Learning selectivity and tolerance computations from natural images

Michael U. Gutmann

University of Helsinki

michael.gutmann@helsinki.fi

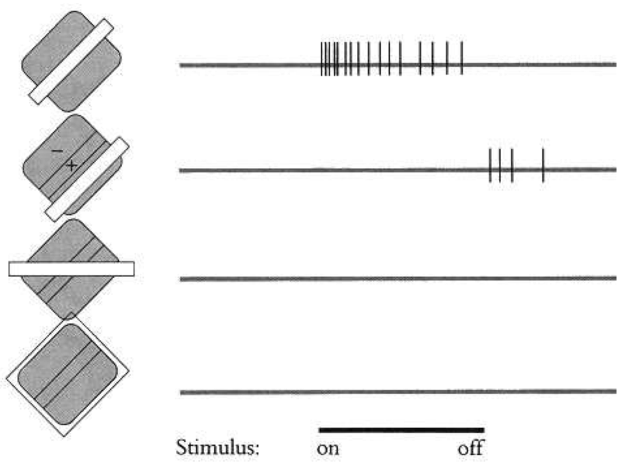
The presentation is based on the paper:

M. Gutmann and A. Hyvärinen,
A three-layer model of natural image statistics,
Journal of Physiology-Paris, 2013, in press.

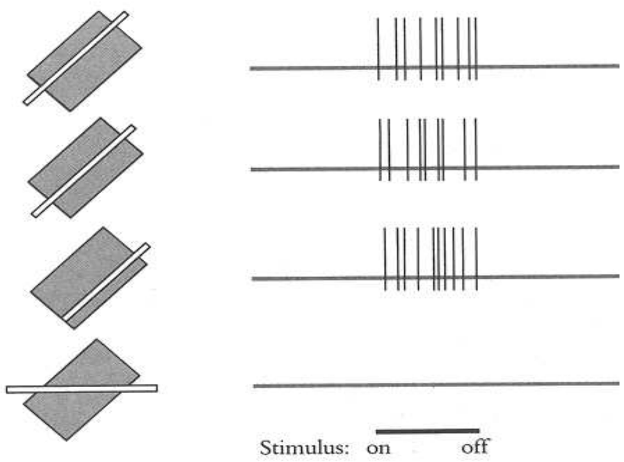
Introduction

Neural selectivity and tolerance

- Some "definitions" of neural selectivity and tolerance:
  - Neurons are selective to certain properties of the stimulus if their response changes strongly when the stimulus properties become present.
  - Neurons are tolerant to them if their response does not change much.
- Example for cells in the primary visual cortex:



Simple cells:
Selective to orientation and location of the bar



Complex cells:
Tolerant to exact location

Pairing of selectivity with tolerance

- It is easy to
  - build an artificial visual system that is highly tolerant (assign the same response to all stimuli)
  - build a highly selective system (nonzero response only if there is an exact match)
- Such visual systems would not be very useful.
- More useful are systems where the opposing requirements of tolerance and selectivity are joined together (“tolerant selectivity”).
- In the visual cortex, tolerant selectivity occurs at multiple levels
  - Higher-level example: Neurons involved in object recognition (see next slide)
  - Lower-level example: Complex cells where orientation selectivity is paired with tolerance to location.

Why should we care about tolerant selectivity?

- Tolerant selectivity on a higher level is thought to be important for reliable object recognition.

  (see for example: DiCarlo and Cox, Trends in Cognitive Sciences, 2007)
- Reason: For reliable object recognition, the neural activity must represent objects in a way that is
  - highly selective to shape, and
  - tolerant to identity-preserving transformations.
- Example: To recognize the face, we need to find visual clues that are
  - specific for the person at hand (selectivity), and
  - somewhat invariant to the facial expressions (tolerance).

(Figure taken from “Facial Expressions – A Visual Reference for Artists” by M. Simon.)

Emergence of higher-level tolerant selectivities (1/3)

- Basic hypothesis: higher-level tolerant selectivities emerge through a sequence of elementary selectivity and tolerance computations.
- Hypothesis goes back to Kunihiko Fukushima's "neocognitron", which is a multi-layer extension of Hubel& Wiesel's simple-cell, complex-cell cascade.

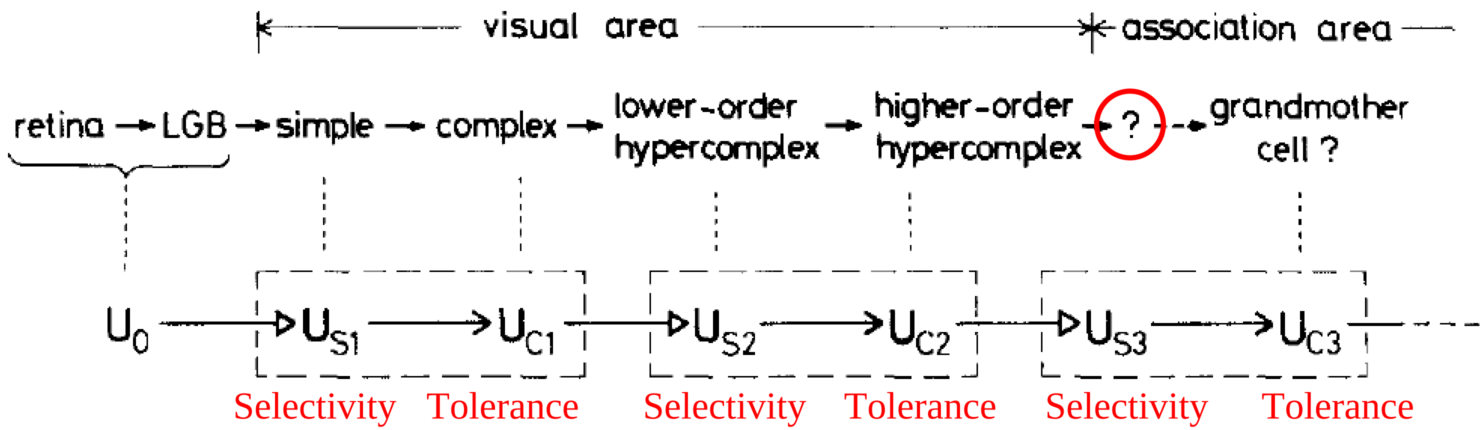
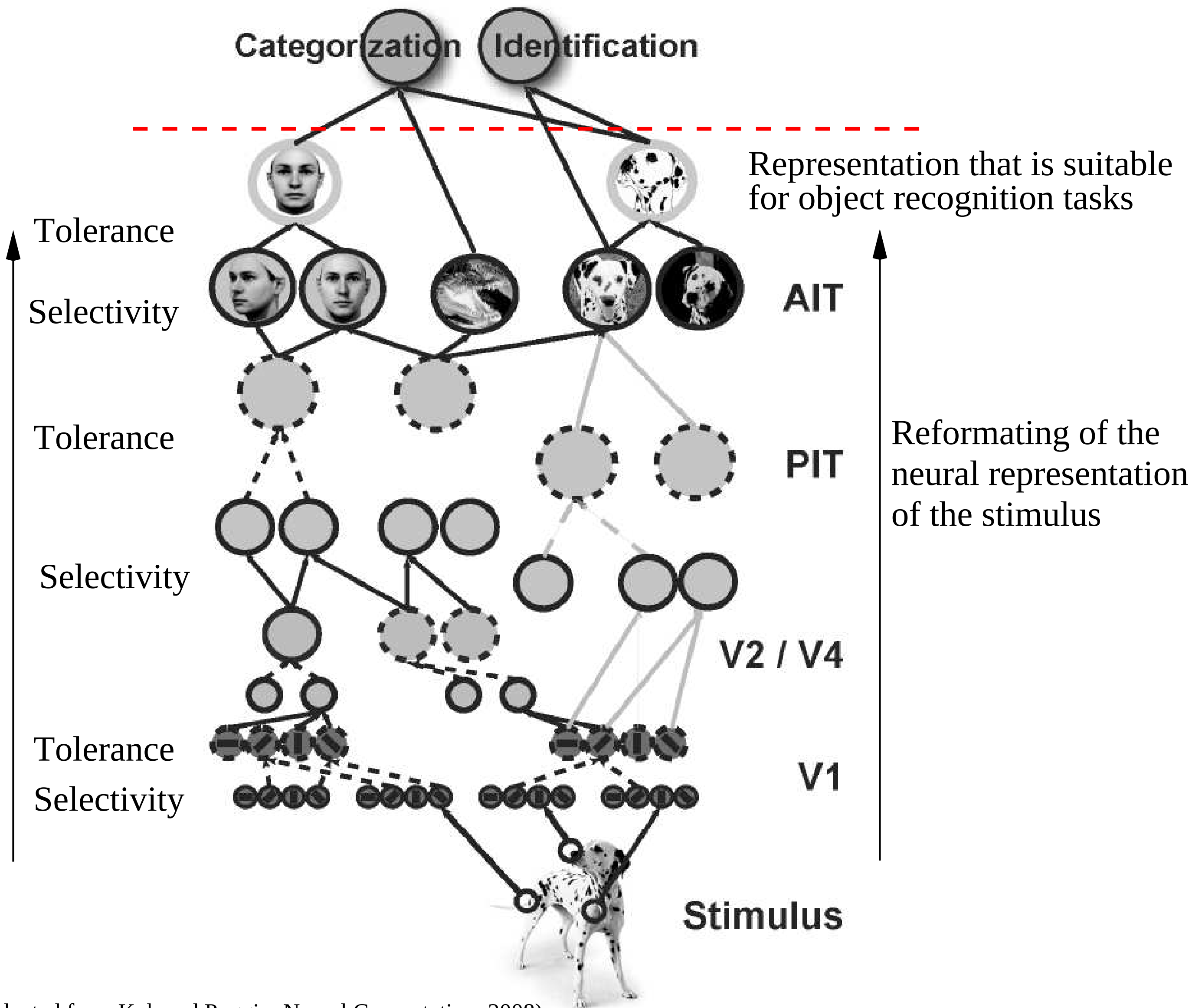


Figure adapted from "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position", Biol Cybernetics, 1980.

Emergence of higher-level tolerant selectivities (2/3)

Similar idea was put forward by Riesenhuber and Poggio, Nature 1999, and others.

Sequence of elementary computations



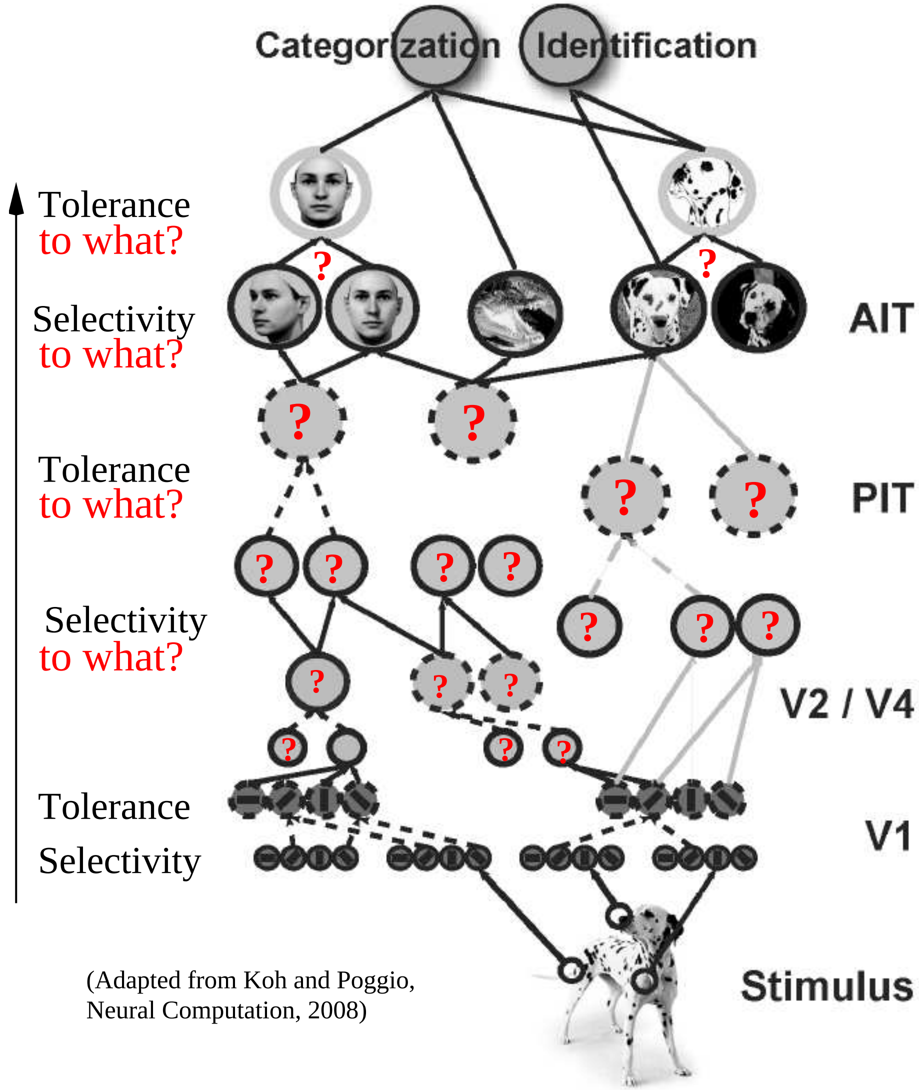
(Adapted from Koh and Poggio, Neural Computation, 2008)

Emergence of higher-level tolerant selectivities (3/3)

- There is (indirect) experimental evidence for an increase in selectivity and tolerance along the ventral pathway

  Rust and DiCarlo, J. Neurosci., 2010

- What remains poorly understood is the nature of the tolerance and selectivity computations along the hierarchy.



(Adapted from Koh and Poggio, Neural Computation, 2008)

Question asked and methodology

- Basic hypothesis:
  Higher level tolerant selectivities emerge through a sequence of elementary selectivity and invariance computations.
- Question asked:
  In a visual system with three processing layers, what should be selected and tolerated at each level of the hierarchy?
- Methodology: Learn the selectivity and invariance computations from images, using as few assumptions as possible.

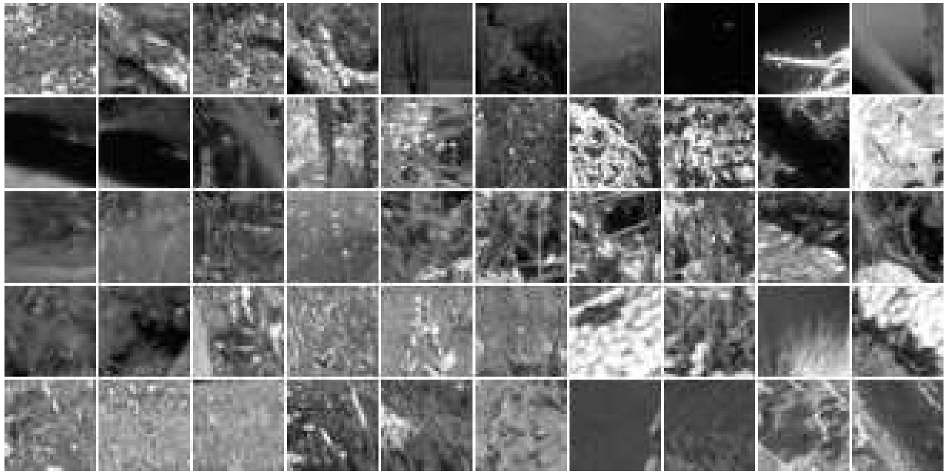
Image data and the three processing layers

Data

We learn the computations for two kinds of image data sets:

1. Image patches of size 32 by 32, extracted from larger images (left).
2. “Tiny images” dataset, converted to gray scale: complete scenes downsampled to 32 by 32 images (right)

   (Torralba et al, TPAMI 2008)



Preprocessing

- Preprocessing consists of three steps
  1. Remove DC component (average pixel value of each image)
  2. Normalize norm after whitening
  3. Reduce the dimension by PCA from $32 \cdot 32 = 1024$ to 600
- Preprocessing can be considered a form of luminance and contrast gain control, followed by low-pass filtering.

The three processing layers (1/2)

- Let $\mathbf{x} \in \mathrm{R}^{600}$ be a vectorized image after preprocessing.
- The three processing layers are:

$$y_i^{(1)} = \text{max}\left(\mathbf{w}_i^{(1)} \cdot \mathbf{x}, 0\right), \qquad i = 1 \ldots 600$$

$$y_i^{(2)} = \ln\left(\mathbf{w}_i^{(2)} \cdot (\mathbf{y}^{(1)})^2 + 1\right), \qquad i = 1 \ldots 100$$

$$\mathbf{z}^{(2)} = \text{gain control}\left(\mathbf{y}^{(2)}\right),$$

$$y_i^{(3)} = \text{max}\left(\mathbf{w}_i^{(3)} \cdot \mathbf{z}^{(2)}, 0\right), \qquad i = 1 \ldots 50$$

  Gain control is similar to the preprocessing: centering, normalizing the norm after whitening, possibly dimension reduction

- Parameters to be learned: feature vectors $\mathbf{w}_i^{(1)}$, $\mathbf{w}_i^{(2)}$, $\mathbf{w}_i^{(3)}$. They govern the computations of the three layers.
- Constraint: the $\mathbf{w}_i^{(2)}$ have nonnegative elements, $w_{ki}^{(2)} \geq 0$.

The three processing layers (2/2)

- First and third layer:
  Linear projection followed by rectification.
  This is a (very) simple model for the steady-state firing rate of neurons.
- Second layer:
  Functional form of the energy model for complex cells
  (Adelson, J Opt Soc Am, A, 1985)
- Linear projections/pooling patterns are not yet specified, but to be learned from the data.
- The specification of the three layer is in line with the multi-layer architecture shown in the introduction.
- The specification imposes only a weak constraint: The large number of free parameters allows to implement a large class of functions.

Learning

General considerations

- The parameters $\mathbf{w}_i^{(1)}$, $\mathbf{w}_i^{(2)}$ and $\mathbf{w}_i^{(3)}$ govern the computations of the three layers.
- We learn the parameters by fitting a probability density function (pdf) to the image data.
- The basic idea is that the overall activity of the feature outputs determines how likely an input image is.
- Why should the outputs be related to the pdf?
  - Object recognition is a classification problem. Knowing the probability density functions of the classes allows for optimal classification.
  - Deeper reason is the theory of Bayesian perception: The visual system is adapted to the properties of the world which it senses. It "knows" about the (statistical) properties of the visual stimuli.

Models for the pdfs of the images

- First, we learn the parameters of layers one and two. Keeping them fixed, learn the parameters of layer three.
- For layer one and two, we fit the pdf

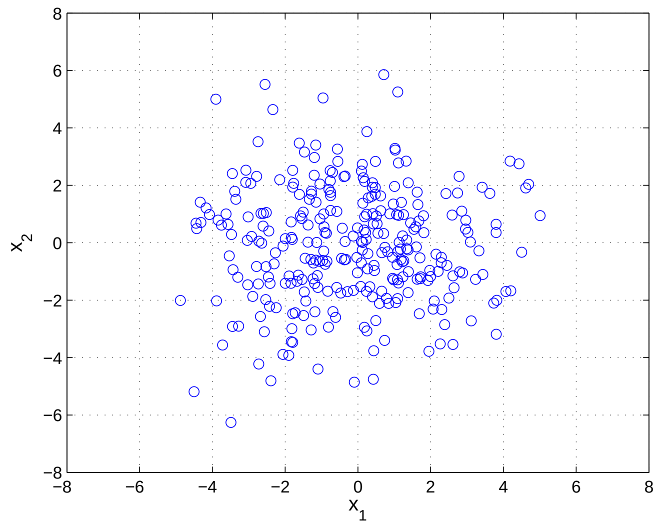
$$p(\mathbf{x}; \mathbf{w}_i^{(1)}, \mathbf{w}_i^{(2)}, b_i^{(2)}) \propto \prod_{i=1}^{100} \exp\left(f_{\text{th}}(y_i^{(2)} + b_i^{(2)})\right).$$

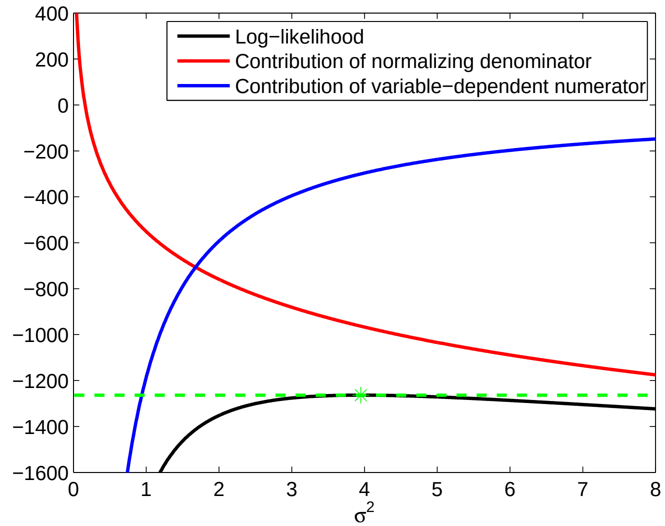
- For layer three, we fit the pdf

$$p(\mathbf{x}; \mathbf{w}_i^{(3)}, b_i^{(3)}) \propto \prod_{i=1}^{50} \exp\left(f_{\text{th}}(y_i^{(3)} + b_i^{(3)})\right).$$

- f_{th} is a smooth version of $\max(u, 0)$. Thresholding is a simple method for enforcing sparsity of the feature outputs.
- We do not know the partition functions (proportionality factors which depend on the parameters): Learning by maximization of the likelihood is not possible.

Importance of the partition function in MLE





- Model of $\mathbf{x} \in \mathbb{R}^2$: $p_m(\mathbf{x};\sigma^2) = \dfrac{\exp\left(-\frac{||\mathbf{x}||^2}{2\sigma^2}\right)}{2\pi\sigma^2}$
- Estimation of σ^2 by maximizing the log-likelihood ℓ of the observed data points $(\mathbf{x}_1 \ldots \mathbf{x}_{T_d})$

$$\ell(\sigma^2) = -T_d \ln(2\pi\sigma^2) + \frac{1}{\sigma^2}\sum_{t=1}^{T_d} \frac{-||\mathbf{x}_t||^2}{2}$$

- Without knowing the partition function $Z(\sigma) = 2\pi\sigma^2$, maximum likelihood estimation (MLE) is not feasible.

Noise-contrastive estimation

(Gutmann and Hyvärinen, Journal of Machine Learning Research, 2012)

- Purpose: learn parameters $\boldsymbol{\theta}$ of a pdf $p_{\boldsymbol{\theta}}$ when you do not know the partition function.
- Intuition: Learn the differences between the data and auxiliary "noise" whose properties you know. Deduce from the differences the properties of the observed data.
- More concrete:
  1. Choose a random variable $\mathbf{z}$ with known pdf $p_{\mathbf{z}}$ where sampling is easy.

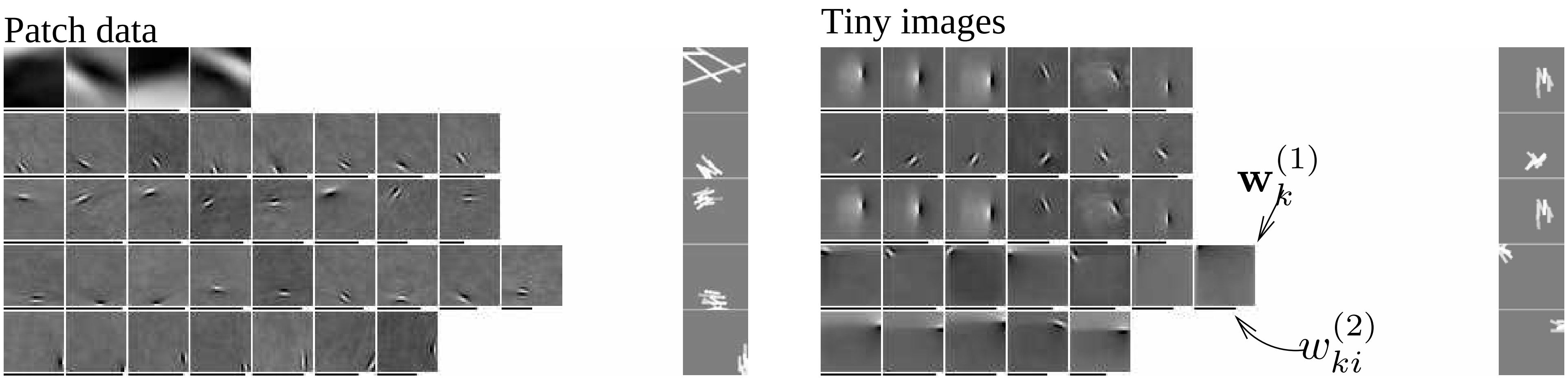
     Here: Uniform distribution in the sphere where the data is defined
  2. Obtain an auxiliary sample of $\mathbf{z}$ ("noise").
  3. Perform logistic regression on the data and the auxiliary "noise"; use the ratio $p_{\boldsymbol{\theta}}/p_{\mathbf{z}}$ in the regression function.
- The procedure provides a consistent estimator of $\boldsymbol{\theta}$.

Results

Selectivity computation in layer one and two

$$y_i^{(2)} = \ln\left(\sum_k w_{ki}^{(2)} (\mathbf{w}_k^{(1)} \cdot \mathbf{x})^2 + 1\right)$$

- The first-layer features $\mathbf{w}_k^{(1)}$ are Gabor-like. (“simple cells”, similar to prev work)
- After learning, the second-layer weight vectors $\mathbf{w}_i^{(2)}$ are sparse:
  - For patch data, 97% of the elements $w_{ki}^{(2)}$ in the vectors $\mathbf{w}_i^{(2)}$ are smaller than the 10^9 fraction of their maximal elements.
  - For tiny images, it is 95%.
- The second layer weights $w_{ki}^{(2)}$ pool similarly oriented and localized first-layer features together. Selectivity to localized oriented image structure.



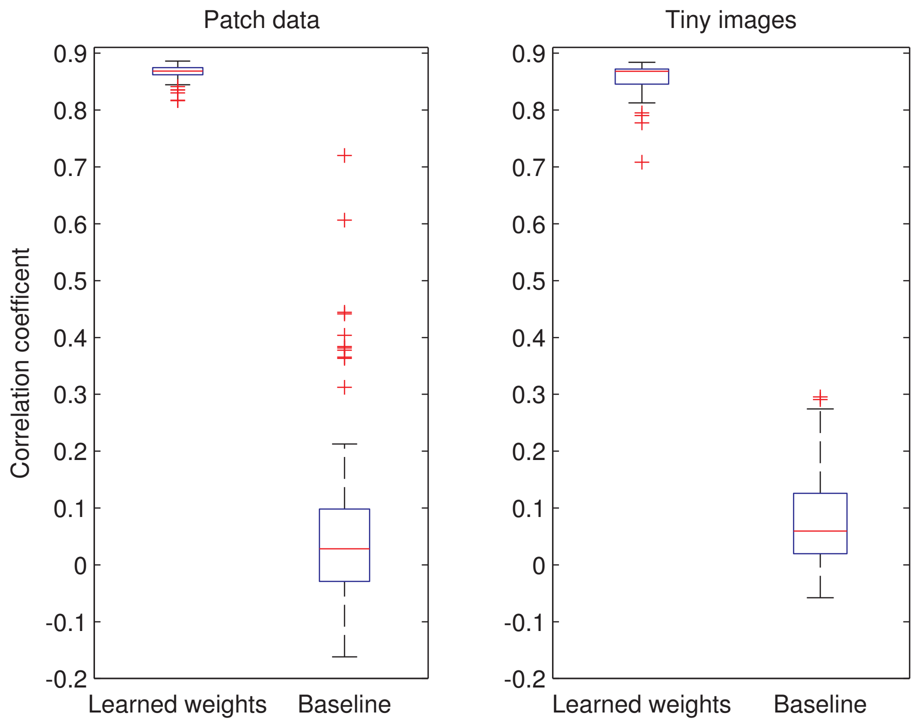
Random subset of features and their icons

Tolerance computation in layer two

- Processing on the second layer: max-like computation over selected first-layer feature outputs $y_k^{(1)}$.
- The learned weights $w_{ki}^{(2)}$ are indices that select over which first-layer outputs to take the max operation.
- Leads to tolerance to exact localization of the stimulus. ("complex cells", similar to prev work)

Distribution of the correlation coefficients

corr$(y_i^{(2)}, \max_{k:w_{ki}^{(2)}>\epsilon_i} |y_k^{(1)}|)$



Layer three results: example unit for patch data

$$\mathbf{z}^{(2)} = \text{gain control}\left(\mathbf{y}^{(2)}\right) \qquad y_i^{(3)} = \max\left(\mathbf{w}_i^{(3)} \cdot \mathbf{z}^{(2)}, 0\right)$$

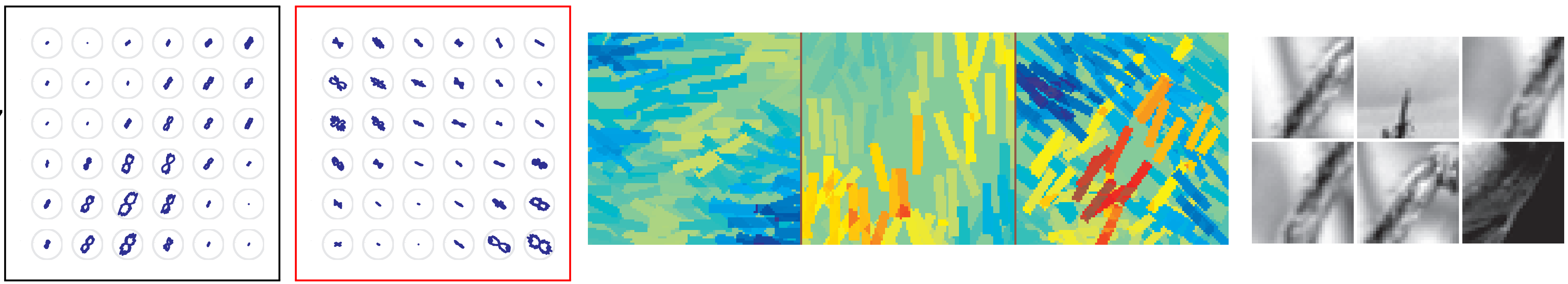
- Black frame: space-orientation receptive field. Visualizes the response to local gratings of different orientations.

  (Anzai et al, Neurons in monkey visual area V2 encode combinations of orientations, Nat Neurosci, 2007)

- Red frame: "inhibitory" space-orientation receptive field. Shows the location and orientation of local gratings which inhibit the units most.
- Optimal 2nd layer activity: Visualizes the activity pattern of layer two which leads to the largest output $y_i^{(3)}$. Red: second-layer units more activated than the population average. Blue means less activity, and green average activity.

Receptive field (RF) | Inhibitory RF | Optimal 2nd layer activity (horizontal, vertical, diagonal) | Strongly activating images

07



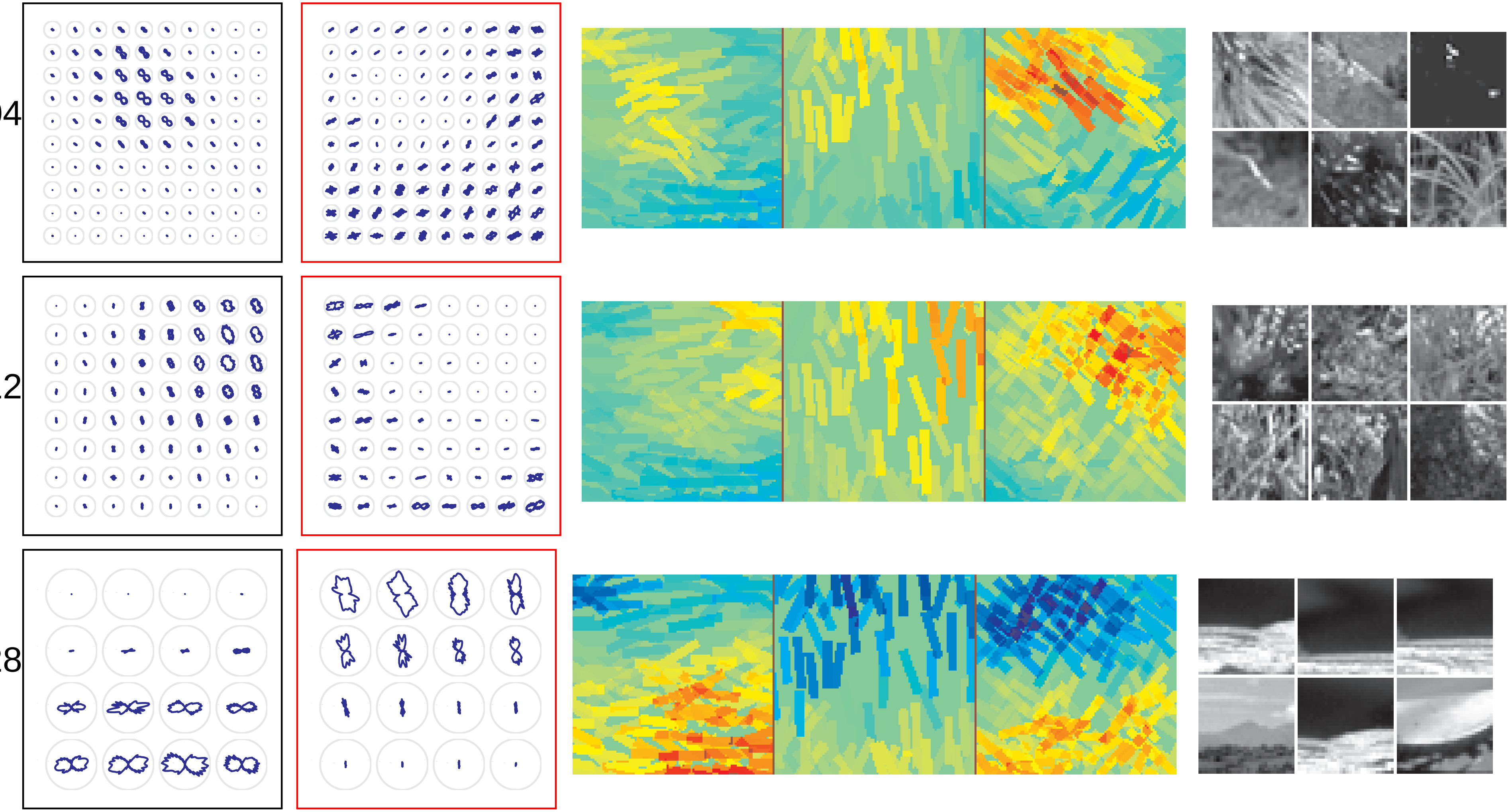
Layer three results: more examples for patch data

Receptive field (RF) | Inhibitory RF | Optimal 2nd layer activity (horizontal, vertical, diagonal) | Strongly activating images

04

12

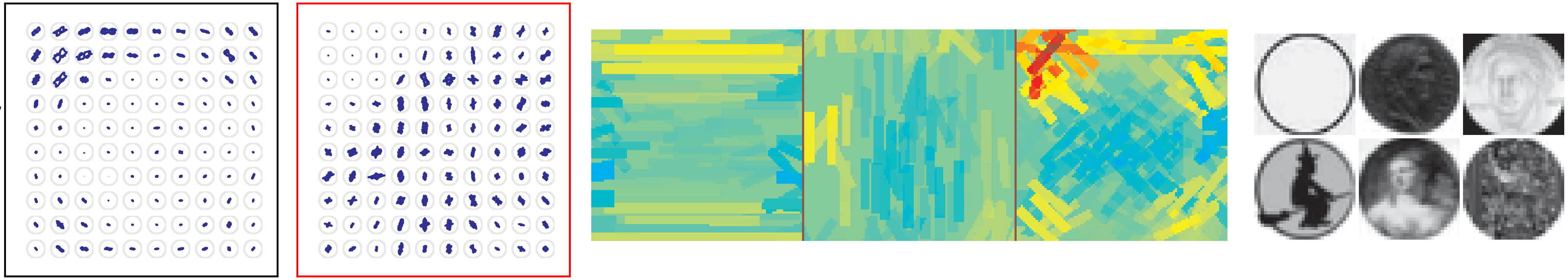
28



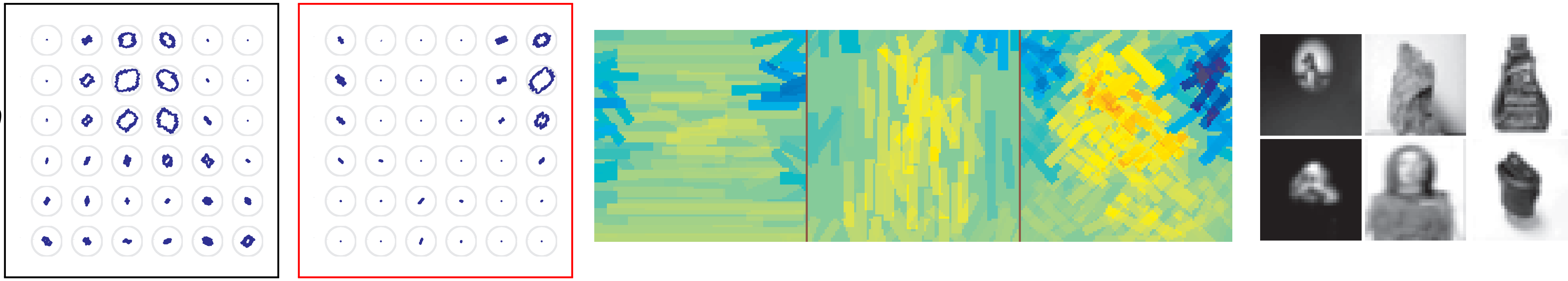
Layer three results: examples for tiny image data

Receptive field (RF) | Inhibitory RF | Optimal 2nd layer activity (horizontal, vertical, diagonal) | Strongly activating images

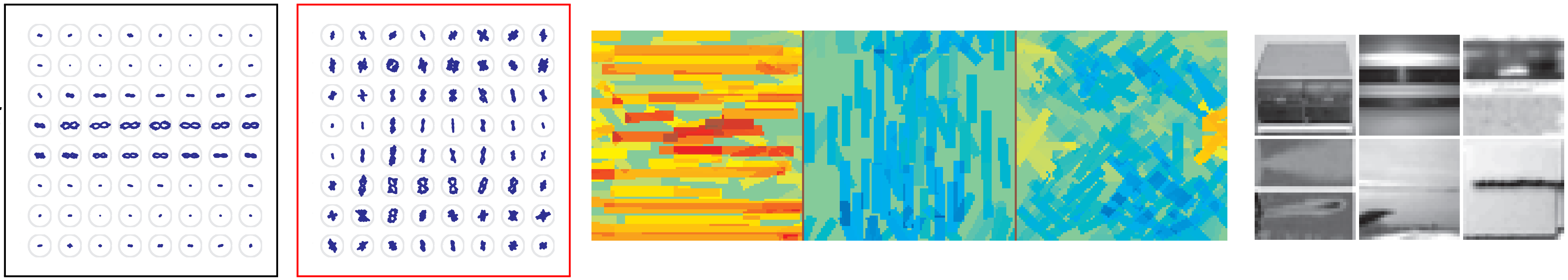
07



09



35

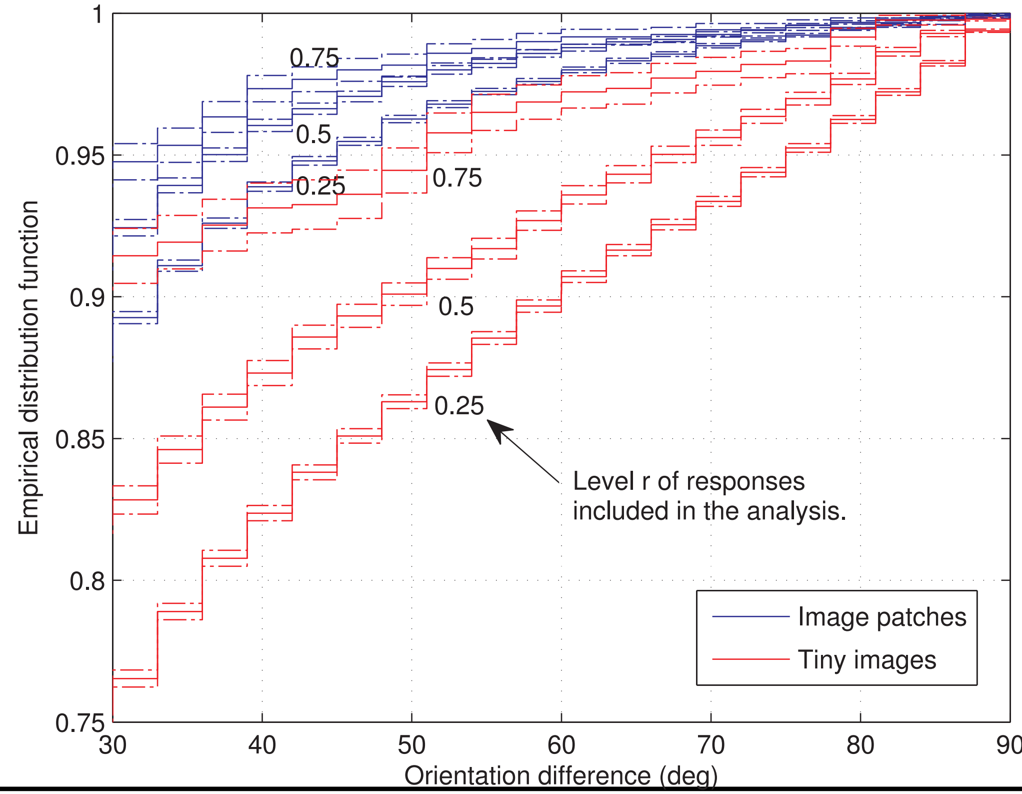


Qualitative observations

- Receptive fields are well structured and often localized.
- Emergence of non-classical receptive fields.
- For tiny images, the receptive fields are more inhomogeneous than for patch data.
- Excitatory and inhibitory gratings form large angles (orientation inhibition).
- Selectivity on the third layer:
  - For patch data: longer contours and texture
  - For tiny images: longer contours, curvatures

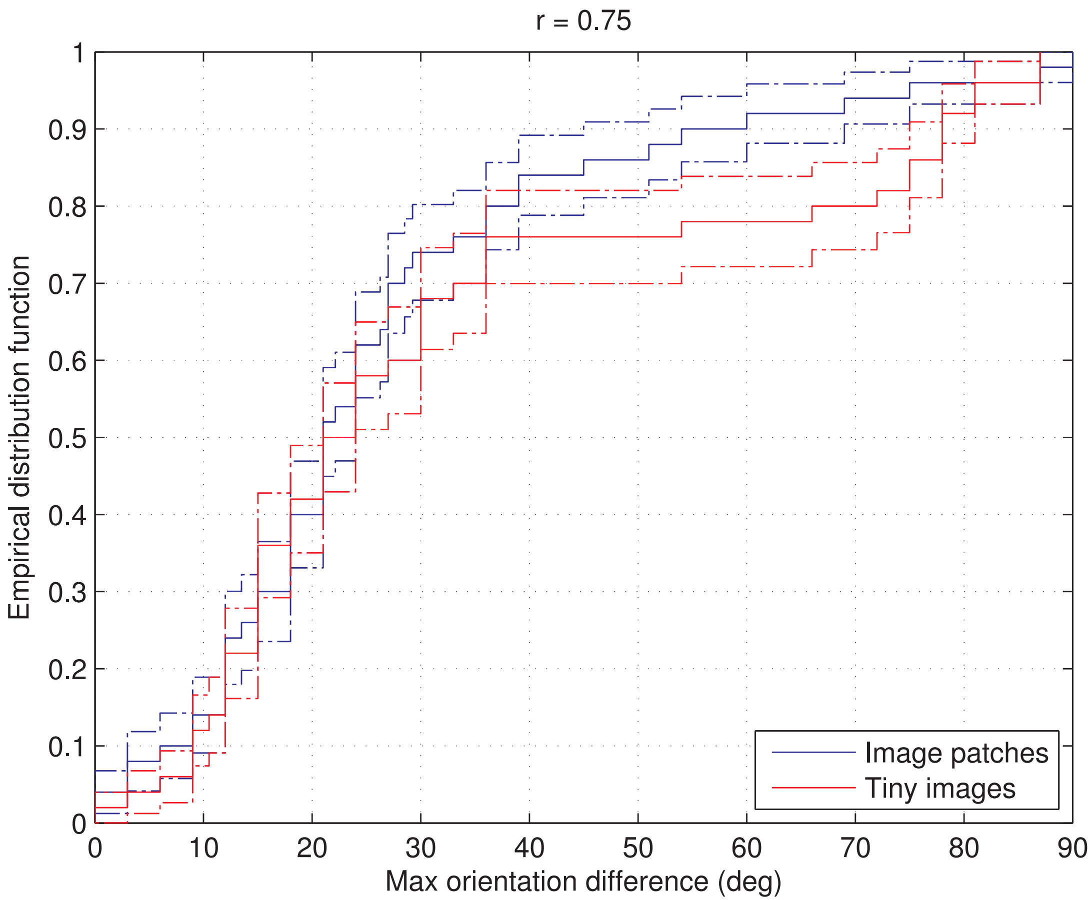
Population analysis of homogeneity (1/2)

- The figure shows the empirical distribution functions for the difference in orientation tuning within a receptive field (RF).
- Locations within a receptive field that yielded a response less than r times the maximal response were excluded.
- Tiny images tend to give more often inhomogeneous RFs than patch data.



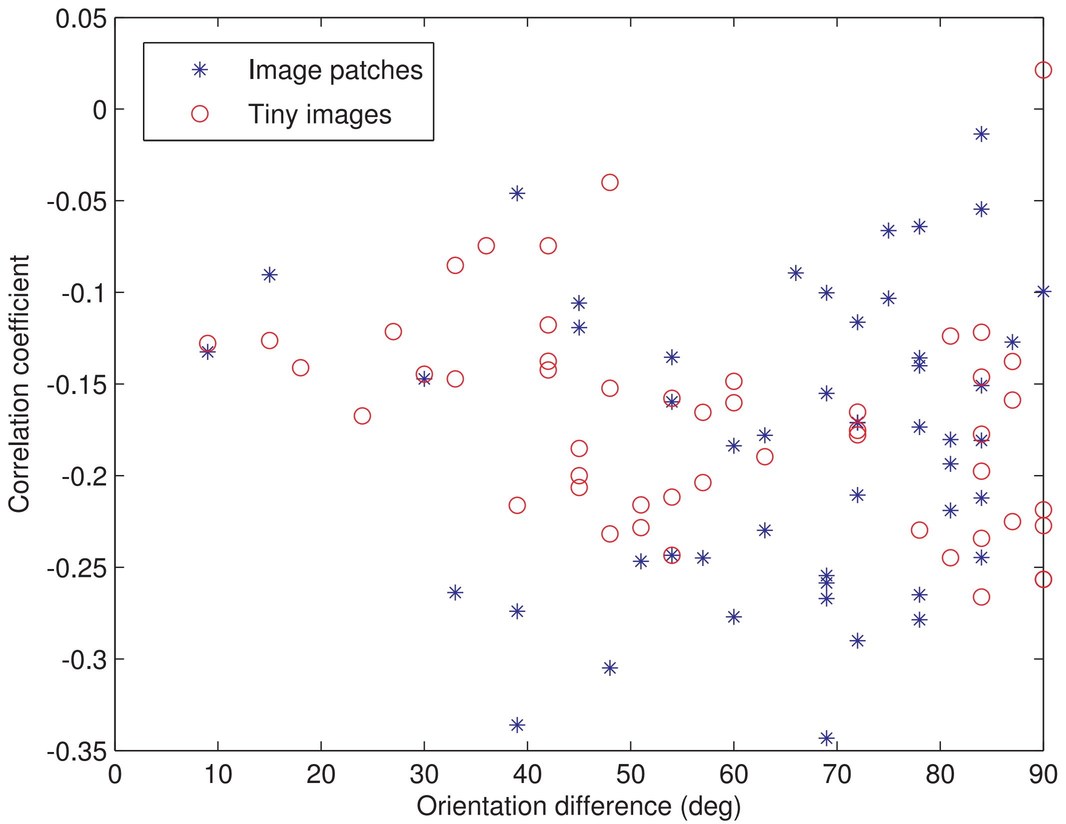
Population analysis of homogeneity (2/2)

- *Maximal* difference δ in orientation tuning within a RF:
  $\delta < 30^\circ : 70\%; \quad \delta > 60^\circ : 10\%$ (patches), 20% (tiny images)
- Experimental findings (V2 in Macaque monkeys):
  - Anzai, 2007: $\delta < 30^\circ : 60 - 70\%; \quad \delta > 60^\circ : 30\%$
  - Tao, 2012: $\delta < 30^\circ : 80\%; \quad \delta > 60^\circ : 5\%$



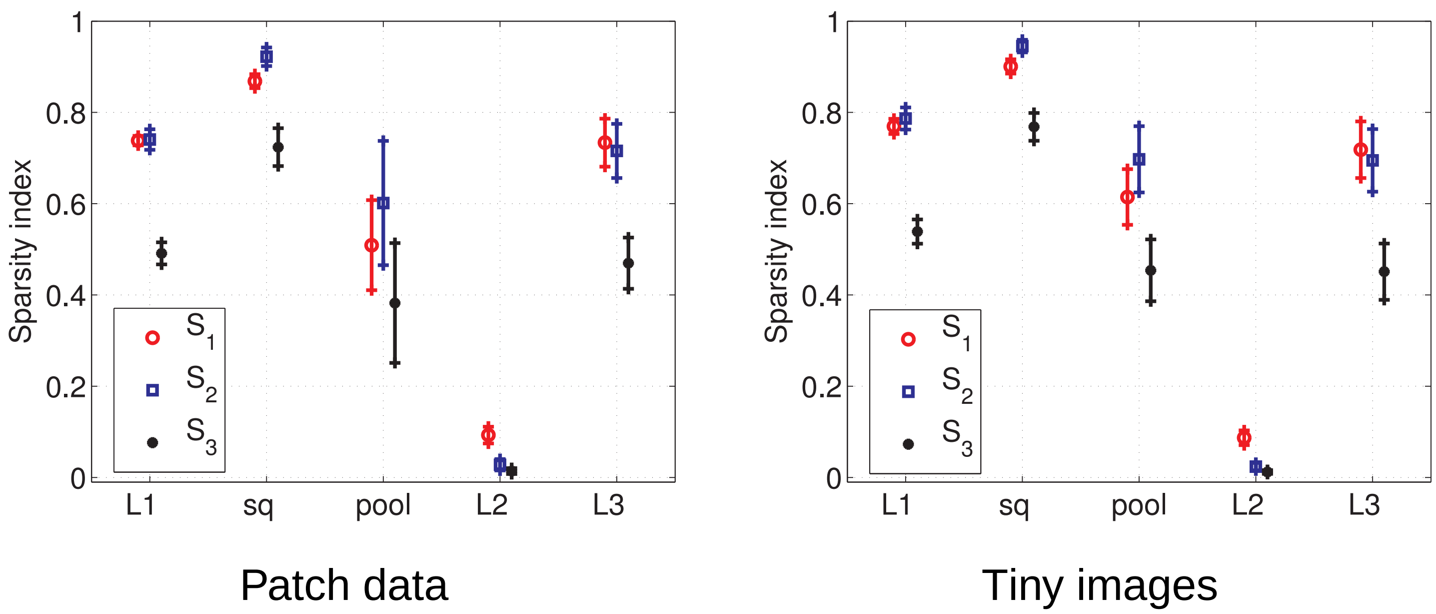
Population analysis of orientation inhibition

- We identified the most and least activated second-layer unit in the optimal activation pattern.
- The plot shows their correlation coefficient and the difference in orientation tuning.
- They are negatively correlated and tend to form large angles.



Lifetime sparsity across the three layers

- We use three different indices S_1, S_2, S_3 to measure lifetime sparsity (see paper for details).
- Sparsity on layer one ("L1") and three ("L3") are about the same.
- Squaring ("sq") increases sparsity. Pooling ("pool") and taking the logarithm ("L2") reduces it.
- Iterating between selectivity and tolerance computations balances sparsity (no net increase).



Conclusions

What the talk was about

- Basic hypothesis of our work is:
  Higher level tolerant selectivities emerge through a sequence of elementary selectivity and invariance computations.
- We asked:
  In a visual system with three processing layers, what should be selected and tolerated at each level of the hierarchy?
- Our approach was:
  Learn the selectivity and invariance computations from images, using as few assumptions as possible.

What we found

- Computations in the first two layers are in line with previous research. For both patch data and tiny images:
  - First layer: Emergence of selectivity to Gabor-like image structure ("simple cells")
  - Second layer: Emergence of tolerance to exact orientation or localization of the stimulus ("complex-cells")
- New kind of features on the third layer:
  - Patch data: Emergence of selectivity to longer contours and, to some extent, texture.
  - Tiny images: Emergence of selectivity to longer contours and, to some extent, curvature.
  - The features are mostly homogeneous, in line with experimental results. They are more inhomogeneous for tiny images than for patch data.
  - Emergence of (orientation) inhibition to facilitate the selectivity computations.
- No net increase of sparsity as we go from layer one to layer three.

What could be done next

- We could analyze more precisely the pooling pattern on the second layer:
  - What is the "radius" over which the first-layer units (Gabor-like features) are pooled together?
  - What is the range of the preferred orientations over which the pooling is happening?
- The input of the third layer was restricted to second-layer outputs. Would it be helpful to also feed first-layer outputs to the third layer?
- We could use the learned features for object recognition. The tiny images dataset includes the CIfAR dataset which is often used to test classifiers.
- We could learn the computations on layer four, five, ...