The final version of this article has been published in
Neural Computation, 18(10):2495-2508 (2006) published by The MIT Press.
This version does not differ significantly from the final version.

What is the Relation Between Slow Feature Analysis and Independent Component Analysis?

Tobias Blaschke, Pietro Berkes*, and Laurenz Wiskott
Institute for Theoretical Biology, Humboldt University Berlin
Invalidenstraße 43, D-10115 Berlin, Germany
{t.blaschke,p.berkes,l.wiskott}@biologie.hu-berlin.de
http://itb.biologie.hu-berlin.de/{~blaschke,~berkes,~wiskott}

Abstract

We present an analytical comparison between linear slow feature analysis and second-order independent component analysis, and show that in the case of one time delay the two approaches are equivalent. We also consider the case of several time delays and discuss two possible extensions of slow feature analysis.

1 Introduction

In data analysis it is often desirable to transform the input signals into a new representation that recovers as much information as possible about the underlying processes. In the classical example of two people speaking simultaneously while being recorded with two microphones, for instance, the observed signal is a mixture of their voices. A more useful representation here would be one where each signal component contains only the information about a single speaker. On the other hand, in the visual domain one might be interested in a representation that is invariant to typical transformations, such as translation or zoom. A variety of linear and nonlinear methods have been developed to extract the interesting features from an observed signal.

In this paper we focus on two methods that consider different properties of the observed signal, namely Independent Component Analysis (ICA) (see Hyvärinen et al., 2001, for an overview) and Slow Feature Analysis (SFA) (Wiskott and Sejnowski, 2002). ICA finds a representation of the data such that signal components are mutually statistically independent, which can be used to separate the two speakers in the example above. SFA on the other hand extracts slowly-varying features, which can be used in the second example to learn visual invariances. At first glance these two methods are very different and even seem to be conflicting, since two slowly-varying signals of finite length are intuitively more likely to have statistical dependencies than quickly-varying ones. However, we will see that ICA and SFA do have common properties, which we are going to point out by comparing the two algorithms mathematically.

*Current address: Gatsby Computational Neuroscience Unit, London WC1N 3AR, UK

To carry out the comparison we have to apply some restrictions. SFA is constrained to non-white signals with a temporal structure (e.g., speech signals) and it is based on second-order statistics. We therefore compare it to ICA algorithms that only use second-order information and need a temporally structured signal as well (Molgedey and Schuster, 1994; Belouchrani et al., 1997; Ziehe and Müller, 1998; Zibulevsky and Pearlmutter, 2000; Nuzillard and Nuzillard, 2003). SFA is usually applied as a nonlinear method: It uses a nonlinear expansion to map the input signal into a feature space and then solves a linear problem there. ICA on the other hand is typically a linear method, since in the nonlinear case the problem is in general underdetermined (because the solution is not unique) and there is thus no guarantee to recover the original sources (Hyvärinen and Pajunen, 1999; Jutten and Karhunen, 2003). (There do however exist some nonlinear approaches that make additional assumptions about the nonlinear mapping or the input data.) To make a comparison between the two methods possible, we will restrict SFA to the linear case. Nevertheless, all calculations in this paper are essentially the same for linear or nonlinear SFA.

2 Linear Mixing and Unmixing

Let $\mathbf{x}(t) = [x_1(t), \ldots, x_N(t)]^T$ be a linear mixture of a multidimensional source signal $\mathbf{s}(t) = [s_1(t), \ldots, s_N(t)]^T$:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)\,, \tag{1}$$

where $\mathbf{A}$ is a square mixing matrix and different components s_i come from statistically independent sources. In the following we will assume that $\mathbf{s}(t)$ and $\mathbf{x}(t)$ have zero mean, without loss of generality. A common linear preprocessing step in many ICA algorithms as well as in linear SFA is the whitening of the input signal $\mathbf{x}(t)$. Whitening results in a signal $\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t)$ with mutually uncorrelated components, $\langle y_i(t) y_j(t)\rangle = 0 \quad \forall\, i \neq j$, unit variance, $\langle y_i(t)^2\rangle = 1$, and zero mean, $\langle y_i(t)\rangle = 0$, where $\langle\cdot\rangle$ denotes averaging over time. It can be shown that after the whitening step an orthogonal transformation $\mathbf{Q}$ on $\mathbf{y}$ is sufficient to yield independent components (Comon, 1994) or slowly-varying features (Wiskott and Sejnowski, 2002). Therefore the output signal $\mathbf{u}(t)$ can be obtained by combining the whitening matrix $\mathbf{W}$ and a rotation matrix $\mathbf{Q}$

$$\mathbf{u}(t) = \mathbf{Q}\mathbf{y}(t) = \mathbf{Q}\mathbf{W}\mathbf{x}(t)\,. \tag{2}$$

In the following we will always assume whitened data $\mathbf{y}(t)$ and focus on finding $\mathbf{Q}$. Since zero mean and whitening are preserved under any orthogonal transformation, the components of $\mathbf{u}(t)$ also satisfy the three conditions:

$$\langle u_i(t)\rangle = 0 \qquad \text{(zero mean)}, \tag{3}$$

$$\langle u_i(t)^2\rangle = 1 \qquad \text{(unit variance)}, \tag{4}$$

$$\forall i \neq j: \quad \langle u_i(t) u_j(t)\rangle = 0 \qquad \text{(decorrelation)}. \tag{5}$$

These properties fulfill the constraints imposed by SFA (cf. Sec. 4) and are a good prerequisite for ICA because they constrain the output signals $u_i(t)$ to be statistically independent in the first and second order.

3 Second-Order Independent Component Analysis

Given the linear mixture (1) ICA tries to retrieve the source signal components $\mathbf{s}(t)$ from the input signal $\mathbf{x}(t)$. The mixing matrix $\mathbf{A}$ is unknown and the source signal components are assumed to be mutually independent. The typical approach is to define an objective function that is a measure

of independence of the estimated source signal components u_i. The problem is then solved by optimizing this function with respect to $\mathbf{Q}$.

There exist different measures of independence. Most algorithms are based on the assumption that two signals are independent if their joint distribution is equal to the product of their marginals (e.g. Cardoso and Souloumiac, 1993; Hyvärinen, 1999; Lee et al., 1999). A corresponding measure in this case is the Kullback-Leibler divergence. We will refer to this approach as *higher-order ICA*.

This definition, however, does not capture all aspects of independence: Consider a signal without temporal auto-correlation (e.g., white noise) and a second signal that is equal to the first one but shifted in time. Applying the measure of independence mentioned above, the two signals appear to be independent although they are actually a time shifted copy of each other and thereby intuitively strongly dependent. This dependence across time can be taken into account using a different measure where two signals are considered statistically independent if all time-delayed correlations are zero (*second-order ICA*) (Molgedey and Schuster, 1994; Belouchrani et al., 1997; Ziehe and Müller, 1998). In order to successfully apply this measure the source signals need to have a time structure (must be non-white), which is also a necessary condition for SFA. An alternative formulation of this idea is to use a model of the sources that includes a dynamic in time and assume that the time series are independent as a whole (Pearlmutter and Parra, 1996). In this paper we are going to study algorithms based on this latter definition of independence, following the formulation by Molgedey and Schuster (1994).

To derive an objective function for second-order ICA we first introduce time-delayed correlation matrices of the estimated source signal $\mathbf{u}(t)$,

$$\mathbf{C}^{(\mathbf{u})}(\tau) := \left\langle \mathbf{u}(t)\mathbf{u}(t+\tau)^T \right\rangle , \tag{6}$$

where τ is the time delay between two signals. We denote the entries of $\mathbf{C}^{(\mathbf{u})}(\tau)$ as $C_{ij}^{(\mathbf{u})}(\tau)$. For a signal $\mathbf{u}(t)$ with independent components, $\mathbf{C}^{(\mathbf{u})}(\tau)$ should be diagonal for all τ. We are therefore looking for an objective function that, when optimized, jointly diagonalizes those matrices.

It is common in practice to use a symmetrized version of the correlation matrices[1]:

$$\mathbf{C}^{(\mathbf{u})}(\tau) \quad := \quad \frac{1}{2}\left[\left\langle \mathbf{u}(t)\mathbf{u}(t+\tau)^T \right\rangle + \left\langle \mathbf{u}(t+\tau)\mathbf{u}(t)^T \right\rangle\right] . \tag{7}$$

Computing the symmetrized matrices is equivalent to applying the algorithm to the original input data and to the data reversed in time (because $\left\langle \mathbf{u}(t+\tau)\mathbf{u}(t)^T \right\rangle = \left\langle \mathbf{u}(t)\mathbf{u}(t-\tau)^T \right\rangle$). This reflects the fact that, with respect to the unmixing problem, the time direction is not important. Moreover, the symmetric form can always be diagonalized with a rotation matrix (while the non-symmetric matrices can have complex eigenvalues and eigenvectors) and has better numerical properties. Note, however, that in some pathological cases the cross-correlation terms can cancel out each other: For example, if $\mathbf{u}(t) = [\sin(t), \cos(t)]^T$ there clearly are cross-correlations but in the symmetrized version the off-diagonal terms in (7) are zero for all τ. The two signals are thus considered independent by the algorithm.

We will first focus on the case of a single time delay τ (Molgedey and Schuster, 1994). The extension to more than one time-delayed correlation matrix is straightforward and will be described in Section 5. Because of the whitening step (5) the correlation matrix with time delay zero is already diagonal. With one time delay the ICA algorithm thus reduces to diagonalizing a single time-delayed correlation matrix $\mathbf{C}^{(\mathbf{u})}(\tau)$. This can be achieved by using the method of Jacobi (Cardoso and Souloumiac, 1996) to minimize the sum of the squared off-diagonal entries, a technique used in several second-order ICA algorithms (Belouchrani et al., 1997; Ziehe and Müller, 1998) as well as

[1] In (Ziehe and Müller, 1998) the correlation matrices are not explicitly defined in the paper but the Matlab implementation made available by the authors uses the symmetric form.

in methods based on higher-order statistics (Cardoso and Souloumiac, 1993). Using this method we can define a simple objective function subject to minimization

$$\Psi_{\mathrm{ICA}} := \sum_{\substack{i,j=1 \\ i \neq j}}^{N} (C_{ij}^{(\mathbf{u})}(\tau))^2 \tag{8}$$

$$= \sum_{i \neq j} \left(\mathbf{q}_i^T \mathbf{C}^{(\mathbf{y})}(\tau) \mathbf{q}_j \right)^2 \tag{9}$$

where $\mathbf{q}_i$ is the i-th row of $\mathbf{Q}$. Ψ_{ICA} is a function of the vectors $\mathbf{q}_i$, which are subject to learning, and of the whitened signal $\mathbf{y}(t)$, which is given. This objective function is optimized by a sequence of elementary rotations within the plane spanned by two axes. A possible optimization procedure has been described by Cardoso and Souloumiac (1996); a more efficient optimization schedule has been derived by Blaschke and Wiskott (2004a).

4 Linear Slow Feature Analysis

Given a whitened input signal $\mathbf{y}(t) = [y_1(t), \ldots, y_N(t)]^T$, linear SFA finds a rotation matrix $\mathbf{Q}$ such that the components u_i of the output signal $\mathbf{u}(t) = \mathbf{Q}\mathbf{y}(t)$ vary as slowly as possible in time and are ordered by decreasing slowness (the first one being the slowest possible, the second one the next slowest uncorrelated to the first, etc.). As a measure of slowness we define (small values indicating slowly-varying signals)

$$\Delta(u_i) := \left\langle \dot{u}_i(t)^2 \right\rangle , \tag{10}$$

which has to be minimized (Wiskott and Sejnowski, 2002). Due to the earlier whitening step, each output signal $u_i(t)$ has zero mean and unit variance (3, 4). This ensures that the solution will not be the trivial solution $u_i(t) = \text{const}$. The decorrelation of the output signals (5) guarantees that different components carry different information.

We will first show how to solve the optimization problem of SFA in a way similar to that described by Wiskott and Sejnowski (2002) and then establish a link between SFA and second-order ICA. For discrete time series the first derivative of $\mathbf{u}(t)$ can be approximated in the first order by

$$\dot{\mathbf{u}}(t) \approx \mathbf{u}(t+1) - \mathbf{u}(t). \tag{11}$$

Using this approximation we can rewrite the SFA objective function (10) as

$$\Delta(u_i) \approx \left\langle (u_i(t+1) - u_i(t))^2 \right\rangle \tag{12}$$

$$= \langle u_i(t+1)u_i(t+1) \rangle + \langle u_i(t)u_i(t) \rangle - \langle u_i(t)u_i(t+1) \rangle - \langle u_i(t+1)u_i(t) \rangle \tag{13}$$

$$= 2 \left\langle u_i(t)^2 \right\rangle - 2 \langle u_i(t)u_i(t+1) \rangle \tag{14}$$

(since $\left\langle u_i(t+1)^2 \right\rangle = \left\langle u_i(t)^2 \right\rangle$ because we average over all t)

$$= 2 - 2 \langle u_i(t)u_i(t+1) \rangle \tag{15}$$

(since $\left\langle u_i(t)^2 \right\rangle = 1$ because $\mathbf{u}(t)$ is white (4)).

Since the constant factor does not matter during optimization, instead of minimizing $\Delta(u_i)$ we can

maximize

$$\begin{aligned}\widetilde{\Delta}\,(u_i) &:= 1 - \frac{1}{2}\Delta\,(u_i) && (16)\\ &= \langle u_i(t)u_i(t+1)\rangle && (17)\\ &= C^{(\mathbf{u})}_{ii}(1) && (18)\\ &= \mathbf{q}_i^T\mathbf{C}^{(\mathbf{y})}(1)\mathbf{q}_i\,. && (19)\end{aligned}$$

The objective function $\widetilde{\Delta}\,(u_i)$ is a function of the rotation matrix $\mathbf{Q}$ and we are thus searching for the orthogonal weight vectors $\mathbf{q}_i$ in (19) that maximize $\widetilde{\Delta}(u_i)$. The solution for $i = 1$ is obviously the eigenvector of the largest eigenvalue of $\mathbf{C}^{(\mathbf{y})}(1)$, which yields the slowest component $u_1(t) = \mathbf{q}_1^T\,\mathbf{y}(t)$. The following eigenvectors in order of decreasing eigenvalue yield the next slowest components, $u_2(t), u_3(t)$, and so forth.

Therefore, to extract all slow components the maximization problem (19) can be formulated as an eigenvalue problem

$$\mathbf{C}^{(\mathbf{y})}(1)\mathbf{Q}^T = \mathbf{Q}^T\mathbf{\Lambda} \tag{20}$$

where $\mathbf{\Lambda}$ denotes a diagonal matrix with Λ_{ii} being the i-th largest eigenvalue and $\mathbf{q}_i$ the corresponding eigenvectors.

In order to allow a better comparison with second-order ICA, we now want to deduce an alternative formulation of SFA, i.e. we want to construct an objective function similar to that of second-order ICA. First, we show the equivalence of solving the eigenvalue problem (20) and the diagonalization of $\mathbf{C}^{(\mathbf{u})}(1)$. If we multiply both sides of (20) with $\mathbf{Q}$ we obtain

$$\mathbf{C}^{(\mathbf{u})}(1) = \mathbf{Q}\mathbf{C}^{(\mathbf{y})}(1)\mathbf{Q}^T = \mathbf{\Lambda}\,. \tag{21}$$

Since $\mathbf{\Lambda}$ is diagonal, $\mathbf{C}^{(\mathbf{u})}(1)$ is diagonal, too. Therefore solving the eigenvalue problem for $\mathbf{C}^{(\mathbf{y})}(1)$ is equivalent to finding a rotation matrix $\mathbf{Q}$ such that the time-delayed correlation matrix $\mathbf{C}^{(\mathbf{u})}(1)$ is diagonal. Second, to perform the diagonalization we minimize all off-diagonal entries of $\mathbf{C}^{(\mathbf{u})}(1)$ using the same Jacobi scheme as for second-order ICA (Sec. 3) and define the following objective function for SFA

$$\begin{aligned}\widetilde{\Psi}_{\mathrm{SFA}} &:= \sum_{i\neq j}(C^{(\mathbf{u})}_{ij}(1))^2 && (22)\\ &= \sum_{i\neq j}\left(\mathbf{q}_i^T\mathbf{C}^{(\mathbf{y})}(1)\mathbf{q}_j\right)^2\,. && (23)\end{aligned}$$

Minimizing this expression produces the same slow components $u_1(t), \ldots, u_N(t)$ as obtained by the eigenvalue problem (20), again assuming an additional sorting step. Note also that this is equivalent to a decorrelation of the time derivatives of the output signal components $u_i(t)$ (cf. Wiskott, 2003) since $\langle \dot{u}_i\dot{u}_j\rangle = -2\,C^{(\mathbf{u})}_{ij}(1)$ for $i \neq j$.

Interestingly, the objective function (23) is identical to the one for ICA (9). With this observation we arrive at the important result that **linear SFA is formally equivalent to second-order ICA with time delay one**.

To bring (22) into a form that can be understood more intuitively in the sense of SFA we can use the fact that the sum of all squared entries of correlation matrices with a given time delay τ is invariant under orthogonal transformations

$$\sum_{i,j}(C^{(\mathbf{u})}_{ij}(\tau))^2 = \sum_{i,j}(C^{(\mathbf{y})}_{ij}(\tau))^2 = \text{const.} \tag{24}$$

We can split this sum in two terms

$$\sum_{i,j}(C_{ij}^{(\mathbf{u})}(\tau))^2 = \sum_{i}(C_{ii}^{(\mathbf{u})}(\tau))^2 + \sum_{i\neq j}(C_{ij}^{(\mathbf{u})}(\tau))^2 = \text{const}\,, \tag{25}$$

so that it is easy to see that the minimization of $\widetilde{\Psi}_{\text{SFA}}$ is equivalent to the maximization of

$$\Psi_{\text{SFA}} := \sum_{i=1}^{N}(C_{ii}^{(\mathbf{u})}(1))^2 \tag{26}$$

$$= \sum_{i=1}^{N}\left(\mathbf{q}_i^T\mathbf{C}^{(\mathbf{y})}(1)\mathbf{q}_i\right)^2 . \tag{27}$$

Having started from minimizing temporal variations (10) as an objective for SFA we now arrived at an objective for maximizing squared auto-correlations (26) at time delay one. This relation can be interpreted intuitively: A signal component with a large squared auto-correlation has a high temporal predictability. If the auto-correlation is positive (i.e., $C_{ii}^{(\mathbf{u})}(1) > 0$), predictability implies that the signal component has to vary slowly.

What if the auto-correlation is negative? This could happen if for example $u_i(t)$ has alternating signs for successive data points. Consider the signal

$$u_i(t) := \begin{cases} -1 & \text{for } t \text{ odd} \\ 1 & \text{for } t \text{ even} \end{cases} \tag{28}$$

with $1 \leq t \leq T$. This signal has zero mean and unit variance and thus fulfills Constraints (3) and (4). Furthermore, it is favorable in terms of the objective (26), since $C_{ii}^{(\mathbf{u})}(1)$ has a large absolute value. On the other hand, this is a very quickly-varying component, which might seem paradoxical since maximizing (26) should result in slowly-varying components. This apparent contradiction can be resolved by studying the constraints imposed on the optimization of (26). Since $\mathbf{Q}$ is an orthogonal matrix, the trace of $\mathbf{C}^{(\mathbf{u})}(1)$ is invariant under the transformation $\mathbf{u}(t) = \mathbf{Q}\mathbf{y}(t)$ (e.g., Zurmühl and Falk, 1997). If we consider all N possible components in the optimization procedure, the decrease of one correlation $C_{ii}^{(\mathbf{u})}(1)$ implies the increase of at least one other correlation $C_{jj}^{(\mathbf{u})}(1)$. Therefore extracting the most slowly-varying signals implies that other extracted components correspond to the most quickly-varying signals. Hence, it is reasonable to further minimize negative correlations since this in turn implies that other correlations will be maximized. As above, a successive sorting step is required to bring the components in order of increasing temporal variation.

5 More than one Time Delay

5.1 Second-order ICA

We know that second-order ICA can always be solved with a single time delay (Tong et al., 1991). However, the delay τ has to be chosen properly so that all eigenvalues of $\mathbf{C}^{(\mathbf{y})}(\tau)$ are distinct. To obtain a more robust method one can consider a certain number T of time-delayed correlation matrices with respective time delays $\tau = 1, 2, \ldots, T$ and diagonalize them jointly (Belouchrani et al., 1997; Ziehe and Müller, 1998). This leads to a straightforward extension of objective (8)

subject to minimization

$$\Psi_{\text{ICAj}} := \sum_{\tau=1}^{T} \kappa_\tau \Psi_{\text{ICA}}(\tau) \tag{29}$$

$$= \sum_{\tau=1}^{T} \kappa_\tau \sum_{i \neq j} (C_{ij}^{(\mathbf{u})}(\tau))^2 \tag{30}$$

$$= \sum_{\tau=1}^{T} \kappa_\tau \sum_{i \neq j} \left(\mathbf{q}_i^T \mathbf{C}^{(\mathbf{y})}(\tau) \mathbf{q}_j \right)^2 , \tag{31}$$

where we additionally introduced positive factors κ_τ that allow us to weight correlation matrices with different time delays differently. In (29) we write ICAj for *joint-diagonalization ICA*. Pham and Garat (1997) have derived a formula closely related to (31) with a maximum likelihood approach.

Extending the objective function of ICA in this way leads to the joint diagonalization of several correlation matrices with different time delays. Decorrelation is thus achieved over a time window of length T. It is intuitively clear that by enlarging the window length the unmixing performance should improve until the width of the autocorrelation function is reached. Exceeding this limit would introduce matrices consisting entirely of zero-mean noise, which would degrade the unmixing performance.

5.2 Linear SFA

5.2.1 Joint Diagonalization

We can use an argument similar to the one used for second-order ICA in order to extend SFA to more than a single time delay. Adding more time-delayed auto-correlations increases the temporal predictability of the signal: Knowing the amplitude of a signal at a given time can give a good prediction for the next T time points since they are strongly correlated. Signals with large temporal predictability are in turn likely to be slowly-varying (cf. the end of Sec. 4). Thus an intuitive extension of the normal SFA objective (26) subject to maximization is

$$\Psi_{\text{SFAj}} := \sum_{\tau=1}^{T} \kappa_\tau \Psi_{\text{SFA}}(\tau) \tag{32}$$

$$= \sum_{\tau=1}^{T} \kappa_\tau \sum_{i=1}^{N} (C_{ii}^{(\mathbf{u})}(\tau))^2 \tag{33}$$

$$= \sum_{\tau=1}^{T} \kappa_\tau \sum_{i=1}^{N} \left(\mathbf{q}_i^T \mathbf{C}^{(\mathbf{y})}(\tau) \mathbf{q}_i \right)^2 . \tag{34}$$

As in (29–31), we have introduced weighting factors κ_τ for the delayed correlation matrices. Note that this new objective (33, 34) is again equivalent to the ICA objective (30, 31) due to the constancy of the sum of all squared entries of each time-delayed correlation matrix (25).

We must be careful, however, with this definition for two reasons. First, while the definition of slowness based on $C_{ii}^{(\mathbf{u})}(1)$ corresponds to our intuition of what a slow signal is, $C_{ii}^{(\mathbf{u})}(2)$ can have a large positive value for signal components that we would not consider to be slow at all. In fact, the alternating signal (28) would yield a maximal value for $C_{ii}^{(\mathbf{u})}(2)$. Second, consider the case where two time-delayed auto-correlations have opposite signs, e.g. $C_{ii}^{(\mathbf{u})}(1) < 0$ and $C_{ii}^{(\mathbf{u})}(2) > 0$. Maximizing objective function (33) would favor a decreasing value of $C_{ii}^{(\mathbf{u})}(1)$ (since it is negative)

and an increasing value of $C_{ii}^{(\mathbf{u})}(2)$. The former would intuitively tend to make the signal faster, while the latter would make it slower. Thus, if the auto-correlations of a component have different signs for different time-delays, the objective function appears to be inconsistent, at least for that component. This conflict cannot be solved as easily as the one discussed at the end of Section 4. However, one can at least monitor the signs of the auto-correlations and diagnose the inconsistent cases. It is not clear to us how often these two problems arise in practice. We believe that by weighting the first auto-correlation stronger than the others, e.g. with an exponential decay of the weights, the inconsistencies can be largely avoided.

5.2.2 Linear Filtering

An alternative to the joint diagonalization of several correlation matrices with different time delays in analogy to second-order ICA is to average over a range of time delays within one correlation matrix and diagonalize just this one matrix. To do so we introduce the following new measure of slowness (cf. 16–19):

$$\widetilde{\Sigma}(u_i) := \left\langle u_i(t)\Big(\sum_{\tau=1}^{T}\kappa_\tau\, u_i(t+\tau)\Big)\right\rangle \tag{35}$$

$$= \sum_{\tau=1}^{T}\kappa_\tau \left\langle u_i(t)u_i(t+\tau)\right\rangle \tag{36}$$

$$= \sum_{\tau=1}^{T}\kappa_\tau C_{ii}^{(\mathbf{u})}(\tau) \tag{37}$$

$$= \mathbf{q}_i^T\Big(\sum_{\tau=1}^{T}\kappa_\tau \mathbf{C}^{(\mathbf{y})}(\tau)\Big)\mathbf{q}_i \tag{38}$$

$$=: \mathbf{q}_i^T\widetilde{\mathbf{C}}^{(\mathbf{y})}\mathbf{q}_i\,, \tag{39}$$

with constants κ_τ that weight different time delays differently. This definition differs from that of (16–19) in that $u_i(t)$ should not only be well correlated to the next data point but to a weighted average over the next T data points. This is a straightforward way of taking several time scales into account. Note that the weighted averaging is a linear-filter operation. As in the joint-diagonalization extension, exponentially decaying weights $\kappa_\tau := \exp(-\gamma\tau)$ for different time delays seem to be a suitable choice. With such weights this measure of slowness is similar to the objective of temporal smoothness used by Stone (1995) and somewhat related to the trace learning rules first introduced by Földiák (1991).

Because of the similarity of (39) with (19) we can apply the steps that led from (19) to (27) and derive the following objective function to be maximized

$$\Psi_{\mathrm{SFAl}} := \sum_{i=1}^{N}\big(\widetilde{C}_{ii}^{(\mathbf{u})}\big)^2 \tag{40}$$

$$= \sum_{i=1}^{N}\big(\mathbf{q}_i^T\widetilde{\mathbf{C}}^{(\mathbf{y})}\mathbf{q}_i\big)^2\,, \tag{41}$$

where $\widetilde{\mathbf{C}}^{(\mathbf{u})}$ is defined analogously to $\widetilde{\mathbf{C}}^{(\mathbf{y})}$ and SFAl stands for *linear-filtering SFA*. Since this objective function is based on just one correlation matrix, it does not have the problems mentioned above for the joint-diagonalization extension (Sec. 5.2.1).

Blaschke (2005, sec. 8.2.2) also considered extending SFA by simultaneously minimizing the variance not only of the first but also of higher-order derivatives, which could result in even more stable signals. This would also lead to (41), because discrete approximations of higher-order derivatives involve multiple time delays. In this case, with positive weights for all derivatives, the constants κ_τ in (38) would have values with alternating signs (positive for odd τ and negative for even τ), which is somewhat counterintuitive. We do not fully understand the implications of this effect but believe that higher-order derivatives do not offer a good way of extending SFA to longer time-scales, even though unmixing performance was actually good in some simple examples.

6 Conclusion

The main result of this work is that linear SFA and second-order ICA with time delay one are formally equivalent, see (23) and (9). This is surprising, because SFA and ICA are based on two very different principles: slowness vs. statistical independence. These principles might seem to contradict each other, because two analog signals of finite length would typically become more statistically dependent if they are more slowly varying.

The formal equivalence of linear SFA and second-order ICA with time delay one allows us to apply the intuition we have gained for one algorithm to deepen our understanding of the other. For example, it is known that higher-order ICA applied to natural images learns linear filters similar to Gabor wavelets (e.g. Bell and Sejnowski, 1997; van Hateren and van der Schaaf, 1998), which in turn resemble receptive fields of simple cells in V1. On the other hand, linear SFA (and therefore also second-order ICA with time delay one) applied to natural image sequences learns filters similar to the principal components of natural images, the first of which are effectively spatial low-pass filters and therefore also generate slowly-varying output signals. This suggests that the solutions found by second-order ICA and higher-order ICA can be very different in practice even though both methods try to maximize statistical independence.

Despite the formal equivalence in the linear case and for time delay one, SFA and ICA have different objectives and differ in the more general case.

First, while in standard SFA the time delay is fixed to one due to the approximation of the time derivative, in ICA it can be chosen freely, or one can rather use several correlation matrices with different time delays simultaneously for optimal unmixing (Sec. 5.1). We have seen (Sec. 5.2.1) that the same extension to several time delays can also be used for SFA, but that the algorithm then becomes inconsistent with respect to the slowness objective if the entries of the time-delayed correlation matrices have different signs for different delays. An extension more consistent with the slowness objective is based on linear filtering before computing the time derivative (Sec. 5.2.2). This also introduces several time delays, but in a different way than used for ICA. Thus, when taking several time delays into account, the conceptual differences between ICA and SFA become relevant.

Second, in the nonlinear case many output signal components can be extracted from a lower dimensional input signal. With SFA they would all be uncorrelated and ordered by slowness, in agreement with the definition in Equations (3–5, 10). With second-order ICA they would not be ordered in any way nor would they be statistically independent for dimensionality reasons. The results would therefore be inconsistent with the ICA objective. Thus, in the nonlinear case the conceptual differences between ICA and SFA also matter.

We believe that the close relation between linear SFA and second-order ICA will lead to a way to combine the two algorithms into a nonlinear method for extracting slowly varying *and* statistically independent components and thereby perform nonlinear blind source separation. This is the subject of current research (Blaschke and Wiskott, 2004b, 2005).

Acknowledgment

This work has been supported by a grant to Laurenz Wiskott from the Volkswagen Foundation.

References

Bell, A. J. and Sejnowski, T. J. (1997). The 'independent components' of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338. 9

Belouchrani, A., Abed Meraim, K., Cardoso, J.-F., and Moulines, E. (1997). A blind source separation technique based on second order statistics. *IEEE Transactions on Signal Processing*, 45(2):434–44. 2, 3, 6

Blaschke, T. (2005). *Independent component analysis and slow feature analysis: Relations and combination*. PhD thesis, Humboldt-University Berlin. 8

Blaschke, T. and Wiskott, L. (2004a). CuBICA: Independent component analysis by simultaneous third- and fourth-order cumulant diagonalization. *IEEE Transactions on Signal Processing*, 52(5):1250–1256. 4

Blaschke, T. and Wiskott, L. (2004b). Independent slow feature analysis and nonlinear blind source separation. In *Proc. of the 5th Int. Conf. on Independent Component Analysis and Blind Signal Separation*, Lecture Notes in Computer Science. Springer Verlag. 9

Blaschke, T. and Wiskott, L. (2005). Nonlinear blind source separation by integrating independent component analysis and slow feature analysis. In Saul, L. K., Weiss, Y., and Bottou, L., editors, *Proc. Advances in Neural Information Processing Systems 17*, pages 177–184. The MIT Press. 9

Cardoso, J.-F. and Souloumiac, A. (1993). Blind beamforming for non Gaussian signals. *IEEE Proceedings-F*, 140:362–370. 3, 4

Cardoso, J.-F. and Souloumiac, A. (1996). Jacobi angles for simultaneous diagonalization. *SIAM J. Mat. Anal. Appl.*, 17(1):161–164. 3, 4

Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing*, 36(3):287–314. Special issue on higher-order statistics. 2

Földiák, P. (1991). Learning invariance from transformation sequences. *Neural Computation*, 3(2):194–200. 8

Hyvärinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634. 3

Hyvärinen, A., Karhunen, J., and Oja, E. (2001). *Independent component analysis*. Wiley Series on Adaptive and Learning Systems for Signal Processing, Communications, and Control. John Wiley & Sons, New York. 1

Hyvärinen, A. and Pajunen, P. (1999). Nonlinear independent component analysis: existence and uniqueness results. *Neural Networks*, 12(3):429–439. 2

Jutten, C. and Karhunen, J. (2003). Advances in nonlinear blind source separation. In *Proc. of the 4th Int. Symposium on Independent Component Analysis and Blind Signal Separation*, pages 245–256. 2

Lee, T.-W., Girolami, M., and Sejnowski, T. (1999). Independent component analysis using an extended Infomax algorithm for mixed sub-Gaussian and super-Gaussian sources. *Neural Computation*, 11(2):409–433. 3

Molgedey, L. and Schuster, G. (1994). Separation of a mixture of independent signals using time-delayed correlations. *Physical Review Letters*, 72(23):3634–3637. 2, 3

Nuzillard, D. and Nuzillard, J.-M. (2003). Second-order blind source separation in the Fourier space of data. *Signal Processing*, 83(3):627–631. 2

Pearlmutter, B. and Parra, L. (1996). A context-sensitive generalization of ICA. In *Proc. of the International Conference on Neural Information Processing*. Springer Verlag. 3

Pham, D. and Garat, P. (1997). Blind separation of mixtures of independent sources through a maximum likelihood approach. *IEEE Transactions on Signal Processing*, 45(7):1712–1725. 7

Stone, J. (1995). A learning rule for extracting spatio-temporal invariances. *Network*, 6(3):1–8. 8

Tong, L., Liu, R., Soon, V. C., and Huang, Y.-F. (1991). Indeterminacy and identifiability of blind identification. *IEEE Transactions on Circuits and Systems*, 38(5):499–509. 6

van Hateren, J. and van der Schaaf, A. (1998). Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc. R. Soc. Lond. B*, 265:359–366. 9

Wiskott, L. (2003). Slow feature analysis: A theoretical analysis of optimal free responses. *Neural Computation*, 15(9):2147–2177. 5

Wiskott, L. and Sejnowski, T. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, 14(4):715–770. 1, 2, 4

Zibulevsky, M. and Pearlmutter, B. (2000). Second order blind source separation by recursive splitting of signal subspaces. In *Proc. of the 2nd Int. Workshop on Independent Component Analysis and Blind Signal Separation*, pages 489–491. 2

Ziehe, A. and Müller, K.-R. (1998). TDSEP—an efficient algorithm for blind separation using time structure. In *Proc. of the 8th Int. Conference on Artificial Neural Networks*, pages 675 – 680. Springer Verlag. 2, 3, 6

Zurmühl, A. and Falk, S. (1997). *Matrizen und ihre Anwendungen*, volume 1. Springer, Berlin, 6th edition. 6