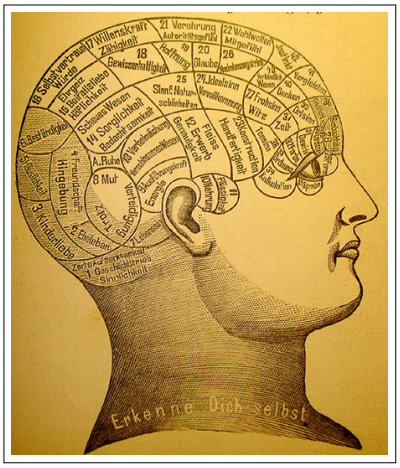
Visual Perception

— Lecture Notes —

Laurenz Wiskott
Institut für Neuroinformatik
Ruhr-Universität Bochum, Germany, EU

15 May 2024

The Human Brain



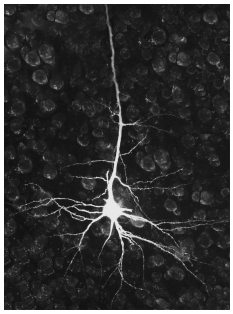
("Phrenologie", Friedrich Eduard Bilz, 1894)

2/48

If applicable, core text and formulas are set in dark red, one can repeat the lecture notes quickly by just reading these; ♦ marks important formulas or items worth remembering and learning for an exam; ◊ marks less important formulas or items that I would usually also present in a lecture; + marks sections that I would usually skip in a lecture.
You can also browse through this material on the internet.

The Human Brain

(WWW, Wikipedia, 2006; from NIH)

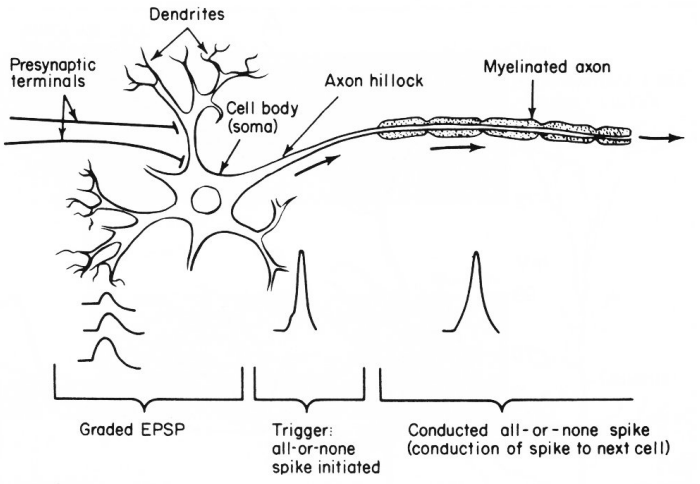


(2007, http://www.jimpryor.net/.../neuron.jpg)

- approx. 100.000.000.000 neurons
- approx. 100.000.000.000.000 synapses

3/48

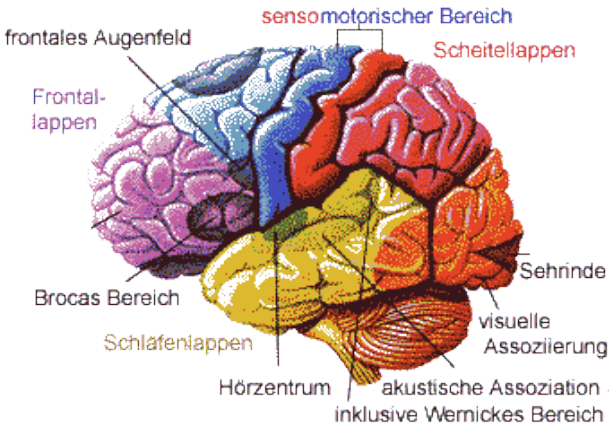
A Neuron



(Churchland & Sejnowski, 1993; from Thompson, 1967)

4/48

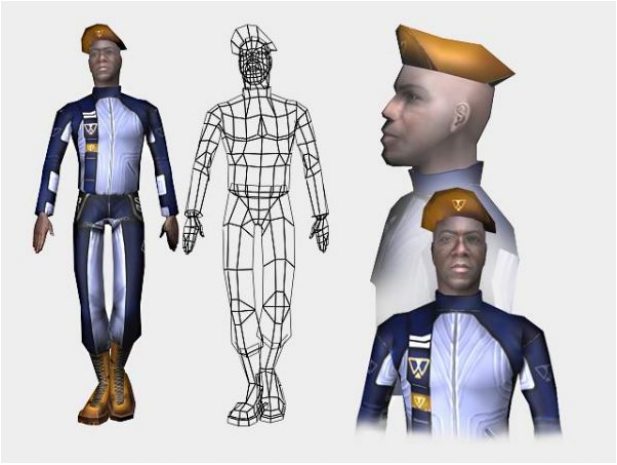
The Human Brain



(WWW, [werner.stangl]s arbeitsblätter, 2006; from GEO)

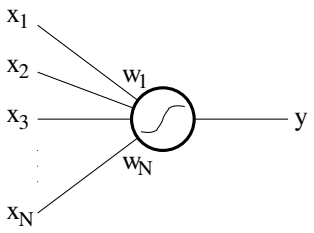
5/48

Computational Models



6/48

Nonlinear model neuron



$$y = \sigma\left(\sum_{i=1}^{N} w_i x_i\right) = \sigma(\boldsymbol{w}^T \boldsymbol{x})$$

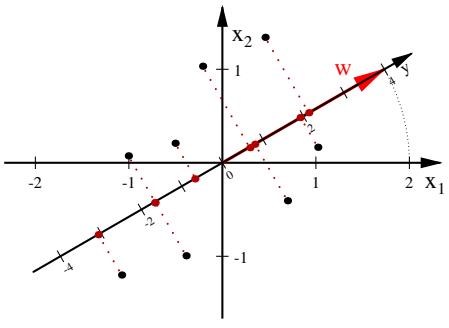
7/48

Linear Model Neuron

A linear model neuron is described by:

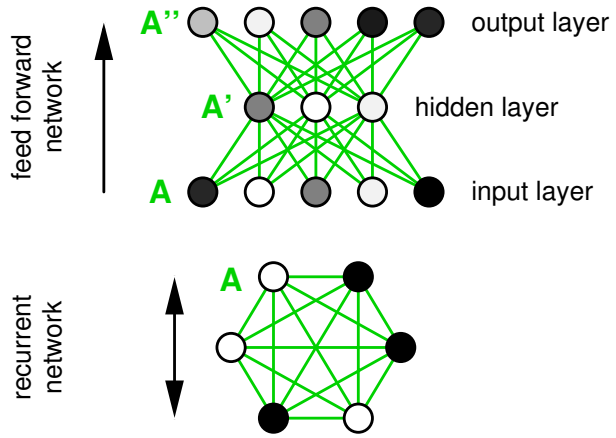
$$y = \sum_{i=1}^{N} w_i x_i = \boldsymbol{w}^T \boldsymbol{x}\,. \tag{1}$$

This corresponds to a projection of the data onto the axis given by $\boldsymbol{w}$ scaled with $|\boldsymbol{w}|$.



8/48

Networks



9/48

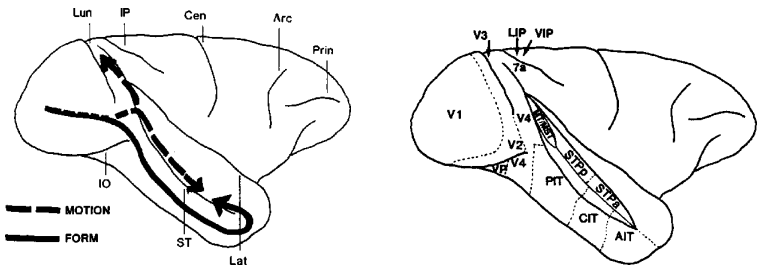
Physiological Receptive Fields

Image: (Alphab.fr, 2007, Wikipedia, © CC BY 2.0, URL)

10/48

Image: (Alphab.fr, 2007, Wikipedia, © CC BY 2.0, URL)[3.1]

Visual Pathways and Areas (Macaque)



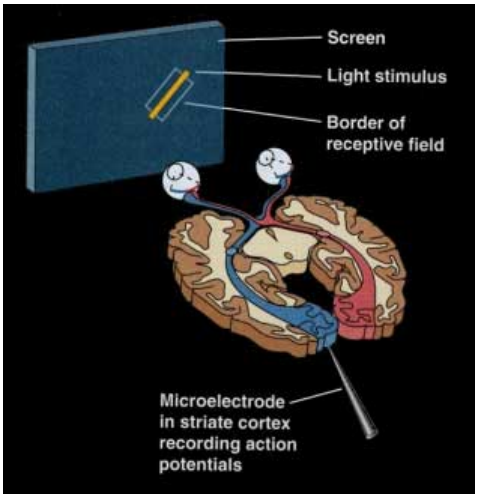
(Oram & Perrett, 1994, Neur. Netw. 7(6-7):945–972)

33/48

Visual input from the retina gets projected through the lateral geniculate nucleus (LGN, not shown here), which is subcortical, to the primary visual cortex (V1, 'V' and '1' indicating 'visual' and 'primary' respectively). From there it goes through V2 and V4 to the inferior temporal cortex (IT), which can be further subdivided into posterior (PIT), central (CIT), and anterior (AIT) IT. IT is thought to be instrumental for object recognition, while V1–V4 extract more elementary features. This path is referred to as the *what*-path, because it tells us what we see.

Another path goes through V2 and V4 to areas MT/MST, which are particularly responsive to motion. This path has been termed the *where*- or *how*-path, because it is thought to tell us where the objects are or how we can handle them, e.g. grasp them. The paths converge in the posterior (STPp) and anterior (STPa) superior temporal polysensory area. Cells in STPa, for instance, have been found to be sensitive to body motion, such as walking. Figure by Oram and Perrett (1994).

Measuring Receptive Fields

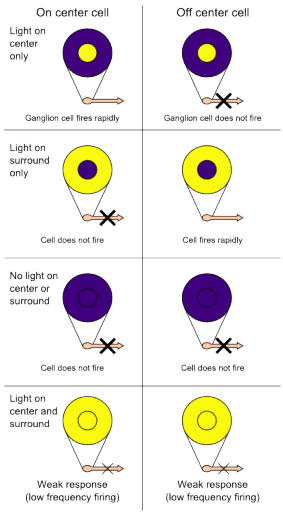


(http://www.medinfo.ufl.edu/... 2001-10-29 (outdated))

12/48

It is possible to make quite detailed measurements of response properties of single cells in awake or anaesthetized animals. To measure visual receptive fields, one typically places an animal in front of a computer monitor, let the animal fixate the center of the screen, presents visual stimuli, and simultaneously records extracellularly from individual neurons. Visually driven neurons usually respond only to stimuli within a particular region, which is referred to as the receptive field. They also only respond to particular shapes or features, such as orientation, color, or motion. One says the cell has a tuning for one or several of these features. One also speaks of such cells as feature detectors.

Center-Surround Cells in Retina and LGN

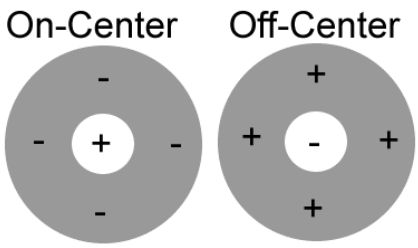


(Delldot/Xoneca, 2005/2008, Wikimedia, © CC0, URL)

13/48

Cells in retina and LGN (lateral geniculate nucleus, which is a relay station between retina and cortex, have center-surround receptive fields. Some of them respond best to a bright spot on a dark background (on-center cell/stimulus), others to a dark spot on a bright background (off-center cell/stimulus). They do not respond well to full field stimuli (dark or bright). Interestingly, an on-center cell gives a response if an off-center stimulus disappears (release-of-inhibition reponse) and the other way around.

Assumed Connectivity

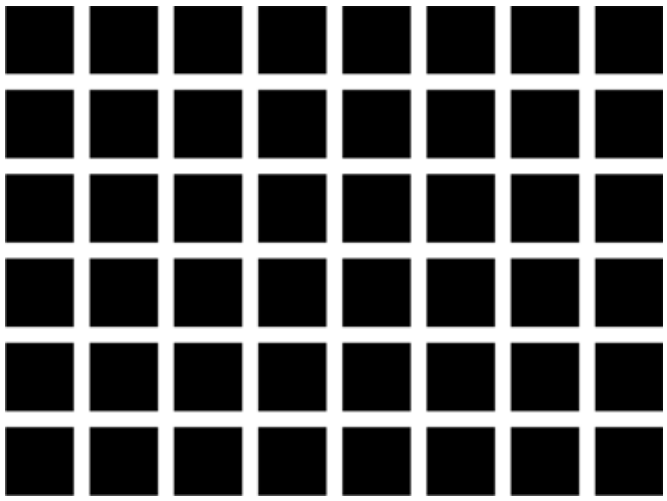


(Paskari, 2007, Wikipedia, © CC0, URL)

14/48

Center-surround receptive fields can be set up easily by a corresponding feedforward connectivity. For an on-center cell, connections coming from the center of the recptive field would be excitatory and those coming from the surround would be inhibitory. For off-center cell it would be the other way around. A canonical way of plotting such a receptive field is to plot the excitatory and inhibitory regions in the visual field (see lower left). Such receptive fields are conceptually linear.

Hermann Grid

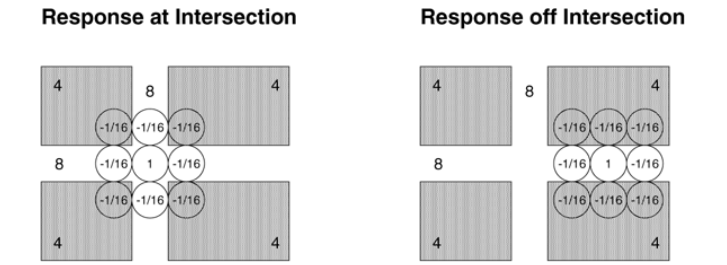


(`http://www-psych.stanford.edu/` 2001-10-29 (outdated))

15/48

A Hermann grid is a white square grid of appropriate size on a black background. If you look at it you might notice that the white looks darkened somehow at the crosses, but only in the periphery and not at the point of fixation. This is an optical illusion that can be explained with the center-surround receptive fields in the retina or LGN.
(`http://www-psych.stanford.edu/` 2001-10-29 (outdated))

Hermann Grid



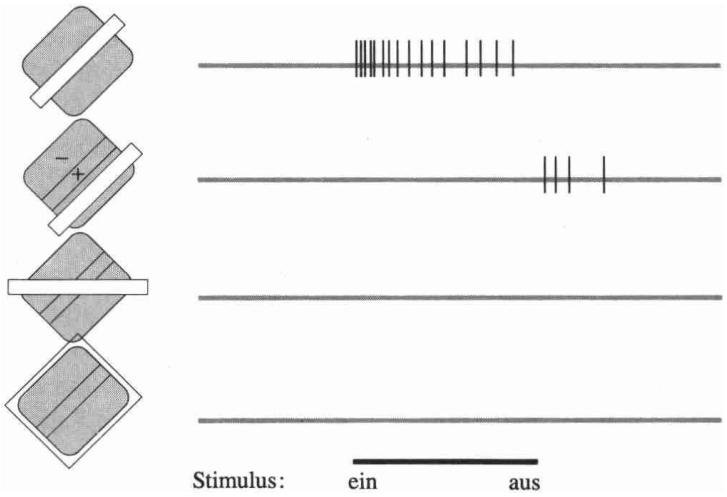
$8(1 + 4(-1/16)) + 4(4(-1/16)) = 5$ $8(1 + 2(-1/16)) + 4(6(-1/16)) = 5.5$

(http://www-psych.stanford.edu/ 2001-10-29 (outdated))

16/48

To explain the Hermann grid illusion, place a simple center-surround receptive field at a cross and at a line. By adding the product of the image gray value with the receptive field weight one gets a somewhat lower response at a cross (value 5) than a line (value 5 1/2) due to the stronger surround inhibition. This effect depends on the width of the stripes compared to the size of the receptive fields. In the fovea, i.e. around the point of fixation, the receptive fields are very small and the Hermann grid illusion cannot be observed with a coarse grid.
(http://www-psych.stanford.edu/ 2001-10-29 (outdated))

Simple Cells in Primary Visual Cortex (V1)



(Hubel, 1989, Auge und Gehirn: Neurobiologie des Sehens, Fig. 4.10)

17/48

In the first cortical area dedicated to visual processing, referred to as primary visual cortex or, for short, V1, one mainly distinguishes between two types of cells based on their receptive fields: simple cells and complex cells. Both cell types prefer oriented stimuli, such as bars and stripes, but simple cells care about the exact location of the stimuli while complex cells don't. Thus, in some sense complex cells have a higher degree of invariance than simple cells.
(Hubel, 1989, Auge und Gehirn: Neurobiologie des Sehens, Fig. 4.10)

Complex Cells in Primary Visual Cortex (V1)

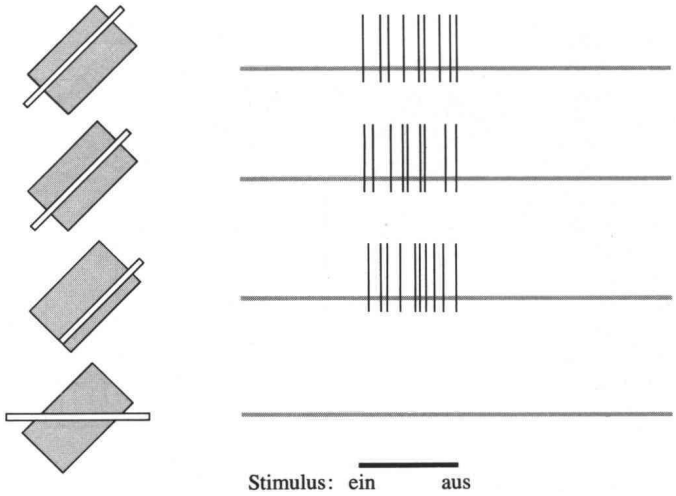
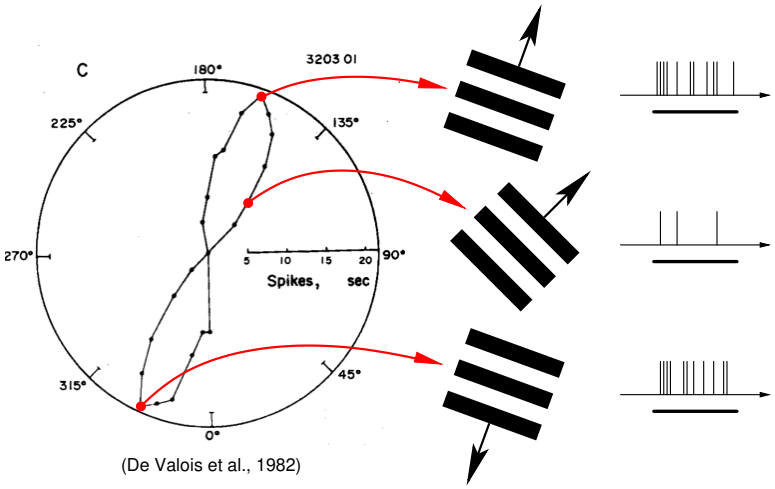


Figure: (Hubel, 1989, Auge und Gehirn: Neurobiologie des Sehens, Fig. 4.13)

18/48

In the first cortical area dedicated to visual processing, referred to as primary visual cortex or, for short, V1, one mainly distinguishes between two types of cells based on their receptive fields: simple cells and complex cells. Both cell types prefer oriented stimuli, such as bars and stripes, but simple cells care about the exact location of the stimuli while complex cells don't. Thus, in some sense complex cells have a higher degree of invariance than simple cells.
Figure: (Hubel, 1989, Auge und Gehirn: Neurobiologie des Sehens, Fig. 4.13)

Orientation Tuning



(De Valois et al., 1982)

19/48

Simple and complex cells usually have preferences for certain orientations. This can be measured by presenting gratings of different orientation to the cell (or rather the animal) and recording the corresponding neural responses. Normally, drifting gratings are used, because the cells respond stronger to moving stimuli.
The responses to different orientations can be conveniently visualized in a polar plot. One simply plots the firing rate in radial direction as a function of orientation in azimuthal direction. The graph shows a standard orientation tuning with one preferred orientation at about 160°, which appears here as two lobes in 180° distance due to the two different drifting directions. Since the two lobes have same size, the cell does not have a preference for a particular drifting direction.

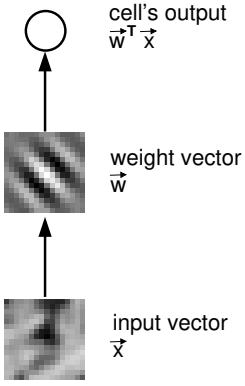
Models of Visual Receptive Fields

Figure: (http://www.uwec.edu/geography/Ivogeler/w111/Images/srilankatopsheet.jpg 2005-11-16 (outdated))

20/48

Figure: (http://www.uwec.edu/geography/Ivogeler/w111/Images/srilankatopsheet.jpg 2005-11-16 (outdated))

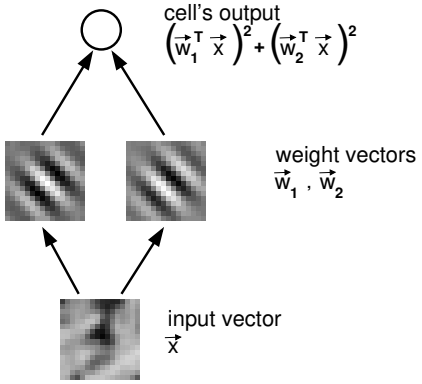
Standard Simple-Cell Model



21/48

The standard model of a simple cell is simply a linear filter having the shape of a wavelet. The response is the inner product $\boldsymbol{w}^T\boldsymbol{x}$ (sum over pointwise products) between the filter (weight vector $\boldsymbol{w}$) and the image (input vector $\boldsymbol{x}$). Such a filter is strongly excited by a bar or grating of the correct frequency (in case of a grating), orientation, and exact position. If the grating is shifted in phase by 180°, or in position by one wavelength orthogonal to the wave fronts, the model unit gives a strong negative response.

Standard Complex-Cell Model



22/48

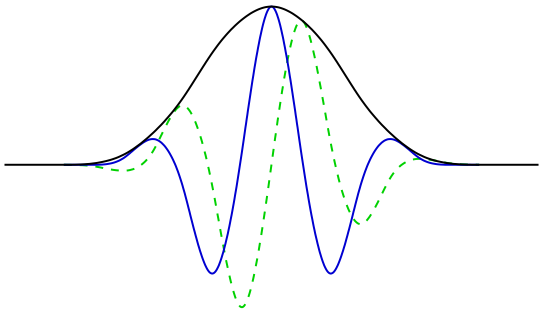
The standard model for a complex cell is the so-called quadrature filter pair model. The response of two standard simple cell models are squared and added. The filters of the two simple cells form a so called quadrature filter pair, in this case two wavelets that differ only by a slight shift of the stripes by half a stripe width. Their relationship is therefore similar to that of sin and cos, for which $\sin(\phi)^2 + \cos(\phi)^2 = 1$ holds, which implies that the square sum is invariant to a change of ϕ. Similarly, the response of the standard complex cell model is approximately invariant to a shift of stimulus. This invariance is the defining property of an ideal complex cell.

Gabor Wavelets

Gabor wavelets (with DC-correction) are defined as

$$\psi_j(\boldsymbol{x}) := \frac{k_j^2}{\sigma^2} \exp\left(-\frac{k_j^2 x^2}{2\sigma^2}\right)\left(\exp(\mathrm{i}\boldsymbol{k}_j^T\boldsymbol{x}) - \exp\left(-\frac{\sigma^2}{2}\right)\right),$$

with wave vectors $\boldsymbol{k}_j$ having different orientations and different frequencies.



Gabor wavelets fulfill the uncertainty relationship exactly.

23/48

Gabor wavelets are often used for image processing and to model simple and complex cells. They are localized in space and frequency, and they actually do that as precise as theoretically possible, i.e. they fulfill Heisenberg's uncertainty relationship exactly (side note for physicists and electrical engineers). A Gabor wavelet is essentially the product of a Gaussian (black solid line) with a (co)sine wave and could therefore be written in its simplest one-dimensional form as $\exp(-x^2)\sin(x)$ (green dashed line) or $\exp(-x^2)\cos(x)$ (blue solid line); together these form a quadrature filter pair.
The equation given on the slide is more complicated and simpler in some aspects for several reasons. This is not essential for the lecture, but for the technically interested reader I explain the differences.

- The $\sin(x)$ and $\cos(x)$ wavelets are combined into one complex wavelet with $\exp(\mathrm{i}x) = \cos(x) + \mathrm{i}\sin(x)$ (second exponential in the equation). This makes in particular the convolution more efficient. Since a convolution is always complex, the second convolution in the imaginary part comes for free.

- The simple x in $\exp(\mathrm{i}x)$ is multiplied by a wave number k_j to allow chosing a spatial frequence different from 1, and index j allows to chose different wave numbers for different cells, yielding $\exp(\mathrm{i}k_j x)$

- In two (or higher) dimensions a wave not only has a frequency but also a direction, thus k_j becomes a two (or higher) dimensional vector and the product $k_j x$ an inner product, yielding $\exp(\mathrm{i}\boldsymbol{k}_j^T\boldsymbol{x})$. (Please note the difference between $\boldsymbol{x}$ representing an image, in which case it might be a 10000-dimensional vector for a 100×100-pixel image, and $\boldsymbol{x}$ representing space, in which case it is just two-dimensional for an image. Here we use the latter version.)

- It is common to add a parameter σ to the Gaussian $\exp(-x^2)$ to control its width, yielding $\exp(-\frac{x^2}{2\sigma^2})$.

- The additional factor k_j^2 in $\exp(-\frac{k_j^2 x^2}{2\sigma^2})$ scales the Gaussian such that all Gabor wavelets look alike, no matter what frequency they have. This is referred to as *self-similarity* of the family of Gabor wavelets with constant σ.

- The term $-\exp(-\frac{\sigma^2}{2})$ at the end pulls the cosine wavelet a bit down in the center to make it really DC-free (DC stands for *direct current* here), i.e. the integral over the whole filter is zero. This is guaranteed for symmetry reasons for the sine filter, but for the cosine filter it must be taken care of explicitely. The filter being DC-free has the advantage that the response of the modeled simple or complex cell does not depend on overall brightness of the image, which is a simple form of visual invariance.

- The prefactor $\frac{k_j^2}{\sigma^2}$ finally scales the Gabor wavelets such that the average magnitude of the responses of the convolution on natural images are more balanced for different k and σ.

Sparseness Principle

(Olshausen & Field, 2004, Curr. Opp. Neurobiol 14:481)

A sparse representation

- can reduce metabolic costs, because fewer units are active,
- can reduce wiring, because fewer units need to be connected,
- can be more robust, because units tend to be more binary,
- can simplify learning and processing, because relevant information is more localized,
- ...

24/48

Olshausen and Field (1996) have argued that the goal of sensory coding is to yield a *sparse* (D: spärliche(?)) representation. A sparse representation is one, where for any given input only few units are strongly active, all others are close to zero. This code might have various advantages for the brain.
The figure (Olshausen and Field, 2004) shows a non-sparse representation at the top and a sparse representation at the bottom.

Sparse Coding

Assumption: Images can be written as a superposition of basis functions,

$$I(\mathbf{x}) = \sum_i a_i \phi_i(\mathbf{x}) \,, \tag{2}$$

with fixed functions $\phi_i(\mathbf{x})$ and variable coefficients a_i.

Objective: Choose the (probably normalized) functions such that the reconstruction error is small and the distribution of coefficients sparse, i.e.

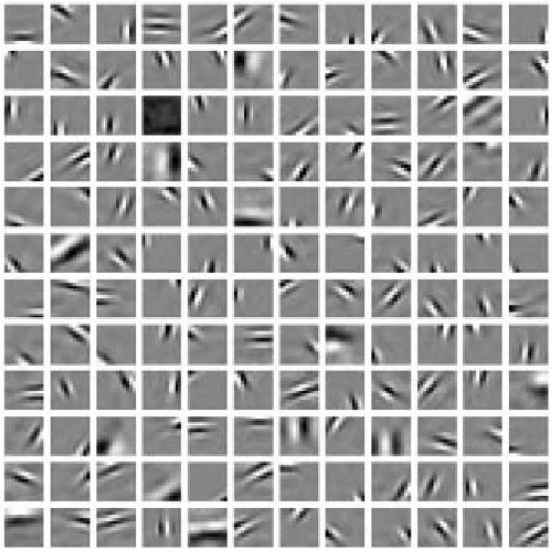
$$\text{minimize} \quad E := \underbrace{\int_{\mathbf{x}} (I(\mathbf{x}) - \sum_i a_i \phi_i(\mathbf{x}))^2 \, \mathrm{d}^2\mathbf{x}}_{\text{reconstruction term}} + \lambda \underbrace{\sum_i |a_i|}_{\text{sparseness term}} \,. \tag{3}$$

(Olshausen & Field, 1996, Nature 381:607–9)

25/48

The model by Olshausen and Field (1996) assumes that images $I(\mathbf{x})$ can be represented by a linear superposition of some fixed basis functions $\phi_i(\mathbf{x})$, wich leads to the first term in the cost function E. The basis functions may be overcomplete, i.e. there may be more functions than pixels in the image, and non-orthogonal, which they must be in cast of an overcomplete set.
The weighting coefficients a_i vary from image to image and should be sparsely distributed, i.e. should be near zero most of the time and only occasionally have a large positive or negative value. The second term in the cost function E formalizes the sparseness objective.
An optimization procedure optimizes both, the basis functions across all images as well as the weighting coefficients for each image individually.

Filters Generating a Sparse Code of Natural Images



(Olshausen & Field, 2004, Curr. Opp. Neurobiol 14:481, Fig. 1a)

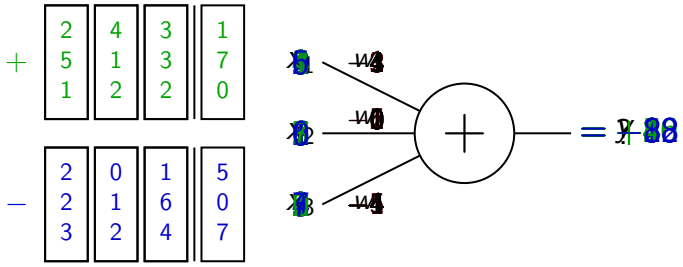
26/48

The filters obtained by optimizing the sparseness of the code in the model by Olshausen and Field (1996) resemble simple cell receptive fields fairly well (figure from Olshausen and Field, 2004).

Neural Network Learning

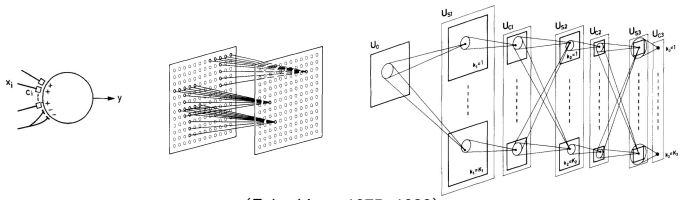
27/48

Perceptron Learning

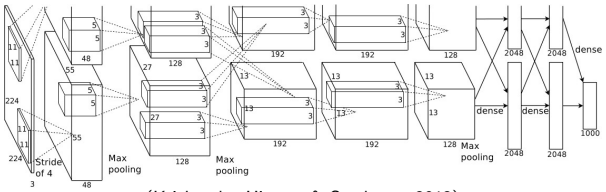


- The unit has inputs $x_i, ...$, weights $w_i, ...$, and an output $y = x_1 w_1 + x_2 w_2 + x_3 w_3$.
- The unit is trained with examples that are supposed to generate a positive or negative output.
- The weights of the unit are initialized randomly.
- If a negative output should actually be positive, the input is added to the weights.
- If a positive output should actually be negative, the input is subtracted from the weights.
- If the task can be solved at all, then the algorithm finds the solution.
- The solution also generalizes to new inputs.

Deep Neural Networks



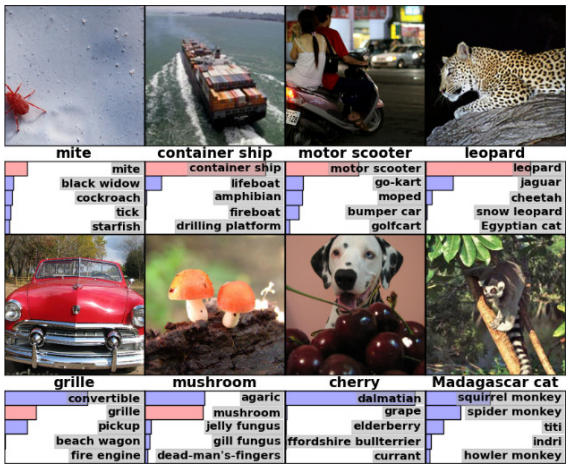
(Fukushima, 1975, 1980)



(Krizhevsky, Hinton, & Sutskever, 2012)

- Artificial neural networks are inspired by the brain.
- They are good at processing sub-symbolic information.
- They can learn from examples.

Alex Net

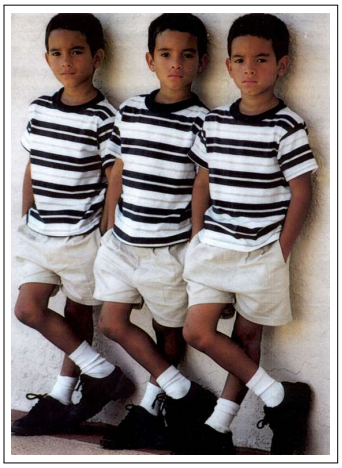


(Krizhevsky, Hinton, & Sutskever, 2012)

- 2012: Deep neural networks achieve super-human recognition rates in many applications.

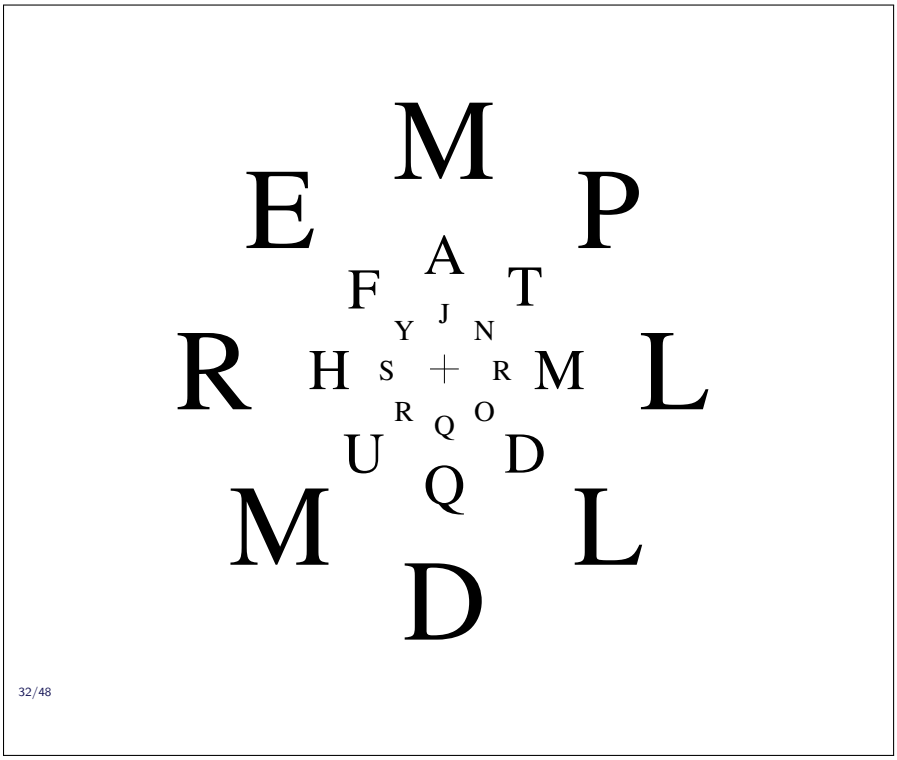
30/48

Visual Invariances



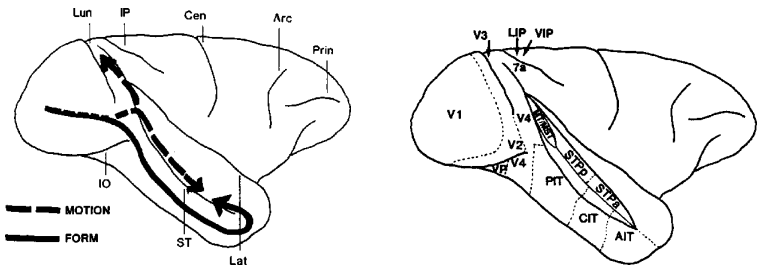
31/48

7 Visual Invariances



32/48

Visual Pathways and Areas (Macaque)

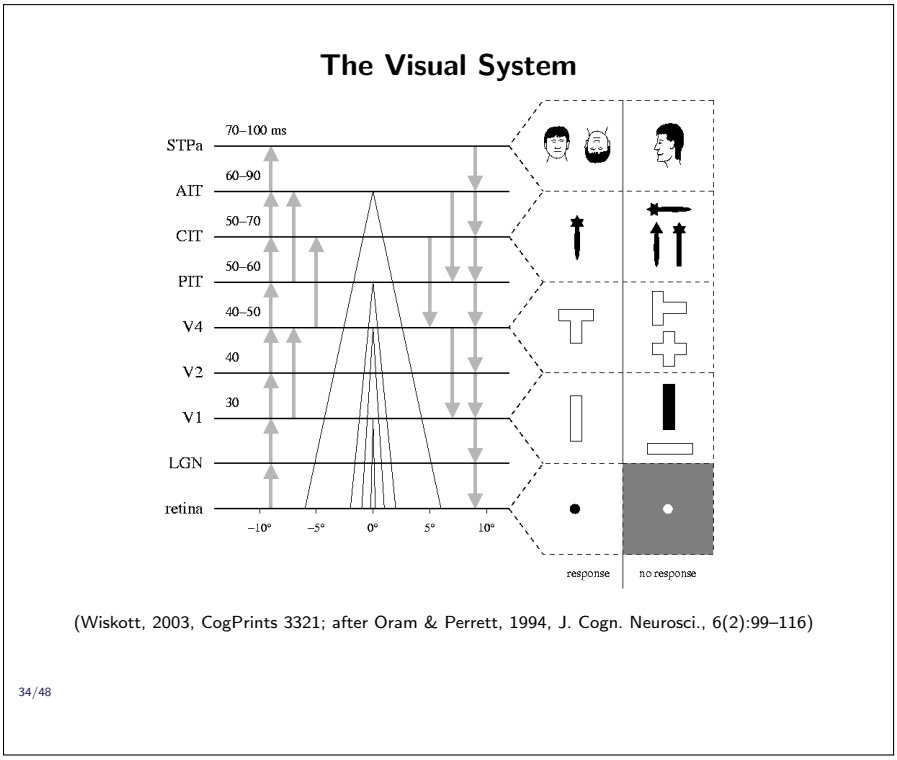


(Oram & Perrett, 1994, Neur. Netw. 7(6-7):945–972)

33/48

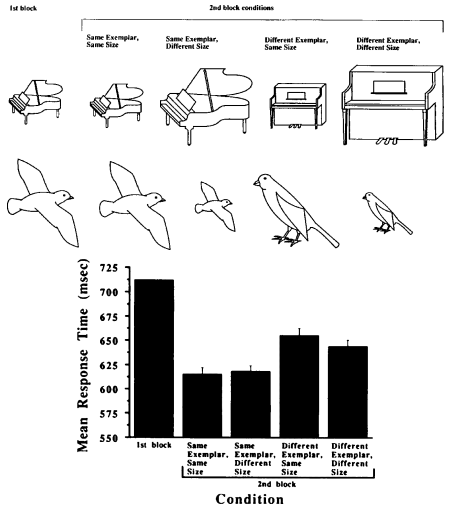
Visual input from the retina gets projected through the lateral geniculate nucleus (LGN, not shown here), which is subcortical, to the primary visual cortex (V1, 'V' and '1' indicating 'visual' and 'primary' respectively). From there it goes through V2 and V4 to the inferior temporal cortex (IT), which can be further subdivided into posterior (PIT), central (CIT), and anterior (AIT) IT. IT is thought to be instrumental for object recognition, while V1–V4 extract more elementary features. This path is referred to as the *what*-path, because it tells us what we see.

Another path goes through V2 and V4 to areas MT/MST, which are particularly responsive to motion. This path has been termed the *where*- or *how*-path, because it is thought to tell us where the objects are or how we can handle them, e.g. grasp them. The paths converge in the posterior (STPp) and anterior (STPa) superior temporal polysensory area. Cells in STPa, for instance, have been found to be sensitive to body motion, such as walking. Figure by Oram and Perrett (1994).



This schematic drawing (Wiskott, 2003) highlights some organizational principles of the visual system. It is hierarchically structured in areas (listed on the left; in models usually referred to as layers, not to be confused with the layers of cortex), which are coupled by feedforward (gray upward arrows) as well as feedback (gray downward arrows) connections with some shortcut connections that skip an area. Processing in each area takes about 10ms (latencies are shown on the left). Along the hierarchy the receptive field sizes increase (indicated by the triangles in the middle), the feature complexity increases (indicated by some typical stimuli on the right to which a neuron might respond or not), and the invariance, e.g. to shift (or translation), scaling, and rotation, increases.

Psychophysical Evidence for Size Invariance

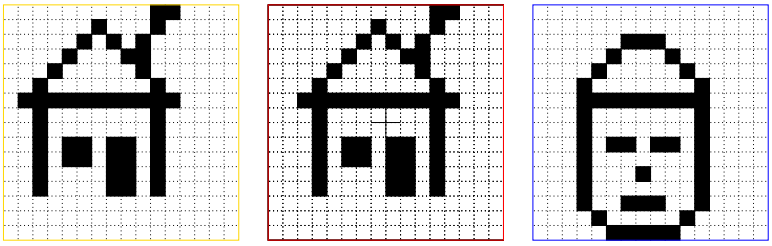


(Biederman & Cooper, 1992)

35/48

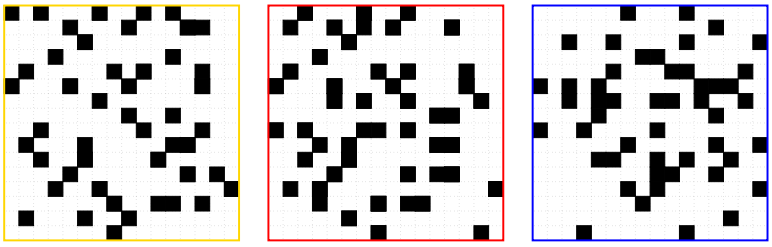
If one has seen an object not too long ago, the naming response to that object is faster than it was the first time, even if one is not aware that one has seen it before. This effect is called priming.
In this experiment naming response times to line drawings of objects were measured. The response times were larger (712ms) for the first presentation ('1st block') than for the second presentation (average 630ms, '2nd block'), which is due to the priming effect. Interestingly, the priming advantage did not depend on the size of the drawing (compare 'Same Size' with 'Different Size'). However, it depends on the object instance of same name (compare 'Same Exemplar' with 'Different Exemplar'), which indicates that the priming effect happens somewhere in the visual processing, where this difference matters, and not somewhere in the 'naming area', where the visual difference should not matter. If the priming advantage would come from the 'naming area', size invariance would be no surprise.
Taken together the results support the idea that visual processing is size invariant within the limits measured here.

The Simple Invariance Problem



36/48

The Difficult Invariance Problem



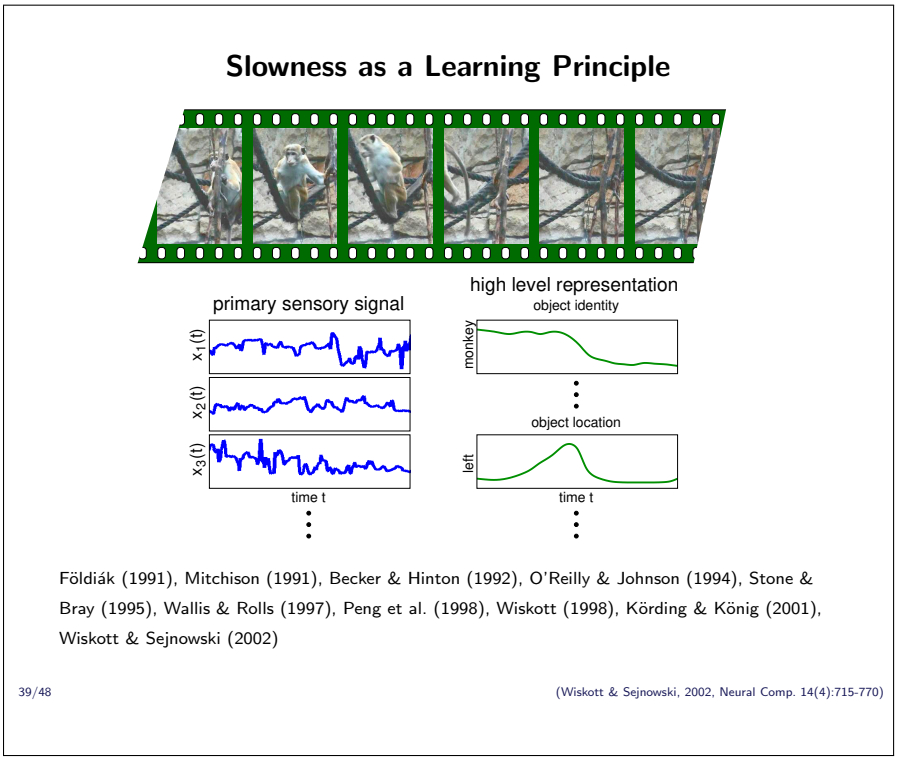
37/48

Slow Feature Analysis

(maxmann, 2016, pixabay, © CC0, URL)

38/48

Image: (maxmann, 2016, pixabay, © CC0, URL)[7.1]



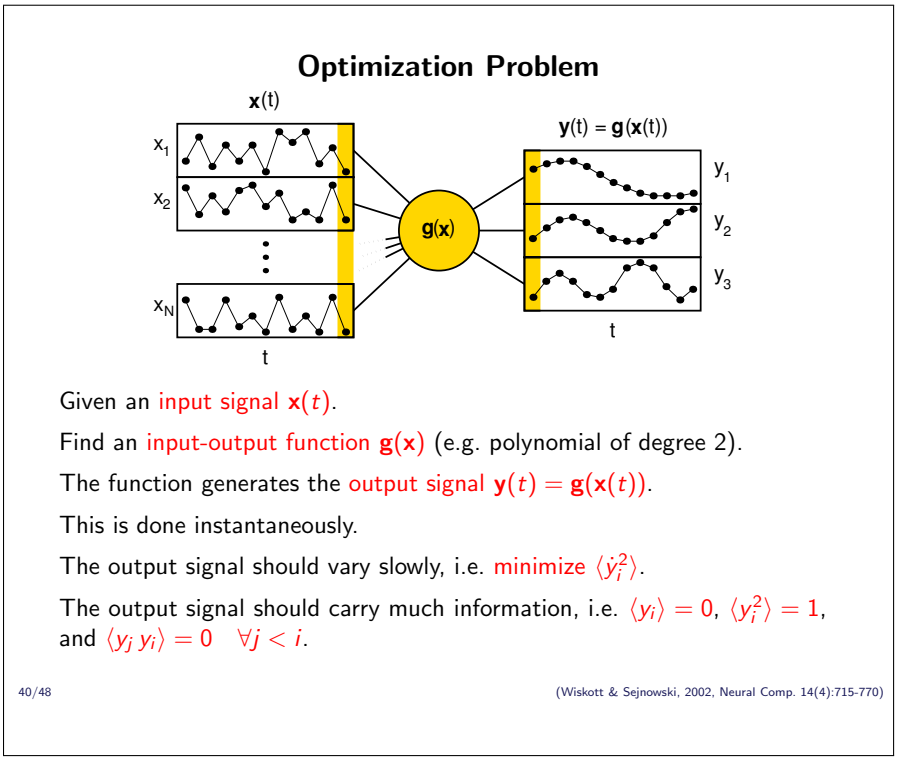
Slowness as a learning principle is based on the observation that different representations of the visual sensory input vary on different time scales. Our visual environment itself is rather stable. It varies on a time scale of seconds.

The primary sensory signal on the hand, e.g. responses of single receptors in our retina or the gray value of a single pixel of a CCD camera, vary on a faster time scale of milliseconds, simply as a consequence of the very small receptive field sizes combined with gaze changes or moving objects. As an example imagine you are looking at a quietly grazing zebra. As your eyes scan the zebra, single receptors rapidly change from black to white and back again because of the stripes of the zebra. But the scenery itself does not change much.

Finally, your internal high-level representation of the environment changes on a similar time scale as the environment itself, namely on a slow time scale. The brain is somehow able to extract the slowly varying high-level representation from the quickly varying primary sensory input. The hypothesis of the slowness learning principle is that the time scale itself provides the cue for this extraction. The idea is that if the system manages to extract slowly varying features from the quickly varying sensory input, then there is a good chance that the features are a good representation of the visual environment.

A number of people have worked along these lines. Slow feature analysis is within this tradition but differs in some significant technical aspects from all previous approaches.

Figure: (Wiskott et al., 2011, Fig. 2, © CC BY 4.0, URL)[7.2]



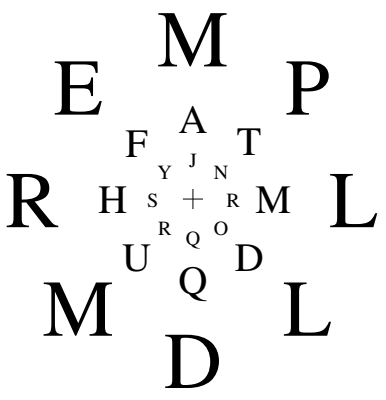
Slow feature analysis is based on a clearcut optimization problem. The goal is to find input-output functions that extract most slowly varying features from a quickly varying input signal.

It is important that the functions are instantaneous, i.e. one time slice of the output signal is based on just one time slice of the input signal (marked in yellow). Otherwise low-pass filtering would be a valid but not particularly useful method of extracting slow output signals. Instantaneous functions also make the system fast after training, as is important in visual processing, for instance. It is also possible to take a few input time slices into account, e.g. to make the system sensitive to motion or to process scalar input signals with a fast dynamics on a short time scale. However, low-pass filtering should never be the main method by which slowness is achieved.

Without any constraints, the optimal but not very useful output signal would be constant. We thus impose the constraints of unit variance $\langle y_i^2 \rangle = 1$ and, for mathematical convenience, zero mean $\langle y_i \rangle = 0$. To make different output signal components represent different information, we impose the decorrelation constraint $\langle y_j y_i \rangle = 0$. Without this constraint, all output components would typically be the same. Notice that the constraint is asymmetric, later components have to be uncorrelated to earlier ones but not the other way around. This induces an order. The first component is the slowest possible one, the second component is the next slowest one under the constraint of being uncorrelated to the first, the third component is the next slowest one under the constraint of being uncorrelated to the first two, etc.

Figure: (Wiskott et al., 2011, Fig. 1, © CC BY 4.0, URL)[7.3]

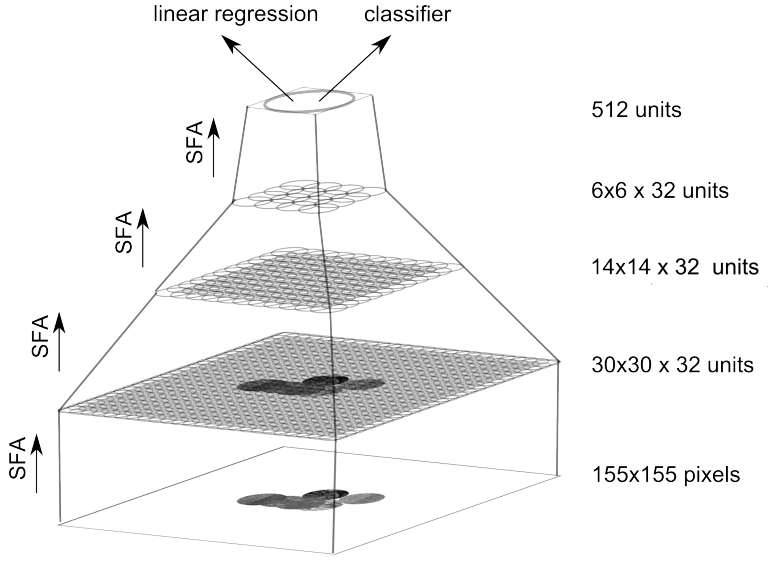
Learning Visual Invariances



Mathias Franzius Niko Wilbert

41/48

Hierarchical Model

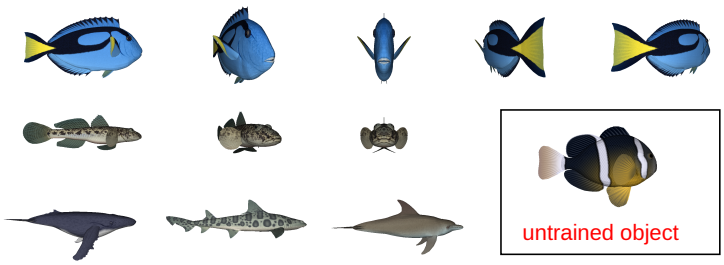


(Franzius, Wilbert, & Wiskott, 2011, Neural Computation 23(9):2289-2323)
cf. (Wallis & Rolls, 1997, Progress in Neurobiol. 51(2):167–194)

42/48

The model is a hierarchical network of SFA modules, each one realizing polynomials of degree two. The input image has a size of 155×155 pixels. The first layer has an array of 30×30 SFA modules with overlapping receptive fields. Each of these modules has 32 units, i.e. it extracts the 32 most slowly varying features, which feed into the next layer of 14×14 modules, again with overlapping receptive fields. Such convergent hierarchical processing proceeds up to the top of the network. Overall the network realizes polynomials of degree 16 and extracts the 512 most slowly varying features. The output is later used for linear regression or to train a classifier.

Stimuli - Fish



(models taken from toucan.web.infoseek.co.jp/3DCG/3ds/FishModelsE.html)

SFA-training with 15 'old' objects.

Random walk in x-position, y-position, scale, and in-depth rotation.

10,000 data points per object.

Additional data for 10 'new' objects, after SFA-training.

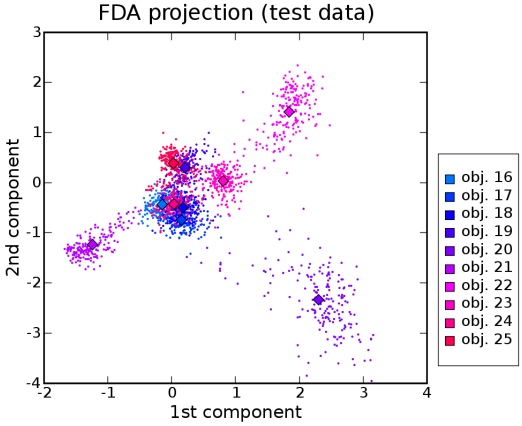
Only grayscale images are used in the experiments.

(Franzius, Wilbert, & Wiskott, 2011, Neural Computation 23(9):2289-2323)

One stimulus set is a set of different fish, sharks, and whales, rendered in 3D. They are shown at different positions, scales, and in-depth rotation angles. Fifteen objects were used for training the network, ten new ones for testing. Color was not used, because it would simplify the task too much.

Classification Results

test images of the 10 new objects
FDA projection (test data)



Performance of Gaussian classifier across position, scale, and in-depth rotation angle: 97.08%, 97.81%, and 95.41% on 15 old, 10 new, and all 25 objects, respectively.

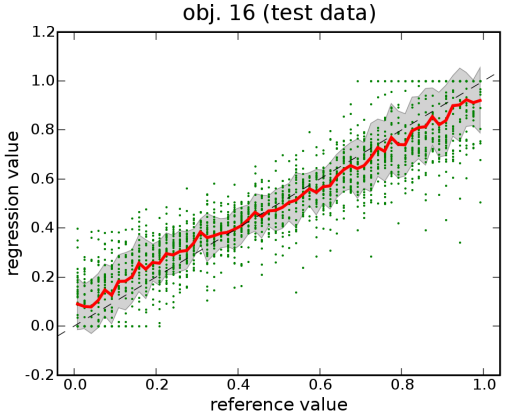
(Franzius, Wilbert, & Wiskott, 2011, Neural Computation 23(9):2289-2323)

A Gaussian classifier was trained on the 512 output components to classify images of the fish at different positions, scales, and in-depth rotation angles with recognition rates of 95% and up. The scatter plot shows the projection of the output onto the first two Fisher discriminants. It can be seen how the data cluster according to object identity.

Regression Results - Scale

test images of 1 new object

obj. 16 (test data)



Regression on 50% of one new object, test on remaining 50%.

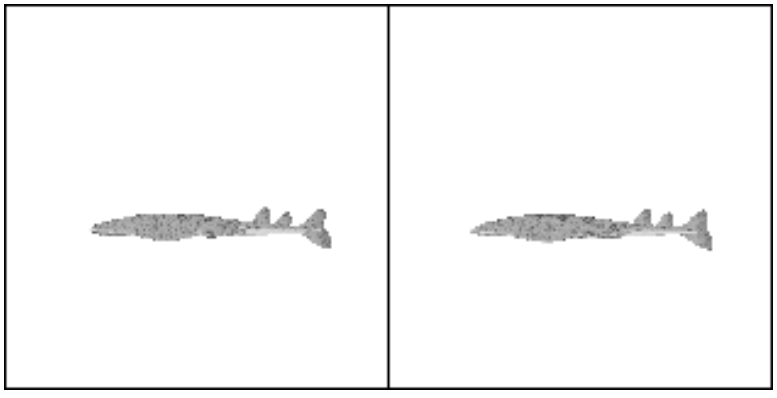
(green: all data points; red: mean; gray: $\pm$ one standard deviation)

(Franzius, Wilbert, & Wiskott, 2011, Neural Computation 23(9):2289-2323)

Animated Results

test input image

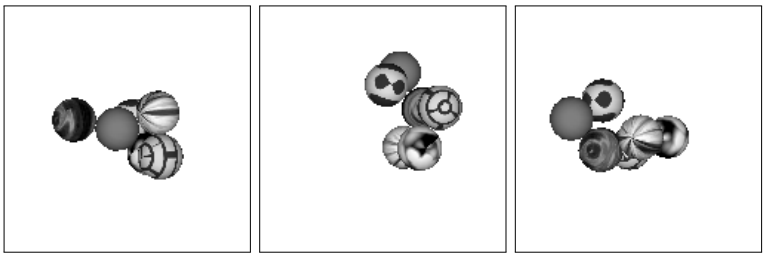
rendered output of classifier and regression



(Franzius, Wilbert, & Wiskott, 2011, Neural Computation 23(9):2289-2323)

When we present an image of a fish to the network, the SFA output allows us to determine the identity of the fish with a Gaussian classifier and estimate its position, scale, and in-depth rotation angle by linear regression. This estimated information can be used to render a new image of a fish. Ideally the rendered image should be identical to the one presented to the network. This animation illustrates the performance of the network by comparing the original image (left) with the newly rendered one (right) for many different fish images.

Stimuli - Textured Sphere Clusters



SFA-training with 5 'old' objects.

Random walk in x-position, y-position plus, in-depth and in-plane rotation angle.

10,000 data points per object.

Additional data for 5 'new' objects, after SFA-training.

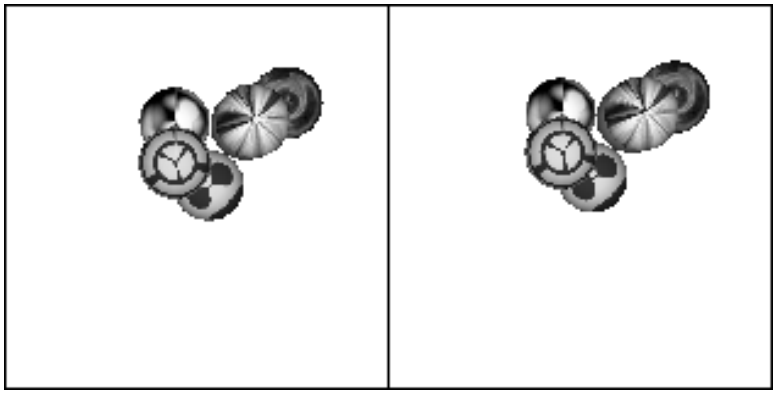
Only grayscale images are used in the experiments.

(Franzius, Wilbert, & Wiskott, 2011, Neural Computation 23(9):2289-2323)

Animated Results

test input image

rendered output of classifier and regression



(Franzius, Wilbert, & Wiskott, 2011, Neural Computation 23(9):2289-2323)

When we present an image of a sphere object to the network, the SFA output allows us to determine the identity of the object with a Gaussian classifier and estimate its position and in-plane as well as in-depth rotation angle by linear regression. This estimated information can be used to render a new image of a sphere object. Ideally the rendered image should be identical to the one presented to the network. This animation illustrates the performance of the network by comparing the original image (left) with the newly rendered one (right) for many different sphere-object images.

References

Olshausen, B. A. and Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609.

Olshausen, B. A. and Field, D. J. (2004). Sparse coding of sensory inputs. *Current Opinion in Neurobiology*, 14(4):481–7.

Oram, M. W. and Perrett, D. I. (1994). Modeling visual recognition from neurobiological constraints. *Neural Networks*, 7(6/7):945–972.

Wiskott, L. (2003). How does our visual system achieve shift and size invariance? Cognitive Sciences EPrint Archive (CogPrints).

Wiskott, L., Berkes, P., Franzius, M., Sprekeler, H., and Wilbert, N. (2011). Slow feature analysis. *Scholarpedia*, 6(4):5282.

Notes

[3.1] Alphab.fr, 2007, Wikipedia, © CC BY 2.0, https://en.wikipedia.org/wiki/File:Rice_fields_near_Sapa,_Vi%C3%AAt_Nam.jpg

[7.1] maxmann, 2016, pixabay, © CC0, https://pixabay.com/en/snail-shell-crawl-mollusk-1330766/

[7.2] Wiskott et al., 2011, Scholarpedia 5(2):1362, Fig. 2, © CC BY 4.0, http://scholarpedia.org/article/Slow_feature_analysis

[7.3] Wiskott et al., 2011, Scholarpedia 5(2):1362, Fig. 1, © CC BY 4.0, http://scholarpedia.org/article/Slow_feature_analysis