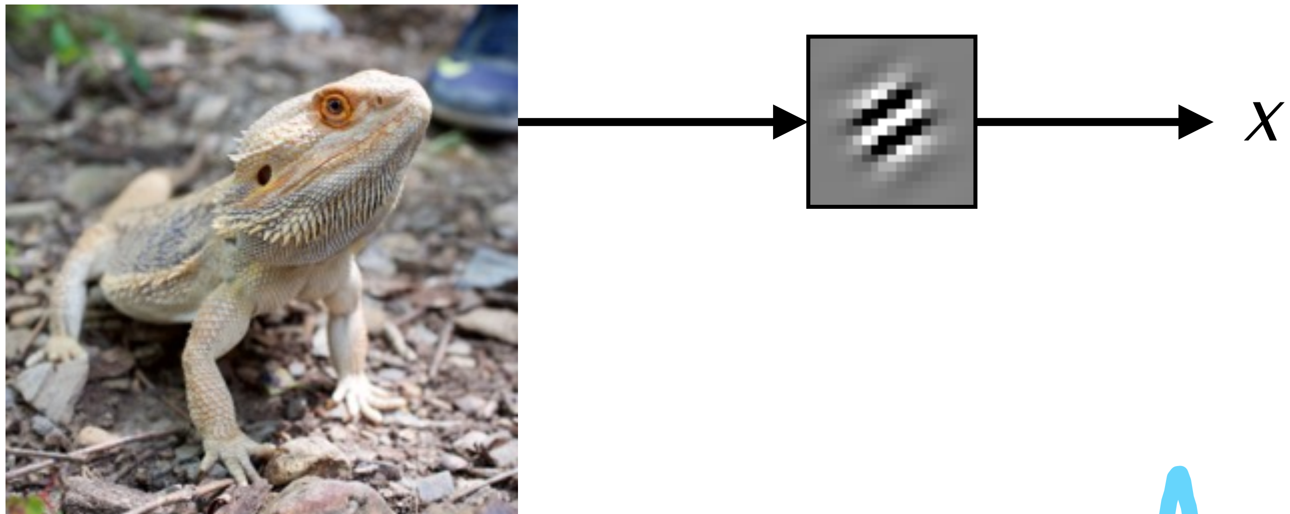
"inferred" density:

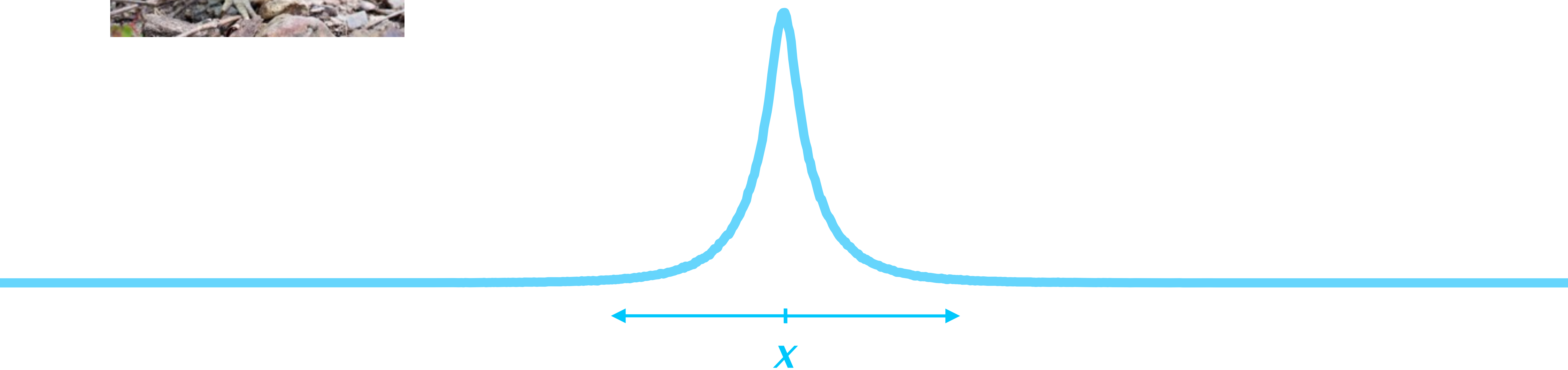
$$p_x(\boldsymbol{x}) = \left| \frac{\partial g(\boldsymbol{x};\boldsymbol{\theta})}{\partial \boldsymbol{x}} \right| \mathcal{N}\big(g(\boldsymbol{x};\boldsymbol{\theta})\big)$$

now need to find optimal transform *g*
no integration required

Friedman, 1984
Chen & Gopinath, 2001
Lyu & Simoncelli, 2009
Laparra et al., 2010
Dinh et al., 2015

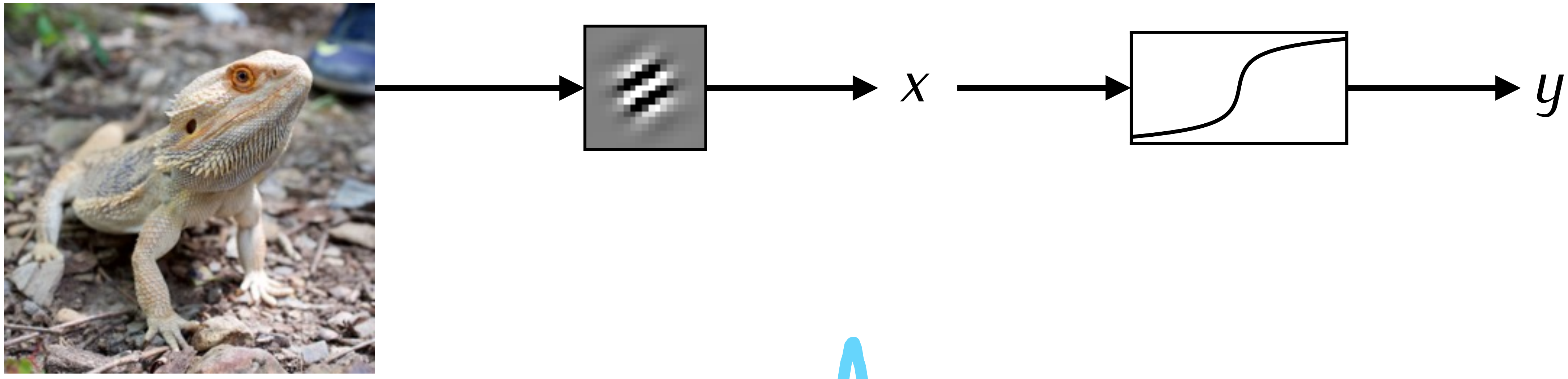
Marginal distribution of linear filter responses

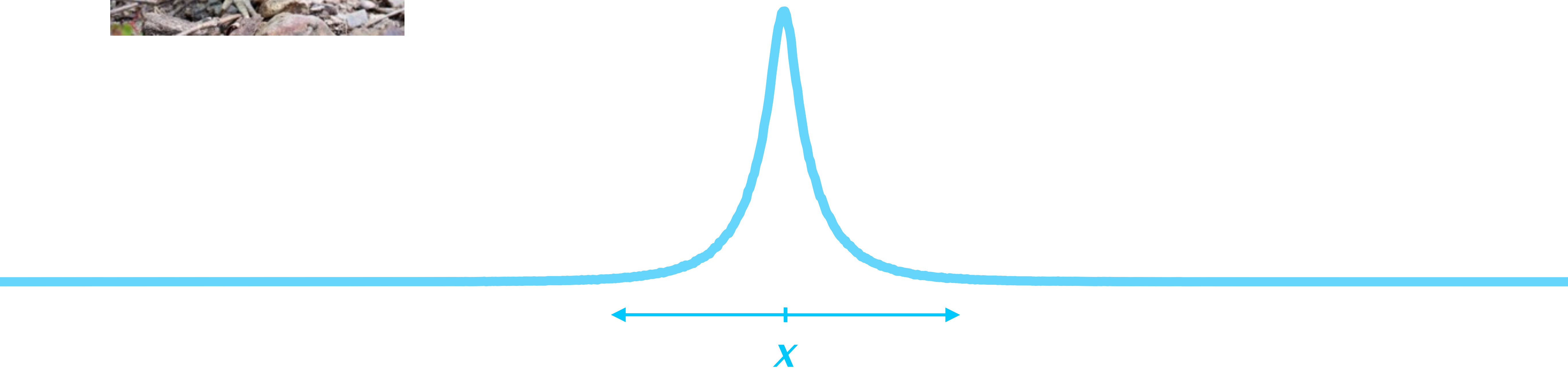




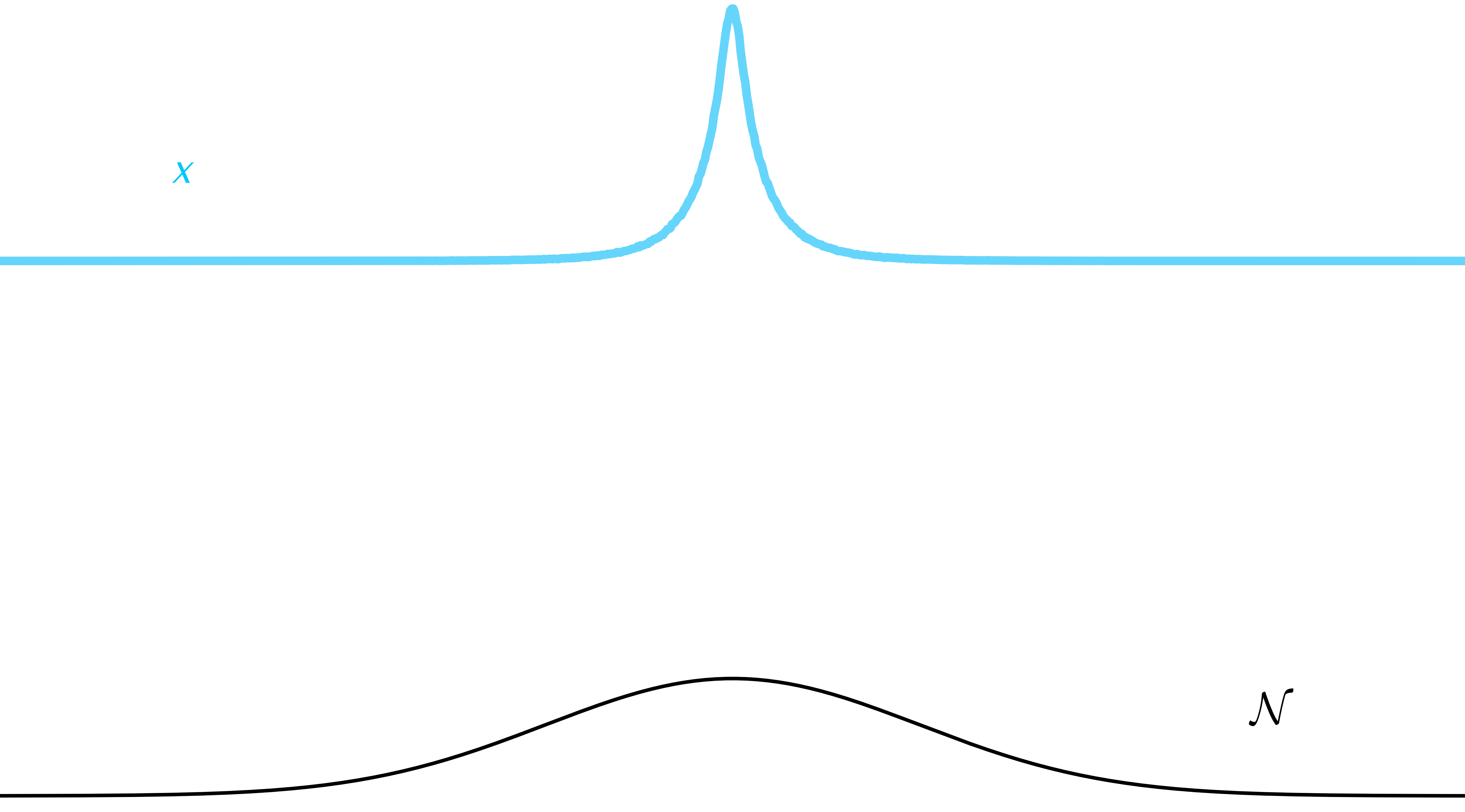
Burt & Adelson, 1981
Field, 1987
Mallat, 1989

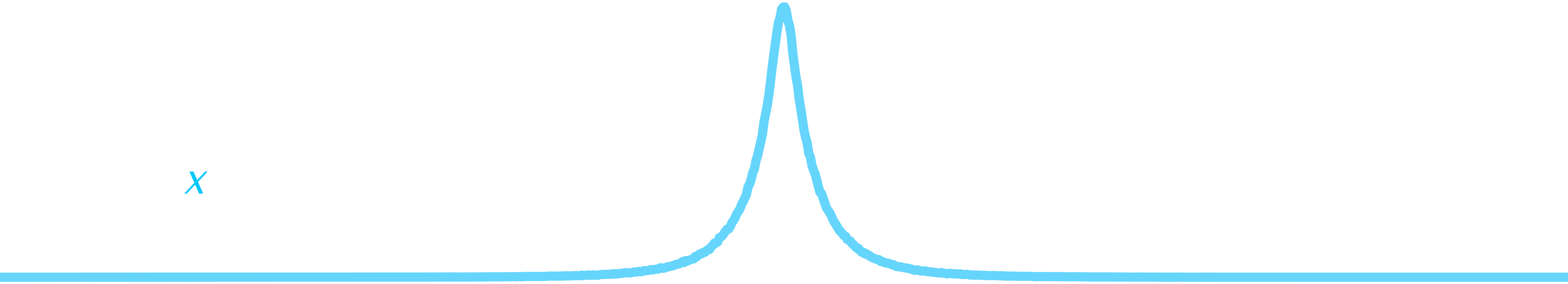
Marginal distribution of linear filter responses

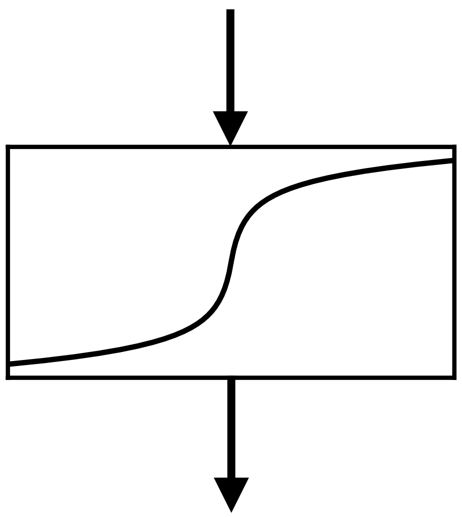




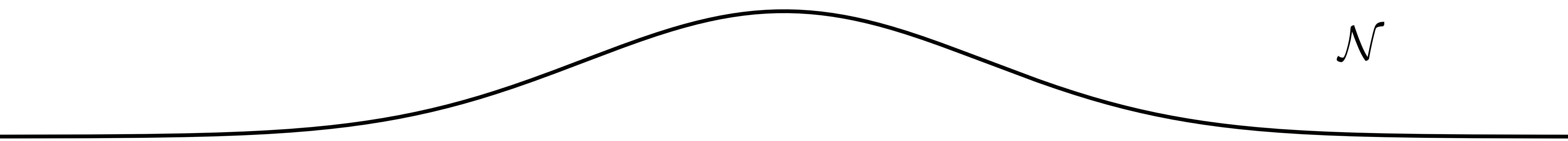
Burt & Adelson, 1981
Field, 1987
Mallat, 1989

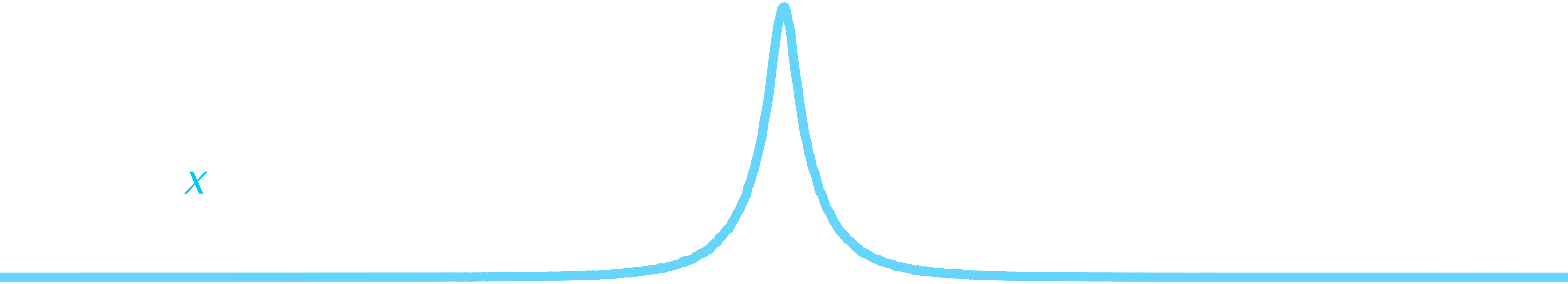

x
$\mathcal{N}$

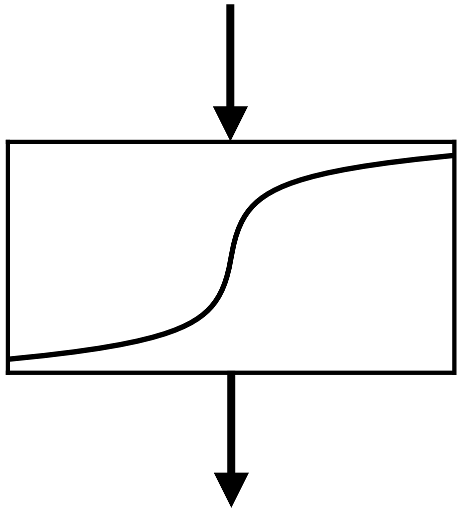




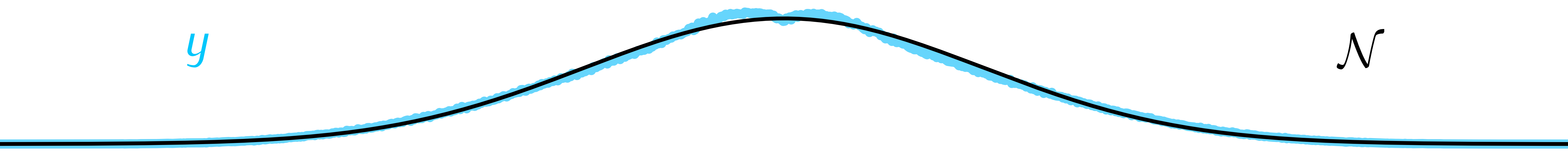
$$y = \frac{x}{\left(\beta + \gamma |x|\right)^{\varepsilon}}$$

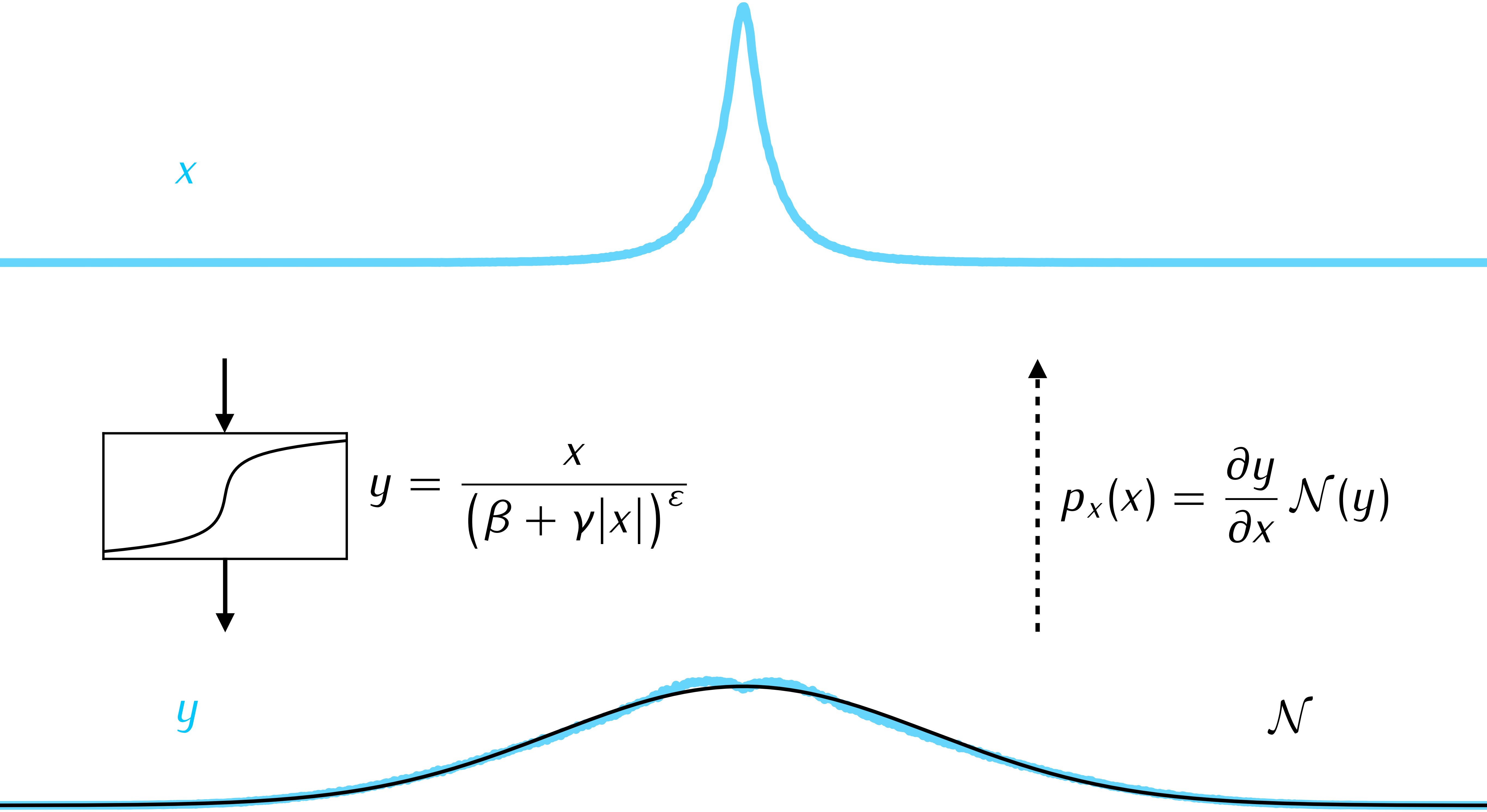


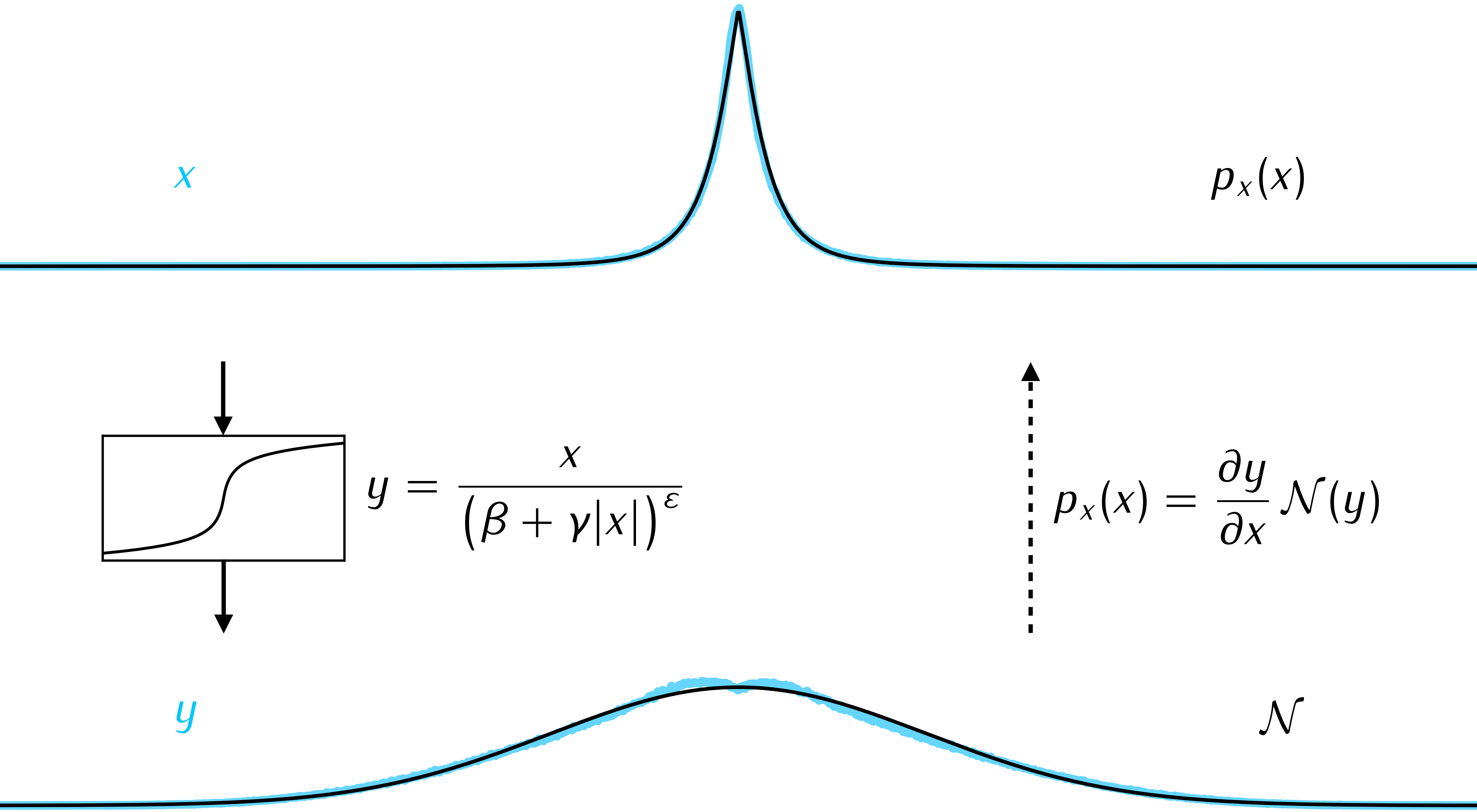




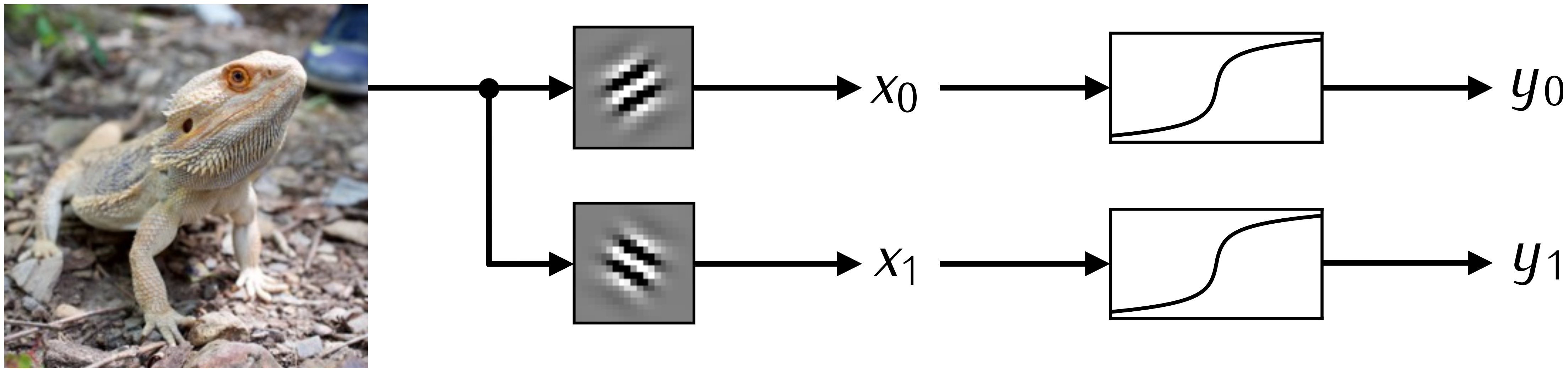
$$y = \frac{x}{\left(\beta + \gamma|x|\right)^{\varepsilon}}$$



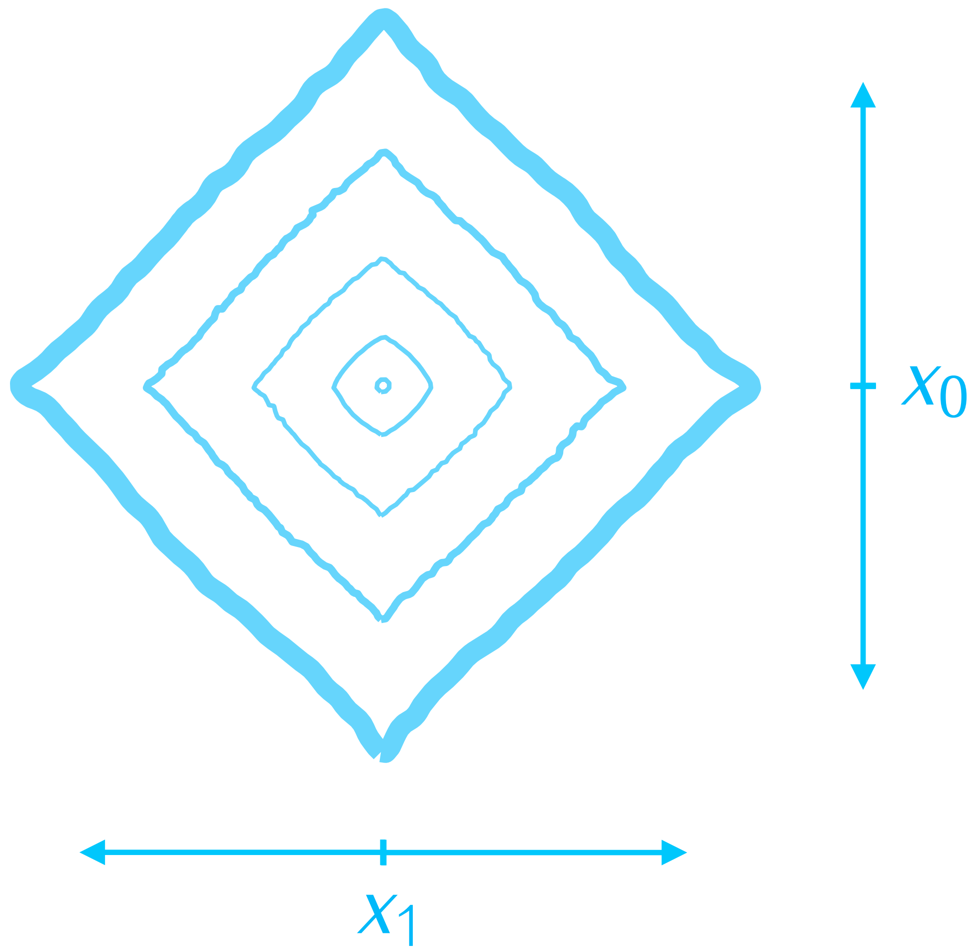

x
$y = \frac{x}{\left(\beta + \gamma|x|\right)^{\varepsilon}}$
$p_x(x) = \frac{\partial y}{\partial x}\mathcal{N}(y)$
y
$\mathcal{N}$

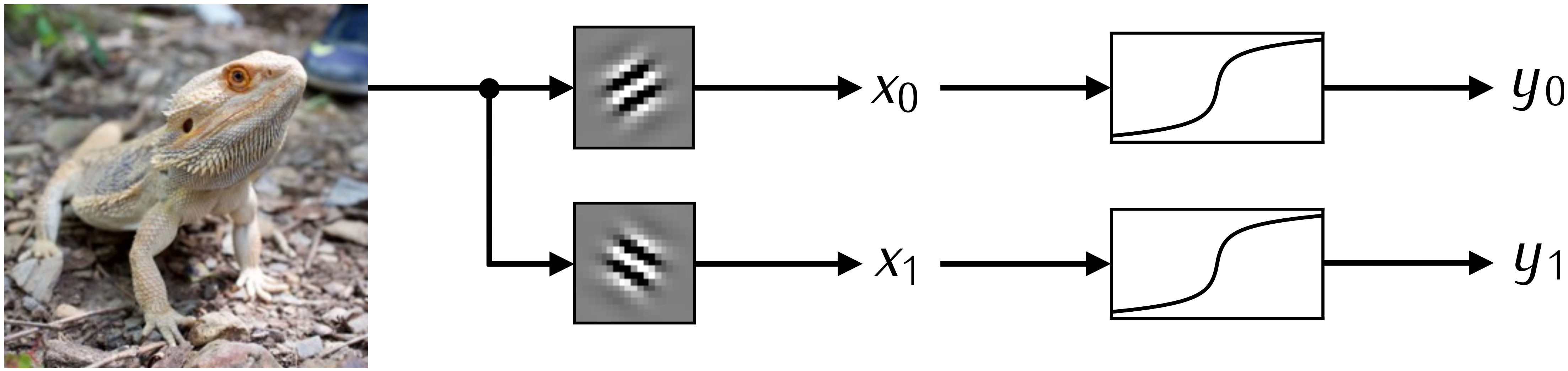

x
$p_x(x)$
$y = \frac{x}{\left(\beta + \gamma|x|\right)^{\varepsilon}}$
$p_x(x) = \frac{\partial y}{\partial x}\mathcal{N}(y)$
y
$\mathcal{N}$

Joint distribution of linear filter responses



contour lines
of joint density

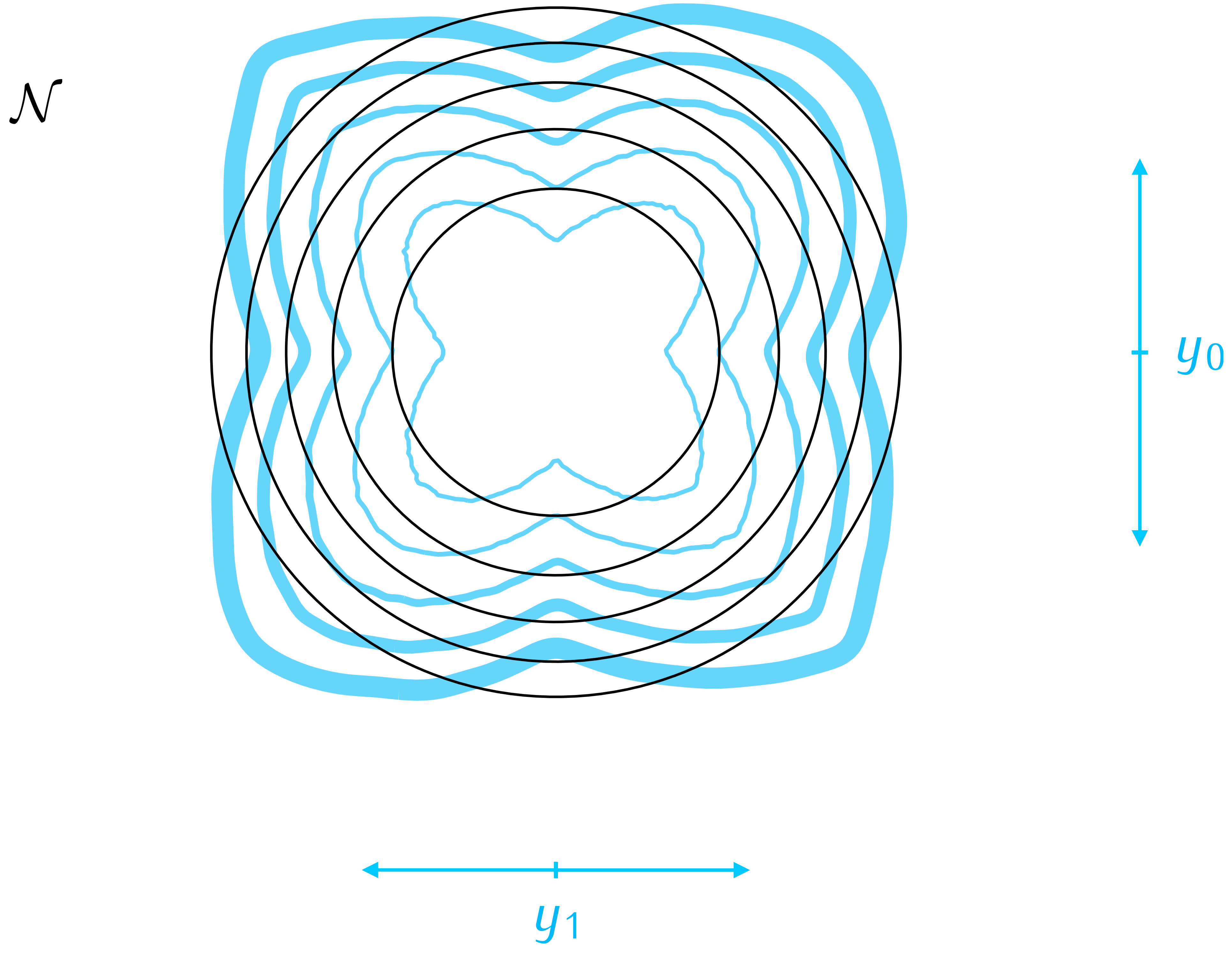




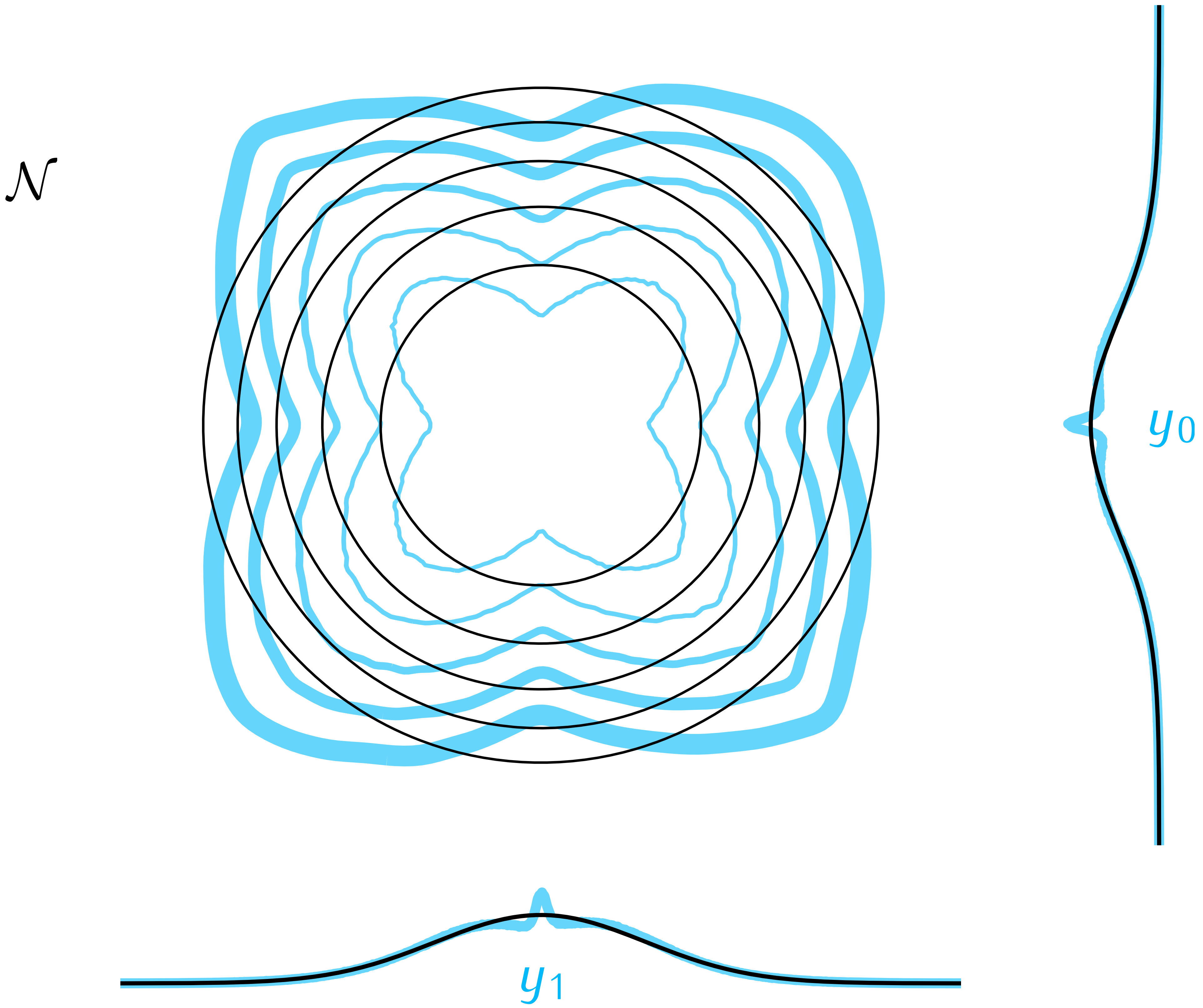
$$y_0 = \frac{x_0}{\left(\beta_0 + \gamma_0 |x_0|^{\alpha_0}\right)^{\varepsilon_0}}$$

$$y_1 = \frac{x_1}{\left(\beta_1 + \gamma_1 |x_1|^{\alpha_1}\right)^{\varepsilon_1}}$$

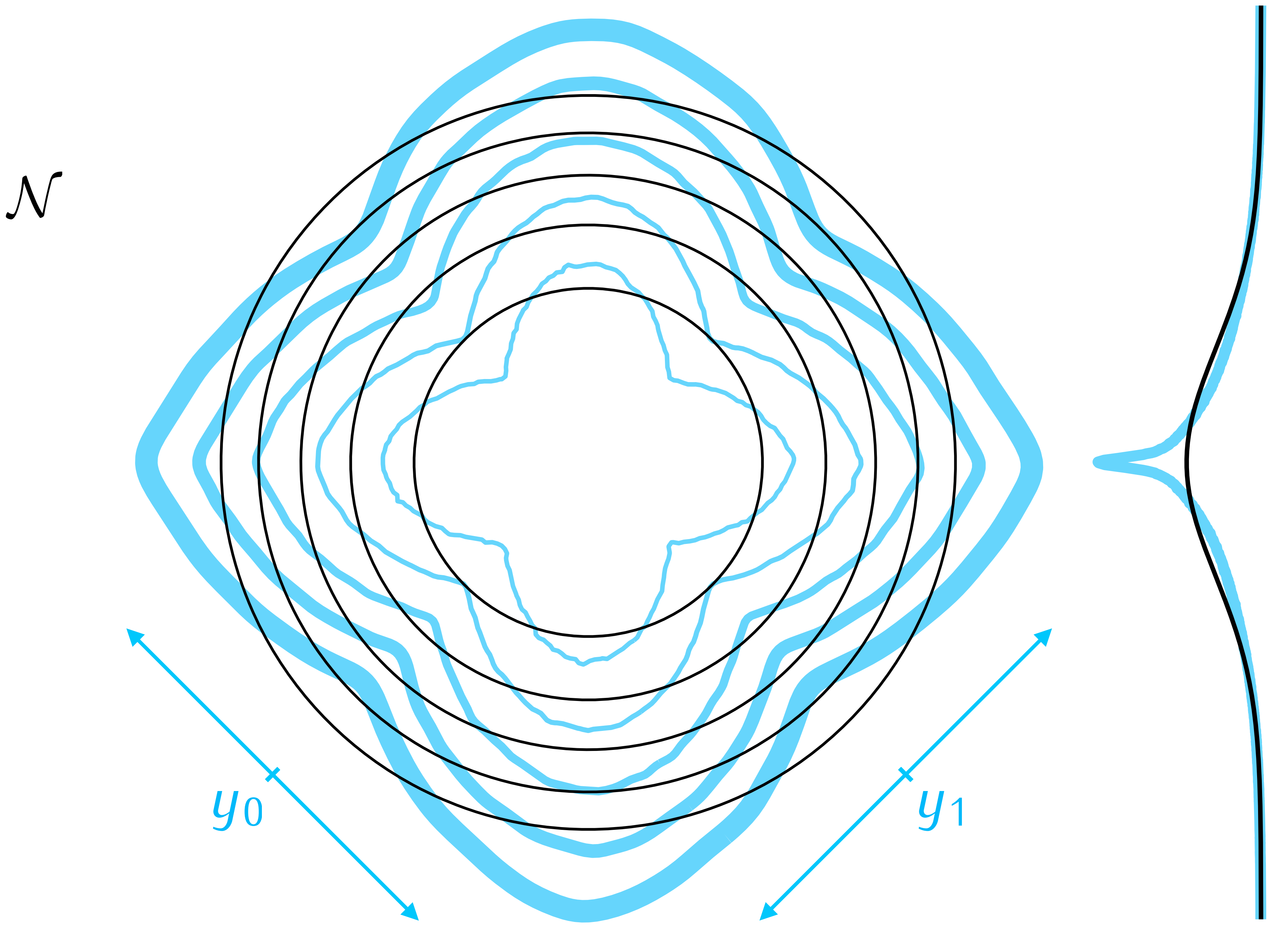
Contour lines, Gaussianized responses

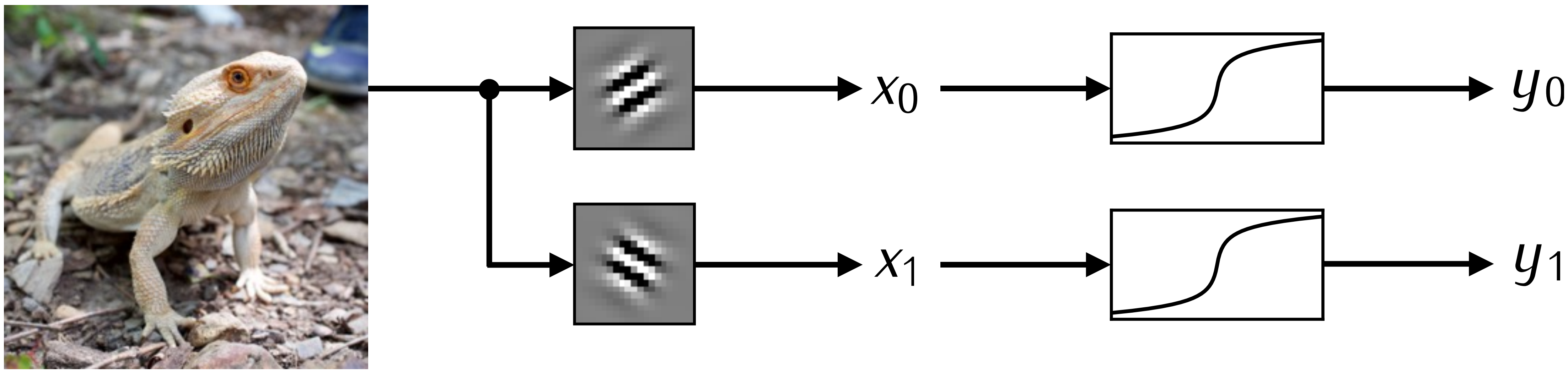


Contour lines, Gaussianized responses



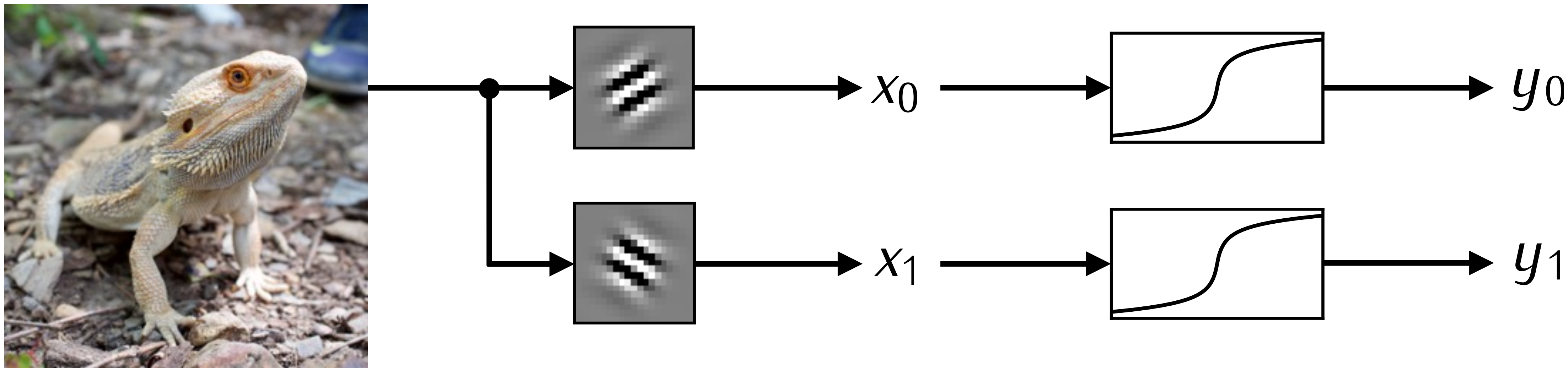
Contour lines, Gaussianized responses





$$y_0 = \frac{x_0}{\left(\beta_0 + \gamma_0 |x_0|^{\alpha_0}\right)^{\varepsilon_0}}$$

$$y_1 = \frac{x_1}{\left(\beta_1 + \gamma_1 |x_1|^{\alpha_1}\right)^{\varepsilon_1}}$$

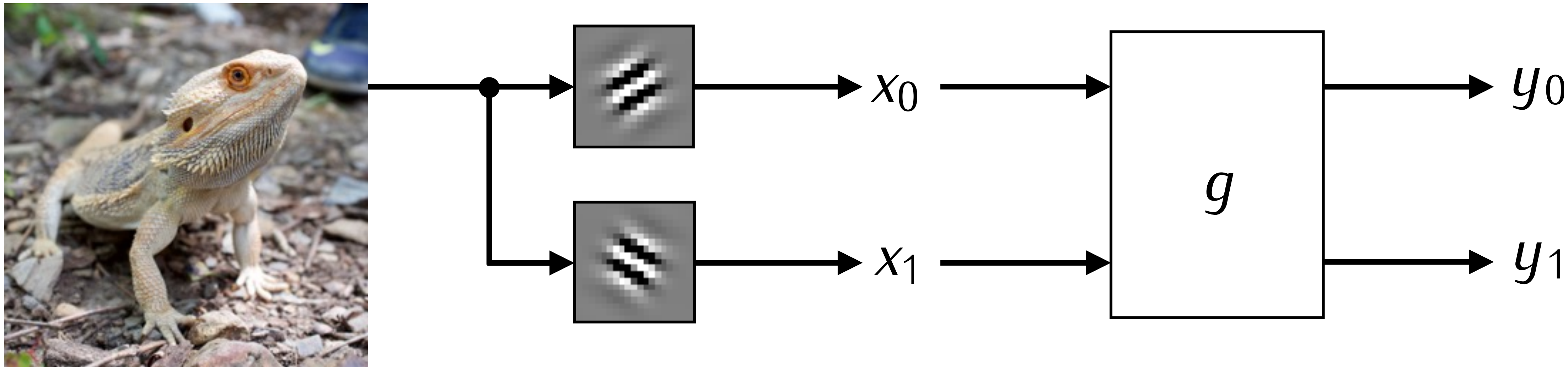


$$y_0 = \frac{x_0}{\left(\beta_0 + \gamma_0 |x_0|^{\alpha_0}\right)^{\varepsilon_0}}$$

$$\uparrow$$

$$y_1 = \frac{x_1}{\left(\beta_1 + \gamma_1 |x_1|^{\alpha_1}\right)^{\varepsilon_1}}$$

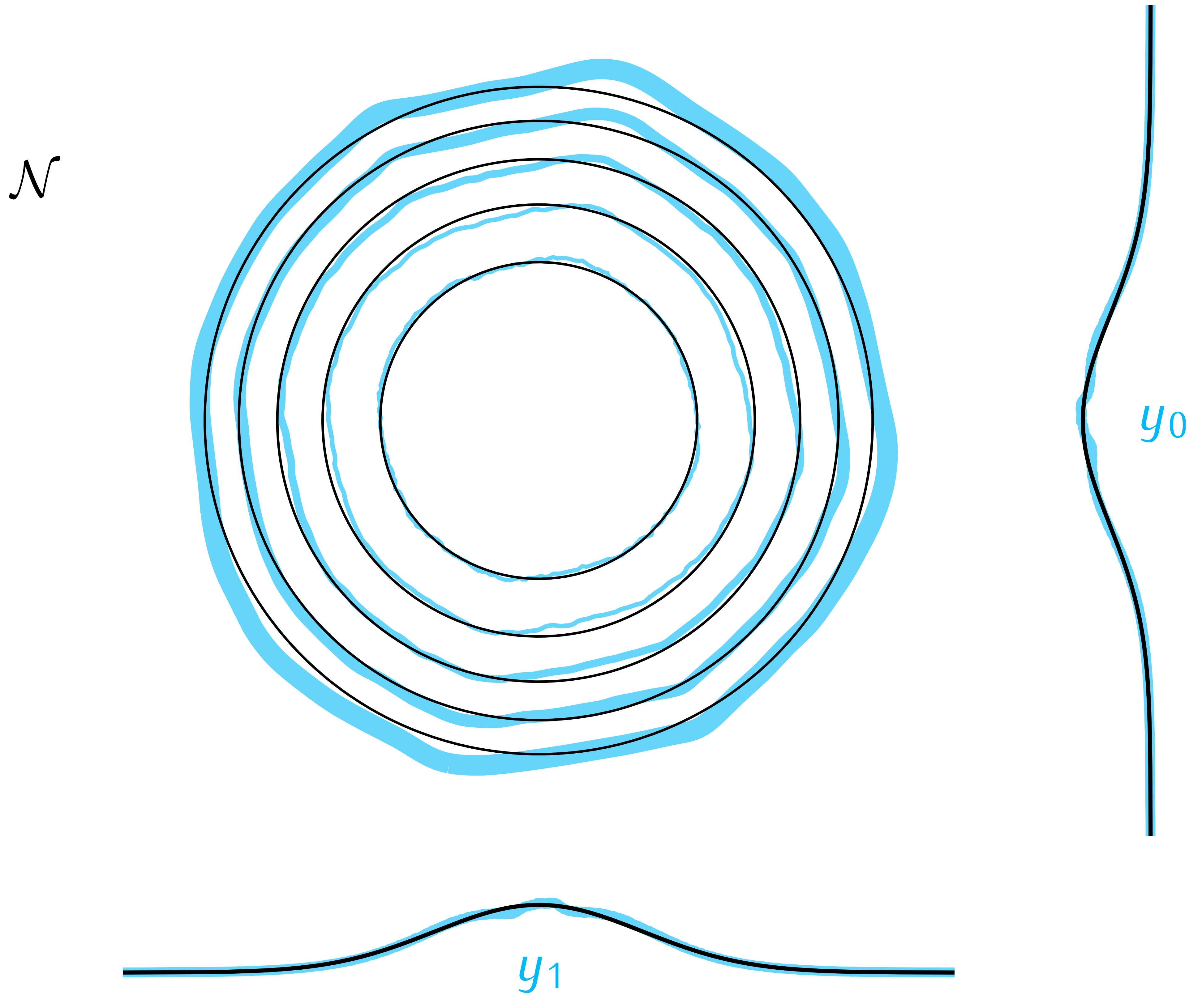
$$\uparrow$$



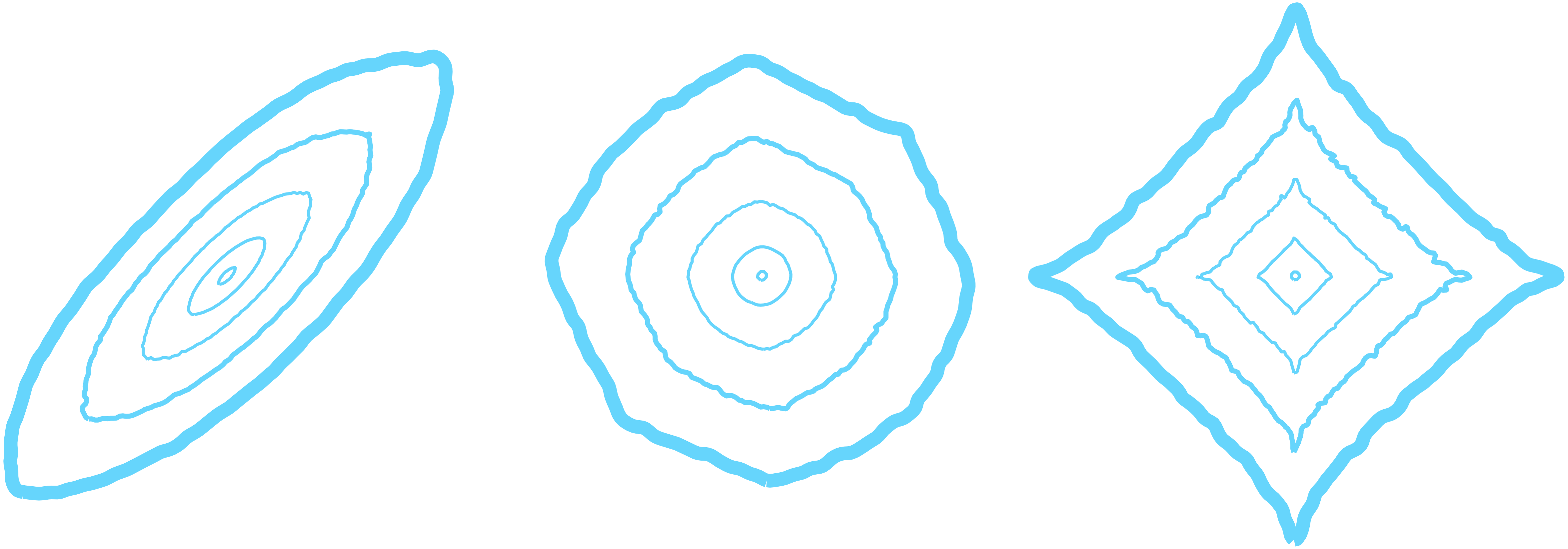
$$y_0 = \frac{x_0}{\left(\beta_0 + \gamma_{01}|x_1|^{\alpha_{01}} + \gamma_{00}|x_0|^{\alpha_{00}}\right)^{\varepsilon_0}}$$

$$y_1 = \frac{x_1}{\left(\beta_1 + \gamma_{10}|x_0|^{\alpha_{10}} + \gamma_{11}|x_1|^{\alpha_{11}}\right)^{\varepsilon_1}}$$

Contour lines, Gaussianized responses



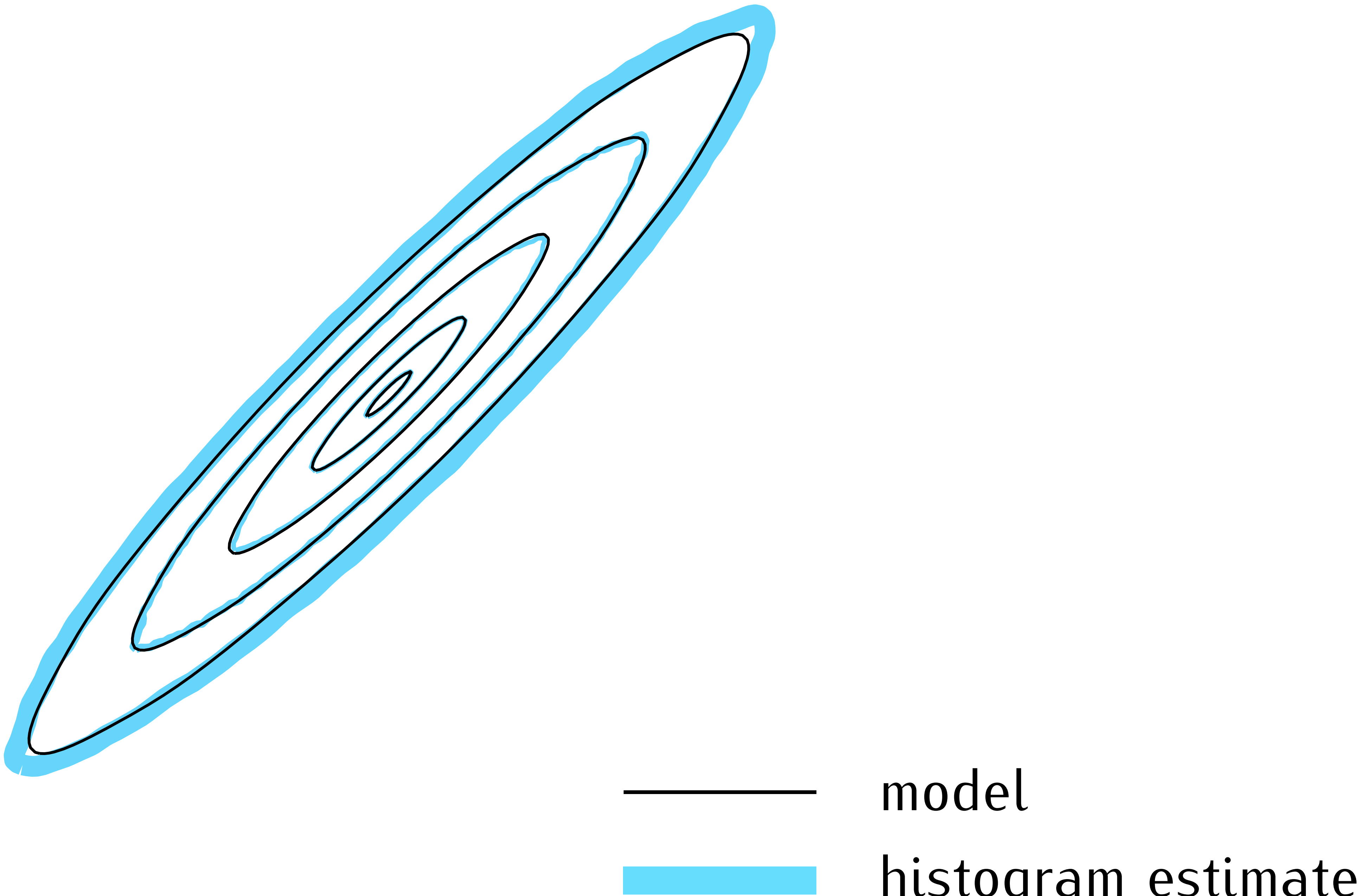
Variety of shapes, joint density of filter responses



elliptical | ? | marginally independent

Lyu & Simoncelli, 2009
Sinz et al., 2009

Contour lines, linear filter responses



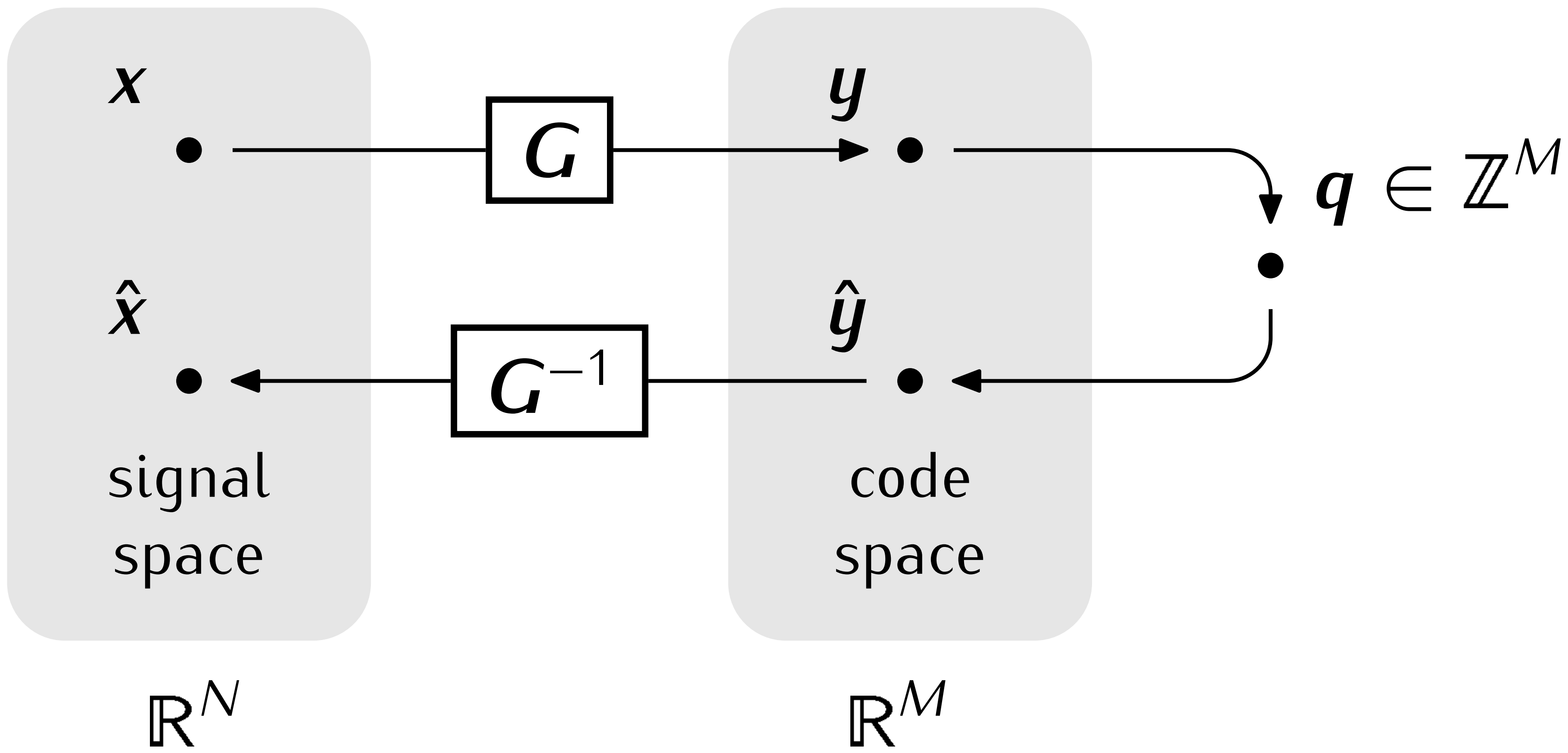
Special cases/related models

- Independent Component Analysis, Cardoso, 2003
- Independent Subspace Analysis, Hyvärinen & Hoyer, 2000
- Weighted normalization model, Schwartz & Simoncelli, 2001
- Topographic ICA, Hyvärinen et al., 2001
- Radial Gaussianization, Lyu & Simoncelli, 2009
- L_p-nested symmetric distributions, Sinz & Bethge, 2010
- "Two-layer model", Köster & Hyvärinen, 2010

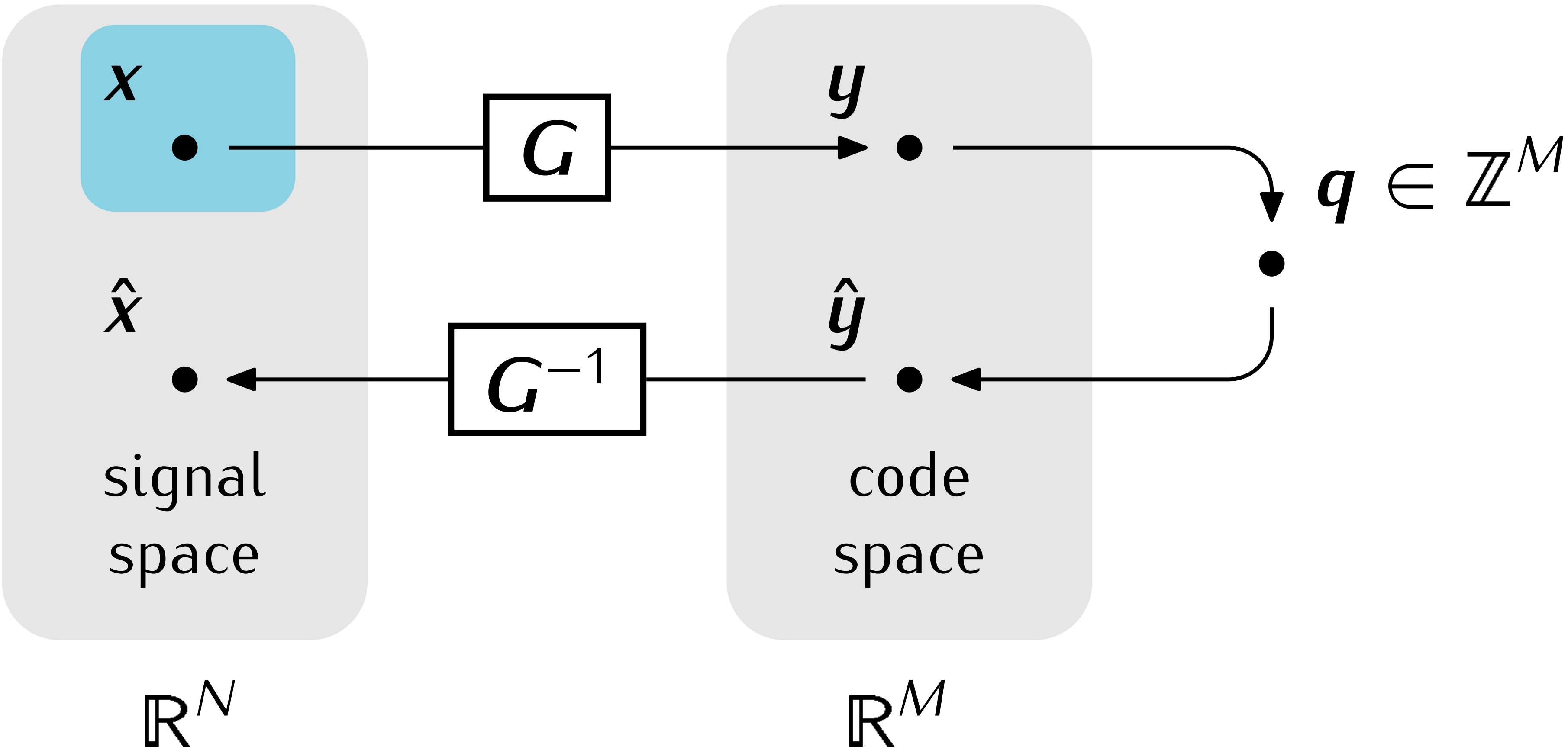
Applications:

perceptual metrics
perceptually optimized rendering
density estimation
image compression
denoising
...

Linear transform coding

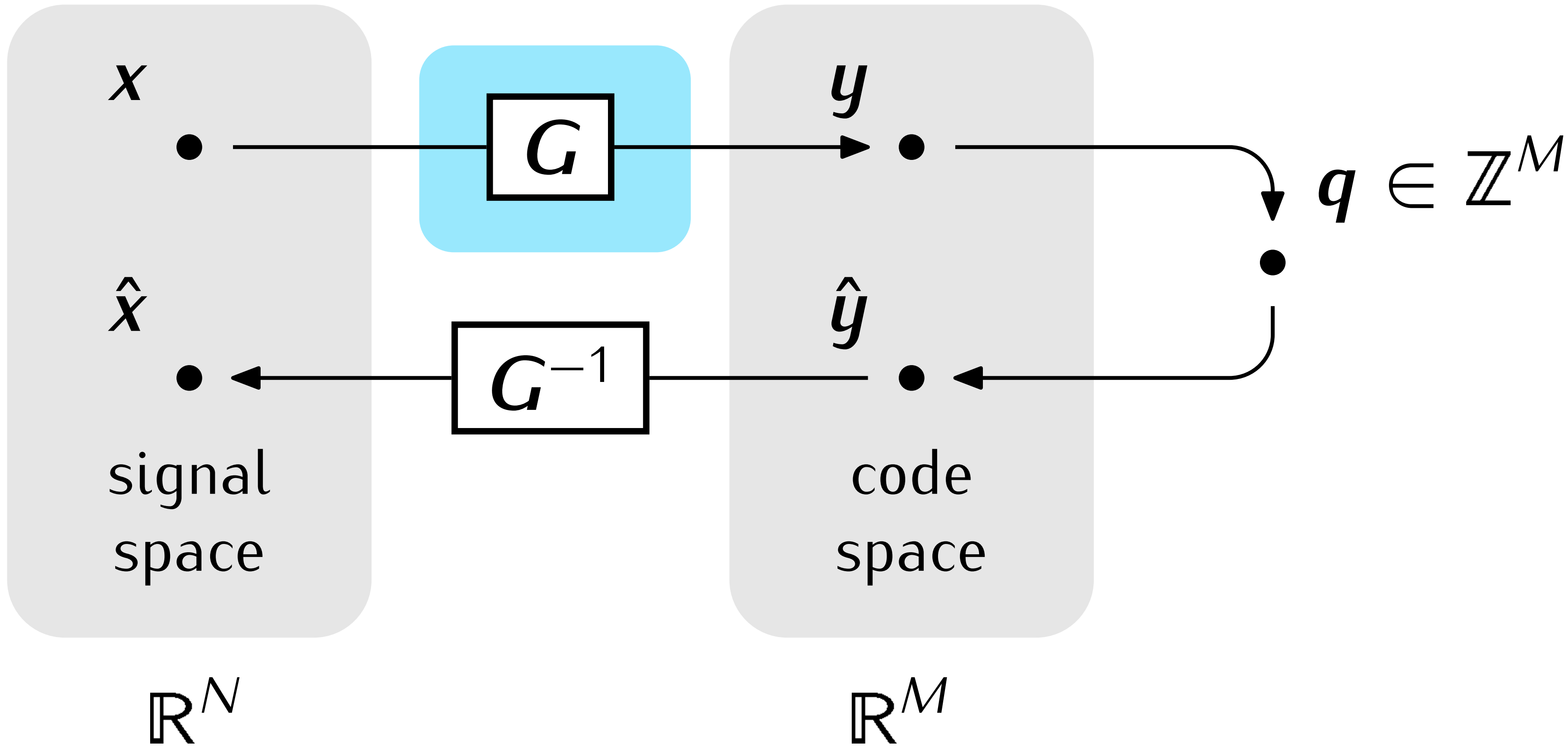


Linear transform coding

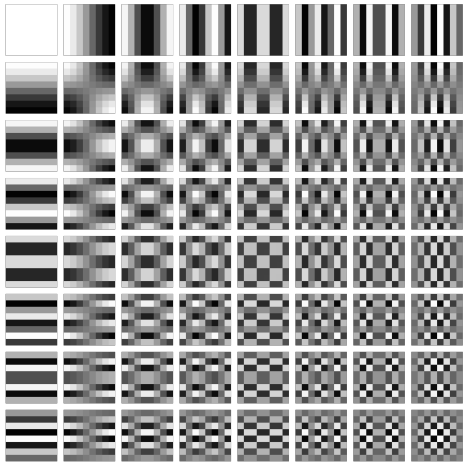


$\boldsymbol{x}$

Linear transform coding

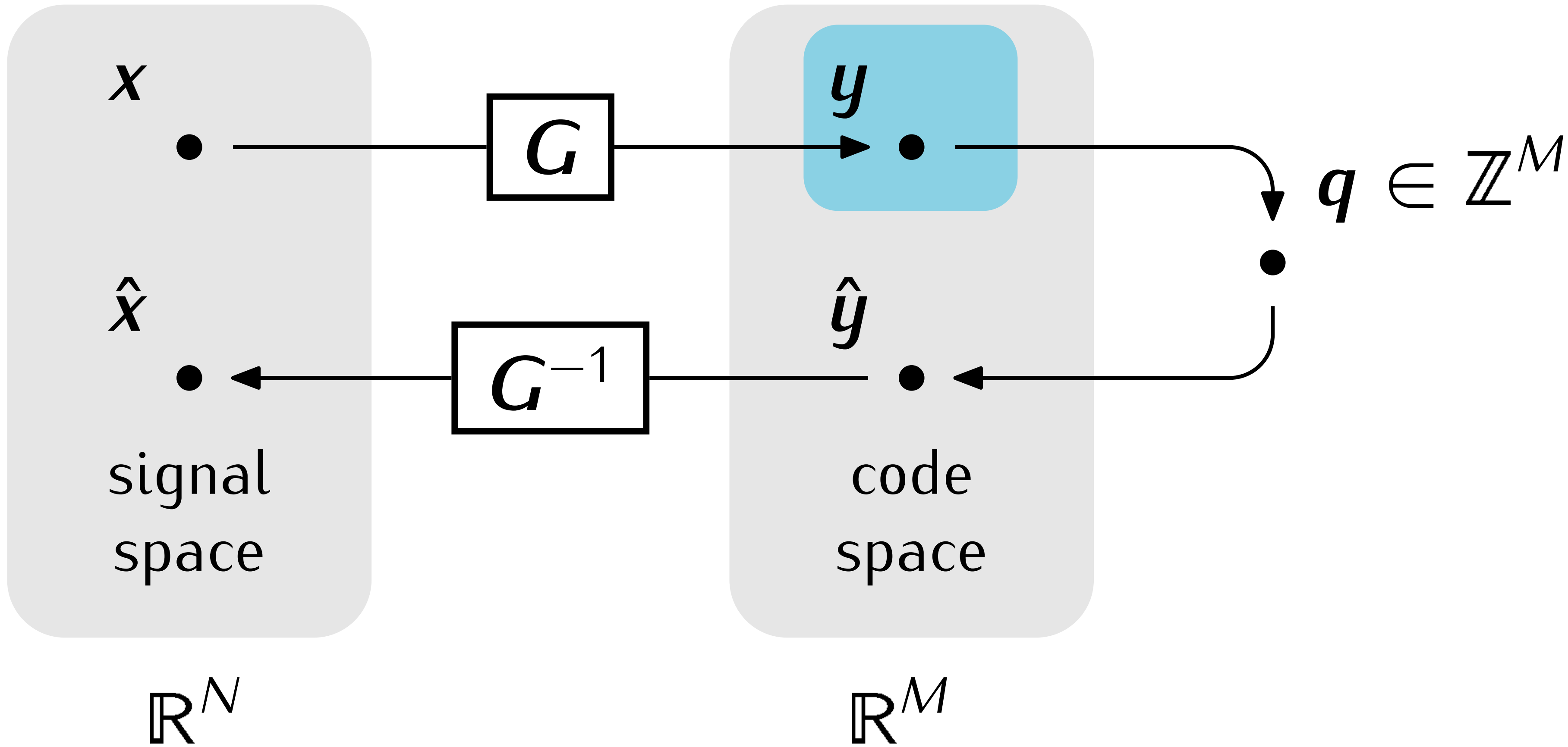


x

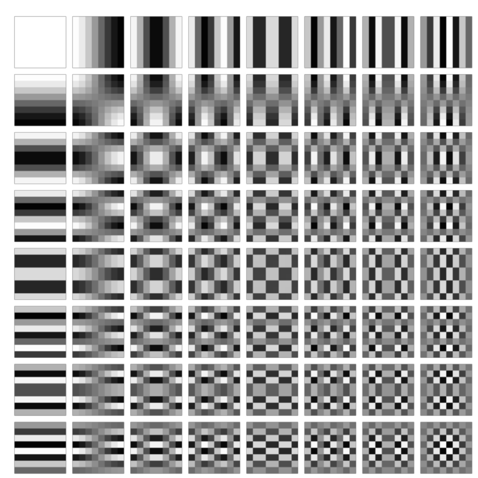


G

Linear transform coding

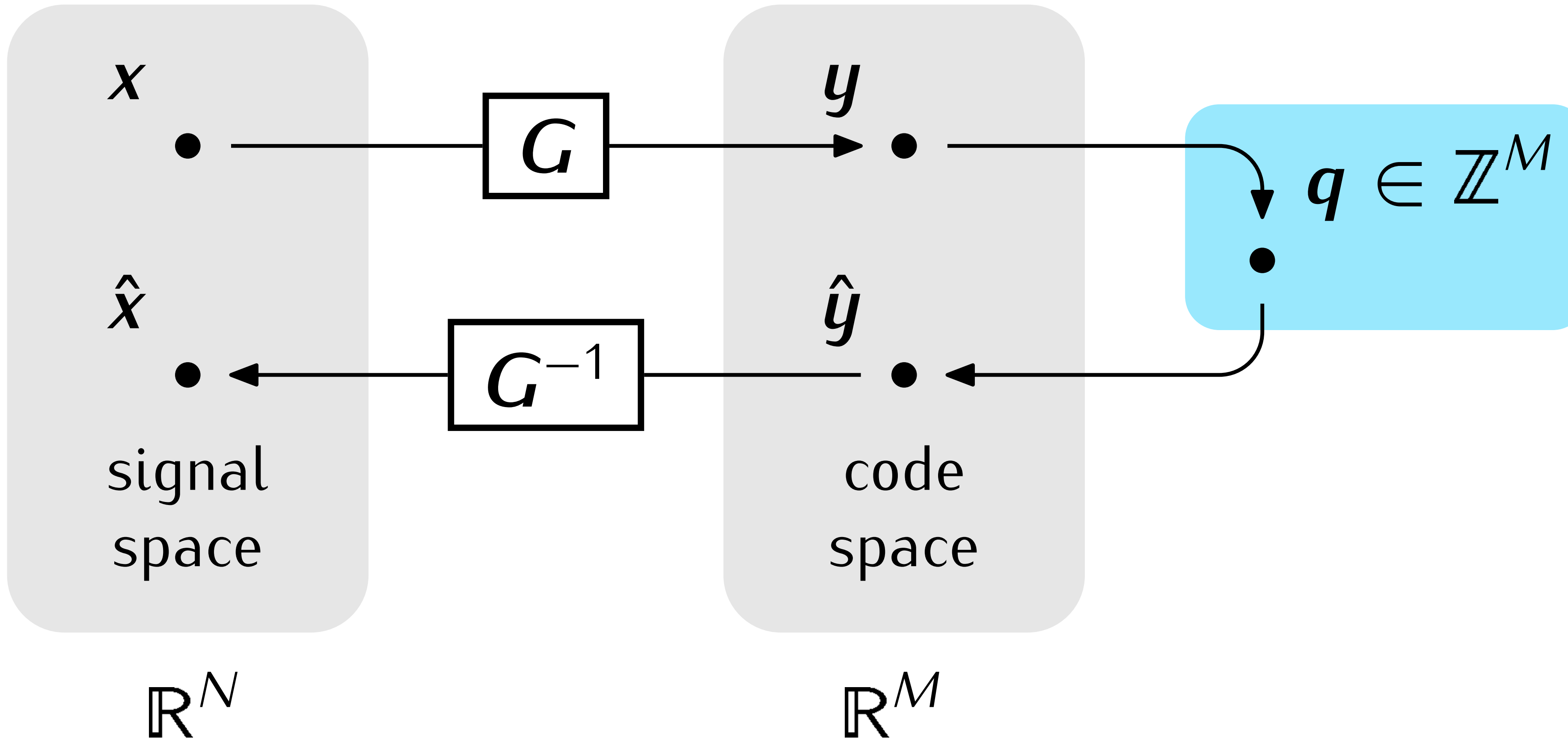


x

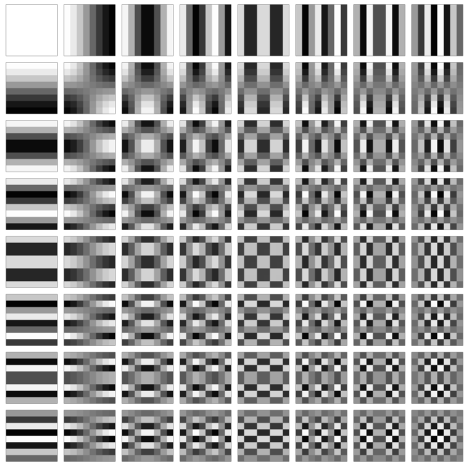


G

Linear transform coding

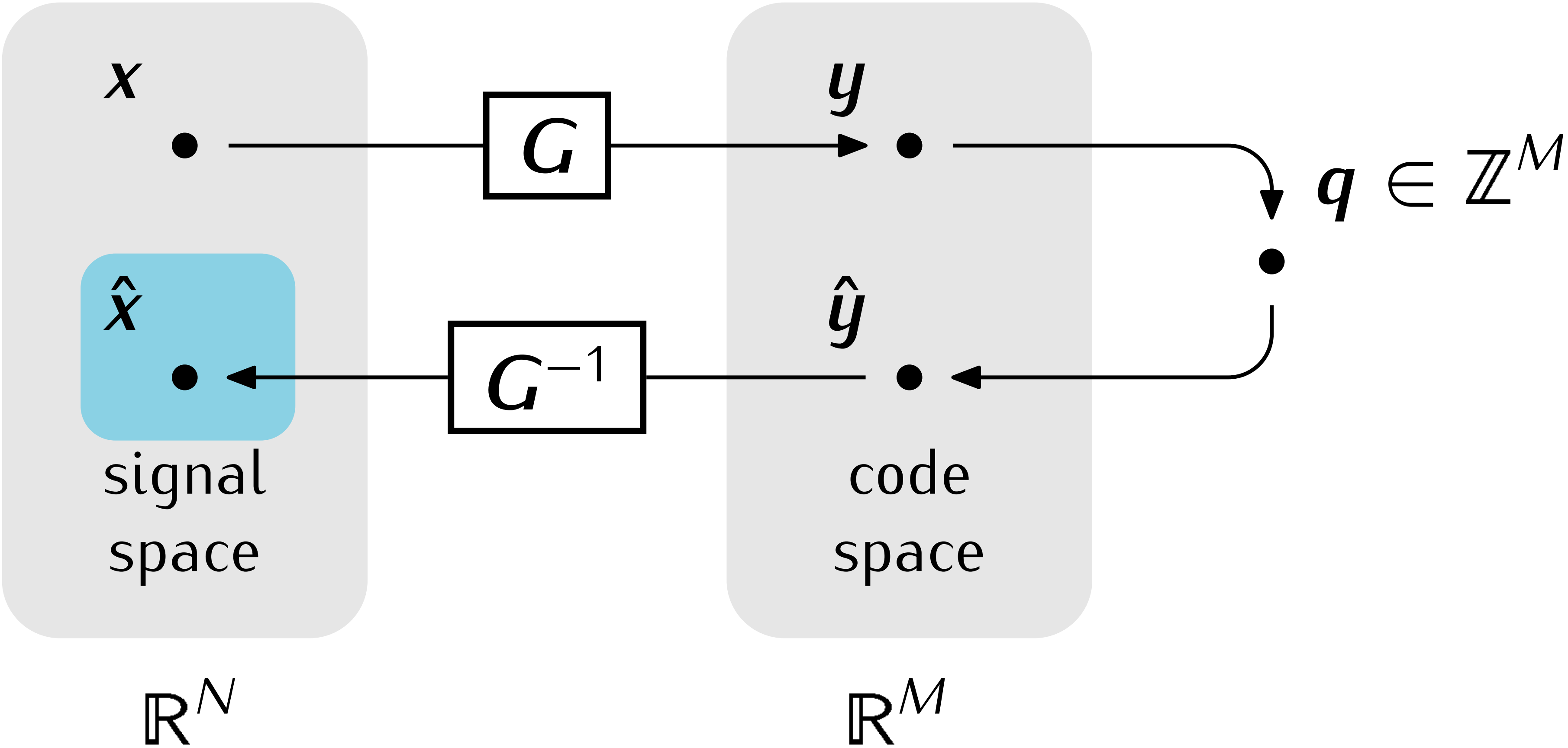


$\boldsymbol{x}$

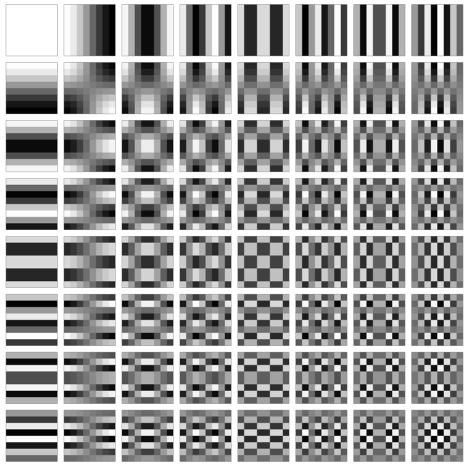


$\boldsymbol{G}$

Linear transform coding



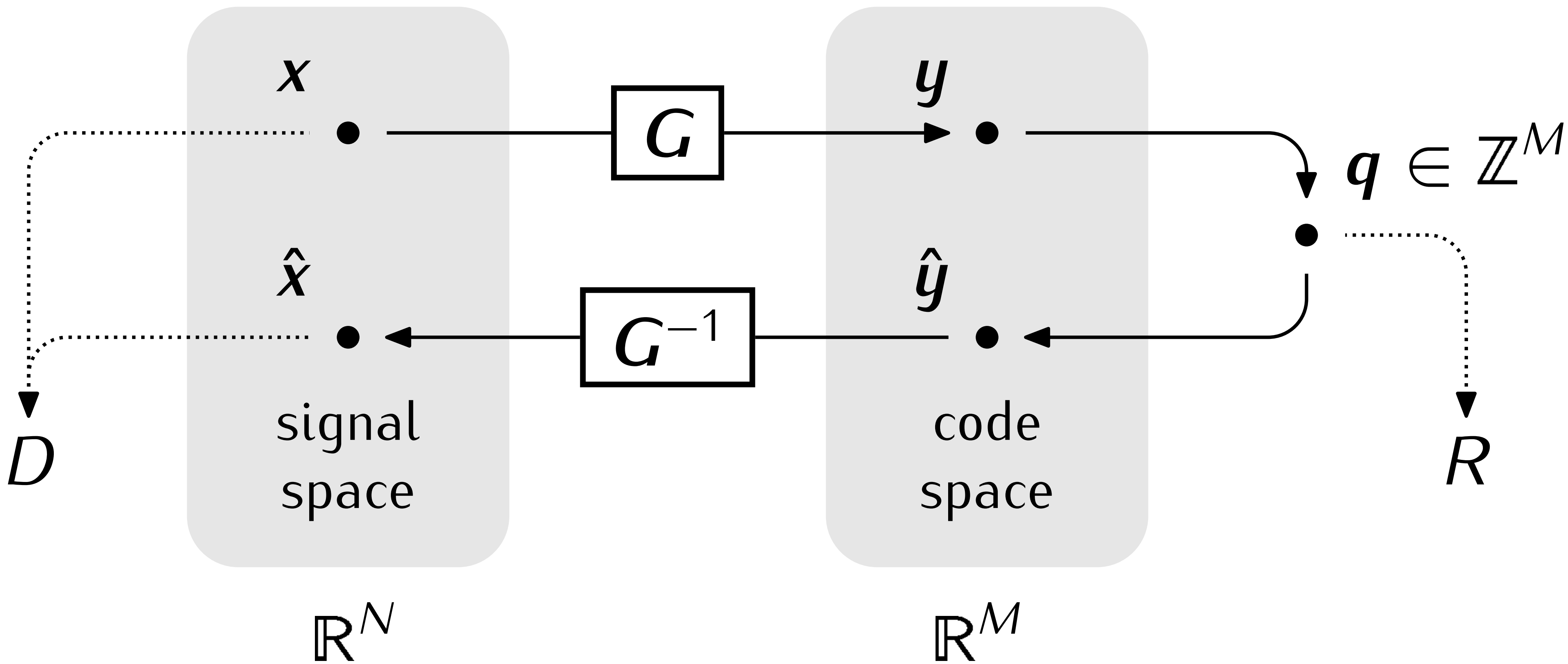
$\boldsymbol{x}$



G

$\hat{\boldsymbol{x}}$

Linear transform coding



D: distortion, e.g. mean squared error
R: rate, ideally close to Shannon entropy of $\boldsymbol{q}$

rate: 0.17 bits/pixel

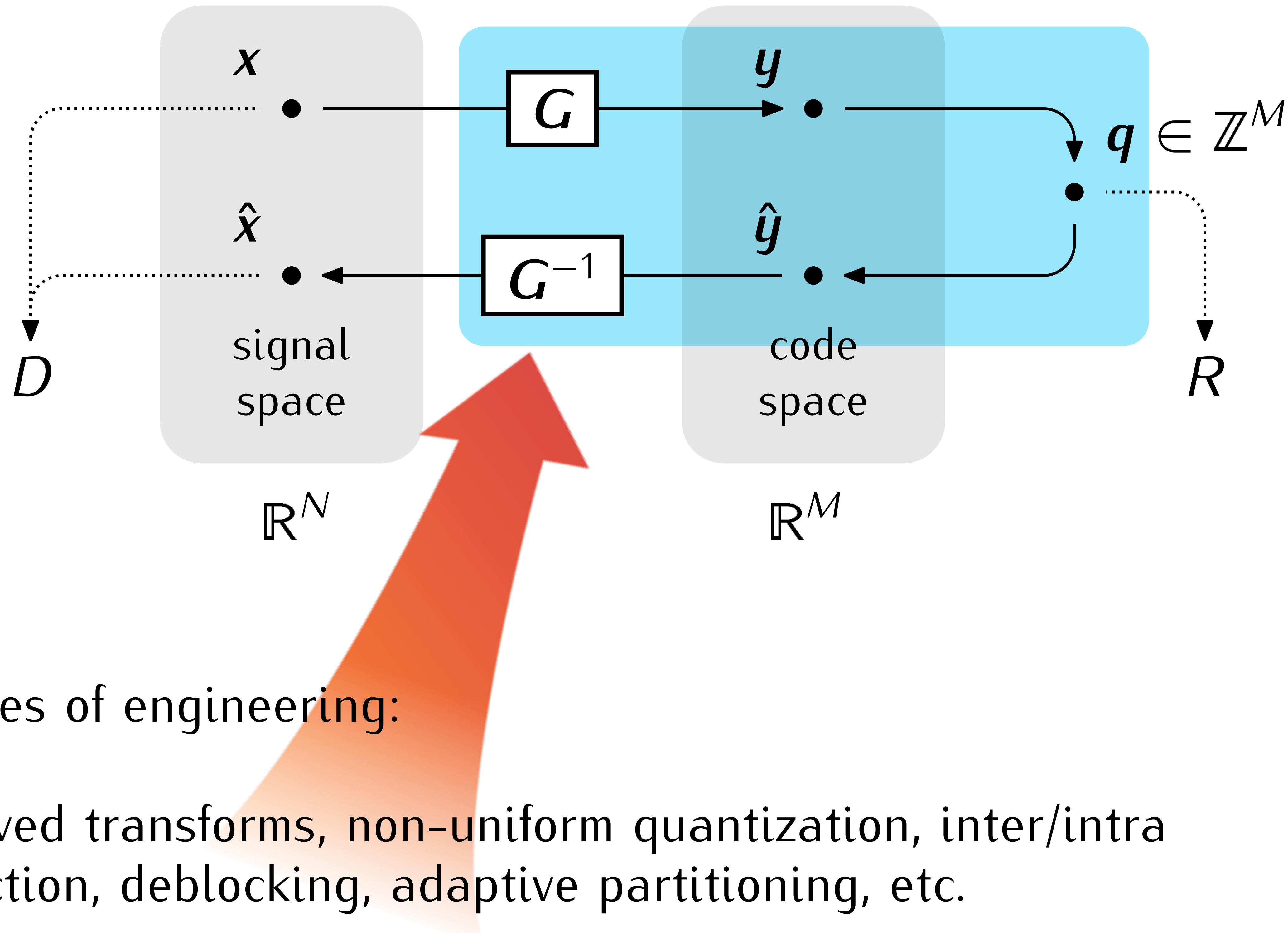
rate: 0.12 bits/pixel

coarser quantization: lower rate, higher distortion

rate: 0.32 bits/pixel

finer quantization: higher rate, lower distortion

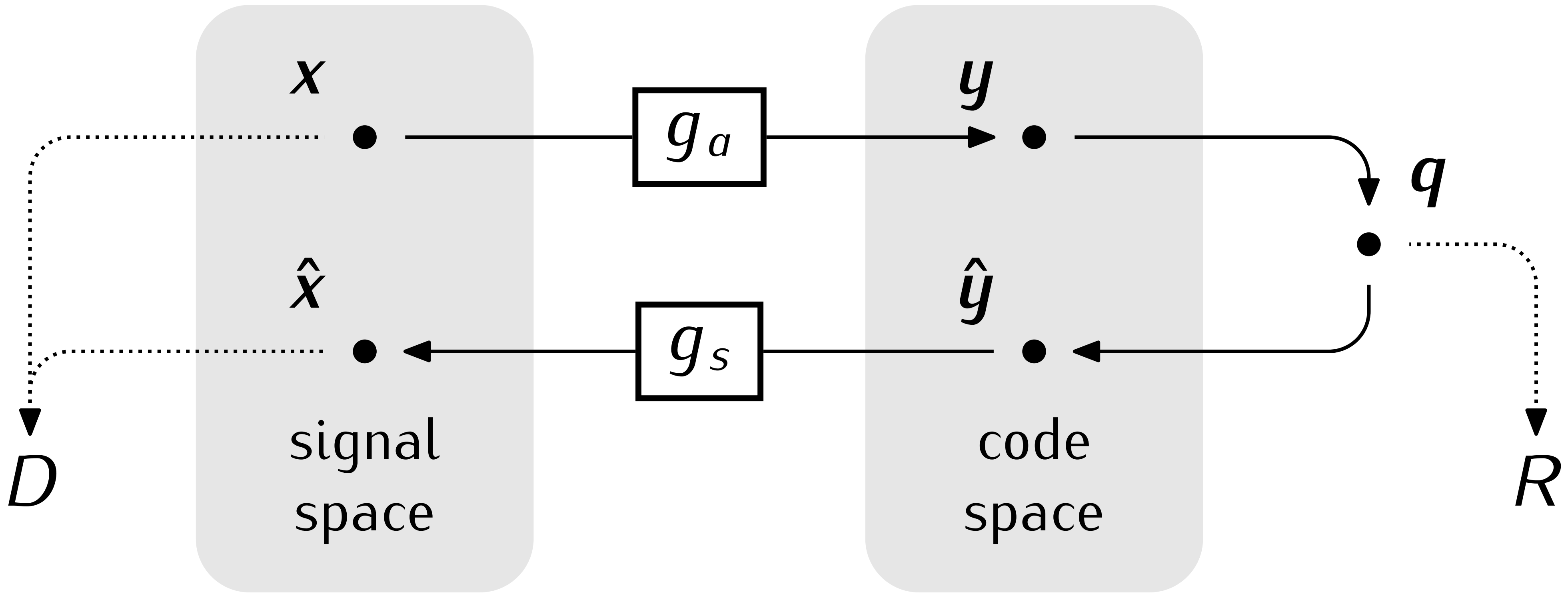
Linear transform coding



decades of engineering:

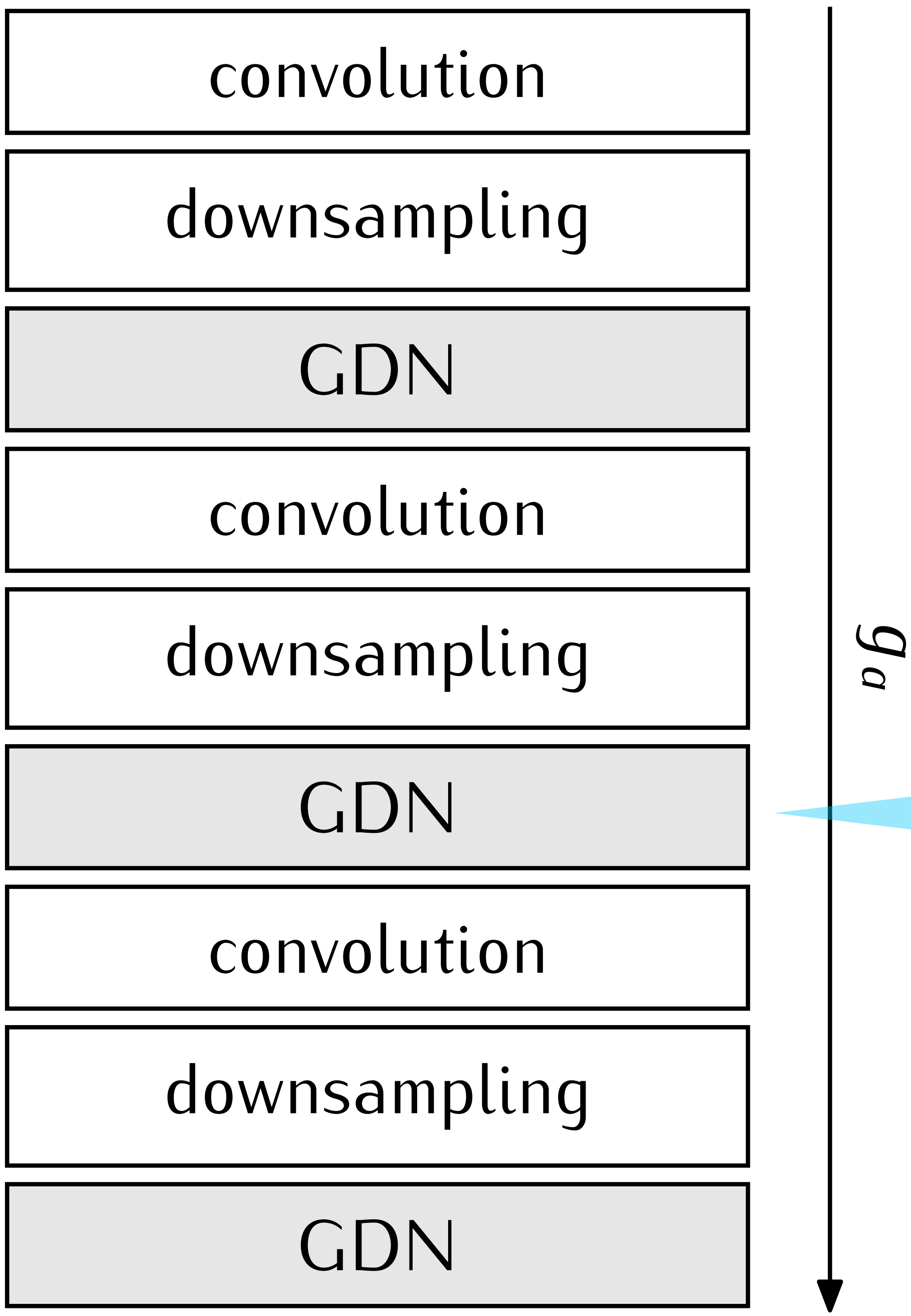
improved transforms, non-uniform quantization, inter/intra prediction, deblocking, adaptive partitioning, etc.

Nonlinear transform coding



g_a, g_s: multivariate, parametric nonlinear functions
(if it helps, think of them as neural networks)

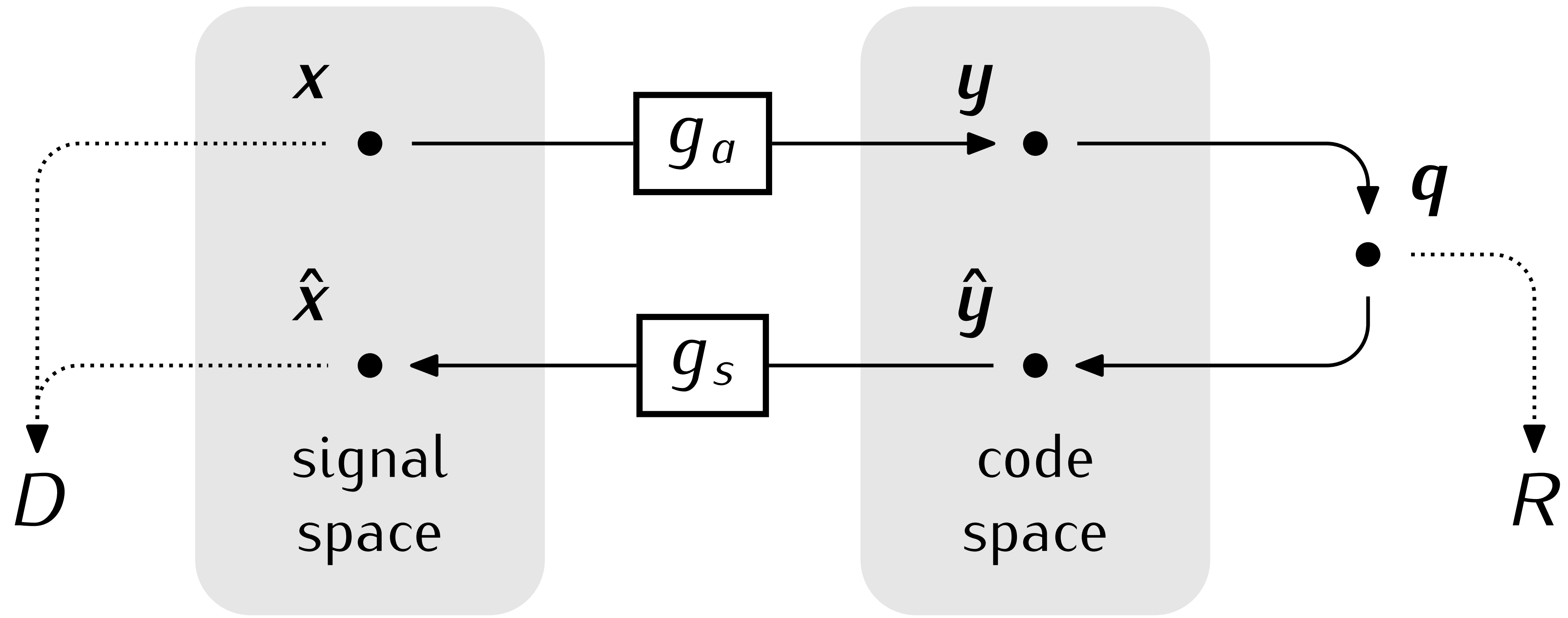
Architecture of transformation



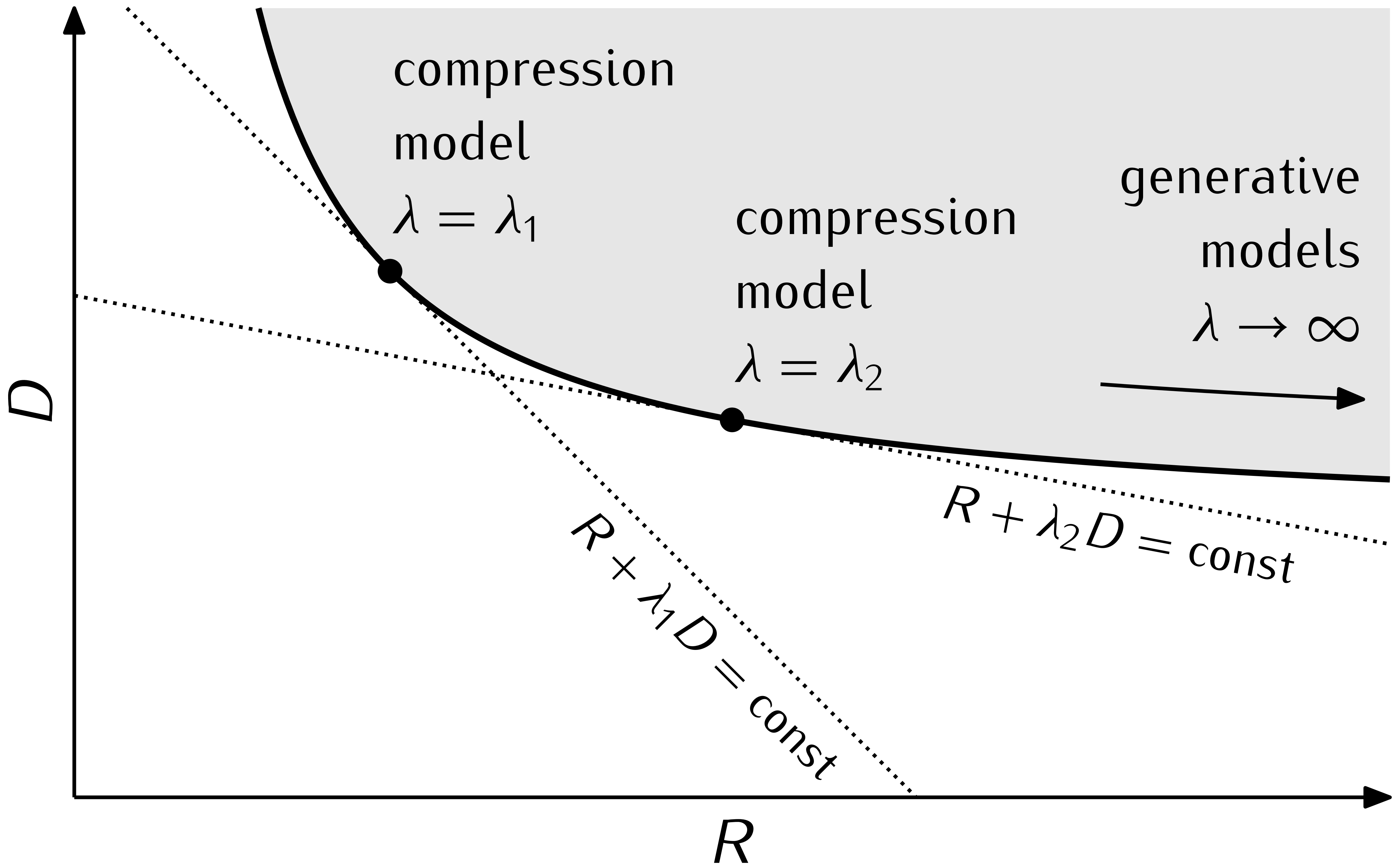
generalized divisive normalization

$$u_i = \frac{w_i}{\sqrt{\beta_i + \sum_j \gamma_{ij} |w_j|^2}}$$

Optimize to find transform parameters for various lambdas



$$L = \mathbb{E}\left[-\sum_i \log_2 P_{q_i}(q_i) + \lambda \left\|\hat{\boldsymbol{x}} - \boldsymbol{x}\right\|_2^2\right]$$



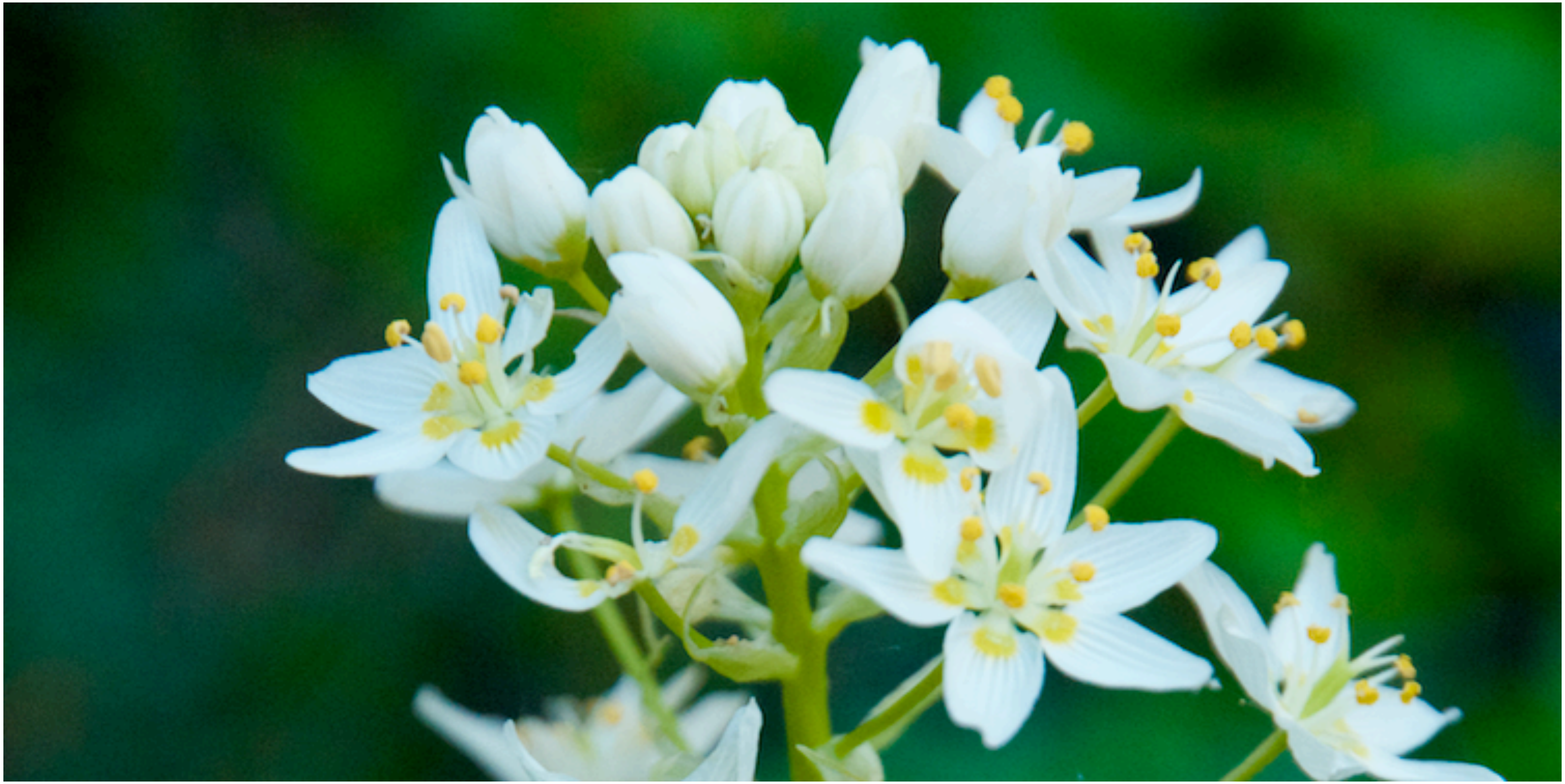
compression
model
$\lambda = \lambda_1$
compression
model
$\lambda = \lambda_2$
generative
models
$\lambda \to \infty$
$R + \lambda_1 D = \text{const}$
$R + \lambda_2 D = \text{const}$
D
R

JPEG @ 0.119 bits/px

JPEG 2000 @ 0.107 bits/px

proposed @ 0.106 bits/px

original



original

proposed

JPEG

JPEG 2000


original
proposed
JPEG
JPEG 2000

Thanks!

Perceptual metric

HV/EI 2016

Density estimation

ICLR 2016
arXiv: 1511.06281

Nonlinear transform coding

ICLR 2017
arXiv: 1611.01704