Finding Dependent and Independent Components from Two Related Data Sets

Juha Karhunen and Tele Hao

Abstract— **Independent component analysis (ICA) and blind source separation (BSS) are usually applied to a single data set. Both these techniques are nowadays well understood, and several good methods based on somewhat varying assumptions on the data are available. In this paper, we consider an extension of ICA and BSS for separating mutually dependent and independent components from two different but related data sets. This problem is important in practice, because such data sets are common in real-world applications. We propose a new method which first uses canonical correlation analysis (CCA) for detecting subspaces of independent and dependent components. Standard ICA and BSS methods can after this be used for final separation of these components. The proposed method performs excellently for synthetic data sets for which the assumed data model holds exactly, and provides meaningful results for real-world robot grasping data. The method has a sound theoretical basis, and it is straightforward to implement and computationally not too demanding. Moreover, the proposed method has a very important by-product: its improves clearly the separation results provided by the FastICA and UniBSS methods that we have used in our experiments. Not only are the signal-to-noise ratios of the separated sources often clearly higher, but CCA preprocessing also helps FastICA to separate sources that it alone is not able to separate.**

I. Introduction

A. Independent component analysis and blind source separation

Independent component analysis (ICA) and related blind source separation (BSS) methods [1], [3], [4], [6] are nowadays already well-known and understood techniques for blind (unsupervised) extraction of useful information from vector-valued data $\mathbf{x}$. They have many applications (see for example [1], [4]) in which they provide much more meaningful results than standard linear techniques based on second-order statistics such as principal component analysis (PCA). Even though the basic data model in ICA is still linear, its proper estimation requires using higher-order statistics, which are taken into account by using appropriately nonlinearities in the ICA estimation algorithms.

The data model used in standard linear ICA is simply

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) = \sum_{i=1}^{n} s_i(t)\mathbf{a}_i \qquad (1)$$

Thus each data vector $\mathbf{x}(t)$ is expressed as a linear combination of scalar coefficients $s_i(t)$, $i = 1, 2, \ldots, n$, which multiply the respective constant basis vectors $\mathbf{a}_i$, $i = 1, 2, \ldots, n$. The scalar coefficients $s_i(t)$, $i = 1, 2, \ldots, n$, are different for each data vector $\mathbf{x}(t)$, depending directly on it. They can be collectively presented as the coefficient vector $\mathbf{s}(t)$ = $[s_1(t), s_2(t), \ldots, s_n(t)]^T$. The constant basis vectors $\mathbf{a}_i$, $i = 1, 2, \ldots, n$, are usually estimated by some criterion from the entire data set $\mathbf{X}$ = $[\mathbf{x}(1), \ldots, \mathbf{x}(N_x)]$, where N_x is the total number of data vectors in $\mathbf{X}$. Hence these basis vectors also depend on the properties of the data, but once they have been estimated, they are the same for all the data vectors belonging to this data set. These basis vectors $\mathbf{a}_i$ are in general linearly independent but non-orthogonal. They can be collectively presented in terms of the mixing (basis) matrix $\mathbf{A}$ = $[\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$.

The scalar coefficients $s_i(t)$ are called independent components or source signals (in short sources) depending on the context. The index t may denote time, position (especially in digital images), or just the number of the sample vector. For simplicity, we assume here that both the data vector $\mathbf{x}(t)$ = $[x_1(t), x_2(t), \ldots, x_n(t)]^T$ and the source vector $\mathbf{s}(t)$ are zero mean n-vectors, and that the mixing matrix $\mathbf{A}$ is a full-rank constant $n \times n$ matrix. In ICA, the column vectors $\mathbf{a}_i$, $i = 1, 2, \ldots, n$ of the matrix $\mathbf{A}$ comprise the basis vectors of ICA, and the components $s_i(t)$ of the source vector $\mathbf{s}(t)$ are respectively independent components corresponding to the data vector $\mathbf{x}(t)$.

In standard linear ICA, the index t can be dropped out, because the order of the data vectors $\mathbf{x}(t)$ is not important and can even be random. This assumption is valid if the data vectors are samples from some multivariate statistical distribution. However, the data vectors $\mathbf{x}(t)$ have often important underlying temporal structure, if they are subsequent samples from a vector-valued time series which is temporally correlated (non-white). Standard ICA can be applied to such time series, too, but it is suboptimal because it does not utilize this temporal information. Alternative methods have been developed for extracting the source signals or independent components in such cases. They usually utilize either temporal autocorrelations directly or smoothly changing nonstationarity of variance; see for example [1], [3], [4], [6].

The application domain and assumptions made in these three major groups of BSS technique vary somewhat [1], [6]. In standard ICA, it is assumed that all the independent components except for possibly one have non-Gaussian distributions and are mutually statistically independent [1], [15]. Then standard ICA methods are able to separate their waveforms, leaving however the order, sign, and scaling of the separated components ambiquous. The scaling indeter-

The authors are with the Department of Information and Computer Science, Aalto Univ. School of Science, P.O. Box 15400, FI-00076 Aalto, Espoo, Finland. Email: {Juha.Karhunen, Tele.Hao}@hut.fi. URL: http://www.cis.hut.fi/juha/.

minacy is usually fixed by normalizing the variances of the separated independent components to unity. The most widely used standard ICA method is currently FastICA [1], [9] due to its efficient implementation and fast convergence which makes it applicable to higher dimensional problems, too. We have used in our experiments the freely downloadable FastICA Matlab software package [22]. Another popular ICA method is the adaptive neural natural gradient method [1], [3], which however converges slowly and requires knowledge of the type of the source signals or independent components; they are either super-Gaussian or sub-Gaussian.

Methods based on temporal autocorrelations of the source signals require that different sources have at least some different non-zero autocorrelations. Contrary to standard ICA, they can then separate even Gaussian sources, but on the other hand they fail if such temporal correlations do not exist, while standard ICA can even in this case separate non-Gaussian sources. Examples of methods based on temporal autocorrelations are the SOBI method [16] and the TDSEP method [8]. A recent review of such methods is [12], containing many more references.

In the third group of BSS methods, it is assumed that the source signals have nonstationary smoothly changing variances. Such methods have been introduced for example in [17], [18]. If the assumptions made in them are valid, they can separate even Gaussian temporally uncorrelated (white) sources that ICA and temporal autocorrelation methods are not able to handle appropriately. A fourth class of BSS methods employs time-frequency representations (see Chapter 11 in [4]), but we shall not discuss them in this paper.

Some attempts have been made to combine different types of BSS methods so that they would be able to separate wider classes of source signals. The JADE$_{TD}$ method introduced in [19] as well as the method in [29] combine non-Gaussianity used in ICA and temporal information. In [10], Hyvärinen developed an approximate method which tries to utilize both higher-order statistics, temporal autocorrelations, and nonstationarity of variances. Only the autocorrelation coefficient corresponding to a single time lag equal to 1 is used there, but the method seems anyway to be able to separate different types of sources. We have used this method, called UniBSS in its Matlab code [23], in addition to FastICA in our experiments.

The simple linear data model (1) used in basic ICA and BSS methods can be generalized in many ways. For example to include additive noise, for the cases of having more or less sources than mixtures, for nonlinear mixtures, convolutive mixtures, and so on. Such generalizations are discussed in [1], [3], [4], [21]. In particular, up-to-date reviews of many such techniques can be found in [4].

B. An overview of our method

In this paper, we consider a generalization in which one tries to find out mutually dependent and independent components from two different but related data sets $\mathbf{X}$ and $\mathbf{Y}$. Data vectors $\mathbf{y}(t)$ belonging to the second related data set $\mathbf{Y} = [\mathbf{y}(1), \ldots, \mathbf{y}(N_y)]$ are assumed to obey a similar basic linear ICA data model

$$\mathbf{y}(t) = \mathbf{B}\mathbf{r}(t) = \sum_{i=1}^{n} r_i(t)\mathbf{b}_i \tag{2}$$

as the data vectors $\mathbf{x}(t)$ in (1). The assumptions that we make on the basis vectors $\mathbf{b}_i$ and source signals $r_i(t)$ are exactly the same as those made on the basis vectors $\mathbf{a}_i$ and source signals $s_i(t)$ in context with Eq. (1).

In our method, we first apply canonical correlation analysis (CCA) [7], [11] to be discussed in more detail in the next section to find the subspaces of dependent and independent sources in the two related data sets. We perform this step by first whitening the data sets and then computing the singular value decomposition (SVD) of their cross-covariance matrix. The data sets are then projected to the subspaces of singular vectors corresponding to the dependent and independent components. After this, any suitable ICA or BSS method can be used for separating the sources. This method will be described mathematically in detail in the next section.

C. Related work

The extension of ICA and BSS for separating dependent and independent source signals from two related data sets has not been studied as much as many other extensions of ICA and BSS mentioned above, but some research on this topic has been carried out.

Canonical correlation analysis (CCA), explained mathematically in the next section, is an old technique [30] which uses second-order statistics only. However, it has been recently applied by several authors to different real-world data analysis problems. This is because CCA often performs surprisingly well in practice, and using higher-order statistics and nonlinear techniques does not necessarily improve the results markedly.

In [14], Ylipaavalniemi et al. have carried out their analysis of biomedical fMRI sources in reverse order compared with our method. They first apply standard ICA to the two related data sets separately. Then they connect dependent sources (independent components) in these data sets using CCA. The method performs pretty well for the analyzed real-world data sets but it has a theoretical weakness: ICA assumes that the sources are non-Gaussian except for possibly one source, but CCA can be derived from a probabilistic latent variable model where all the involved random variables (vectors) are Gaussian [20]. Thus the assumptions made in ICA and CCA are theoretically contradictory.

The authors of the paper [14] have themselves noticed this theoretical weakness and improved their method in two later papers. In [24], they apply to the results first provided by ICA a nonparametric CCA type model where Gaussian distributions are not assumed, getting improved results. In another more theoretical paper [25] the authors show on a general level how to apply a probabilistic CCA type model without assuming Gaussian distributions, using instead of them any noise model belonging to the exponential family of probability distributions.

In [13], the authors use standard CCA and its extension to multiple data sets for the analysis of medical imaging data, discussing the advantages of such approaches and comparing their performances to standard ICA that has been successfully applied to this type of problems.

Koetsier et al. have presented in [26] an unsupervised neural algorithm called Exploratory Correlation Analysis for the extraction of common features in multiple data sources. This method is closely related with canonical correlation analysis.

Akaho and his co-authors [5] have considered an ICA style generalization of canonical correlation analysis which they call multimodal independent component analysis. In their method, standard linear ICA is first applied to both data sets $\mathbf{x}$ and $\mathbf{y}$ separately. Then the corresponding dependent components of the two ICA expansions are identified using a natural gradient type learning rule.

Furhermore, several authors have developed constrained ICA methods for extracting source signals which are contrained to be similar to some reference signals. This requires, however, some prior knowledge on the reference signals. In [27], Van Hulle introduces three ways to perform constrained ICA. In one of them he tries to find dependent components between two data sets by generalizing CCA, with a small-scale biomedical application. More references on constrained ICA approaches can be found in [27].

Finally, the first author of this paper tried to generalize cross-correlation analysis based on singular value decomposition in ICA style to take into account higher-order statistics in [2]. In this paper, we modify that method so that its performance is clearly improved, and a theoretical weakness of this earlier method vanishes.

II. The new method

A. *Canonical correlation analysis and singular value decomposition*

Canonical correlation analysis (CCA) [7], [11] measures the linear relationships between two multidimensional datasets $\mathbf{X}$ and $\mathbf{Y}$ using their second-order statistics, that is, autocovariances and cross-covariances. It finds two bases, one for both $\mathbf{X}$ and $\mathbf{Y}$, that are optimal with respect to correlations and it also finds the corresponding correlations. In other words, CCA finds the two bases in which the cross-correlation matrix between the data sets $\mathbf{X}$ and $\mathbf{Y}$ becomes diagonal and the correlations of the diagonal are maximized. An important property of canonical correlations is that they are invariant to affine transformations of the variables, which does not hold for ordinary correlation analysis [11].

Consider first the case where only one pair of basis vectors is sought, namely the ones corresponding to the largest canonical correlation. For this, consider the linear combinations $x = \mathbf{x}^T\mathbf{w_x}$ and $y = \mathbf{y}^T\mathbf{w_y}$ of the random vectors $\mathbf{x} \in \mathbf{X}$ and $\mathbf{y} \in \mathbf{Y}$. The dimensions of the vectors $\mathbf{x}$ and $\mathbf{y}$ can be different, but they are assumed to have zero means. The function to be maximized in CCA is the normalized correlation coefficient ρ between these two projections:

$$\rho = \frac{\mathrm{E}\{xy\}}{\sqrt{\mathrm{E}\{x^2\}\mathrm{E}\{y^2\}}} = \frac{\mathbf{w_x}^T\mathbf{C_{xy}}\mathbf{w_y}}{\sqrt{\mathbf{w_x}^T\mathbf{C_{xx}}\mathbf{w_x}\mathbf{w_y}^T\mathbf{C_{yy}}\mathbf{w_y}}} \tag{3}$$

where $\mathbf{C_{xy}} = \mathrm{E}\{\mathbf{xy}^T\}$ is the cross-covariance matrix of $\mathbf{x}$ and $\mathbf{y}$, and $\mathbf{C_{xx}} = \mathrm{E}\{\mathbf{xx}^T\}$ as well as $\mathbf{C_{xy}} = \mathrm{E}\{\mathbf{yy}^T\}$ are their autocovariance matrices. The maximum of ρ with respect to the weight vectors $\mathbf{w_x}$ and $\mathbf{w_y}$ defines the maximum canonical correlation.

The i:th canonical correlation is defined for $\mathbf{x}$ by the weight vector $\mathbf{w}_{xi}$: $x_i = \mathbf{x}^T\mathbf{w}_{xi}$, and for $\mathbf{y}$ by $\mathbf{w}_{yi}$: $y_i = \mathbf{y}^T\mathbf{w}_{yi}$. Different canonical correlations are uncorrelated: $\mathrm{E}\{x_ix_j\} = \mathrm{E}\{y_iy_j\} = \mathrm{E}\{x_iy_j\} = 0$. These canonical correlations can be computed solving the eigenvalue equations [7], [11]

$$\begin{aligned} \mathbf{C_{xx}^{-1}}\mathbf{C_{xy}}\mathbf{C_{yy}^{-1}}\mathbf{C_{yx}}\mathbf{w_x} &= \rho^2\mathbf{w_x} \\ \mathbf{C_{yy}^{-1}}\mathbf{C_{yx}}\mathbf{C_{xx}^{-1}}\mathbf{C_{xy}}\mathbf{w_y} &= \rho^2\mathbf{w_y} \end{aligned} \tag{4}$$

where $\mathbf{C_{yx}} = \mathrm{E}\{\mathbf{yx}^T\}$. The eigenvalues ρ^2 are squared canonical correlations and the eigenvectors $\mathbf{w_x}$ and $\mathbf{w_x}$ are normalized canonical correlation basis vectors. Only non-zero solutions to these equations are usually of interest, and their number is equal to the smaller of the dimensions of the vectors $\mathbf{x}$ and $\mathbf{y}$.

The solution (4) can be simplified if the data vectors $\mathbf{x}$ and $\mathbf{y}$ are prewhitened [1], which is the usual practice in many ICA algorithms, for example in FastICA. After prewhitening, both $\mathbf{C_{xx}}$ and $\mathbf{C_{yy}}$ become unit matrices, and noting that $\mathbf{C_{yx}} = \mathbf{C_{xy}^{\mathit{T}}}$ Eqs. (4) become

$$\begin{aligned} \mathbf{C_{xy}}\mathbf{C_{xy}^{\mathit{T}}}\mathbf{w_x} &= \rho^2\mathbf{w_x} \\ \mathbf{C_{yx}}\mathbf{C_{yx}^{\mathit{T}}}\mathbf{w_y} &= \rho^2\mathbf{w_y} \end{aligned} \tag{5}$$

But these are just the defining equations for the singular value decomposition (SVD) [28] of the cross-covariance matrix $\mathbf{C_{xy}}$:

$$\mathbf{C_{xy}} = \mathbf{U\Sigma V}^T = \sum_{i=1}^{L} \rho_i\mathbf{u}_i\mathbf{v}_i^T \tag{6}$$

There $\mathbf{U}$ and $\mathbf{V}$ are orthogonal square matrices ($\mathbf{U}^T\mathbf{U} = \mathbf{I}$, $\mathbf{V}^T\mathbf{V} = \mathbf{I}$) containing the singular vectors $\mathbf{u}_i$ and $\mathbf{v}_i$. In our case, these singular vectors are the basis vectors providing canonical correlations. In general, the dimensionalities of the matrices $\mathbf{U}$ and $\mathbf{V}$ and consequently the singular vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ are different corresponding to different dimensions of the data vectors $\mathbf{x}$ and $\mathbf{y}$. The pseudodiagonal matrix

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{7}$$

consists of a diagonal matrix $\mathbf{D}$ containing the non-zero singular values appended with zero matrices so that the matrix $\mathbf{\Sigma}$ is compatible with the different dimensions of $\mathbf{x}$ and $\mathbf{y}$. These non-zero singular values are just the non-zero canonical correlations. If the cross-covariance matrix $\mathbf{C_{xy}}$ has full rank, their number L is the smaller one of the dimensions of the data vectors $\mathbf{x}$ and $\mathbf{y}$.

B. The proposed method

We first separately preprocess the data vectors $\mathbf{x} \in \mathbf{X}$ and $\mathbf{y} \in \mathbf{Y}$ by subtracting their mean vectors from them if they are non-zero. After this, these data vectors are whitened separately:

$$\mathbf{v_x} = \mathbf{V_x x}, \quad \mathbf{v_y} = \mathbf{V_y y} \tag{8}$$

Whitening can be carried out in many ways [1], [3], typically standard principal component analysis (PCA) is used to that end. That is, whitening is based on the eigendecompositions of the autocovariance matrices $\mathbf{C_{xx}}$ and $\mathbf{C_{yy}}$. The whitening matrix for $\mathbf{x}$ is then

$$\mathbf{V_x} = \mathbf{\Lambda}^{-1/2}\mathbf{E} \tag{9}$$

where the columns of the matrix $\mathbf{E}$ contain the eigenvectors of $\mathbf{C_{xx}}$, and the diagonal matrix $\mathbf{\Lambda}$ contains the respective eigenvalues in the same order. The whitening matrix $\mathbf{V_y}$ for $\mathbf{y}$ is computed similarly using the eigendecomposition of $\mathbf{C_{yy}}$. After whitening, the cross-covariances (cross-correlations) of different components of $\mathbf{v_x}$ and $\mathbf{v_y}$ are zero, while their variances equal to 1. Thus whitening normalizes the data with respect to its second-order statistics. When PCA in (9) is used for whitening, it is also possible to compress the dimensionality of the data and possibly filter out some noise by retaining in $\mathbf{\Lambda}$ only the largest PCA eigenvalues and in $\mathbf{E}$ the corresponding principal eigenvectors, but we don't use this option.

After whitening, we estimate the cross-covariance matrix $\mathbf{C_{v_x v_y}}$ of the whitened data vectors $\mathbf{v_x}$ and $\mathbf{v_y}$ in standard manner:

$$\widehat{\mathbf{C}}_{\mathbf{v_x v_y}} = \sum_{t=1}^{N} \mathbf{v_x}(t)\mathbf{v_y}^T(t) \tag{10}$$

There N is the smaller of the numbers N_x and N_y of the data vectors in the two data sets $\mathbf{X}$ and $\mathbf{Y}$, respectively. In practice and in principle, too, these numbers can be different, but the cross-covariance matrix (10) can be estimated over corresponding pairs of vectors $\mathbf{v_x}(t)$ and $\mathbf{v_y}(t)$ only.

We then perform singular value decomposition of the estimated cross-covariance matrix $\widehat{\mathbf{C}}_{\mathbf{v_x v_y}}$ quite similarly as for $\mathbf{C_{xy}}$ in (6). Inspecting the magnitude of the singular values in the pseudodiagonal matrix $\mathbf{\Sigma}$, we then divide the matrices $\mathbf{U}$ and $\mathbf{V}$ of singular vectors into two submatrices:

$$\mathbf{U} = [\mathbf{U}_1 \, \mathbf{U}_2], \quad \mathbf{V} = [\mathbf{V}_1 \, \mathbf{V}_2] \tag{11}$$

There $\mathbf{U}_1$ and $\mathbf{V}_1$ correspond to dependent components for which the respective singular values are large, and $\mathbf{U}_2$ and $\mathbf{V}_2$ to the independent components for which the respective singular values are small. The data are then projected using these submatrices into subspaces corresponding to the dependent and independent components by computing

$$\mathbf{U}_1^T\mathbf{X}, \quad \mathbf{U}_2^T\mathbf{X}, \quad \mathbf{V}_1^T\mathbf{Y}, \quad \mathbf{V}_2^T\mathbf{Y} \tag{12}$$

where $\mathbf{X} = [\mathbf{x}(1), \ldots, \mathbf{x}(N_x)]$ and $\mathbf{Y} = [\mathbf{y}(1), \ldots, \mathbf{y}(N_y)]$.

Finally, we apply any suitable ICA or BSS method separately to each of these 4 projected data sets for separating the source signals contained in these subspaces. It should be noted that we include in the submatrices $\mathbf{U}_2$ and $\mathbf{V}_2$ also the singular vectors corresponding to small or even zero singular values for being able to separate all the sources in $\mathbf{X}$ and $\mathbf{Y}$. As mentioned in the introduction, we have thus far applied for post-processing (final separation) FastICA [1], [9] and the more general UniBSS method introduced in [10].

We have used the Matlab implementation UniBSS.m [23] of this method and different types of random source signals defined in that code in our experiments. Running this program revealed, however, some drawbacks of this method. First, it requires at least of the order of 1000 samples to perform appropriately, while for example FastICA needs much less samples for providing pretty good estimates of the sources if there are just a few of them. Second, the UniBSS method requires many iterations and it does not converge uniformly. It may already provide good estimates but then still with more iterations move far away from a good solution, giving then rather poor estimates of the source signals. This can happen several times until the method eventually permanently converges to a good solution. We have not yet studied the scalability of this method and how many different sources it can separate in practice. Still another drawback of the UniBSS method is that just like the natural gradient algorithm, it requires different types of nonlinearities for super-Gaussian and sub-Gaussian source signals. Thus one should know or somehow be able to estimate how many super-Gaussian and sub-Gaussian sources the data set contains. FastICA does not suffer from this limitation.

In the following, we present several somewhat intuitive and heuristic justifications to the proposed method which anyway in our opinion should largely explain its good performance.

First, let us denote the separating matrices after the whitening step in (8) by $\mathbf{W}_\mathbf{x}^T$ for $\mathbf{v_x}$ and respectively by $\mathbf{W}_\mathbf{y}^T$ for $\mathbf{v_y}$. A basic result in the theory of ICA and BSS [1], [15] is that after whitening the separating matrices $\mathbf{W_x}$ and $\mathbf{W_y}$ become orthogonal: $\mathbf{W}_\mathbf{x}^T\mathbf{W_x} = \mathbf{I}$, $\mathbf{W}_\mathbf{y}^T\mathbf{W_y} = \mathbf{I}$. Thus

$$\mathbf{v_x} = \mathbf{W}_\mathbf{x}^T\mathbf{V_x x} = \mathbf{W}_\mathbf{x}^T\mathbf{V_x A s} = \mathbf{s} \tag{13}$$

where we have for simplicity assumed that the separated sources appear in the same order as the original sources $\mathbf{s}$. Assuming that there are as many linearly independent mixtures $\mathbf{x}$ and $\mathbf{W_y}$ as sources $\mathbf{s}$, so that the mixing matrix $\mathbf{A}$ is a full-rank square matrix, we get from (13)

$$\mathbf{A} = (\mathbf{W}_\mathbf{x}^T\mathbf{V_x})^{-1} = \mathbf{V}_\mathbf{x}^{-1}\mathbf{W_x} \tag{14}$$

due to the orthogonality of the matrix $\mathbf{W_x}$. Quite similarly, we get for the another mixing matrix $\mathbf{B}$ in (2) similar result $\mathbf{B} = \mathbf{V}_\mathbf{y}^{-1}\mathbf{W_y}$.

Consider now the cross-covariance matrix after whitening. It is

$$\mathbf{C_{v_x v_y}} = \mathbf{V_x}\mathrm{E}\{\mathbf{xy}\}\mathbf{V}_\mathbf{y}^T = \mathbf{V_x A Q B}^T\mathbf{V}_\mathbf{y}^T \tag{15}$$

Here the matrix $\mathbf{Q} = \mathrm{E}\{\mathbf{sr}^T\}$ is a diagonal matrix, if the sources signals in the source vectors $\mathbf{s}$ and $\mathbf{r}$ are pairwise dependent but otherwise independent of each other. Inserting

$\mathbf{A} = (\mathbf{V}_\mathbf{x})^{-1}\mathbf{W}_\mathbf{x}$ and $\mathbf{B} = (\mathbf{V}_\mathbf{y})^{-1}\mathbf{W}_\mathbf{y}$ into (15) yields finally

$$\mathbf{C}_{\mathbf{v}_\mathbf{x}\mathbf{v}_\mathbf{y}} = \mathbf{W}_\mathbf{x}\mathbf{Q}\mathbf{W}_\mathbf{y}^T \tag{16}$$

But this is exactly the same type of expansion as the singular value decomposition of the whitened cross-covariance matrix $\mathbf{C}_{\mathbf{v}_\mathbf{x}\mathbf{v}_\mathbf{y}}$ (cf. Eq. (6)), because the matrices $\mathbf{W}_\mathbf{x}$ and $\mathbf{W}_\mathbf{y}$ are orthogonal matrices and $\mathbf{Q}$ is diagonal matrix. Thus on the assumptions made above the SVD of the whitened cross-covariance matrix provides a solution that has the same structure as the separating solution. Even though we cannot from this result directly deduce that the SVD of the whitened cross-covariance matrix (that is, CCA) would provide a separating solution, this seems to hold in simple cases at least as shown by our experiments in the next section. At least CCA when applied to the data sets $\mathbf{X}$ and $\mathbf{Y}$ using (12) provides already partial separation, helping FastICA and UniBSS to achieve clearly better results.

Another justification is that CCA, or SVD of whitened data vectors, uses second-order statistics (cross-covariances) only for separation, while standard ICA algorithms such as FastICA use for separation higher-order statistics only after the data has been normalized with respect to their second-order statistics by whitening them. Combining both second-order statistics and higher-order statistics by first performing CCA and then post-processing the results using a suitable ICA or BSS method can be expected to provide better results than using solely second-order or higher-statistics only for separation.

Our third justification is that dividing the separation problem into subproblems using the matrices in (12) may help. Probably solving two lower dimensional subproblems is easier than solving a higher dimensional separation problem.

We can somewhat heuristically modify the SVD based method introduced above to include higher-order statistics via nonlinearities by using instead of the plain cross-covariance matrix $\mathbf{C}_{\mathbf{v}_\mathbf{x}\mathbf{v}_\mathbf{y}} = \mathrm{E}\{\mathbf{v}_\mathbf{x}\mathbf{v}_\mathbf{y}^T\}$ the generalized cross-covariance matrices

$$\mathbf{G}_{\mathbf{v}_\mathbf{x}\mathbf{v}_\mathbf{y}} = \mathrm{E}\{\mathbf{v}_\mathbf{x}\mathbf{v}_\mathbf{y}^T + \mathbf{f}(\mathbf{v}_\mathbf{x})\mathbf{v}_\mathbf{y}^T + \mathbf{v}_\mathbf{x}\mathbf{f}(\mathbf{v}_\mathbf{y}^T)\} \tag{17}$$

where $\mathbf{f}(\mathbf{z})$ is a suitably chosen nonlinearity applied componentwise to its argument vector $\mathbf{z}$; we have tried $\mathbf{f}(\mathbf{z}) = \tanh(\mathbf{z})$ (suitably scaled). Similarly, we can include temporal correlations into the computations by using

$$\mathbf{G}_{\mathbf{v}_\mathbf{x}\mathbf{v}_\mathbf{y}} = \mathrm{E}\{\mathbf{v}_\mathbf{x}(t)\mathbf{v}_\mathbf{y}^T(t) + \mathbf{v}_\mathbf{x}(t-d)\mathbf{v}_\mathbf{y}^T(t) + \mathbf{v}_\mathbf{x}(t)\mathbf{v}_\mathbf{y}^T(t-d)\} \tag{18}$$

where d is the chosen time delay. In our experiments, the tanh nonlinearity in (17) had hardly any effect on the results, while a suitably chosen time delay d in (18) can improve the separation results.

III. Experimental results

A. Simulated data

We first made some experiments with artificially generated data in which there were 4 mixtures of 4 sources in both the data sets $\mathbf{X}$ and $\mathbf{Y}$. Such data is useful, because the true source signals are known, allowing evaluation of the performance of the methods studied. For real-world data, the true sources are usually unknown.

The sources $\mathbf{s}(t)$ used to generate the mixtures (1) providing the data vectors $\mathbf{x}(t) \in \mathbf{X}$ were all sub-Gaussian, consisting of uniformly distributed white noise, a sinusoidal signal, a ramp signal, and a fourth deterministic sub-Gaussian source. This type of deterministic sources are often used in ICA and BSS experiments because for them it is easy to inspect visually the quality of achieved separation results. Two of the sources $\mathbf{r}(t)$ used to generate the data vectors (2) of the other data set $\mathbf{Y}$ were the same as in the first data set $\mathbf{Y}$ and sub-Gaussian, namely uniformly distributed white noise and a sinusoid and thus completely dependent on the same sources in the data set $\mathbf{X}$. The two remaining sources in $\mathbf{Y}$ were deterministic super-Gaussian sources that are completely independent of the other sources, like the 3rd and 4th source in the first data set $\mathbf{X}$. The number of samples was moderate, 400, and we used the standard FastICA algorithm which is able to separate such sources.

For these sources, our method performed excellently. It is able to separate the sources very well even in difficult cases when the mixing matrix was almost singular and the power of some of the sources was very small compared with the power of the other source signals. This is possible because the data models (1) and (2) now hold respectively for the data vectors $\mathbf{x}(t)$ and $\mathbf{y}(t)$ exactly. If this were not true or there were even a small amount of additive noise, blind separation would not be possible in such pathological cases.

In another series of experiments which we describe in more detail here, we use stochastic source signals which are clearly more difficult to separate, defined in the Matlab code UniBSS.m [23] and explained in [10]. There are a total of 6 source signals which are all stochastic, containing at least some random component. Such sources are more appropriate than the deterministic sources used in the first experiments, but visual inspection of the quality of the separation results is more difficult for them. The four first sources are generated using a first-order autoregressive model so that the two first of them are super-Gaussian and the third and fourth source are Gaussian. Furthermore, the first and third source had identical temporal autocovariances, and similarly the second and fourth source. The fifth and sixth source have smoothly changing variances.

These six sources have been purposedly designed so that standard ICA methods such as FastICA or the natural gradient method [1] based on non-Gaussianity and higher-order statistics are able to separate the two first sources only. Methods based on temporal statistics such as [8], [16] are not able to separate any of them because there is no source with a unique temporal autocovariance sequence. Method utilizing smoothly changing variances such as [17], [18] are able to separate only the fifth and sixth source. Methods combining temporal correlations and non-Gaussianity [19], [29] would be able to separate the 4 first sources. Only the approximative method introduced in [10] could separate all these 6 sources.

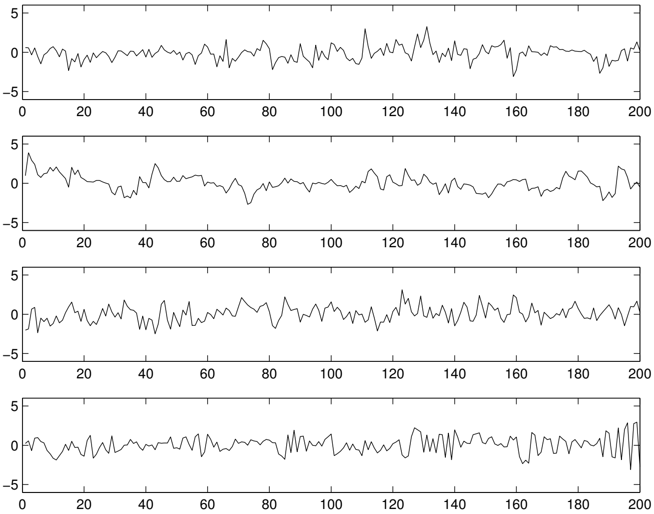

Fig. 1. 200 samples of the original source signals in first data set $\mathbf{X}$. The two first sources are non-Gaussian, the third one is temporally correlated Gaussian, and the last source has smoothly changing nonstationary variance.

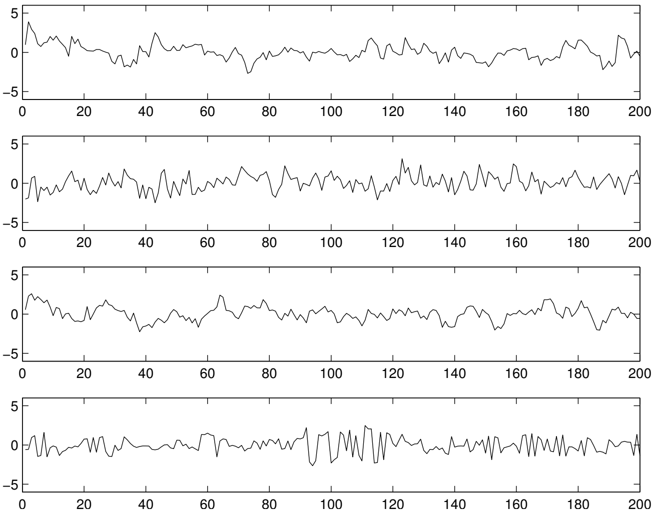

Fig. 2. 200 samples of the original source signals in second data set $\mathbf{Y}$. The first one is non-Gaussian, the third and fourth sources are temporally correlated Gaussians, and the last source has smoothly changing nonstationary variance. The two first sources in $\mathbf{Y}$ are the same as the second and third source in the first data set $\mathbf{X}$.

We picked the first three sources and the fifth source from the UniBSS.m code [23] to the first data set $\mathbf{X}$. One statistical realization of these sources is shown in Figure 1. We took the second and third source to the second data set $\mathbf{Y}$, added with the fourth and sixth source in [23]. These sources are shown similarly in Figure 2. Thus in the data sets $\mathbf{X}$ and $\mathbf{Y}$ there are two completely dependent sources, while the remaining two sources in them are statistically independent of all the other sources.

In this series of experiments, we used 2000 data vectors and source signal values ($t = 1, 2, \ldots, 2000$) for providing enough data[1] to the UniBSS method [10]. Because the results can vary a lot for different statistical realizations of these sources and their mixtures, we computed the averages of the signal-to-noise ratios of the separated sources over 100 random realizations of the sources and the data sets $\mathbf{X}$ and $\mathbf{Y}$. In each realization, the elements of the 4×4 mixing matrices were Gaussian random numbers.

The signal-to-noise ratios (SNR's) of the estimated source signals were computed for each realization of the data sets and each source from the formula

$$\mathrm{SNR}(i) = 10 \log_{10} \frac{\frac{1}{N}\sum_{t=1}^{N} s_i(t)^2}{\frac{1}{N}\sum_{t=1}^{N}[s_i(t) - \hat{s}_i(t)]^2} \qquad (19)$$

where the numerator is the average power of the i:th source $s_i(t)$ over the N samples, and the denominator is the respective power of the difference $s_i(t) - \hat{s}_i(t)$ between the source signal $s_i(t)$ and its estimate $\hat{s}_i(t)$. We computed the averages of these SNR's over the 100 realizations for each source and its estimate, and quite similarly for the sources $r_i(t)$ of the other data set $\mathbf{Y}$.

TABLE I

SIGNAL-TO-NOISE RATIOS (DB) OF DIFFERENT METHODS FOR THE SOURCE SIGNALS 1-4 IN THE FIRST DATA SET $\mathbf{X}$.

Method	Source 1	Source 2	Source 3	Source 4
CCA	10.8	10.4	10.8	10.9
FastICA	17.4	9.3	7.0	7.9
CCA + FastICA	25.8	16.4	16.8	25.5
CCA + UniBSS	29.4	44.9	34.7	29.2
UniBSS	28.6	36.8	22.6	23.6

TABLE II

SIGNAL-TO-NOISE RATIOS (DB) OF DIFFERENT METHODS FOR THE SOURCE SIGNALS 5-8 IN THE SECOND DATA SET $\mathbf{Y}$.

Method	Source 5	Source 6	Source 7	Source 8
CCA	10.4	10.8	11.6	11.9
FastICA	6.3	3.5	3.0	4.0
CCA + FastICA	17.5	18.3	13.2	13.6
CCA + UniBSS	45.4	36.1	26.0	28.2
UniBSS	32.3	22.0	24.1	24.2

The separation results for the four sources 1-4 contained in the first data set $\mathbf{X}$ are shown in Table I, and for the 4 sources in the other data set $\mathbf{Y}$ in Table II. For clarity, we have numbered these sources from 5 to 8. Based on the visual inspection of the results, we set (somewhat arbitrarily) the threshold of successful separation to 10 dB. Inspecting these results, one can see that plain canonical correlation analysis (CCA) alone is in case able to marginally separate all the 8 sources. It yields rather uniform separation results for all the sources, ranging from 10.4 dB to 11.9 dB. Plain FastICA provides clearly worse results. It can separate only the first source, though by a clear margin. But for sources 2-5, it makes already takes a long step towards separation. The results for these sources are clearly superior over a random guess, and less so also for the sources 6-8.

[1] In the original paper [10] the number of samples was even larger, 5000.

The combined CCA followed by FastICA method is clearly superior compared with both plain CCA and plain FastICA. It can separate all the 8 sources by a clear margin, and improves the results for *all these 8 sources* quite significantly especially when compared with plain FastICA. The UniBSS method [10] performs well for all these 8 sources, with over 20 dB separation quality. Combining it with CCA preprocessing gives in many cases even clearly better separation results which are already of excellent quality, see Tables I-II.

TABLE III
SIGNAL-TO-NOISE RATIOS (DB) OF DIFFERENT METHODS FOR THE SOURCES 1,2, AND 3 IN THE FIRST DATA SET $\mathbf{X}$.

Method	Source 1	Source 2	Source 3
CCA	5.8	7.1	6.3
FastICA	12.7	11.1	5.5
CCA+FastICA	20.4	20.0	14.0
CCA+UniBSS	25.1	31.8	38.3
UniBSS	24.3	25.6	33.7

TABLE IV
SIGNAL-TO-NOISE RATIOS (DB) OF DIFFERENT METHODS FOR THE SOURCES 4,5, AND 6 IN THE FIRST DATA SET $\mathbf{X}$.

Method	Source 4	Source 5	Source 6
CCA	7.1	7.1	7.3
FastICA	3.3	3.3	4.2
CCA+FastICA	14.0	11.1	11.5
CCA+UniBSS	28.7	22.4	23.9
UniBSS	13.6	14.2	21.3

To study the effect of the number of the sources signals and the difficulty of the separation problem on the results and our method, we designed another two data sets $\mathbf{X}$ and $\mathbf{Y}$. This time they both consisted of 6 random mixtures of 6 sources which were selected among a total of 9 different sources. Six of the sources were the same as in the previous experiment, but we now generated three additional sources, one of each type, namely one super-Gaussian, one temporally correlated Gaussian, and one source having non-stationary variance. They were generated in a similar manner as in [10], [23]. Due to the design of these source signals, this separation problem is very difficult, and almost all the methods introduced thus far except for UniBSS fail to separate at least some of the sources theoretically.

The first data set $\mathbf{X}$ consisted of 6 mixtures of 3 super-Gaussian sources, two temporally correlated sources, and one source with smoothly changing non-stationarity. The second data set $\mathbf{Y}$ contained 6 mixtures of 2 super-Gaussian sources, 2 temporally correlated Gaussian sources, and two sources having smoothly changing nonstationary variances. The two super-Gaussian sources and the first temporally correlated Gaussian source in this data set $\mathbf{Y}$ were the same as in the first data set $\mathbf{X}$. The remaining 3 sources in both these data sets were statistically independent of all the other sources. The number of samples was again 2000 and we computed the average results over 100 different realizations of these sources and their random mixtures.

For clarity, we have ordered the source signals in the first data set $\mathbf{X}$ so that their numbers are $1, 2, \ldots, 6$, and the 6 sources in the related data set $\mathbf{Y}$ are numbered $7, 8, \ldots, 12$. The results for different methods are presented in Tables III-VI. Plain CCA is now not able to separate any of the sources in this very difficult separation problem. But it again provides rather similar results for all the sources ranging from 5.6 dB to 7.1 dB. This is anyway a big step towards separation compared with a mere random guess. Looking next at the results of FastICA, we see that it is able to separate only the first two sources with a rather small margin only. For many of the sources, plain FastICA gives poorer results than plain CCA.

Combining CCA and FastICA improves again greatly the performance. This combined method can separate 9 first of these sources, though sources 5 and 6 marginally only. It fails to separate the sources 10-12, and the results for these sources are no better than for plain CCA. There are two possible explanations to this phenomenon. The first one is that plain FastICA performs for these sources so poorly that it cannot help CCA at all in separation. The second one is that for these sources these methods utilize same type of information, and combining them does not help any more.

Finally, both the UniBSS method and the combination of CCA and UniBSS are able to separate all the 12 sources. However, in this very difficult BSS problem CCA preprocessing improves the results for most sources, and quite markedly for some of them. UniBSS alone achieves about the same separation quality only for the first, sixth, and tenth sources.

TABLE V
SIGNAL-TO-NOISE RATIOS (DB) OF DIFFERENT METHODS FOR THE SOURCES 7,8, AND 9 IN THE SECOND DATA SET $\mathbf{Y}$.

Method	Source 7	Source 8	Source 9
CCA	7.1	6.3	7.1
FastICA	6.7	2.9	1.9
CCA+FastICA	20.0	14.1	14.1
CCA+UniBSS	31.8	38.5	29.0
UniBSS	25.0	30.2	17.5

TABLE VI
SIGNAL-TO-NOISE RATIOS (DB) OF DIFFERENT METHODS FOR THE SOURCES 10, 11, AND 12 IN THE SECOND DATA SET $\mathbf{Y}$.

Method	Source 10	Source 11	Source 12
CCA	5.8	5.6	6.6
FastICA	1.5	1.8	2.0
CCA+FastICA	6.8	5.9	5.8
CCA+UniBSS	23.8	23.5	23.1
UniBSS	23.2	20.0	20.2

To give an idea about the practical quality of separation corresponding to different SNR's, Figure 3 shows one original source signal and its estimates provided by different methods for a single realization of the data sets. This source signal is a super-Gaussian source which is the third source in the first set $\mathbf{s}(t)$ of sources and the second dependent source in the second set $\mathbf{r}(t)$ of sources. Only 200 first samples are shown to make the details of the estimates better discernible. The signal-to-noise ratios of the estimates provided by different methods are 1.92 dB for plain FastICA, 15.5 dB for the combination of CCA and FastICA, 37.9 dB for the combination of CCA and UniBSS, 33.5 dB for plain UniBSS, and 3.5 dB for plain CCA.

Inspecting Figure 3 visually shows that even though the SNR of plain CCA is poor, 3.5 dB only, it is anyway able to approximate some parts of the original source signal, for example the last samples, but for the other parts it fails. The combination of CCA and FastICA is clearly able to separate the source adequately with the SNR of 15.5 dB. The much better SNR's of the UniBSS method and the method in which CCA is combined with UniBSS do not show up in the visual quality of separation results notably. Obviously finding differences in the quality of these estimates would require looking at fine details of the separation results.

B. Robot grasping data

Our real-world robot data consists of samples from a robot arm that is used for picking off garbage from a conveyor belt. In this experimental setting there are several sensors in different parts of the robot arm. The sensor data used in our experiments consist of two data sets. First, there is the ”wrist” which guides the arm of the robot to turn so that its grasping hand containing three “fingers” moves to a right position. This force sensitive wrist data set $\mathbf{X}$ consists of four attributes: three of them are used to represent the status of movement in Euclidean three dimensional space. The fourth attribute is used to represent the status of the rotation in one direction. The other related data set $\mathbf{Y}$ consists of 7-dimensional position information about the wrist using Euclidean distance measure and standard quaternion representation in computer graphics and robotics. A mathematical model describing the relationship between these two data sets is not known. Quite probably the data model of this paper and its assumptions hold as an approximation only.

We can argue that there should be some dependent changes in the force sensitive wrist data set $\mathbf{X}$ as well as independent changes with respect to the position information data set $\mathbf{Y}$. For instance, when the wrist is sent to grab a rather heavy object, the wrist sensor data set not only expresses the position information, but also provides some feedback on holding a heavy thing in the robot arm. Furthermore, when the arm is moving into some direction, the wrist sensor data should indicate the status of the wrist along with the change in the position. Therefore, these robot data sets should suit well for testing our methods. The goal is to separate the wrist signals to the dependent parts, which have strong relationships between the relative moments, and to the independent parts, showing the impacts from the external world, such as grabbing a heavy object.

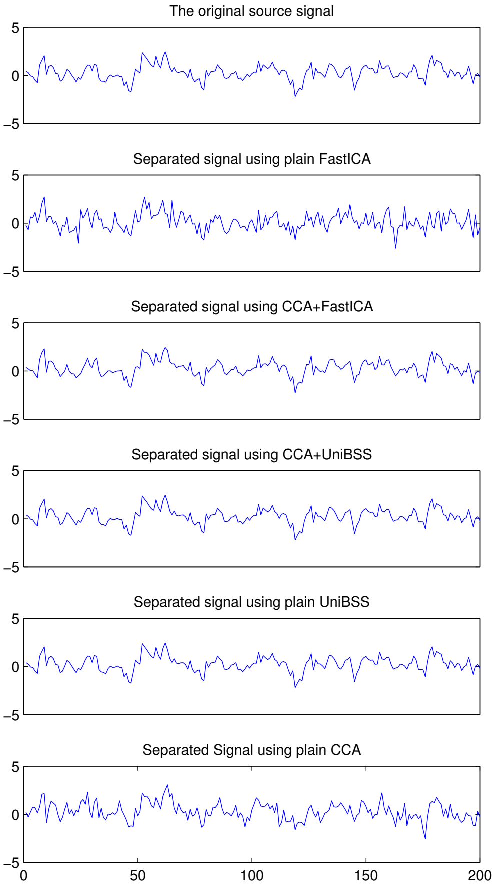

Fig. 3. First 200 samples of the original super-Gaussian 3rd source signal in the data set $\mathbf{X}$ and its estimates given by different methods.

We first preprocessed both the data sets by making their means zero and by whitening them. Furthermore, the second originally 7-dimensional position data set $\mathbf{Y}$ was transformed to a 4-dimensional data set, too, by converting the 4-dimensional quaternion representation to Euler angles in space. Furthermore, we found in our experiments that better results are obtained by using first-order and second-order differences of subsequent values of each component for the latter data set $\mathbf{Y}$. Because the original components represent position information, these first-order differences approximate their first derivative with respect to time, which is local velocity. Second-order differences approximate the second derivative of position with respect to time, which is local acceleration in the direction of the respective coordinate.

Using second-order differences can be justified by the classical law of physics: The force $F = ma$, where m is the mass of the object and a is its acceleration. Here F is the external force applied to the objects handled by the robot, and it is thus linearly proportional to the acceleration.

Figures 4 and 5 show the results for the experiment where

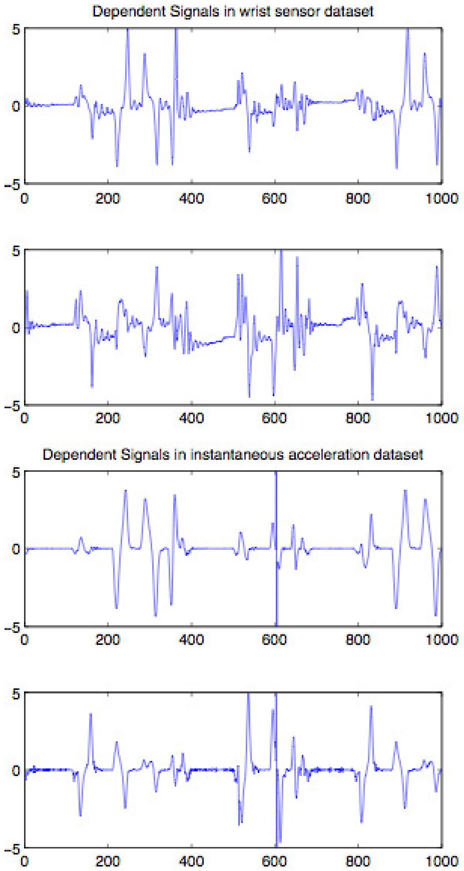

Fig. 4. Dependent signals in the robot data sets for the wrist data (2 top subfigures) and for the instantaneous acceleration data (2 bottom subfigures.

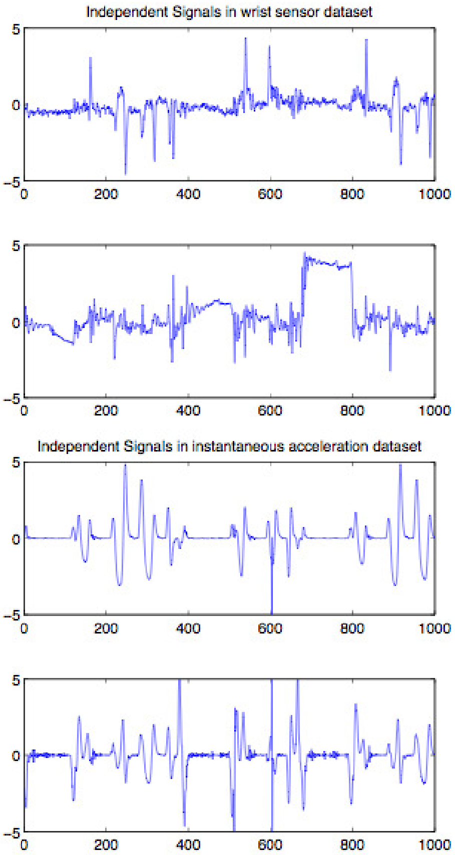

Fig. 5. Independent signals in the robot data sets for the wrist data (2 top subfigures) and for the instantaneous acceleration data (2 bottom subfigures.

we used second-order differences for the position data set $\mathbf{Y}$ using then CCA followed by the UniBSS method. The 4 singular values in the diagonal matrix $\mathbf{D}$ in (6) were 0.580, 0.340, 0.132, and 0.035. Thus it is clear that the first two singular values correspond to mutually dependent components, and the last quite small one to independent components in the two data sets $\mathbf{X}$ and $\mathbf{Y}$. The third singular value 0.132 is relatively small, and it was deemed to correspond another pair of independent components. Inspecting the first and second dependent components of the data sets depicted in Figure 4 shows clearly their dependence. Here one must recall the sign ambiguity in ICA and BSS methods: if the separated source has different sign than the original one, peaks correspond to bumps and vice versa. On the other hand, especially the second components in Figure 5 are quite clearly independent.

IV. Discussion and conclusions

This paper presents first results on a new technique for independent component analysis (ICA) and blind source separation (BSS) in which canonical correlation analysis (CCA) is used for preprocessing two data sets that are related. The final goal is to find both the dependent and independent components in these data sets. We used for final separation after CCA two ICA and BSS methods, the highly popular FastICA method [1], [9], [22] and the general UniBSS method [10], [23] which should be able to separate different categories of source signals. Our experimental results with synthetically generated data sets that consist of mixtures of source signals that are quite difficult to separate show that CCA preprocessing improves clearly the performance of both FastICA and UniBSS methods. Especially the performance of FastICA is improved for many sources dramatically: not only is the separation quality measured using signal-to-noise ratio much better, but with CCA preprocessing FastICA can separate many sources for which it alone fails in separation. Furthermore, we noticed in our experiments that after CCA preprocessing FastICA often converges much faster, requiring much less iterations than without it.

The simpler method combining CCA and FastICA is often preferable in practice for several reasons, even though the combination of CCA and UniBSS is more general and yields better results in difficult problems. First, FastICA requires much less samples to converge than the UniBSS method. Second, in the UniBSS method different types of nonlinearities must be used for separating sub-Gaussian and super-Gaussian sources [10], and therefore one should know in advance or be able to estimate how many sources belong to these categories. Third, FastICA requires usually much less iterations than the UniBSS method for convergence. Fourth, the combined CCA followed by FastICA method is computationally not demanding, and scales well with the dimensionality of the problem.

Canonical correlation analysis is based on second-order statistics, that is, autocovariances and cross-covariances of the two related data sets. Furthermore, like PCA it can be derived from a probabilistic model in which all the involved random vectors are Gaussian [20]. In our method, this is not so great limitation as one might expect, because all the information including higher-order statistics and non-Gaussianity contained in the two related data sets are retained in mapping them to the subspaces corresponding to their dependent and independent components in (12). The division into these subspaces is now based on inspection of the magnitudes of singular values of the cross-covariance matrix of whitened data sets. One could argue that also higher-order statistics should be taken into account in determining these subspaces. However, even this is often not critical because the final goal is to separate all the sources in the related two data sets irrespective of how dependent or independent they are from each other and in which way they are divided into these subspaces.

In real-world there exist many data sets which are related, and hence there should be many applications to our method. Thus far we have studied robot grasping data and presented some results for it at the end of this paper. Our next application which is currently under study is biomedical data consisting of magnetic and electric brain signal measurements. However, for these real world data sets the correct results are usually unknown, and one can often mainly assess the meaningfulness of the results only.

Acknowledgements The authors are grateful to Dr. Harri Valpola for providing us the robot grasping data set and helping to interpret the results on it. Dr. Valpola is with ZenRobotics Ltd, and also with the Department of Information and Computer Science in Aalto University School of Science, Espoo, Finland. This work was supported by the Adaptive Informatics Research Centre of Excellence of the Academy of Finland.

References

[1] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis.* Wiley, 2001.

[2] J. Karhunen and T. Ukkonen, “Extending ICA for finding jointly dependent components from two related data sets,” *Neurocomputing*, vol. 70, pp. 2969–2979, 2007.

[3] A. Cichocki and S.-I.Amari, *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications*, Wiley, 2002.

[4] P. Comon and C. Jutten (Eds.), *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Academic Press, 2010.

[5] S. Akaho, Y. Kiuchi, and S. Umeyama, ”MICA: Multidimensional Independent Component Analysis,” in *Proc. of the 1999 Int. Joint Conf. on Neural Networks (IJCNN’99)*, Washington, DC, USA, July 1999. IEEE Press, 1999, pp. 927–932.

[6] J.-F. Cardoso, ”The three easy routes to independent component analysis; contrasts and geometry”, in *Proc. of the 3rd Int. Conf. on Independent Component Analysis and Blind Signal Separation (ICA2001)*, San Diego, California, USA, December 2001, pp. 1–6.

[7] A. Rencher, *Methods of Multivariate Analysis, 2nd ed.*, Wiley, 2002.

[8] A. Ziehe and K.-R. Müller, “TDSEP - an efficient algorithm for blind source separation using time structure,” in *Proc. of the Int. Conf. on Artificial Neural Networks (ICANN’98)*, Skövde, Sweden, pp. 675–680, 1998.

[9] A. Hyvärinen, “Fast and robust fixed-point algorithms for independent component analysis,” *IEEE Trans. on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.

[10] A. Hyvärinen, “A unifying model for blind separation of independent sources,“ *Signal Processing*, vol. 85, no. 7, pp. 1419–1427, 2005.

[11] M. Borga, “Canonical correlation: a tutorial”, Linköping University, Linköping, Sweden, 2001, 12 pages. Available at http://www.imt.liu.se/∼magnus/cca/tutorial/.

[12] A. Yeredor, “Second-order methods based on color”. Chapter 7 in P. Comon and C. Jutten (Eds.), *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Academic Press, 2010, pp. 227–279.

[13] N. Correa, T. Adali, Y.-Q. Li, and V. Calhoun, “Canonical correlation analysis for data fusion and group inferences”, *IEEE Signal Processing Magazine*, vol. 27, no. 4, July 2010, pp. 39–50.

[14] J. Ylipaavalniemi et al., “Dependencies between stimuli and spatially independent fMRI sources: towards brain correlates of natural stimuli”, *NeuroImage*, vol. 48, 2009, pp. 176–185.

[15] P. Comon, “Independent component analysis - a new concept?”, *Signal Processing*, vol. 36, pp. 287–314, 1994.

[16] A. Belouchrani et al., “A blind source separation technique based on second order statistics,” *IEEE Trans. on Signal Processing*, vol. 45, no. 2, pp. 434–444, 1997.

[17] D.-T. Pham and J.-F, Cardoso, “Blind separation of instantaneous mixtures of non stationary sources”, *IEEE Trans. on Signal Processing*, vol. 49, no. 9, 1837–1848, 2001.

[18] A. Hyvärinen, “Blind source separation by nonstationarity of variance: a cumulant-based approach,” *IEEE Trans. on Neural Networks*, vol. 12, no. 6, pp. 1471-1474, 2001.

[19] K.-R. Müller, P. Philips, and A. Ziehe, “JADE$_{TD}$: Combining higher-order statistics and temporal information for blind source separation (with noise)”, in *Proc. of Int. Workshop on Independent Component Analysis and Signal Separation (ICA’99)*, Aussois, France, 1999, pages 87–92.

[20] F. Bach and M. Jordan, “A probabilistic interpretation of canonical correlation analysis”. Technical Report 688, Dept. of Statistics, Univ. of California, Berkeley, CA, USA, 2005.

[21] S. Roberts and R. Everson (Eds.), *Independent Component Analysis: Principles and Practice*, Cambridge Univ. Press, 2001.

[22] A. Hyvärinen et al., “The FastICA package for Matlab”, Helsinki Univ. of Technology, Espoo, Finland, 2005. Available at http://www.cis.hut.fi/projects/ica/fastica/ .

[23] A. Hyvärinen, “Basic Matlab code for the unifying model for BSS”, Univ. of Helsinki, Dept. of Mathematics and Statistics and Dept. of Computer Science, Helsinki, Finland, 2003–2006. Available at http://www.cs.helsinki.fi/u/ahyvarin/code/UniBSS.m .

[24] E. Savia, A. Klami, and S. Kaski, “Fast dependent components for fMRI analysis”, in *Proc. of, the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP’09)*, Taipei, Taiwan, April 2009, pp. 1737–1740.

[25] A. Klami, S. Virtanen, and S. Kaski, “Bayesian exponential family projections for coupled data sources”, in P. Grunwald and P. Spirtes (Eds.), *Proc. of the 26th Conf. on Uncertainty in Artificial Intelligence (UAI 2010*, Catalina Island, California, USA, July 2010. AUAI Press, Corvallis, Oregon, USA, 2010, pp. 286–293.

[26] J. Koetsier, D. MacDonald, D. Charles, and C. Fyfe, “Exploratory correlation analysis”, in *Proc. of the 10th European Symposium on Artificial Neural Networks (ESANN2002)*, Bruges, Belgium, April 2002, pp. 483–488.

[27] M. Van Hulle,”Constrained subspace ICA based on mutual information optimization directly”, *Neural Computation*, vol. 20, no. 4, 2008, pp. 964–973.

[28] S. Haykin, *Modern Filters.* MacMillan, 1989.

[29] A. Hyvärinen, “Complexity pursuit: separating interesting components from time-series,” *Neural Computation*, vol. 13, no. 4, pp. 883–898, 2001.

[30] H. Hotelling, “Relations between two sets of variates”, *Biometrika*, vol. 28, pp. 321–377, 1936.