

Perception as Signal Processing

October 16, 2018

What is theory for?

To answer *why*?

What is theory for?

To answer *why*?

There are two sorts of answer in the context of neuroscience.

What is theory for?

To answer *why*?

There are two sorts of answer in the context of neuroscience.

Constructive or mechanistic – why is the sky blue?

- ▶ provides a mechanistic understanding of observations
- ▶ links structure to function
- ▶ helps to codify, organise and relate experimental findings

What is theory for?

To answer *why*?

There are two sorts of answer in the context of neuroscience.

Constructive or mechanistic – why is the sky blue?

- ▶ provides a mechanistic understanding of observations
- ▶ links structure to function
- ▶ helps to codify, organise and relate experimental findings

Normative or teleological – why do we see light between 390 to 700 nm?

- ▶ provides an understanding of the purpose of function
- ▶ only sensible in the context of evolutionary selection

Sensation and Perception

Two dominant ways of thinking about sensory systems and perception.

Signal processing – falls between normative and mechanistic

- ▶ a succession of filtering and feature-extraction stages that arrives at a 'detection' or 'recognition' output.
- ▶ dominated by feed-forward metaphors
 - ▶ temporal processing often limited to integration
 - ▶ some theories may incorporate local recurrence and also feedback for feature selection or attention
- ▶ behavioural and neural theory is dominated by information-like quantities

Inference – strongly normative

- ▶ parse sensory input to work out the configuration of the world
- ▶ fundamental roles for lateral interaction, feedback and dynamical state
- ▶ behavioural theory is well understood and powerful; neural underpinnings are little understood.

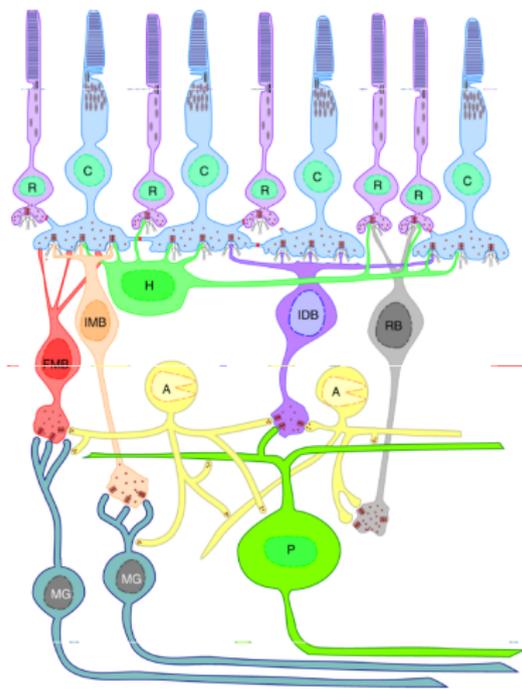
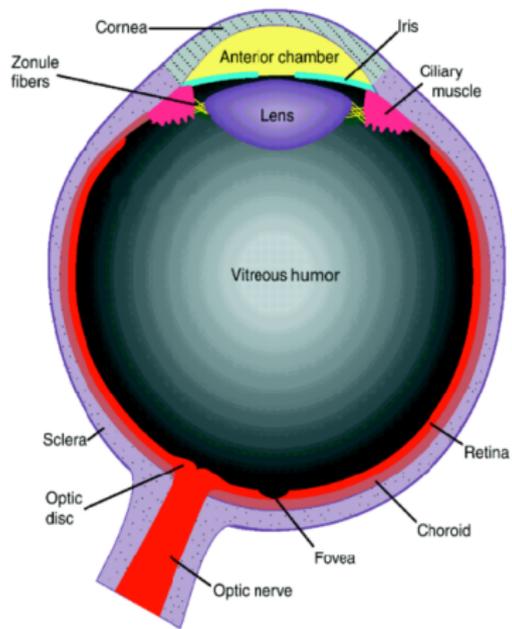
Signal-processing paradigms

- 1 filtering
- 2 (efficient) coding
- 3 feature detection

Signal-processing paradigms

- 1 filtering
- 2 (efficient) coding
- 3 feature detection

The eye and retina



Academic Press notes and derived items
copyright © 1999 by Academic Press

Centre-surround receptive fields

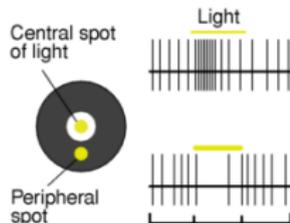
A On center field



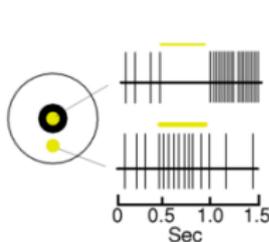
F Off center field



B On center cell responses



G Off center cell responses



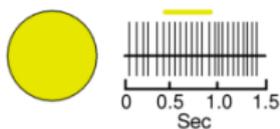
C Central illumination



D Annular illumination



E Diffuse illumination

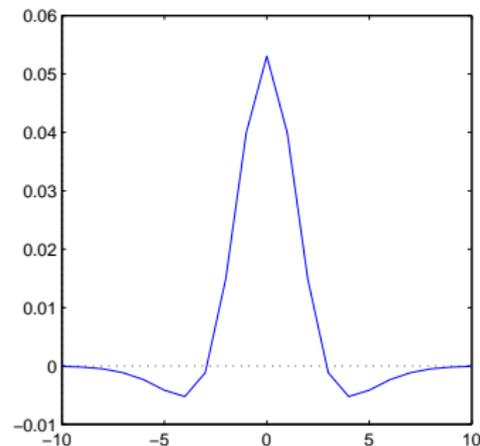
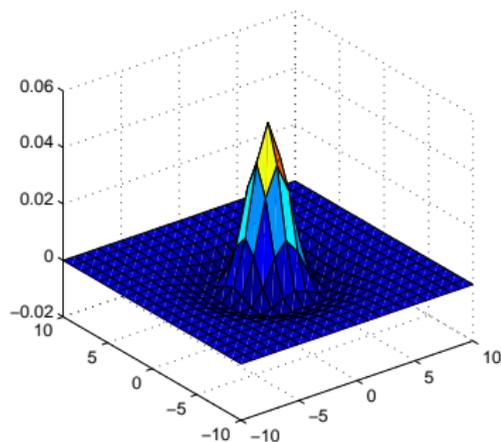


Centre-surround models

Centre-surround receptive fields are commonly described by one of two equations, giving the scaled response to a point of light shone at the retinal location (x, y) .

A difference-of-Gaussians (DoG) model:

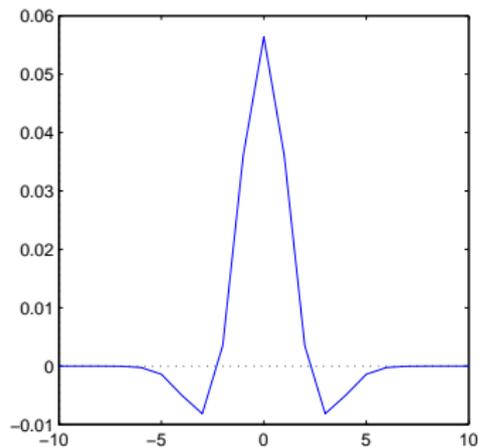
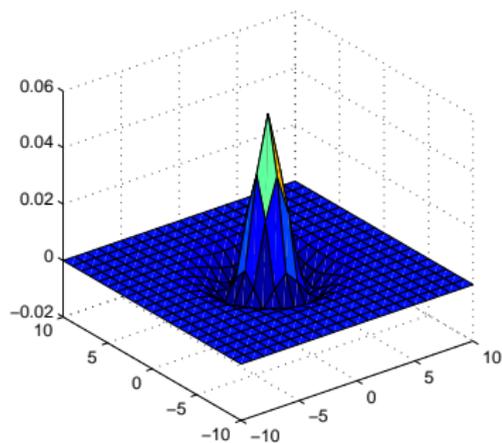
$$D_{\text{DoG}}(x, y) = \frac{1}{2\pi\sigma_c^2} \exp\left(-\frac{(x - c_x)^2 + (y - c_y)^2}{2\sigma_c^2}\right) - \frac{1}{2\pi\sigma_s^2} \exp\left(-\frac{(x - c_x)^2 + (y - c_y)^2}{2\sigma_s^2}\right)$$



Centre-surround models

... or a Laplacian-of-Gaussian (LoG) model:

$$D_{\text{LoG}}(x, y) = -\nabla^2 \left[\frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x - c_x)^2 + (y - c_y)^2}{2\sigma^2}\right) \right]$$



Linear receptive fields

The linear-like response apparent in the prototypical experiments can be generalised to give a predicted firing rate in response to an arbitrary stimulus $s(x, y)$:

$$r(c_x, c_y; s(x, y)) = \int dx dy D_{c_x, c_y}(x, y)s(x, y)$$

The receptive field centres (c_x, c_y) are distributed over visual space.

If we let $D()$ represent the RF function centred at 0, instead of at (c_x, c_y) , we can write:

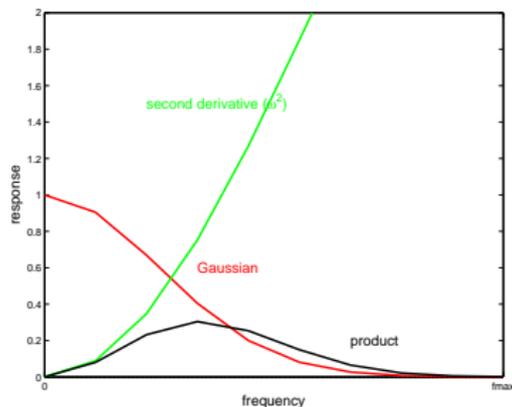
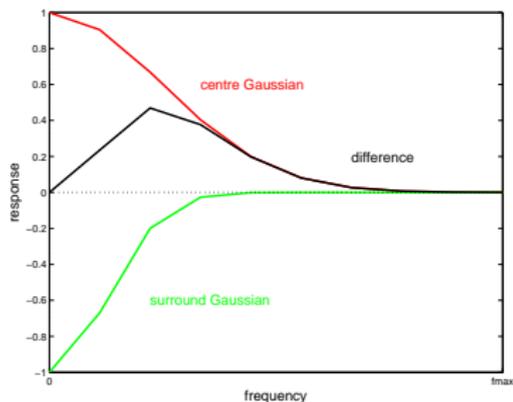
$$r(c_x, c_y; s(x, y)) = \int dx dy D(c_x - x, c_y - y)s(x, y)$$

which looks like a convolution.

Transfer functions

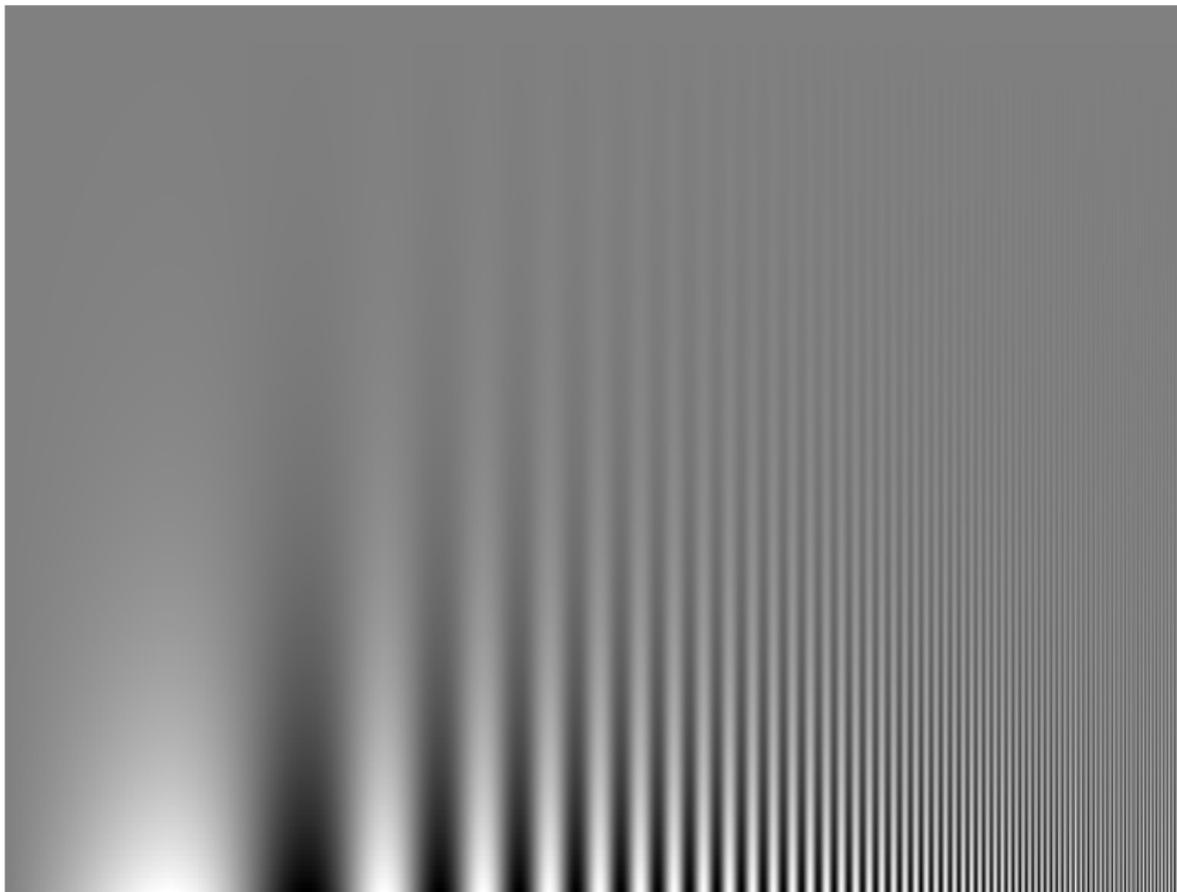
Thus a repeated linear receptive field acts like a spatial filter, and can be characterised by its frequency-domain transfer function. (Indeed, much early visual processing is studied in terms of linear systems theory.)

Transfer functions for both DoG and LoG centre-surround models are **bandpass**. Taking 1D versions:



This accentuates mid-range **spatial** frequencies.

Transfer functions



Edge detection

Bandpass filters emphasise edges:



original image

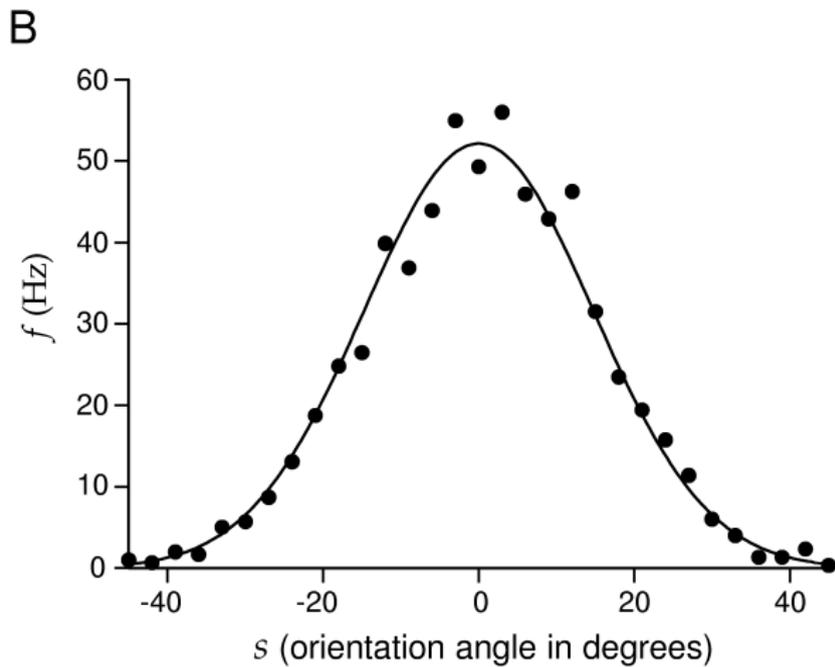
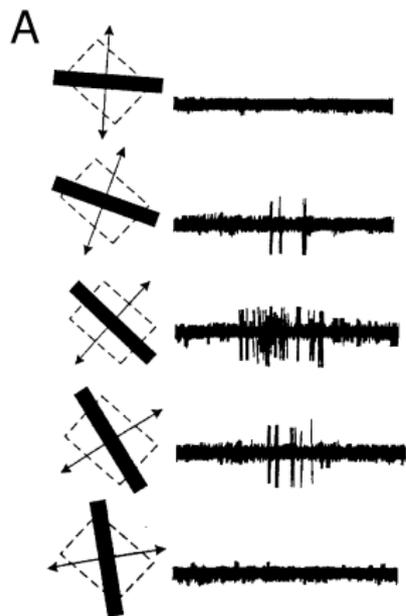


DoG responses



thresholded

Orientation selectivity



Linear receptive fields – simple cells

Linear response encoding:

$$r(t_0, s(x, y, t)) = \int_0^{\infty} d\tau \int dx dy s(x, y, t_0 - \tau) D(x, y, \tau)$$

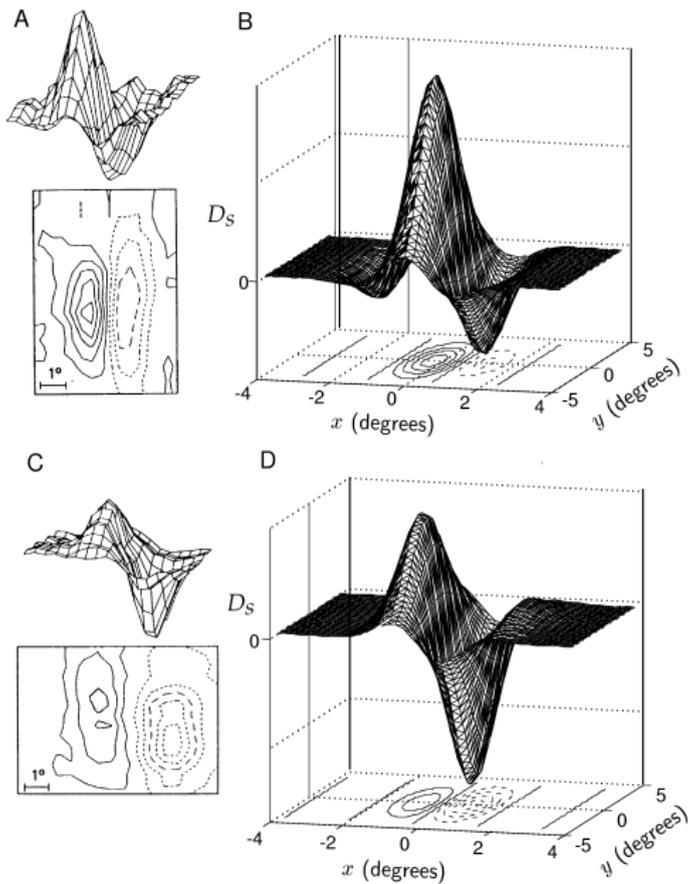
For separable receptive fields:

$$D(x, y, \tau) = D_s(x, y) D_t(\tau)$$

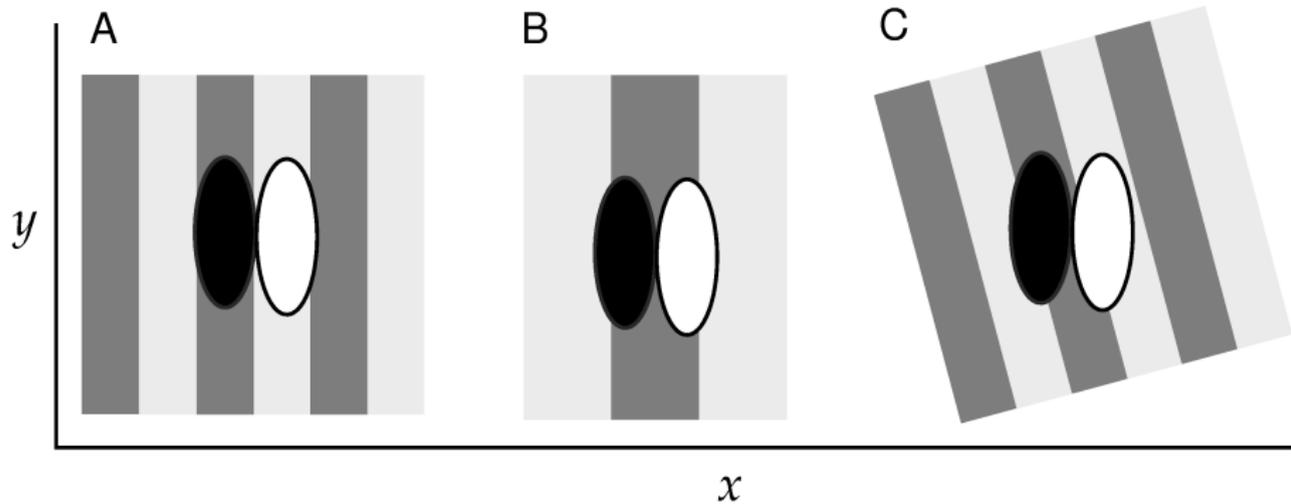
For simple cells:

$$D_s = \exp\left(-\frac{(x - c_x)^2}{2\sigma_x^2} - \frac{(y - c_y)^2}{2\sigma_y^2}\right) \cos(kx - \phi)$$

Linear response functions – simple cells



Simple cell orientation selectivity



2D Fourier Transforms

Again, the best way to look at a filter is in the frequency domain, but now we need a 2D transform.

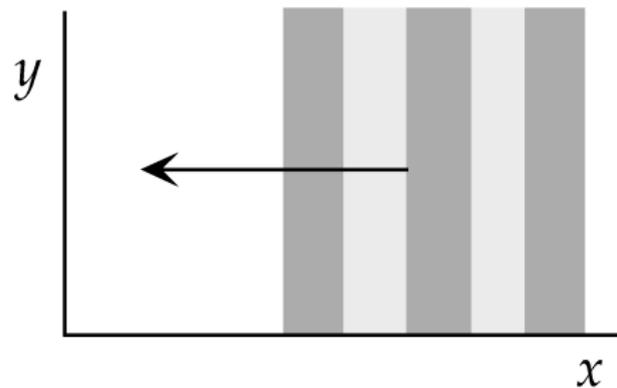
$$D(x, y) = \exp\left(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}\right) \cos(kx)$$

$$\begin{aligned}\tilde{D}(\omega_x, \omega_y) &= \int dx dy e^{-i\omega_x x} e^{-i\omega_y y} \exp\left(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}\right) \cos(kx - \phi) \\ &= \int dx e^{-i\omega_x x} e^{-x^2/2\sigma_x^2} \cos(kx - \phi) \cdot \int dy e^{-i\omega_y y} e^{-y^2/2\sigma_y^2} \\ &= \sqrt{2\pi}\sigma_x \left[e^{-\sigma_x^2 \omega_x^2 / 2} \circ \pi [\delta(\omega_x - k) + \delta(\omega_x + k)] \right] \sqrt{2\pi}\sigma_y e^{-\sigma_y^2 \omega_y^2 / 2} \\ &= 2\pi^2 \sigma_x \sigma_y \left[e^{-\frac{1}{2}[(\omega_x - k)^2 \sigma_x^2 + \omega_y^2 \sigma_y^2]} + e^{-\frac{1}{2}[(\omega_x + k)^2 \sigma_x^2 + \omega_y^2 \sigma_y^2]} \right]\end{aligned}$$

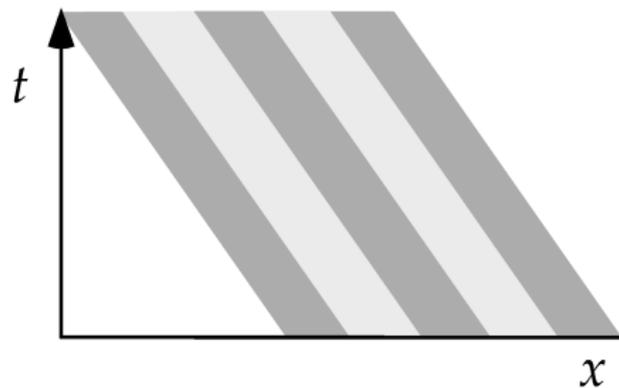
Easy to read spatial frequency tuning, bandwidth; orientation tuning and (for homework) bandwidth.

Drifting gratings

A

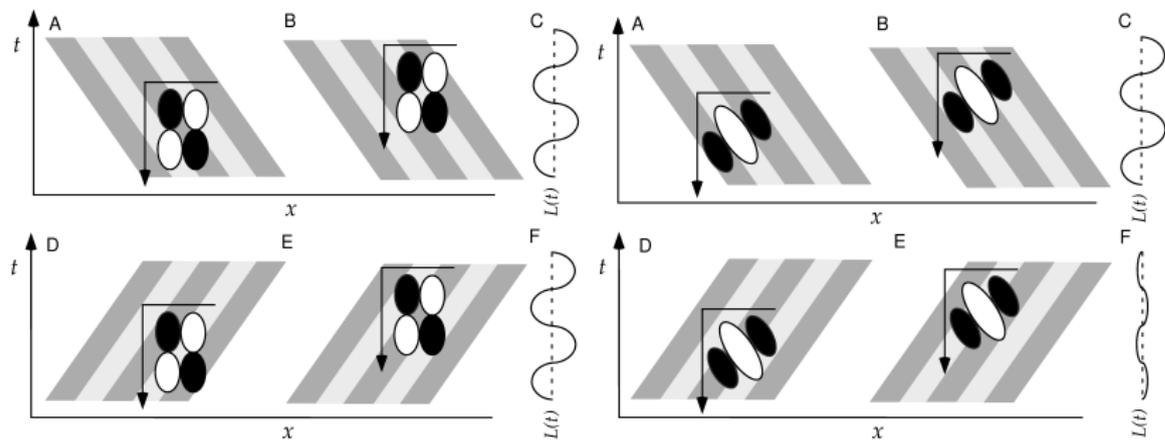


B



$$s(x, y, t) = G + A \cos(kx - \omega t - \phi)$$

Separable and inseparable response functions



Separable: motion sensitive;
not direction sensitive

Inseparable: motion sensitive;
and direction sensitive

Complex cells

Complex cells are sensitive to orientation, but, supposedly, not phase.

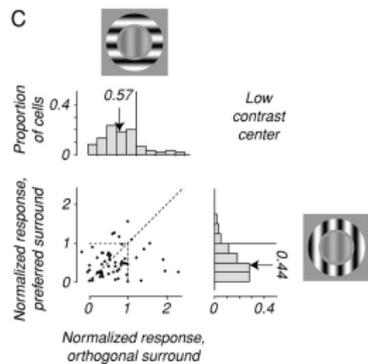
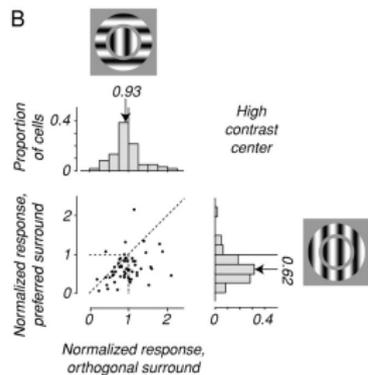
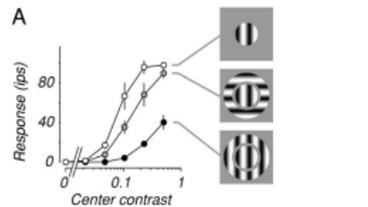
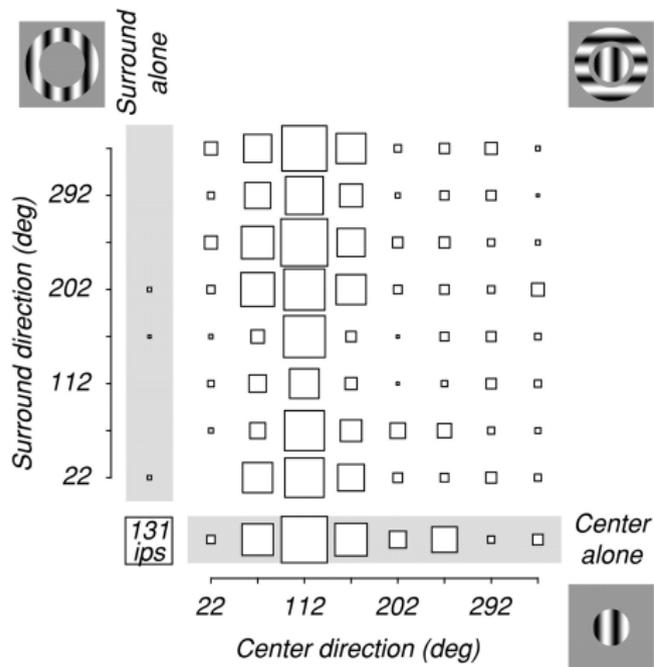
One model might be (neglecting time)

$$r(s(x, y)) = \left[\int dx dy s(x, y) \exp \left(-\frac{(x - c_x)^2}{2\sigma_x^2} - \frac{(y - c_y)^2}{2\sigma_y^2} \right) \cos(kx) \right]^2 \\ + \left[\int dx dy s(x, y) \exp \left(-\frac{(x - c_x)^2}{2\sigma_x^2} - \frac{(y - c_y)^2}{2\sigma_y^2} \right) \cos(kx - \pi/2) \right]^2$$

But many cells do have some residual phase sensitivity. Quantified by (f_1/f_0 ratio).

Stimulus-response functions (and constructive models) for complex cells are still a matter of debate.

Other V1 responses: surround effects



Other V1 responses

- ▶ end-stopping (hypercomplex)
- ▶ blobs and colour
- ▶ ...

Signal-processing paradigms

- 1 filtering
- 2 (efficient) coding
- 3 feature detection

Information

What does a neural response tell us about a stimulus?

Shannon theory:

- ▶ Entropy: *bits* needed to specify an exact stimulus.
- ▶ Conditional entropy: bits needed to specify the exact stimulus after we see the response.
- ▶ (Average mutual) information: the difference (information gained from the response)
- ▶ Mutual information is bounded by the entropy of the response \Rightarrow maximum entropy encoding and decorrelation.

Discrimination theory:

- ▶ *How accurately* (squared-error) can the stimulus be estimated from the response.
- ▶ Cramér-Rao bound relates this to the Fisher Information – a differential measure of how much the response distribution changes with the stimulus.
- ▶ Fisher information can often be optimised directly.

Linked by rate-distortion theory and by asymptotic (large population) arguments.

Entropy maximisation

$$I[\tilde{S}; R] = \underbrace{H[R]}_{\text{marginal entropy}} - \underbrace{H[R|\tilde{S}]}_{\text{noise entropy}}$$

Entropy maximisation

$$I[\tilde{S}; R] = \underbrace{H[R]}_{\text{marginal entropy}} - \underbrace{H[R|\tilde{S}]}_{\text{noise entropy}}$$

If noise is small and “constant” \Rightarrow maximise marginal entropy \Rightarrow maximise $H[\tilde{S}]$

Entropy maximisation

$$I[\tilde{S}; R] = \underbrace{H[R]}_{\text{marginal entropy}} - \underbrace{H[R|\tilde{S}]}_{\text{noise entropy}}$$

If noise is small and “constant” \Rightarrow maximise marginal entropy \Rightarrow maximise $H[\tilde{S}]$

Consider a (rate coding) neuron with $r \in [0, r_{\max}]$.

$$h(r) = - \int_0^{r_{\max}} dr p(r) \log p(r)$$

Entropy maximisation

$$I[\tilde{S}; R] = \underbrace{\mathbf{H}[R]}_{\text{marginal entropy}} - \underbrace{\mathbf{H}[R|\tilde{S}]}_{\text{noise entropy}}$$

If noise is small and “constant” \Rightarrow maximise marginal entropy \Rightarrow maximise $\mathbf{H}[\tilde{S}]$

Consider a (rate coding) neuron with $r \in [0, r_{\max}]$.

$$h(r) = - \int_0^{r_{\max}} dr p(r) \log p(r)$$

To maximise the marginal entropy, we add a Lagrange multiplier (μ) to enforce normalisation and then differentiate

$$\frac{\delta}{\delta p(r)} \left[h(r) - \mu \int_0^{r_{\max}} p(r) \right] = \begin{cases} -\log p(r) - 1 - \mu & r \in [0, r_{\max}] \\ 0 & \text{otherwise} \end{cases}$$

Entropy maximisation

$$I[\tilde{S}; R] = \underbrace{H[R]}_{\text{marginal entropy}} - \underbrace{H[R|\tilde{S}]}_{\text{noise entropy}}$$

If noise is small and “constant” \Rightarrow maximise marginal entropy \Rightarrow maximise $H[\tilde{S}]$

Consider a (rate coding) neuron with $r \in [0, r_{\max}]$.

$$h(r) = - \int_0^{r_{\max}} dr p(r) \log p(r)$$

To maximise the marginal entropy, we add a Lagrange multiplier (μ) to enforce normalisation and then differentiate

$$\frac{\delta}{\delta p(r)} \left[h(r) - \mu \int_0^{r_{\max}} p(r) \right] = \begin{cases} -\log p(r) - 1 - \mu & r \in [0, r_{\max}] \\ 0 & \text{otherwise} \end{cases}$$

$\Rightarrow p(r) = \text{const}$ for $r \in [0, r_{\max}]$

Entropy maximisation

$$I[\tilde{S}; R] = \underbrace{H[R]}_{\text{marginal entropy}} - \underbrace{H[R|\tilde{S}]}_{\text{noise entropy}}$$

If noise is small and “constant” \Rightarrow maximise marginal entropy \Rightarrow maximise $H[\tilde{S}]$

Consider a (rate coding) neuron with $r \in [0, r_{\max}]$.

$$h(r) = - \int_0^{r_{\max}} dr p(r) \log p(r)$$

To maximise the marginal entropy, we add a Lagrange multiplier (μ) to enforce normalisation and then differentiate

$$\frac{\delta}{\delta p(r)} \left[h(r) - \mu \int_0^{r_{\max}} p(r) \right] = \begin{cases} -\log p(r) - 1 - \mu & r \in [0, r_{\max}] \\ 0 & \text{otherwise} \end{cases}$$

$\Rightarrow p(r) = \text{const}$ for $r \in [0, r_{\max}]$

i.e.

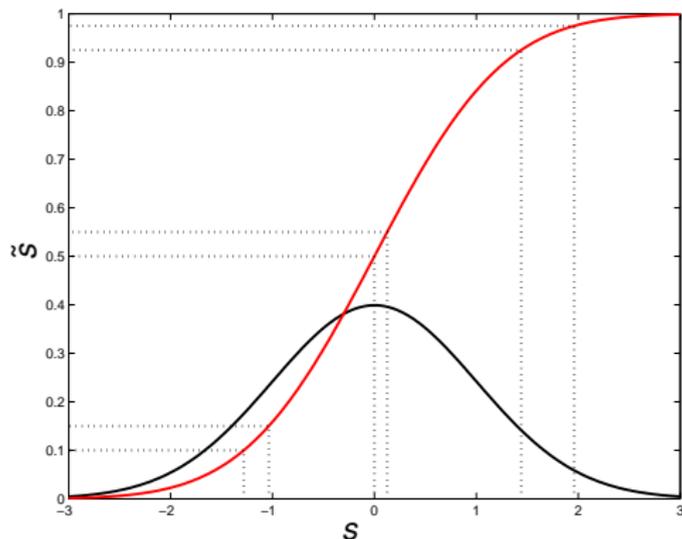
$$p(r) = \begin{cases} \frac{1}{r_{\max}} & r \in [0, r_{\max}] \\ 0 & \text{otherwise} \end{cases}$$

Histogram Equalisation

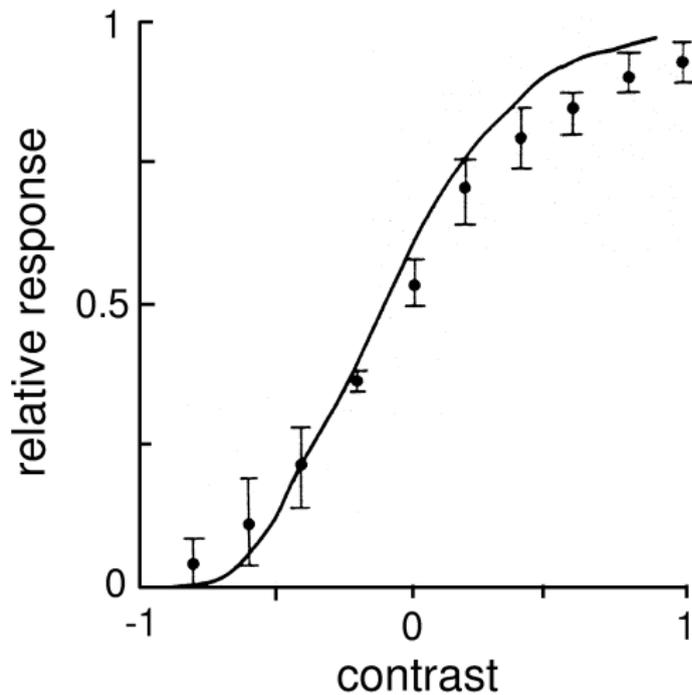
Suppose $r = \tilde{s} + \eta$ where η represents a (relatively small) source of noise. Consider deterministic encoding $\tilde{s} = f(s)$. How do we ensure that $p(r) = 1/r_{\max}$?

$$\frac{1}{r_{\max}} = p(r) \approx p(\tilde{s}) = \frac{p(s)}{f'(s)} \quad \Rightarrow f'(s) = r_{\max} p(s)$$

$$\Rightarrow f(s) = r_{\max} \int_{-\infty}^s ds' p(s')$$



Histogram Equalisation



Laughlin (1981)

Decorrelation at the retina

Atick and Redlich (1992) argued that the retina decorrelates natural spatial statistics.

Decorrelation at the retina

Atick and Redlich (1992) argued that the retina decorrelates natural spatial statistics. RGCs exhibit roughly linear (centre-surround) processing:

$$r_{\mathbf{a}} - \langle r_{\mathbf{a}} \rangle = \int d\mathbf{x} \underbrace{D_s(\mathbf{x} - \mathbf{a})}_{\text{filter}} \underbrace{s(\mathbf{x})}_{\text{stimulus}}$$

Decorrelation at the retina

Atick and Redlich (1992) argued that the retina decorrelates natural spatial statistics. RGCs exhibit roughly linear (centre-surround) processing:

$$r_{\mathbf{a}} - \langle r_{\mathbf{a}} \rangle = \int d\mathbf{x} \underbrace{D_s(\mathbf{x} - \mathbf{a})}_{\text{filter}} \underbrace{s(\mathbf{x})}_{\text{stimulus}}$$

Therefore the correlation (covariance) between cells is

$$\begin{aligned} Q_r(\mathbf{a}, \mathbf{b}) &= \left\langle \int d\mathbf{x} d\mathbf{y} D_s(\mathbf{x} - \mathbf{a}) D_s(\mathbf{y} - \mathbf{b}) s(\mathbf{x}) s(\mathbf{y}) \right\rangle \\ &= \int d\mathbf{x} d\mathbf{y} D_s(\mathbf{x} - \mathbf{a}) D_s(\mathbf{y} - \mathbf{b}) \underbrace{\langle s(\mathbf{x}) s(\mathbf{y}) \rangle}_{Q_s(\mathbf{x}, \mathbf{y})} \end{aligned}$$

Decorrelation at the retina

Atick and Redlich (1992) argued that the retina decorrelates natural spatial statistics. RGCs exhibit roughly linear (centre-surround) processing:

$$r_{\mathbf{a}} - \langle r_{\mathbf{a}} \rangle = \int d\mathbf{x} \underbrace{D_s(\mathbf{x} - \mathbf{a})}_{\text{filter}} \underbrace{s(\mathbf{x})}_{\text{stimulus}}$$

Therefore the correlation (covariance) between cells is

$$\begin{aligned} Q_r(\mathbf{a}, \mathbf{b}) &= \left\langle \int d\mathbf{x} d\mathbf{y} D_s(\mathbf{x} - \mathbf{a}) D_s(\mathbf{y} - \mathbf{b}) s(\mathbf{x}) s(\mathbf{y}) \right\rangle \\ &= \int d\mathbf{x} d\mathbf{y} D_s(\mathbf{x} - \mathbf{a}) D_s(\mathbf{y} - \mathbf{b}) \underbrace{\langle s(\mathbf{x}) s(\mathbf{y}) \rangle}_{Q_s(\mathbf{x}, \mathbf{y})} \end{aligned}$$

Using (spatial) stationarity, we can transform to the Fourier domain:

$$\tilde{Q}_r(\mathbf{k}) = |\tilde{D}_s(\mathbf{k})|^2 \tilde{Q}_s(\mathbf{k})$$

Decorrelation at the retina

Atick and Redlich (1992) argued that the retina decorrelates natural spatial statistics. RGCs exhibit roughly linear (centre-surround) processing:

$$r_{\mathbf{a}} - \langle r_{\mathbf{a}} \rangle = \int d\mathbf{x} \underbrace{D_s(\mathbf{x} - \mathbf{a})}_{\text{filter}} \underbrace{s(\mathbf{x})}_{\text{stimulus}}$$

Therefore the correlation (covariance) between cells is

$$\begin{aligned} Q_r(\mathbf{a}, \mathbf{b}) &= \left\langle \int d\mathbf{x} d\mathbf{y} D_s(\mathbf{x} - \mathbf{a}) D_s(\mathbf{y} - \mathbf{b}) s(\mathbf{x}) s(\mathbf{y}) \right\rangle \\ &= \int d\mathbf{x} d\mathbf{y} D_s(\mathbf{x} - \mathbf{a}) D_s(\mathbf{y} - \mathbf{b}) \underbrace{\langle s(\mathbf{x}) s(\mathbf{y}) \rangle}_{Q_s(\mathbf{x}, \mathbf{y})} \end{aligned}$$

Using (spatial) stationarity, we can transform to the Fourier domain:

$$\tilde{Q}_r(\mathbf{k}) = |\tilde{D}_s(\mathbf{k})|^2 \tilde{Q}_s(\mathbf{k})$$

and thus output decorrelation requires

$$|\tilde{D}_s(\mathbf{k})|^2 \propto \frac{1}{\tilde{Q}_s(\mathbf{k})}$$

Decorrelation at the retina

Spatial correlations of natural images fall off with f^{-2} :

$$\tilde{Q}_s(\mathbf{k}) \propto \frac{1}{|\mathbf{k}|^2 + k_0^2}$$

and the optical filter of the eye introduces (crudely) a low-pass term $\propto e^{-\alpha|\mathbf{k}|}$.

So decorrelation requires

$$|\tilde{D}_s(\mathbf{k})|^2 \propto \frac{|\mathbf{k}|^2 + k_0^2}{e^{-\alpha|\mathbf{k}|}}$$

Decorrelation at the retina

Spatial correlations of natural images fall off with f^{-2} :

$$\tilde{Q}_s(\mathbf{k}) \propto \frac{1}{|\mathbf{k}|^2 + k_0^2}$$

and the optical filter of the eye introduces (crudely) a low-pass term $\propto e^{-\alpha|\mathbf{k}|}$.

So decorrelation requires

$$|\tilde{D}_s(\mathbf{k})|^2 \propto \frac{|\mathbf{k}|^2 + k_0^2}{e^{-\alpha|\mathbf{k}|}}$$

But: not all input is signal.

Decorrelation at the retina

Spatial correlations of natural images fall off with f^{-2} :

$$\tilde{Q}_s(\mathbf{k}) \propto \frac{1}{|\mathbf{k}|^2 + k_0^2}$$

and the optical filter of the eye introduces (crudely) a low-pass term $\propto e^{-\alpha|\mathbf{k}|}$.

So decorrelation requires

$$|\tilde{D}_s(\mathbf{k})|^2 \propto \frac{|\mathbf{k}|^2 + k_0^2}{e^{-\alpha|\mathbf{k}|}}$$

But: not all input is signal.

Photodetection introduces noise. Therefore, cascade linear filters:

$$\mathbf{s} + \boldsymbol{\eta} \xrightarrow{D_\eta} \hat{\mathbf{s}} \xrightarrow{D_s} \mathbf{r}$$

with

$$\tilde{D}_\eta(\mathbf{k}) = \frac{\tilde{Q}_s(\mathbf{k})}{\tilde{Q}_s(\mathbf{k}) + \tilde{Q}_\eta(\mathbf{k})} \quad (\text{Wiener filter})$$

Decorrelation at the retina

Spatial correlations of natural images fall off with f^{-2} :

$$\tilde{Q}_s(\mathbf{k}) \propto \frac{1}{|\mathbf{k}|^2 + k_0^2}$$

and the optical filter of the eye introduces (crudely) a low-pass term $\propto e^{-\alpha|\mathbf{k}|}$.

So decorrelation requires

$$|\tilde{D}_s(\mathbf{k})|^2 \propto \frac{|\mathbf{k}|^2 + k_0^2}{e^{-\alpha|\mathbf{k}|}}$$

But: not all input is signal.

Photodetection introduces noise. Therefore, cascade linear filters:

$$\mathbf{s} + \boldsymbol{\eta} \xrightarrow{D_\eta} \hat{\mathbf{s}} \xrightarrow{D_s} \mathbf{r}$$

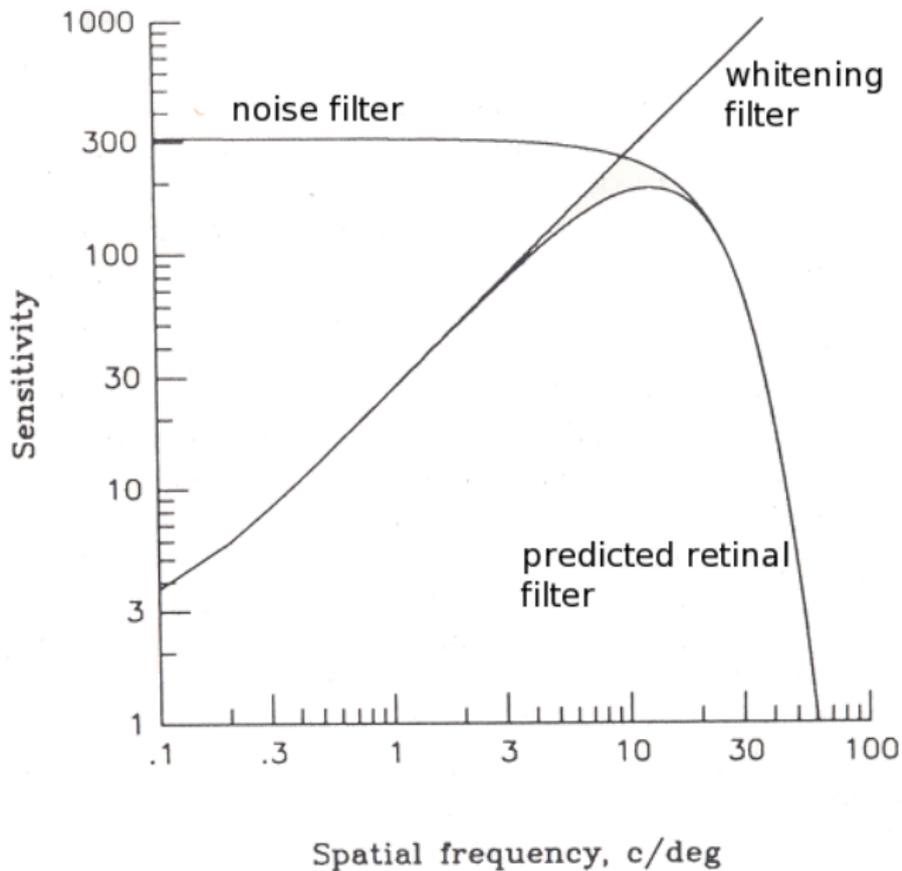
with

$$\tilde{D}_\eta(\mathbf{k}) = \frac{\tilde{Q}_s(\mathbf{k})}{\tilde{Q}_s(\mathbf{k}) + \tilde{Q}_\eta(\mathbf{k})} \quad (\text{Wiener filter})$$

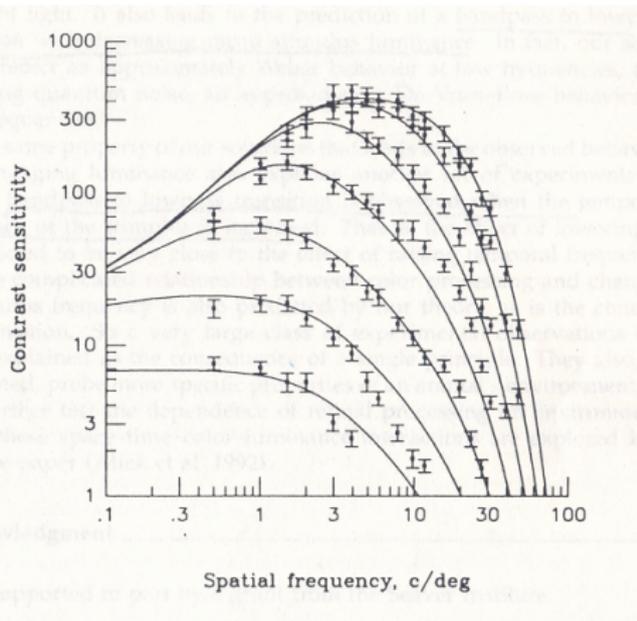
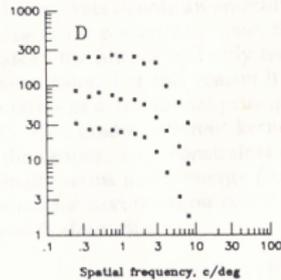
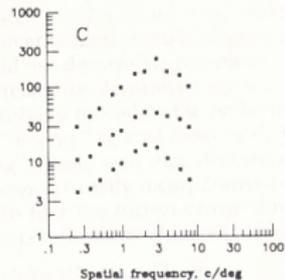
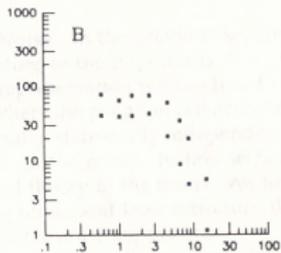
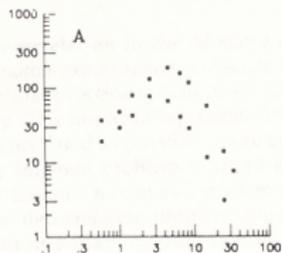
Thus the combined RGC filter is predicted to be:

$$|\tilde{D}_s(\mathbf{k})| \tilde{D}_\eta(\mathbf{k}) \propto \frac{\sqrt{\tilde{Q}_s(\mathbf{k})}}{\tilde{Q}_s(\mathbf{k}) + \tilde{Q}_\eta(\mathbf{k})}$$

Decorrelation at the retina



Decorrelation at the retina



Tuning curves

We often consider the way that the firing rate of a cell r represents a single (possibly multidimensional) stimulus value s :

$$r = f(s).$$

Even if s and r are embedded in time-series we assume:

1. that coding is instantaneous (with a fixed lag),
2. that r (and therefore s) is constant over a short time Δ .

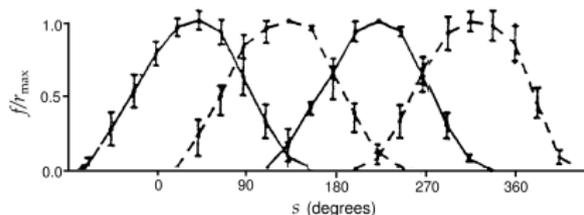
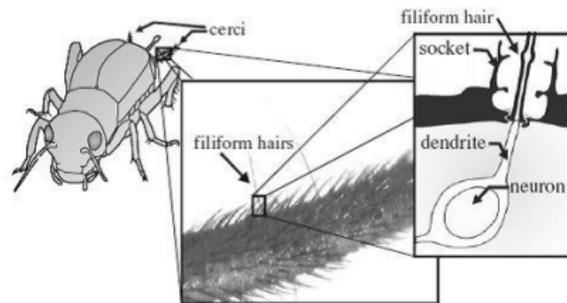
The function $f(s)$ is known as a tuning curve.

Tuning curves

Commonly assumed mathematical forms for (1D) tuning curves:

- Gaussian $r_0 + r_{\max} \exp \left[-\frac{1}{2\sigma^2} (x - x_{\text{pref}})^2 \right]$
- (Thresholded) Ramp $r_0 + \Theta(x - x_{\text{thr}}) r_{\max} \rho \cdot (x - x_{\text{thr}})$
- Cosine $r_0 + r_{\max} \cos(\theta - \theta_{\text{pref}})$
- Wrapped Gaussian $r_0 + r_{\max} \sum_n \exp \left[-\frac{1}{2\sigma^2} (\theta - \theta_{\text{pref}} - 2\pi n)^2 \right]$
- von Mises (“circular Gaussian”) $r_0 + r_{\max} \exp [\kappa \cos(\theta - \theta_{\text{pref}})]$
- periodic (grid) $f(s) = f_1(\sin(2\pi s/\lambda))$

Decoding – the Cricket cercal system



$$r_a(s) = r_a^{\max} [\cos(\theta - \theta_a)]_+ = r_a^{\max} [\mathbf{c}_a^T \mathbf{v}]_+$$

$$\mathbf{c}_1^T \mathbf{c}_2 = 0$$

$$\mathbf{c}_3 = -\mathbf{c}_1$$

$$\mathbf{c}_4 = -\mathbf{c}_2$$

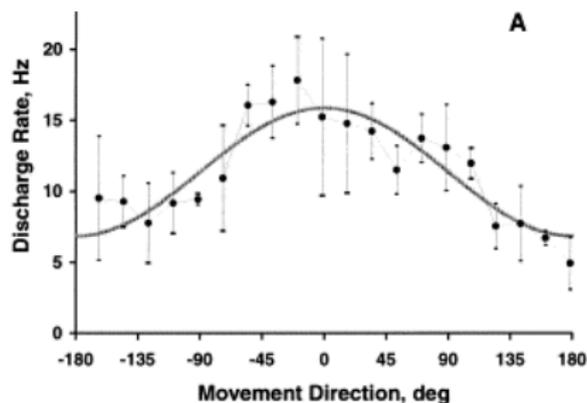
So, writing $\tilde{r}_a = r_a/r_a^{\max}$:

$$\begin{pmatrix} \tilde{r}_1 - \tilde{r}_3 \\ \tilde{r}_2 - \tilde{r}_4 \end{pmatrix} = \begin{pmatrix} \mathbf{c}_1^T \\ \mathbf{c}_2^T \end{pmatrix} \mathbf{v}$$

$$\mathbf{v} = (\mathbf{c}_1 \mathbf{c}_2) \begin{pmatrix} \tilde{r}_1 - \tilde{r}_3 \\ \tilde{r}_2 - \tilde{r}_4 \end{pmatrix} = \tilde{r}_1 \mathbf{c}_1 - \tilde{r}_3 \mathbf{c}_3 + \tilde{r}_2 \mathbf{c}_2 - \tilde{r}_4 \mathbf{c}_4 = \sum_a \tilde{r}_a \mathbf{c}_a$$

This is called **population vector decoding**.

Motor cortex (simplified)



Cosine tuning, randomly distributed preferred directions.

In general, population vector decoding works for

- ▶ cosine tuning
- ▶ cartesian or dense (*tight*) directions

But:

- ▶ is it optimal?
- ▶ does it generalise? (Gaussian tuning curves)
- ▶ how accurate is it?

Measuring the potential quality of a representation

Consider a (one dimensional) stimulus that takes on continuous values (e.g. angle).

- ▶ contrast
- ▶ orientation
- ▶ motion direction
- ▶ movement speed

Suppose a neuron fires n spikes in response to stimulus s according to some distribution

$$P(n|f(s)\Delta)$$

Given an observation of n , how well can we estimate s ?

Cramér-Rao bound

Suppose the neural response can be described by a probability distribution $P(r|s)$. The Fisher information measures how this distribution changes with s :

$$J(s^*) = - \left\langle \frac{d^2 \log P(r|s)}{ds^2} \Big|_{s^*} \right\rangle_{s^*} = \left\langle \left(\frac{d \log P(r|s)}{ds} \Big|_{s^*} \right)^2 \right\rangle_{s^*}$$

The Cramér-Rao bound states that for any N , any *unbiased* estimator $\hat{s}(\{n_i\})$ of s will have the property that

$$\langle (\hat{s}(\{n_i\}) - s^*)^2 \rangle_{n_i|s^*} \geq \frac{1}{J(s^*)}.$$

Thus, Fisher Information gives a lower bound on the variance of any unbiased estimator.

[For estimators with *bias* $b(s^*) = \langle \hat{s}(\{n_i\}) - s^* \rangle$ the bound is:

$$\langle (\hat{s}(\{n_i\}) - s^*)^2 \rangle_{n_i|s^*} \geq \frac{(1+b'(s^*))^2}{J(s^*)} + b^2(s^*)]$$

The Fisher Information is the most common tool to analyse optimality in populations.

Fisher Info and tuning curves

$$n = r\Delta + \text{noise}; \quad r = f(s) \Rightarrow$$

$$J(s^*) = \left\langle \left(\left. \frac{d}{ds} \right|_{s^*} \log P(n|s) \right)^2 \right\rangle_{s^*}$$

Fisher Info and tuning curves

$n = r\Delta + \text{noise}; \quad r = f(s) \Rightarrow$

$$\begin{aligned} J(s^*) &= \left\langle \left(\left. \frac{d}{ds} \right|_{s^*} \log P(n|s) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\left. \frac{d}{dr\Delta} \right|_{f(s^*)} \log P(n|r\Delta) \Delta f'(s^*) \right)^2 \right\rangle_{s^*} \end{aligned}$$

Fisher Info and tuning curves

$$n = r\Delta + \text{noise}; \quad r = f(s) \Rightarrow$$

$$\begin{aligned} J(s^*) &= \left\langle \left(\left. \frac{d}{ds} \right|_{s^*} \log P(n|s) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\left. \frac{d}{dr\Delta} \right|_{f(s^*)} \log P(n|r\Delta) \Delta f'(s^*) \right)^2 \right\rangle_{s^*} \\ &= J_{\text{noise}}(r\Delta) \Delta^2 f'(s^*)^2 \end{aligned}$$

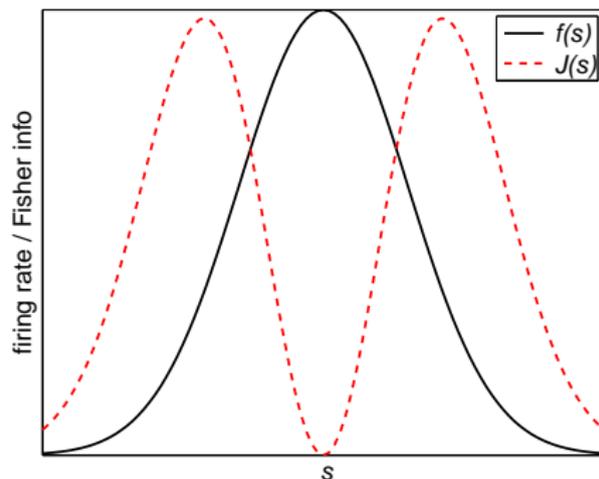
Fisher Info and tuning curves

$$n = r\Delta + \text{noise}; \quad r = f(s) \Rightarrow$$

$$J(s^*) = \left\langle \left(\left. \frac{d}{ds} \right|_{s^*} \log P(n|s) \right)^2 \right\rangle_{s^*}$$

$$= \left\langle \left(\left. \frac{d}{dr\Delta} \right|_{f(s^*)} \log P(n|r\Delta) \Delta f'(s^*) \right)^2 \right\rangle_{s^*}$$

$$= J_{\text{noise}}(r\Delta) \Delta^2 f'(s^*)^2$$



Fisher info for Poisson neurons

For Poisson neurons

$$P(n|r\Delta) = \frac{e^{-r\Delta}}{(r\Delta)^n} n!$$

so

$$J_{\text{noise}}[r\Delta] = \left\langle \left(\left. \frac{d}{dr\Delta} \right|_{r^*\Delta} \log P(n|r\Delta) \right)^2 \right\rangle_{s^*}$$

Fisher info for Poisson neurons

For Poisson neurons

$$P(n|r\Delta) = \frac{e^{-r\Delta}}{(r\Delta)^n} n!$$

so

$$\begin{aligned} J_{\text{noise}}[r\Delta] &= \left\langle \left(\left. \frac{d}{dr\Delta} \right|_{r^*\Delta} \log P(n|r\Delta) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\left. \frac{d}{dr\Delta} \right|_{r^*\Delta} - r\Delta + n \log r\Delta - \log n! \right)^2 \right\rangle_{s^*} \end{aligned}$$

Fisher info for Poisson neurons

For Poisson neurons

$$P(n|r\Delta) = \frac{e^{-r\Delta}}{(r\Delta)^n} n!$$

so

$$\begin{aligned} J_{\text{noise}}[r\Delta] &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} \log P(n|r\Delta) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} - r\Delta + n \log r\Delta - \log n! \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(-1 + n/r^*\Delta \right)^2 \right\rangle_{s^*} \end{aligned}$$

Fisher info for Poisson neurons

For Poisson neurons

$$P(n|r\Delta) = \frac{e^{-r\Delta}}{(r\Delta)^n} n!$$

so

$$\begin{aligned} J_{\text{noise}}[r\Delta] &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} \log P(n|r\Delta) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} - r\Delta + n \log r\Delta - \log n! \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(-1 + n/r^*\Delta \right)^2 \right\rangle_{s^*} \\ &= \left\langle \frac{(n - r^*\Delta)^2}{(r^*\Delta)^2} \right\rangle_{s^*} \end{aligned}$$

Fisher info for Poisson neurons

For Poisson neurons

$$P(n|r\Delta) = \frac{e^{-r\Delta}}{(r\Delta)^n} n!$$

so

$$\begin{aligned} J_{\text{noise}}[r\Delta] &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} \log P(n|r\Delta) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} - r\Delta + n \log r\Delta - \log n! \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(-1 + n/r^*\Delta \right)^2 \right\rangle_{s^*} \\ &= \left\langle \frac{(n - r^*\Delta)^2}{(r^*\Delta)^2} \right\rangle_{s^*} \\ &= \frac{r^*\Delta}{(r^*\Delta)^2} \end{aligned}$$

Fisher info for Poisson neurons

For Poisson neurons

$$P(n|r\Delta) = \frac{e^{-r\Delta}}{(r\Delta)^n} n!$$

so

$$\begin{aligned} J_{\text{noise}}[r\Delta] &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} \log P(n|r\Delta) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} - r\Delta + n \log r\Delta - \log n! \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(-1 + n/r^*\Delta \right)^2 \right\rangle_{s^*} \\ &= \left\langle \frac{(n - r^*\Delta)^2}{(r^*\Delta)^2} \right\rangle_{s^*} \\ &= \frac{r^*\Delta}{(r^*\Delta)^2} = \frac{1}{r^*\Delta} \end{aligned}$$

Fisher info for Poisson neurons

For Poisson neurons

$$P(n|r\Delta) = \frac{e^{-r\Delta}}{(r\Delta)^n} n!$$

so

$$\begin{aligned} J_{\text{noise}}[r\Delta] &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} \log P(n|r\Delta) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} - r\Delta + n \log r\Delta - \log n! \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(-1 + n/r^*\Delta \right)^2 \right\rangle_{s^*} \\ &= \left\langle \frac{(n - r^*\Delta)^2}{(r^*\Delta)^2} \right\rangle_{s^*} \\ &= \frac{r^*\Delta}{(r^*\Delta)^2} = \frac{1}{r^*\Delta} \quad [\text{not surprising! } \widehat{r^*\Delta} = n \text{ and } \mathbb{V}[n] = r^*\Delta] \end{aligned}$$

Fisher info for Poisson neurons

For Poisson neurons

$$P(n|r\Delta) = \frac{e^{-r\Delta}}{(r\Delta)^n} n!$$

so

$$\begin{aligned} J_{\text{noise}}[r\Delta] &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} \log P(n|r\Delta) \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(\frac{d}{dr\Delta} \Big|_{r^*\Delta} - r\Delta + n \log r\Delta - \log n! \right)^2 \right\rangle_{s^*} \\ &= \left\langle \left(-1 + n/r^*\Delta \right)^2 \right\rangle_{s^*} \\ &= \left\langle \frac{(n - r^*\Delta)^2}{(r^*\Delta)^2} \right\rangle_{s^*} \\ &= \frac{r^*\Delta}{(r^*\Delta)^2} = \frac{1}{r^*\Delta} \quad [\text{not surprising! } \widehat{r^*\Delta} = n \text{ and } \mathbb{V}[n] = r^*\Delta] \end{aligned}$$

and, referred back to the stimulus value:

$$J[s^*] = f'(s^*)^2 \Delta / f(s^*)$$

Population Fisher Info

Fisher Informations for independent random variates add:

$$\begin{aligned} J_{\mathbf{n}}(s) &= \left\langle -\frac{d^2}{ds^2} \log P(\mathbf{n}|s) \right\rangle \\ &= \left\langle -\frac{d^2}{ds^2} \sum_a \log P(n_a|s) \right\rangle \\ &= \sum_a \left\langle -\frac{d^2}{ds^2} \log P(n_a|s) \right\rangle = \sum_a J_{n_a}(s). \\ &= \Delta \sum_a \frac{f'_a(s)^2}{f_a(s)} \quad \text{[for Poisson cells]} \end{aligned}$$

Optimal tuning properties

A considerable amount of work has been done in recent years on finding optimal properties of tuning curves for rate-based population codes. Here, we reproduce one such argument (from Zhang and Sejnowski, 1999).

Consider a population of cells that codes the value of a D dimensional stimulus, \mathbf{s} . Let the a th cell emit r spikes in an interval τ with probability distribution that is conditionally independent of the other cells (given \mathbf{s}) and has the form

$$P_a(r | \mathbf{s}, \tau) = S(r, f^a(\mathbf{s}), \tau).$$

Also let the tuning curve of the a th cell, $f^a(\mathbf{s})$, be circularly symmetric:

$$f^a(\mathbf{s}) = F \cdot \phi((\xi^a)^2); \quad (\xi^a)^2 = \sum_i^D (\xi_i^a)^2; \quad \xi_i^a = \frac{s_i - c_i^a}{\sigma},$$

where F is a maximal rate and the function ϕ is monotonically decreasing. The parameters \mathbf{c}^a and σ give the centre of the a th tuning curve and the (common) width.

Optimal tuning properties

Now, the (ij) th term in the FI matrix for the a th cell is (by definition)

$$J_{ij}^a(\mathbf{s}) = \mathbb{E} \left[\frac{\partial}{\partial s_i} \log P^a(r | \mathbf{s}, \tau) \frac{\partial}{\partial s_j} \log P^a(r | \mathbf{s}, \tau) \right]$$

Applying the chain rule repeatedly, we find that

$$\begin{aligned} \frac{\partial}{\partial s_i} \log P^a(r | \mathbf{s}, \tau) &= \frac{1}{S(r, f^a(\mathbf{s}), \tau)} \frac{\partial}{\partial s_i} S(r, f^a(\mathbf{s}), \tau) \\ &= \frac{S^{(2)}(r, f^a(\mathbf{s}), \tau)}{S(r, f^a(\mathbf{s}), \tau)} \frac{\partial}{\partial s_i} f^a(\mathbf{s}) \end{aligned}$$

(where $S^{(2)}$ indicates differentiation with respect to the second argument)

$$\begin{aligned} &= \frac{S^{(2)}(r, f^a(\mathbf{s}), \tau)}{S(r, f^a(\mathbf{s}), \tau)} F\phi'((\xi^a)^2) \frac{\partial}{\partial s_i} \sum_i^D (\xi_i^a)^2 \\ &= \frac{S^{(2)}(r, f^a(\mathbf{s}), \tau)}{S(r, f^a(\mathbf{s}), \tau)} F\phi'((\xi^a)^2) \frac{2(s_i - c_i^a)}{(\sigma_i^a)^2} \end{aligned}$$

Optimal tuning properties

So,

$$\begin{aligned} J_{ij}^a(\mathbf{s}) &= \mathbb{E} \left[\left(\frac{S^{(2)}(r, f^a(\mathbf{s}), \tau)}{S(r, f^a(\mathbf{s}), \tau)} \right)^2 \right] 4F^2 (\phi'((\xi^a)^2))^2 \frac{(s_i - c_i^a)(s_j - c_j^a)}{\sigma^4} \\ &= A_\phi((\xi^a)^2, F, \tau) \frac{(s_i - c_i^a)(s_j - c_j^a)}{\sigma^4} \end{aligned}$$

where the function A_ϕ does not depend explicitly on σ .

Optimal tuning properties

We assumed neurons were independent \Rightarrow Fisher information adds. Approximate by integral over the tuning curve centres, assuming uniform density η of neurons.

$$\begin{aligned} J_{ij}(\mathbf{s}) &= \sum_a J_{ij}^a(\mathbf{s}) \\ &\approx \int_{-\infty}^{+\infty} dc_1^a \cdots \int_{-\infty}^{+\infty} dc_D^a \eta J_{ij}^a(\mathbf{s}) \\ &= \int_{-\infty}^{+\infty} dc_1^a \cdots \int_{-\infty}^{+\infty} dc_D^a \eta A_\phi((\xi^a)^2, F, \tau) \frac{(s_i - c_i^a)(s_j - c_j^a)}{\sigma^4} \end{aligned}$$

Change variables: $c_i^a \rightarrow \xi_i^a$

$$\begin{aligned} &= \int_{-\infty}^{+\infty} \sigma d\xi_1^a \cdots \int_{-\infty}^{+\infty} \sigma d\xi_D^a \eta A_\phi((\xi^a)^2, F, \tau) \frac{\xi_i^a \xi_j^a}{\sigma^2} \\ &= \frac{\sigma^D}{\sigma^2} \eta \int_{-\infty}^{+\infty} d\xi_1^a \cdots \int_{-\infty}^{+\infty} d\xi_D^a A_\phi((\xi^a)^2, F, \tau) \xi_i^a \xi_j^a \end{aligned}$$

Now, if $i \neq j$, integral is odd in both ξ_i^a and ξ_j^a , and thus vanishes. If $i = j$, then the integral has some value $D \cdot K_\phi(F, \tau, D)$, independent of σ . Thus,

$$J_{ii} = \sigma^{D-2} \eta D K_\phi(F, \tau, D)$$

and the total Fisher information is proportional to σ^{D-2} .

Optimal tuning properties

Thus optimal tuning width depends on the stimulus dimension through the interplay of two effects:

slope:	$f'(s) \propto \sigma^{-1} \Rightarrow J_a(s) \propto \sigma^{-2}$	<i>per cell</i>
number of cells:	$N(s) \propto \sigma^D \Rightarrow J(s) \propto \sigma^{D-2}$	<i>population</i>

- ▶ $D = 1$
 - ⇒ $\sigma \rightarrow 0$ (although a lower limit is encountered when the tuning width falls below the inter-cell spacing)
- ▶ $D = 2$
 - ⇒ J independent of σ .
- ▶ $D > 2$
 - ⇒ $\sigma \rightarrow \infty$ (actual limit set by valid stimuli).

Optimal tuning properties

Thus optimal tuning width depends on the stimulus dimension through the interplay of two effects:

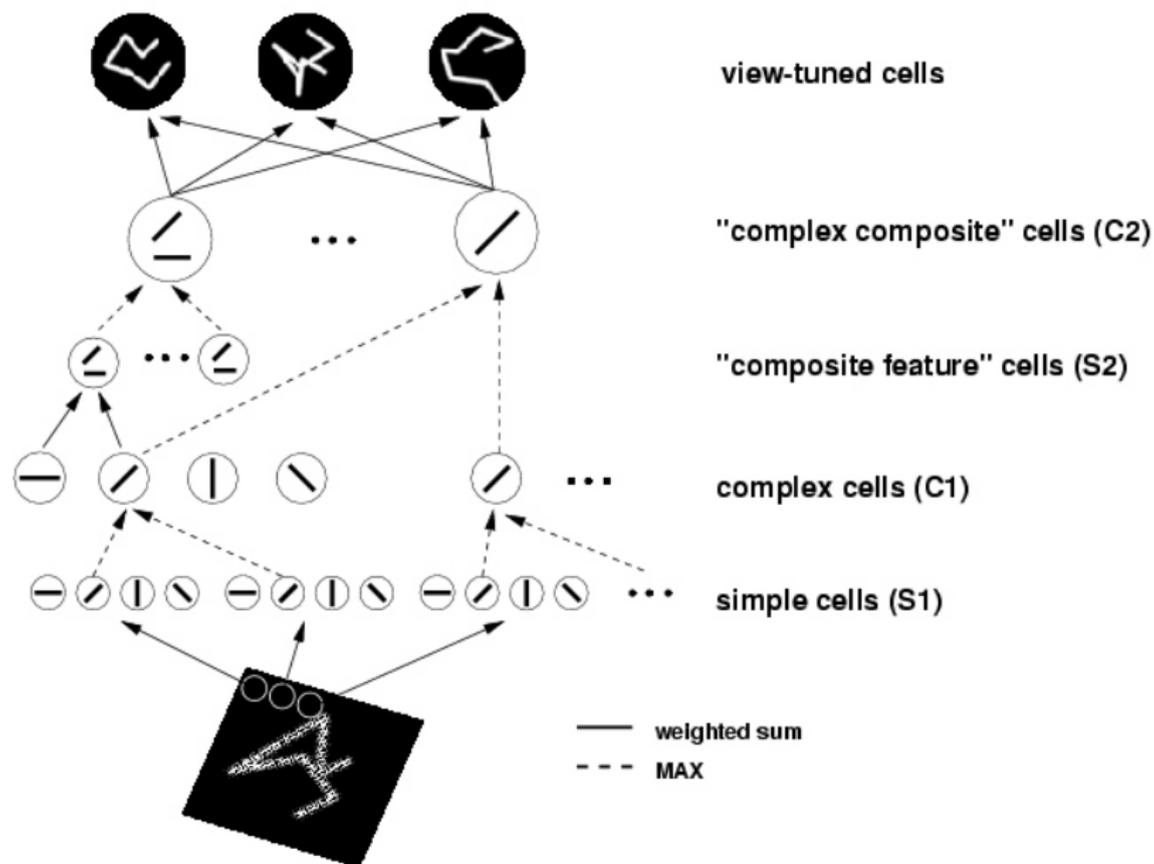
slope:	$f'(s) \propto \sigma^{-1} \Rightarrow J_a(s) \propto \sigma^{-2}$	<i>per cell</i>
number of cells:	$N(s) \propto \sigma^D \Rightarrow J(s) \propto \sigma^{D-2}$	<i>population</i>

- ▶ $D = 1$
 - ⇒ $\sigma \rightarrow 0$ (although a lower limit is encountered when the tuning width falls below the inter-cell spacing)
- ▶ $D = 2$
 - ⇒ J independent of σ .
- ▶ $D > 2$
 - ⇒ $\sigma \rightarrow \infty$ (actual limit set by valid stimuli).
- ▶ If circular symmetry is relaxed to allow different scales in each dimension for different cells then solution is a Cartesian code (narrow in one dimension, wide in others).
- ▶ Single-bump constraint is essential to analysis. Fisher information cannot address ambiguity between bumps.
- ▶ Single coded value – analysing multiple values or distributions is more complex.

Signal-processing paradigms

- 1 filtering
- 2 (efficient) coding
- 3 feature detection

Feature detection and representation



Trained network models

