# Slow feature analysis

Laurenz Wiskott et al. (2011), Scholarpedia, 6(4):5282. doi:10.4249/scholarpedia.5282 revision #137965 [link to/cite this article]

- **Prof. Laurenz Wiskott**, Institut für Neuroinformatik, Ruhr-Universität Bochum, Bochum, Germany
- **Pietro Berkes**, Volen Center for Complex Systems, Brandeis University, Waltham, MA, USA
- **Mathias Franzius**, Honda Research Institute Europe
- **Henning Sprekeler**, Laboratory of Computational Neuroscience, EPFL, Lausanne, Switzerland
- **Mr. Niko Wilbert**, Humboldt Universität zu Berlin, Berlin, Germany

**Slow feature analysis (SFA)** is an unsupervised learning algorithm for extracting slowly varying features from a quickly varying input signal. It has been successfully applied, e.g., to the self-organization of complex-cell receptive fields, the recognition of whole objects invariant to spatial transformations, the self-organization of place-cells, extraction of driving forces, and to nonlinear blind source separation.

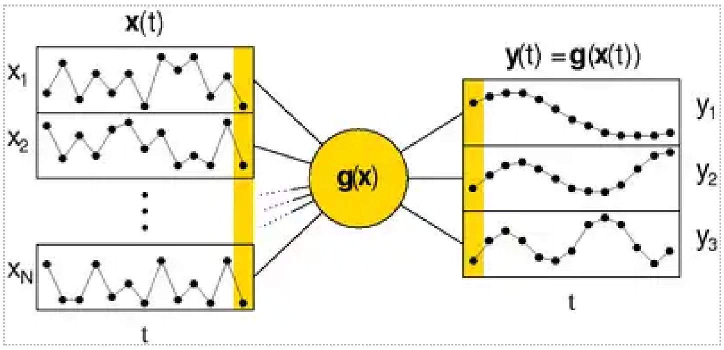

Figure 1: **Schematics of the optimization problem solved by Slow Feature Analysis.** Given a set of time-varying input signals, **x**(t), SFA learns instantaneous, non-linear functions **g**(**x**) that transform **x** into slowly-varying output signals, **y**(t). The optimization procedure guarantees that SFA returns the global optimum for **g** (i.e., the slowest output signal) in a given function space. As the transformations must be instantaneous, trivial solution like low-pass filtering are not possible.

Contents

# Slow feature analysis

## The slowness principle

From a computational point of view, one can think of perception as the problem of reconstructing the external causes of the sensory input to allow generation of adequate behaviour (see AlHacen, Alī al-Ḥasan ibn al-Ḥasan ibn al-Haytham, 10th century, cited in Smith, 2001; Helmholtz, 1910).

For example, when looking at a picture on a computer screen, we see the objects that are present on it and their relative position in the image, rather than the color of the individual pixels. An active area of research in computational neuroscience is concerned with the way the brain learns to form a representation of these external causes from raw sensory input. An important idea in the field is that objects in the world have common structure, which results in statistical regularities in the sensory input. Using these regularities as a guide, the brain is able to form a meaningful representation of its environment.

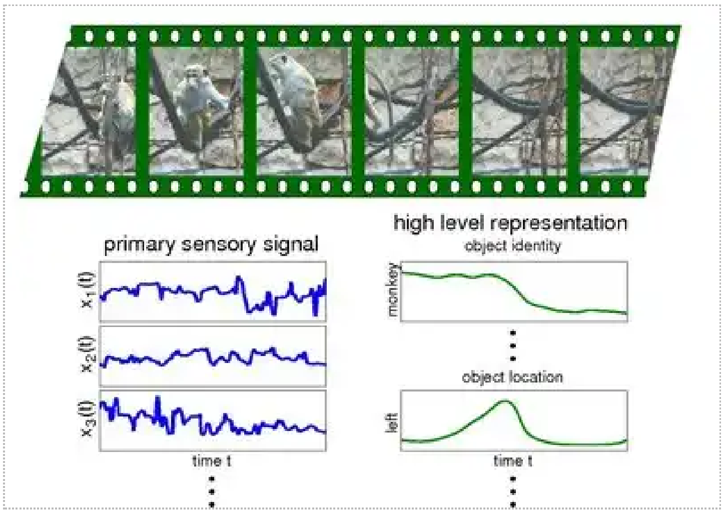

Figure 2: **Illustration of the slowness principle.** The movie strip (top) represents a natural visual scene, in which a monkey stands up from a suspended rope, then leaves the field of view to the left. As observers, we experience this visual scene only through the activity of sensory receptors in the retina, whose response is comparable to raw measurements of light intensity in localized parts the visual field (blue traces). While the information about the monkey is contained in their activity, it is distributed in a non-linear way over millions of receptors. To produce meaningful behavior, the brain needs to reconstruct the high-level signals regarding, for instance, the monkey's presence and position (green traces). Since objects in the world are persistent and their appearance changes mostly in a continuous way, these high-level signals change smoothly over time, which can be leveraged to extract them from the sensory input in an unsupervised way. In other words, the brain may use this insight to learn the transformation that maps the blue traces into the green ones.

At the heart of the slowness principle is one of these regularities, namely that external causes are persistent in time. For example, behaviourally relevant visual elements (objects and their attributes) are visible for extended periods of time and change with time in a continuous fashion, on a time scale of seconds. On the other hand, the primary sensory signal, like the responses of individual retinal receptors or the gray-scale values of a single pixel in a video camera, are sensitive to very small changes in the environment, and thus vary on a much faster time scale ( Figure 2). If it is to explicitly represent the original visual elements, the internal representation of the environment in the brain should vary on a slow time scale again.

This difference in time scales leads to the central idea of the slowness principle: By finding and extracting slowly varying output signals from the quickly varying input signal we seek to recover the underlying external causes of the sensory input. The slowness principle provides a natural hypothesis for the functional organization of visual cortex and possibly also other sensory areas.

## The optimization problem

The Slow Feature Analysis algorithm formalizes the general intuition behind the slowness principle as a non-linear optimization problem ( Figure 1): Given a (potentially high-dimensional) input signal $\mathbf{x}(t)$, find functions $g_j(\mathbf{x})$ such that the output signals

$$y_j(t) := g_j(\mathbf{x}(t)) \tag{1}$$

minimize

$$\Delta(y_j) := \langle \dot{y}_j^2 \rangle_t \tag{2}$$

under the constraints

$$\langle y_j \rangle_t = 0 \quad (3)$$

```
(zero mean),
```

$$\langle y_j^2 \rangle_t = 1 \quad (4)$$

```
(unit variance),
```

$$\forall i < j : \langle y_i y_j \rangle_t = 0 \quad (5)$$

```
(decorrelation and order).
```

The angular brackets, $\langle \cdot \rangle_t$, indicate averaging over time and $\dot{y}$ is the derivative of $y$ with respect to time. The $\Delta$-value defined by (2) is the objective of the optimization problem, and measures the slowness of an output signal as the time average of its squared derivative. A low value indicates small variations over time, and therefore slowly-varying signals.

The $\Delta$-value is optimized under three constraints: Constraints (3) and (4) normalize all output signals to a common scale, which makes their temporal derivative directly comparable. Constraint (5) requires that the output signals are decorrelated from one another and guarantees that different output signal components code for different information.

The SFA formulation of the slowness principle also avoids two uninteresting solutions of the optimization problem. Firstly, constraints (3) and (4) avoid the trivial constant solution, which is infinitely slow but does not carry any information. Secondly, although SFA seeks to maximize slowness over time, the functions $g_j$ must extract the output signals $y_j$ instantaneously (1). Solutions that would produce slowly-varying output signals by smoothing the input over time, for example by computing a moving average of $x_i(t)$ or by low-pass filtering the signals, are thus excluded.

It is this tension between instantaneous processing and slowly-varying output that makes SFA useful in extracting slowly-varying features. For example, in Figure 2, a function $g(t)$ returning the presence of the monkey could be optimal in the SFA sense as it would compute its output instantaneously, yet produce a slowly-varying signal as the external cause of the signal (the monkey) enters and leaves the visual scene on a slow time scale. (Abruptly but rarely changing features are considered slowly-varying on average as well as defined by (2).)

## The algorithm

The optimization problem as stated above is one of variational calculus (http://en.wikipedia.org/wiki/Calculus_of_variations) , as it requires optimizing over functions, $g_j$, rather than over a set of parameters. If one confines the functions $g_j$ to a finite dimensional function space, such as all polynomials of degree two, one can transform the variational problem into a more conventional optimization over the coefficients of the basis of the function space (e.g., all monomials of degree 1 and 2). In this way, the problem becomes simpler to solve, and one can use algebraic methods, which are the basis of the slow feature analysis algorithm, as shown in the following (Wiskott and Sejnowski, 2002).

Consider as a simple example the two dimensional input signal $x_1(t) := \sin(t) + \cos(11t)^2$ and $x_2(t) := \cos(11t)$ (Figure 3). Both components are quickly varying, but hidden in the signal is the slowly varying 'feature' $y(t) = x_1(t) - x_2(t)^2 = \sin(t)$, which can be extracted with a polynomial of degree two, namely $g(\mathbf{x}) = x_1 - x_2^2$.

To find the function $g(\mathbf{x})$ that extracts the slow feature from the input signal, one proceeds as follows.

1. **The non-linear problem is transformed to a linear one by expanding the input into the space of nonlinear functions one is considering.** For example, for polynomials of degree two, expand $\mathbf{x}(t)$ (Figure 3A) into new signal components $\tilde{z}_1 := x_1, \tilde{z}_2 := x_2, \tilde{z}_3 := x_1^2, \tilde{z}_4 := x_1 x_2, \tilde{z}_5 := x_2^2$ (Figure 3B). Any polynomial of degree two in $x_1$ and $x_2$ can be written as a linear combination of these five components, making the problem linear (the constant term is missing, and will be taken care of in the next step).

2. **The expanded signal is normalized such that constraints (3)-(5) are satisfied.** Subtract the mean (this determines the missing constant of the previous step) and apply a linear transformation such that the resulting signal $\mathbf{z}(t)$ has unit variance in all directions (Figure 3C). This is referred to as whitening or sphering and can be done with principal component analysis (http://en.wikipedia.org/wiki/Principal_components_analysis) . If one projects the sphered signal onto any direction, the resulting signal has zero mean and unit variance; if one projects the sphered signal onto two orthogonal directions, the two resulting signals are linearly uncorrelated, as required by the constraints.
3. **Temporal variation is measured in the normalized space.** Calculate the time derivative $\dot{\mathbf{z}}(t)$ of the sphered signal (Figure 3D).
4. **The slowest-varying directions are extracted.** Find the direction of least variance of the time derivative signal (see $\mathbf{w}_1$-axis (solid line) in Figure 3D). This is the direction in which the sphered signal varies most slowly, because on average the square of the time derivative is smallest. If more than one output component is needed, take orthogonal directions with the next smallest variance (dashed lines in Figure 3D). Finding these directions can again be done with principal component analysis. The directions are then the principal components with the smallest eigenvalues, and the eigenvalues are exactly the $\Delta$-values of the projected signals as defined by (2).

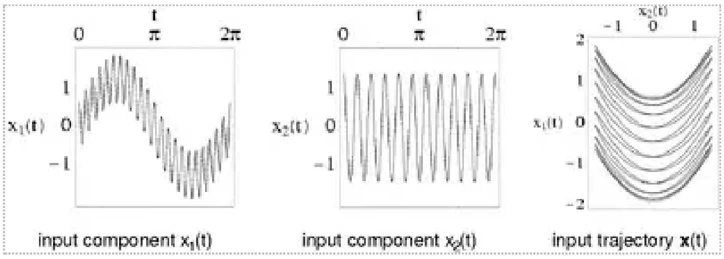

Figure 3: **Input signal of the simple example described in the text.** The panels on the left and center show the two individual input components, $x_1(t)$ and $x_2(t)$ . On the right, the joint 2D trajectory $\mathbf{x}(t) = (x_1(t), x_2(t))$ is shown.

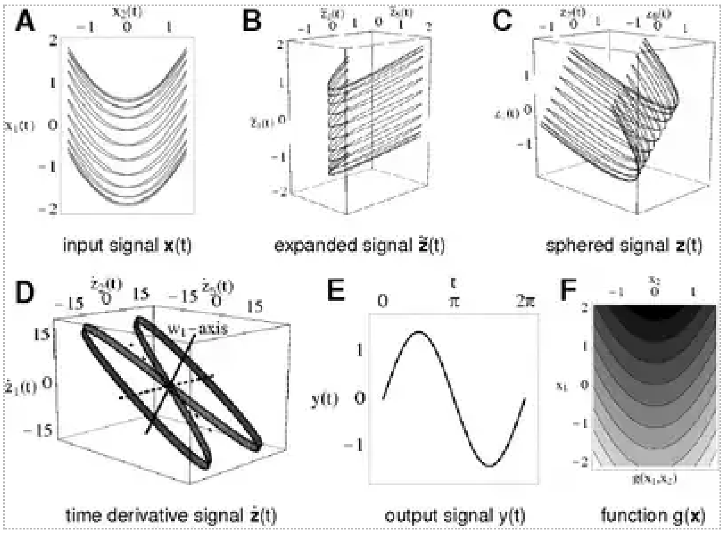

Figure 4: **Step-by-step illustration of the SFA algorithm.** The input signal $\mathbf{x}(t)$ (A) is first expanded in a non-linear function space (B), and then normalized (or "sphered") to zero mean and unit variance (C). In the normalized space, a projection on orthogonal directions always produces output signals that fulfill the constraints of the SFA optimization problem. To find among all possible projections the one that gives signals with slowest variation, one considers the derivative of the expanded signal (D), and computes the directions of smallest variance (D, inner axes). The projection on the first (slowest) of these directions, indicated by the solid line in (D), corresponds to the underlying slowly-varying signal in this example (E). The steps (B-E) are equivalent to finding a quadratic function in the original input space (F).

The sphered signal projected onto the direction of least variance of the time derivative signal is the desired slow feature (Figure 3E). Combining all the steps above (nonlinear expansion, whitening, projection onto the direction of least variance of the time derivative signal) yields the function $g(\mathbf{x})$, see Figure 3F. Evaluating $g(\mathbf{x})$ along the trajectory $\mathbf{x}(t$

yields $y(t) = g(\mathbf{x}(t))$ as shown in Figure 3E.

It is possible to combine steps 2 and 4 in one by solving a generalized eigenvalue problem (Berkes and Wiskott, 2005).

## Historical remarks and relations to other approaches

Probably the first explicit mentioning of slowness (there referred to as smoothness) as a possible objective for unsupervised learning can be found in (Hinton, 1989, on page 208). Early connectionist models were presented by Földiák (1991) and Mitchison (1991), who introduced neural networks with local learning rules that minimize temporal variation in the output units, resulting in invariance to input transformations. Földiák's (1991) approach was based on earlier models of conditioning, and became quite popular for its simplicity and biological plausibility.

The slow feature analysis algorithm (Wiskott, 1998; Wiskott and Sejnowski, 2002) was developed independently of these earlier approaches and is distinct in several important aspects:

- Slow feature analysis has a closed form solution while previous approaches use incremental or online learning rules, often in the form of a gradient ascent/descent method.
- Slow feature analysis is guaranteed to find a globally optimal solution within the finite-dimensional function space and thus does not suffer from local optima.
- Slow feature analysis yields a set of uncorrelated output signals that are ordered by slowness. In online learning rules the output signals are either not ordered, i.e. it is not guaranteed that any of the output signal components is the slowest possible one, or it might take a long time to find the faster signal components, because the slower ones have to converge before the faster ones can.
- Due to the nonlinear expansion, slow feature analysis suffers from a variant of the curse of dimensionality (http://en.wikipedia.org/wiki/Curse_of_dimensionality) , meaning that even moderately large input data cannot be handled anymore, because it becomes prohibitively large through the expansion. This problem can be ameliorated by hierarchical processing, which breaks down a large input space into many smaller ones, cf. Grid, place, head-direction, and view cells in the hippocampus and Invariant visual object recognition.

As for most learning objectives, the slowness principle can be formalized as a signal processing, probabilistic, or information-theoretical problem, resulting in closely related algorithms, which give different perspectives on the same, core idea:

- The signal processing aspect of temporal slowness is captured by the objective function of the SFA algorithm (Wiskott and Sejnowski, 2002). Although the constraints on input signals is minimal (the variance of the signals and of its derivatives must be defined), proving that SFA recovers the "real" underlying features of the input requires additional assumptions (see the sections Nonlinear blind source separation and Theory of slow feature analysis).
- The probabilistic perspective builds a slowness model constructively, assuming that the input signals have been generated by a linear combination of a number of (Markovian) slowly-varying causes, which defines a probability distribution of possible trajectories in the input space. Such a model is known as a Linear Dynamical System, Gaussian State Space model, or Kalman filter. The parameters of the linear combination can be learned by maximizing the probability of the observed signals under the model. Turner and Sahani (2007) have shown that SFA can be derived as the deterministic limit of this model. The equivalence between the two approaches is limited to the linear case, beyond which the two solutions diverge. Probabilistic models explicitly represent the uncertainty about the variables, and are thus more flexible when dealing with missing and noisy data.
- The information theoretical point of view is based on learning a compressed representation $\mathbf{y}(t)$ of the input $\mathbf{x}(t)$ that maximizes information about the *next* input, $\mathbf{x}(t+1)$. A direct relation to SFA can be derived by assuming Gaussian inputs with reversible statistics (Shaw, 2003; Creutzig and Sprekeler, 2008), where reversible means that the data have the same statistics when they are played backwards. As for the probabilistic perspective, this relation is restricted to the linear case.

Finally, Slow feature analysis can also be used for blind source separation, and can be related to some independent component analysis algorithms based on minimizing temporal correlations (Blaschke et al., 2006).

# Applications in computational neuroscience

## Complex-cell receptive fields in primary visual cortex

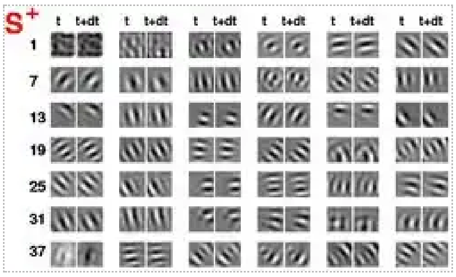

Figure 5: Optimal spatio-temporal stimuli for the first 42 quadratic functions found by SFA for natural image sequences. Stimuli are ordered by decreasing slowness of the corresponding output signal. The optimal stimuli resemble those of complex cells in primary visual cortex: they represent oriented edges, and they are invariant to small translations of the optimal stimuli (not shown).

Visual processing in our brain goes through a number of stages, starting from the retina, through the thalamus, and first reaching cortical layers at the primary visual cortex, also called V1. Neurons in V1 are sensitive to input from small patches of the visual input, their *receptive field*, and most of them respond particularly well to elementary features such as edges and gratings. Cells in V1 are divided into two classes: simple cells and complex cells. Both types respond well to edges and gratings, but simple cells are sensitive to the exact location of the stimulus while complex cells are invariant to stimulus shifts within their receptive field. Both types also show an orientation tuning, i.e., they respond strongly to edges and gratings of one orientation and not at all to some other orientation.

Units reproducing many of the properties of complex cells can be obtained by extracting the slowly-varying features of natural image sequences, suggesting that temporal slowness may be one of the principles underlying the organization of the visual system (Koerding *et al.*, 2004, Berkes and Wiskott, 2005). To model complex cells with slow feature analysis, one first creates input signals by moving a small window across natural images by translation, rotation, and zoom, thereby imitating the natural visual input. One then applies SFA to this input with polynomials of degree two as the nonlinear expansion. The resulting functions take small image patches in the size of the receptive field as an input and yield a scalar value as an output. If one interprets the scalar output value as a mean firing rate, one can compare the functions directly with neurons and indeed finds that they share many properties with complex cells in V1 (Berkes and Wiskott, 2005).

Figure 5 shows optimal stimuli, i.e., stimuli of fixed energy that yield the strongest output, for the first 42 functions found by SFA. They come in pairs to illustrate how the optimal stimulus should ideally change from one time frame to the next. The optimal stimuli have the shape of localized gratings and are known to be ideal also for simple and complex cells. The functions also show invariance to a shift of the stripes within a localized grating, one of the defining properties of complex cells (not shown).

Figure 6 shows the orientation tuning of selected functions in comparison to neurons in V1. These are in good agreement, and SFA reproduces a variety of different types, such as secondary response lobes (bottom right), and direction selectivity (bottom left). Some functions also show end- and side-inhibition, i.e., sensitivity to the length or width of the input stimulus (not shown).

Some of these results can be derived analytically based on the second-order statistics of natural images, see The "Harmonic Oscillation" Result.

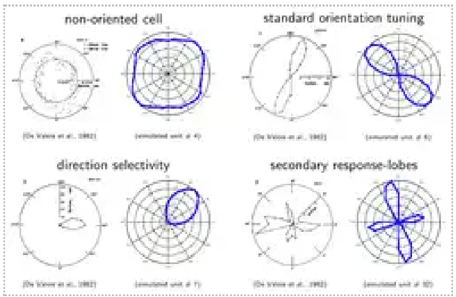

Figure 6: Some illustrative examples of the orientation tuning curves of complex cells (left in each pair, black lines) compared with corresponding functions found by SFA (right, blue lines). Responses are plotted in radial direction as a function of the orientation of the input gratings in azimuthal direction. Physiological tuning curves are reproduced from (De Valois *et al.*, 1982), Vision Research (http://www.sciencedirect.com/science/jou , Vol 22, with permission from Elsevier.

## Hierarchical SFA networks for high-dimensional data

As mentioned above, non-linear SFA suffers from the curse of dimensionality, since the dimensionality of the expanded function space increases very fast with the number of input signals. This is especially a problem for domains that naturally have a high dimensionality, like for instance visual data. For example, quadratic expansion of an input image of 100 by 100 pixels yields a dimensionality of 50,015,000, clearly too large to be handled by modern computers.

One natural solution to this problem is to apply SFA to subsets of the input, extract the slowest-varying features for each subset, and then use the concatenation of these solutions as the input for another iteration of SFA. At each step, a larger fraction of the input data is integrated into the new solution. In this way, the curse of dimensionality can be avoided, although, in general, the final slow features extracted need not be identical to the global solution obtained with the original, complete input. Thus, the splitting of the data into smaller patches relies on the locality of feature correlations in the input data, which typically holds for natural images.

This strategy results in hierarchical networks that resemble the feedforward organization of the visual system ( Figure 7). As we consider increasingly high layers, the effective receptive field size becomes larger, and it is possible to extract increasingly complex features (like whole objects). This is facilitated by the accumulation of computational power with each layer. For example, in a three-layer network with quadratic expansion in each layer the whole network can be represented by functions from a subset of the polynomials of degree up to $2^3 = 8$. For SFA this was already used in (Wiskott and Sejnowski, 2002) and was later applied to more complex stimuli for the modeling of Grid, place, head-direction, and view cells in the hippocampus and Invariant visual object recognition.

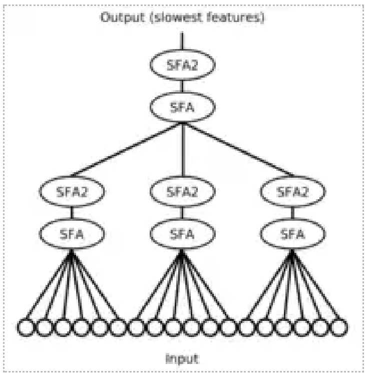

Figure 7: Schematic of a hierarchical SFA Network with two layers. Linear SFA is applied to each receptive field for dimensionality reduction, followed by quadratic SFA (linear SFA after a quadratic expansion).

## Grid, place, head-direction, and view cells in the hippocampus

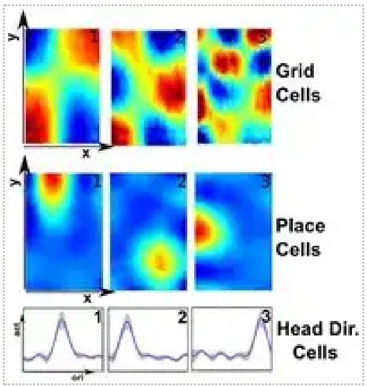

Figure 8: Examples for different simulated cell types from the hippocampal formation: grid cells (top), place cells (middle), and head direction cells (bottom). The upper two graphs show unit activity color coded as a function of location averaged over head direction; the bottom graph shows activity as a function of head direction averaged over location.

The hippocampus is a brain structure important for episodic memory and navigation. In the hippocampus and neighboring areas, a number of cell types have been identified, whose responses correlate with the animal's position and head direction in space. These "oriospatial" cells include place cells, grid cells, head direction cells, and spatial view cells (Figure 8). Grid cells show a regular firing activity on a hexagonal grid in real space (the grid is rectangular in the model). Place cells are typically localized in space, i.e. they fire only in one or few contiguous places. Head direction cells fire in most areas of the environment but each one only near its preferred head direction, while grid and place cells are insensitive to the orientation of the animal. These cells are driven by input from different modalities, such as vision, smell, audition etc. In comparison with the rapidly changing visual input during an animal's movement in a natural environment, the firing rates of oriospatial cells change relatively slowly. This observation is the basis of a model of unsupervised formation of such cells based on visual input with slow feature analysis and sparse coding (Franzius, Sprekeler, Wiskott 2007). A closely related model has earlier been presented by Wyss et al (2006).

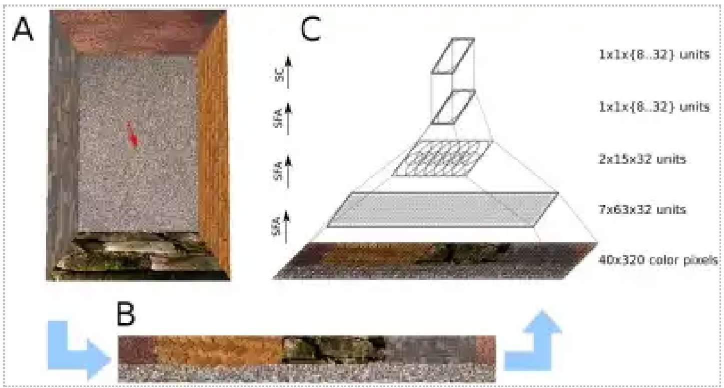

Figure 9: Architecture of the hierarchical model for spatial learning. For a given position and orientation of the virtual rat (red arrow in A) in the naturally textured virtual-reality environment (A) input views are generated (B) and processed in a hierarchical network (C). Units in the lower three layers all perform the same sequence (D) of linear SFA (for dimensionality reduction), expansion, additive noise, linear SFA (for feature extraction), and clipping; the last layer performs linear sparse coding.

The model architecture is depicted in Figure 9C. It consists of a hierarchical network, the first three layers of which are trained with SFA with a quadratic expansion. The last layer, which is linear, is optimized to maximize sparseness, meaning that as few units as possible should be active at any given time while still representing the input faithfully. The network is trained with visual input (Figure 9B) as perceived by a virtual rat running through a textured environment (Figure 9A). It is easy to imagine that the color value of each pixel of such an input fluctuates on a fast time scale while the rat changes position and orientation on a much slower time scale. Since SFA extracts slow features, it computes a representation of position and orientation from the fluctuating pixel values. Depending on the time scales of rotation and translation of the virtual rat, this can either be a spatial code invariant to the head direction or a directional code invariant to spatial position, the more slowly changing parameter dominates the code. With slow translation, SFA alone gives rise to regular firing activity on a spatial grid, see Figure 8 top. Sparse coding then generates responses as known from place cells, see Figure 8 middle. With slow rotation, SFA and sparse coding lead to responses as known from head direction cells, see Figure 8 bottom.

The model computes its spatial representation based on current visual input. There is no temporal delay or integration involved, which is consistent with the rapid firing onset of place and head direction cells when lights are switched on in a previously dark room. However, animals can approximately determine their current position also in a dark room by integrating their own movement from an initially known position, a process called path integration or dead reckoning. For instance, when a rat starts in one corner of a dark room and goes ten steps along one wall, then takes a 90 degree turn and goes another 5 steps into the room, it knows where it is even without any visual input. These two different techniques, sensory driven navigation and path integration, complement each other in real animals, but only the first one is modeled here.

## Invariant visual object recognition

In object recognition tasks the identity of objects is typically not the only relevant information. Just as important is the configuration of the objects (e.g. the position and orientation of an object). The identities of objects and their configurations are typically slow features in the sense of SFA. After training a hierarchical SFA network with visual input data showing single objects moving about, the network should therefore be able to extract features like object identity and configuration. Another important aspect is that ideally the individual features should be independent of each other, i.e., one wants a position representation that is invariant under changes in object orientation. It has been shown that for simple situations a hierarchical SFA network is indeed able to directly extract the desired features (Figure 10).

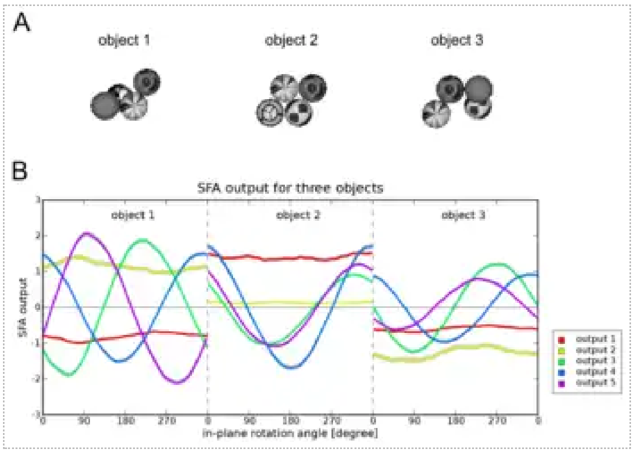

Figure 10: Output of a hierarchical SFA network used for object recognition. The five slowest features (B) are plotted over the in-plane angle for the three different objects (A) on which the network has been trained. Object identity can easily be deduced from the first two outputs (yellow and red lines), which resemble step functions and are largely angle invariant. The following outputs contain angle information. All five outputs are practically translation invariant (the gray areas around the lines show the standard deviation under translation). The movement range of the objects is about one object diameter both horizontally and vertically.

In more complicated situations (e.g., more objects or more configuration features) it is generally not possible to directly interpret the output of a hierarchical SFA network in terms of identity and attributes of individual objects. Nevertheless, the relevant features are much more accessible after the data has been processed by the SFA network and can be easily recovered with an additional post-processing step, using simple supervised or unsupervised methods like linear regression (Franzius et al. 2008) or reinforcement learning (Legenstein, Wilbert, Wiskott 2010). Other examples for the use of slowness for object recognition can be found in (Wallis et al. 1997) and (Einhäuser et al. 2005).

# Technical applications

## Extraction of driving forces from nonlinear dynamical systems

Nonlinear dynamical systems (http://www.scholarpedia.org/article/Encyclopedia_of_dynamical_systems) can be observed by monitoring one or several of their variables over time. The resulting time series can be quite complex and difficult to analyze. Dynamical systems usually have some internal parameters. If these parameters change slowly over time, they are called driving forces, and the analysis of the resulting time series is even more difficult. Since the driving forces usually change more slowly than the variables of the system, they can be estimated in an unsupervised fashion by slow feature analysis (Wiskott, 2003b). Knowing the time course of the driving forces can be useful in itself or can subsequently simplify the analysis of the dynamical system.

As a simple example consider an iterative tent-map $f(w)$ (Figure 11, black curve). Starting from an arbitrary value $w_0$ between 0 and 1 and then repeatedly applying $f(w)$ results in a discrete time series $w_1 = f(w_0), w_2 = f(w_1), w_3 = f(w_2), \ldots$ The tent-map can be cyclically shifted by $\gamma$ within the interval [0,1] (Figure 11, red curve). If this shift is slower than the dynamics of the system, it is a driving force. Figure 12 (top) shows the resulting time series for $\gamma$ changing like shown by the solid line

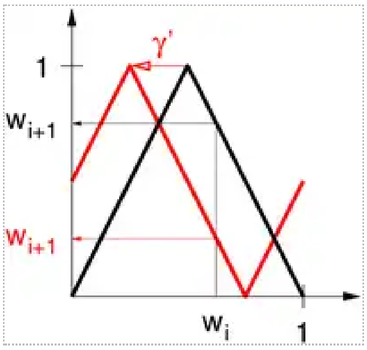

Figure 11: Tent map function (black) and its cyclically shifted version (red).

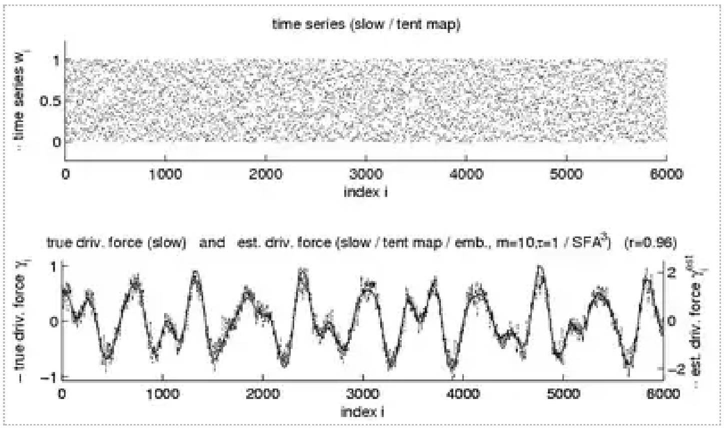

Figure 12: Time series of the iterative tent map (top) produced with a slowly varying cyclic shift (bottom, solid line). The first SFA output (bottom, dots) is highly correlated ($r = 0.96$) with the true driving force.

in the bottom graph of the same figure. There is no obvious indication of the changing driving force in this time series.

A problem in analyzing this time series with SFA is that it is only one-dimensional, so that a single data point does not carry much information about the current state of the system and its driving force. Such a problem is commonly solved by time embedding, i.e. by considering several successive time points simultaneously. In this case 10 successive time points are taken to form a 10-dimensional input vector, with a shift by one time point from one to the next input vector. Let $w_i$ indicate the one-dimensional time series, with $i$ indicating time in units of iterations. Then the input vectors are $\vec{x}_i := (w_{i-4}, w_{i-3}, w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}, ..., w_{i+4}, w_{i+5})^T$. To these input vectors slow feature analysis can be applied successfully with polynomials of degree 3. The dots in the bottom graph of Figure 12 show the first SFA component, which is highly correlated with the true driving force (correlation $r = 0.96$). Thus, SFA was able to extract the driving force from the observed time series in an unsupervised manner.

## Nonlinear blind source separation (xSFA)

The task in blind source separation (BSS) (http://en.wikipedia.org/wiki/Blind_signal_separation) is to recover *source* signals from observed time series where these signals have been mixed together. An illustrative example involves two persons (the sources) in a room talking simultaneously while recorded by two separate microphones (yielding the mixtures). Generally, the sources are assumed to be statistically independent. If the mixtures are linear in the sources, the problem is reduced to that of independent component analysis (ICA) (http://en.wikipedia.org/wiki/Independent_component_analysis) , for which powerful algorithms are readily available. If the relation between the mixtures and the sources is nonlinear, however, the problem is much harder, because many nonlinear transformations of the mixtures generate independent signals.

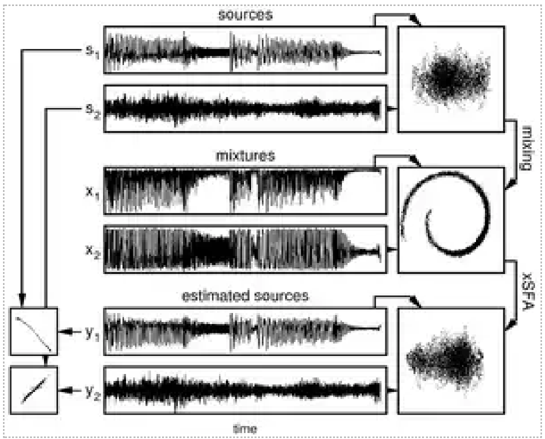

Figure 13: An application of the xSFA algorithm to a nonlinear mixture of audio signals. $s_1$ and $s_2$ are the original sources; $x_1$ and $x_2$ are the nonlinear mixtures; $y_1$ and $y_2$ are the estimated sources extracted with xSFA. On the right are shown scatter plots of each pair of signals. The scatter plots at the lower left illustrate the correlation between extracted and original sources.

Two insights make SFA a good candidate for nonlinear blind source separation:

- A nonlinear transformation of a time-varying signal typically varies more quickly than the original signal. Therefore, SFA will prefer the sources over nonlinearly distorted versions of the sources.
- A linear mixture of two signals varies more quickly than the slower of the two. This suggests that SFA tends to separate the sources.

As a consequence, the slowest signal that is found by applying SFA to the nonlinearly expanded mixture is likely to be the slowest source (or, more precisely, an invertible transformation thereof). This serves as the starting point for extended Slow Feature Analysis (xSFA), an algorithm for nonlinear blind source separation (Sprekeler et al., 2010). The idea is that once the first source is known, it can be removed from the mixture. The slowest signal that can be extracted from the remaining, reduced mixture is the slowest of the remaining sources. After both the first and the second source are removed from the data, SFA should extract the third source. Iteration of this scheme should in principle yield all the sources. See Figure 13 for an example with two sources. The algorithm is closely related to the kTDSEP algorithm proposed by Harmeling et al. (2003).

The algorithm rests on a solid theoretical foundation (see section Statistically Independent Sources).

# Theory of slow feature analysis

## The "Harmonic Oscillation" Result

The SFA objective allows deriving analytical solutions for some interesting cases. Wiskott (2003a) has shown that, without any constraints from the input, the optimal output signals of SFA are harmonic oscillations, with faster signals (higher index $j \Leftrightarrow$ larger $\Delta$-value) oscillating at higher frequencies. Although such optimal output signals do not depend on the input signals, and as such can occur only in case of extreme overfitting, this result helps significantly in the interpretation of many simulation results:

- Complex cells: The orientation and frequency tuning of the complex cell units simulated by Berkes and Wiskott (2005) can be understood as a means of generating harmonic oscillations when the input patches are rotated or zoomed at constant velocity.
- Place- and Head-direction cells: The position and head-direction dependence of the highest SFA units in the hierarchical spatial learning architecture is such that when the animal moves or turns at constant speed, the output signals are harmonic oscillations.
- Invariant Object Recognition: The orientation dependence of the invariant object recognition system leads to sinusoidal output signals if the objects are rotated (in-plane) with constant velocity.

## Input Signals from a Manifold

The intuition gained from the "harmonic oscillation" result are supported by further theoretical analysis of the case where the input data lie on a smooth manifold (http://en.wikipedia.org/wiki/Manifold) , i.e. on a curved surface that is embedded in the potentially high-dimensional input space (Franzius et al, 2007; Sprekeler et al., 2010). The assumptions are:

- The training data are sampled from a smooth manifold with a probability distribution $p(s, \dot{s})$. Here, $s$ is either the input data or an arbitrary parametrization of the manifold and $\dot{s}$ is its derivative. The theory treats the limit of infinite amount of training data, where this distribution is fully sampled.
- The function space of SFA is sufficiently rich to generate arbitrary (smooth) functions on this input manifold.

Under these assumptions, the optimal functions $g_j(s)$ can be thought of as standing waves on the input manifold. In mathematical terms, they are the solutions of a partial differential eigenvalue problem (Franzius et al., 2007)

$$-\nabla_s \cdot p(s) K(s) \nabla_s g_j(s) = \Delta_j p(s) g_j(s)$$

where $K(s)$ is a matrix that contains the second moments of the velocity, conditioned on $s$ :

$K(s) = \langle \dot{s}\dot{s}^T \rangle_{\dot{s}|s} = \int p(\dot{s}|s) \dot{s}\dot{s}^T \mathrm{d}\dot{s}$.

The eigenvalue equation is complemented by von Neumann boundary conditions, i.e., for every boundary point $s$ with normal vector $n(s)$ :

$n(s) \cdot K(s) \nabla_s g_j = 0$ .

The eigenvalue equation has the structure of a Sturm-Liouville problem, i.e. it is a generalized wave equation. In line with the harmonic oscillation result, the optimal functions are oscillatory eigenmodes on the input manifold, with the $\Delta$-value $\Delta_j$ corresponding to the squared oscillation frequency. In this respect, the optimal functions of SFA bear similarities with the Fourier modes of the input manifold. The mathematical structure of the eigenvalue equation ensures that the output signals of the eigenfunctions obey the zero mean and the decorrelation constraint.

In cases where the input manifold is low-dimensional, the optimal functions can be calculated analytically. The highest SFA modules in the hierarchical spatial learning architecture are an illustrative example: The visual input signal is fully determined by the position and head direction of the simulated rat. Therefore, these three parameters form a parametrization of the input manifold. The theoretically optimal functions on this manifold (assuming a uniform distribution $p(s,\dot{s}) = p(s)$) are indeed given by the Fourier modes. This theoretical prediction closely matches the simulation results, as shown in Figure 14.

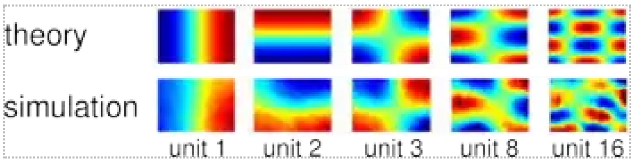

Figure 14: Comparison of the analytically derived (top) and simulated (bottom) position dependence of the optimal functions for the hierarchical spatial learning architecture .

## Recovering Statistically Independent Sources

These theoretical results are instrumental in proving that, under certain conditions, SFA is able to reconstruct the original sources of the input signals, even when the sources were non-linearly mixed. This is the case for statistically independent (http://en.wikipedia.org/wiki/Independence_(probability_theory)) sources, for a set of signals, for which no individual signal conveys information about the others. The mathematical form of this assumption is that both the different components $s_i$ of the manifold parametrization and their derivatives are statistically independent, so that the probability $p(\mathbf{s})$ factorizes and the matrix $K$ of the second moments of the velocities is diagonal. In this case, the optimal functions $g_j$ can be shown to be products of functions $f_{i\alpha}(s_i)$, each of which depends on only one of the sources. The functions $f_{i\alpha}$ are again the solution of a Sturm-Liouville problem

$$-\frac{d}{ds_i}\, p(s_i) K_i(s_i)\, \frac{d}{ds_i}\, f_{i\alpha}(s_i) = \lambda_\alpha p(s_i) f_{i\alpha}(s_i)\ , \tag{6}$$

again with von Neumann boundary conditions.

Although the optimal functions for the full SFA problem are products of the functions $f_{i\alpha}$, it turns out that the functions themselves are also optimal functions that will be part of the full solution, suggesting that some of the output signals of SFA should depend on one of the sources only. Moreover, it can be shown that the slowest non-constant function $f_{i1}$ has a monotonic dependence on its source, suggesting that some of the output signals of SFA are relatively undistorted versions of the original sources. This property can be exploited for the reconstruction of the sources, even when the input data are highly nonlinear mixtures of the sources (see section on Nonlinear blind source separation).

## Transformation-Based Input Signals

Analytical results for SFA can also be obtained for restricted function spaces, when the training data are generated by applying continuous transformations to a set of static templates. The complex cell simulations (Berkes and Wiskott, 2005) are a good example: The input data are generated by moving, rotating and zooming a set of static natural images. For an analytical treatment of SFA for this class of data, it is necessary to assume that (a) the transformations form a Lie group (http://en.wikipedia.org/wiki/Lie_group) and (b) that the statistics of the training data is invariant with respect to these transformations. An example for the invariance assumption would be translation invariance in natural images, where the invariance would mean that the statistics of the full image ensemble (not the statistics of any given image) remains untouched, if all images are shifted by the same amount.

As for manifold-based input signals, the optimal functions $g_j$ are the solutions of an eigenvalue equation

$$Dg_j = \lambda g_j\ , \tag{7}$$

where the operator $D$ is a quadratic form in a set of operators that are often denoted as the generators $G_\alpha$ of the transformation group

$$D = -\sum_{\alpha,\beta} \langle v_\alpha v_\beta \rangle G_\alpha G_\beta\ . \tag{8}$$

Here, $v_\alpha$ denotes the velocity of the transformation that is associated with the generator $G_\alpha$. More information about generators can be found here (http://en.wikipedia.org/wiki/Lie_group) (sections on Lie algebras and the exponential map).

A central result of the theory is that the eigenvalue equation (7) is independent of the statistics of the templates and relies purely on the nature and velocity statistics of the transformations. This explains the observation of Berkes and Wiskott (2005) that the structure of the simulated receptive fields is strongly affected by the nature of the transformations but largely independent of higher order image statistics.

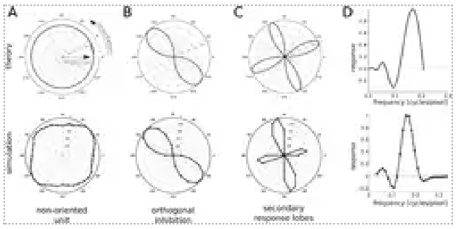

Figure 15: Comparison of the analytically derived (top) and simulated (bottom) orientation and frequency tuning of the optimal functions.

For the concrete example of learning complex cells with second-order SFA, the generators are known and the eigenvalue equation (7) can be solved analytically for the case of translation-invariant functions. The analytical solution reproduces several properties of the simulated receptive fields, including the grating-structure of the optimal stimuli and their orientation and frequency tuning (Figure 15). Side- and end-inhibition effects can be interpreted as a weak breaking of translation invariance.

## Software

The Slow Feature Analysis algorithm is publicly available in Python and Matlab.

- The Modular Toolkit for Data Processing (MDP) (http://mdp-toolkit.sourceforge.net/) is an open-source Python implementation that allows to perform nonlinear SFA, to build hierarchical SFA networks, and to easily combine SFA with other algorithms, for example for classification or dimensionality reduction.
- sfa-tk (http://people.brandeis.edu/~berkes/software/sfa-tk/index.html) is an open-source Matlab implementation of SFA. The library defines functions for linear and quadratic SFA, and allows to define non-linear expansions in custom function spaces.

## References

- Becker, S and Hinton, GE (1992). A self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature* 355(6356): 161-163. doi:10.1038/355161a0 (http://dx.doi.org/10.1038/355161a0) .
- Berkes, P and Wiskott, L (2005). Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of Vision* 5(6): 579-602. doi:10.1167/5.6.9 (http://dx.doi.org/10.1167/5.6.9) .
- Blaschke, T and Berkes, P, and Wiskott (2006). What is the relationship between slow feature analysis and independent component analysis? *Neural Computation* 18(10): 2495-2508. doi:10.1162/neco.2006.18.10.2495 (http://dx.doi.org/10.1162/neco.2006.18.10.2495) .
- Creutzig, F and Sprekeler, H (2008). Predictive coding and the slowness principle: An information-theoretic approach. *Neural Computation* 20(4): 1026-41. doi:10.1162/neco.2008.01-07-455 (http://dx.doi.org/10.1162/neco.2008.01-07-455) .
- De Valois, RL; Yund, EW and Hepler, N (1982). The orientation and direction selectivity of cells in macaque visual cortex. *Vision Research* 22(5): 531-544. doi:10.1016/0042-6989(82)90112-2 (http://dx.doi.org/10.1016/0042-6989(82)90112-2) .
- Einhhäuser, W, and Hipp and J, and Eggert, J, and Körner E, and König P, (2005). Learning viewpoint invariant object representations using a temporal coherence principle. *Biological Cybernetics* 93: 79–90.
- Földiák, P (1991). Learning invariance from transformation sequences. *Neural Computation* 3(2): 194-200. doi:10.1162/neco.1991.3.2.194 (http://dx.doi.org/10.1162/neco.1991.3.2.194) .
- Franzius, M and Sprekeler, H, and Wiskott (2007). Slowness and sparseness lead to place, head-direction, and spatial-view cells (http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.0030166) . *PLoS Computational Biology* 3(8): e166. doi:10.1371/journal.pcbi.0030166 (http://dx.doi.org/10.1371/journal.pcbi.0030166) .
- Franzius, M and Wilbert, N, and Wiskott (2008). Invariant object recognition with slow feature analysis (http://www.springerlink.com/content/239862780068tw81/) . *Proc. 18th Int. Conf. on Artificial Neural Networks* ICANN 08: Prague. doi:10.1007/978-3-540-87536-9_98 (http://dx.doi.org/10.1007/978-3-540-87536-9_98) .
- Harmeling, S and Ziehe, A (2003). Kernel-Based Nonlinear Blind Source Separation (http://people.kyb.tuebingen.mpg.de/harmeling/pubs/article_on_ktdsep.pdf) *Neural Computation* 15(5): 1089-1124. doi:10.1162/089976603765202677 (http://dx.doi.org/10.1162/089976603765202677) .

- Helmholtz, HL (1910). Treatise on Physiological Optics, III: The Perceptions of Vision *Southall JPC, ed. Rochester N.Y.: Optical Society of America. Laplace, Pierre Simon, Philosophical Essay on Probabilities, translated from the fifth French edition of 1825 by Andrew I. Dale. Springer-Verlag: New York (1995)* 1: 120.
- Hinton, GE (1989). Connectionist learning procedures. *Artificial Intelligence* 40(1-3): 185-234. doi:10.1016/0004-3702(89)90049-0 (http://dx.doi.org/10.1016/0004-3702(89)90049-0) .
- Körding, KP; Kayser, C; Einhäuser, W and König, P (2004). How are complex cell properties adapted to the statistics of natural scenes? *Journal of Neurophysiology* 91(1): 206-212. doi:10.1152/jn.00149.2003 (http://dx.doi.org/10.1152/jn.00149.2003) .
- Legenstein, R and Wilbert, N and Wiskott, L (2010). (http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.1000894) Reinforcement Learning on Slow Features of High-Dimensional Input Streams. PLoS Comput. Biol. 2010; 6(8): e1000894. doi:10.1371/journal.pcbi.1000894.
- Mitchison, G (1991). Removing time variation with the anti-Hebbian differential synapse. *Neural Computation* 3(3): 312-320. doi:10.1162/neco.1991.3.3.312 (http://dx.doi.org/10.1162/neco.1991.3.3.312) .
- Shaw, J (2003). Predictive coding with temporal invariance. (http://hdl.handle.net/1802/1318) Technical report UR CSD - TR 859, University of Rochester.
- Smith, AM (2001). Alhacen's Theory of Visual Perception: A Critical Edition, with English Translation and Commentary, of the First Three Books of Alhacen's De aspectibus, the Medieval Latin Version of Ibn al-Haytham's Kitab al-Manazir, volume 91, parts 4-5 *Transactions of the American Philosophical Society* 91: 4-5. doi:10.1086/376125 (http://dx.doi.org/10.1086/376125) .
- Sprekeler, H and Zito, T and Wiskott, L (2010). An extension of slow feature analysis for nonlinear blind source separation, in preparation.
- Turner, RE and Sahani, M (2007). A maximum-likelihood interpretation for slow feature analyis. (http://www.gatsby.ucl.ac.uk/~turner/Publications/TSNCOMP2006v9.pdf) *Neural Computation* 19(4): 1022-1038.
- Wallis, G and and ET, Rolls (1997). Invariant face and object recognition in the visual system. *Progress in Neurobiology* 51(2): 167-194. doi:10.1016/s0301-0082(96)00054-8 (http://dx.doi.org/10.1016/s0301-0082(96)00054-8) .
- Wiskott, L (1998). Learning invariance manifolds. (http://itb.biologie.hu-berlin.de/~wiskott/Abstracts/Wis98a.html) Proc. 5th Joint Symposium on Neural Computation, JSNC'98, San Diego, May 16, publ. University of California, San Diego, pp. 196-203.
- Wiskott, L (2003a). Slow feature analysis: A theoretical analysis of optimal free responses. *Neural Computation* 15(9): 2147-2177. doi:10.1162/089976603322297331 (http://dx.doi.org/10.1162/089976603322297331) .
- Wiskott, L (2003b). Estimating driving forces of nonstationary time series with slow feature analysis. (http://arxiv.org/abs/cond-mat/0312317) *arXiv.org e-Print archive* http://arxiv.org/abs/cond-mat/0312317: 1-8.
- Wiskott, L and Sejnowski, T (2002). Slow feature analysis: Unsupervised learning of invariances. (http://itb.biologie.hu-berlin.de/~wiskott/Abstracts/WisSej2002.html) *Neural Computation* 14(4): 715-770.
- Wyss, R; König, P and Verschure, P (2006). A model of the ventral visual system based on temporal stability and local memory. *PLoS Biology* 4: 120. doi:10.1371/journal.pbio.0040120 (http://dx.doi.org/10.1371/journal.pbio.0040120) .

## External links

Laurenz Wiskott's website (http://www.ini.rub.de/PEOPLE/wiskott/)

Bibliography on Slow Feature Analysis by Laurenz Wiskott (http://www.ini.rub.de/PEOPLE/wiskott/References/SFA-index.html)

Sponsored by: Eugene M. Izhikevich, Editor-in-Chief of Scholarpedia, the peer-reviewed open-access encyclopedia

Reviewed by (http://www.scholarpedia.org/w/index.php?title=Slow_feature_analysis&oldid=80442) : Peter König, Inst. Cognitive Science, Universität Osnabrück, Germany

Reviewed by (http://www.scholarpedia.org/w/index.php?title=Slow_feature_analysis&oldid=87486) : Konrad Paul Kording, Rehabilitation Institute of Chicago, Northwestern University, Chicago, IL, USA

Accepted on: 2011-04-11 21:50:07 GMT (http://www.scholarpedia.org/w/index.php?

title=Slow_feature_analysis&oldid=87486)

Categories: Computational Neuroscience | Vision | Multiple Curators

*This page was last modified on 11 December 2013, at 16:37.*

*This page has been accessed 82,139 times.*