ECSE-626

Statistical Computer Vision

Independent Components Analysis

High-Order Statistical Models
- Independent Components Analysis (ICA)

Principal Component Analysis aims to find the axes of projection which maximize the variance of the projected data. Thus, PCA only deals with the second order statistics of the data.

Often we are concerned with variables whose distributions are non-Gaussian, and hence are described by higher order statistics.

One property of interest is the dependence or independence of random variables.

Dependence is often confused with correlation.

These are not always the same!

A set of random variables are said to be decorrelated if the covariance matrix of the random variables is a diagonal matrix.

Random variables are said to be independent if their joint probability can be factored into a product of individual probabilities:

$$p(\vec{x}) = \prod_{i=1}^{n} p(x_i)$$

For a Gaussian distribution decorrelation implies independence since for a decorrelated set of random variables with a joint Gaussian distribution we have that:

$$p(\vec{x}) = \frac{1}{(2\pi\prod_i \sigma_i)^{n/2}} \exp\left(-\sum_i \frac{(x_i - \mu_i)^2}{2\sigma_i^2} \right)$$

$$= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(x_i - \mu_i)^2}{2\sigma_i^2} \right) = \prod_{i=1}^{n} p(x_i)$$

For non-Gaussian joint probabilities, decorrelation does not necessarily imply independence.

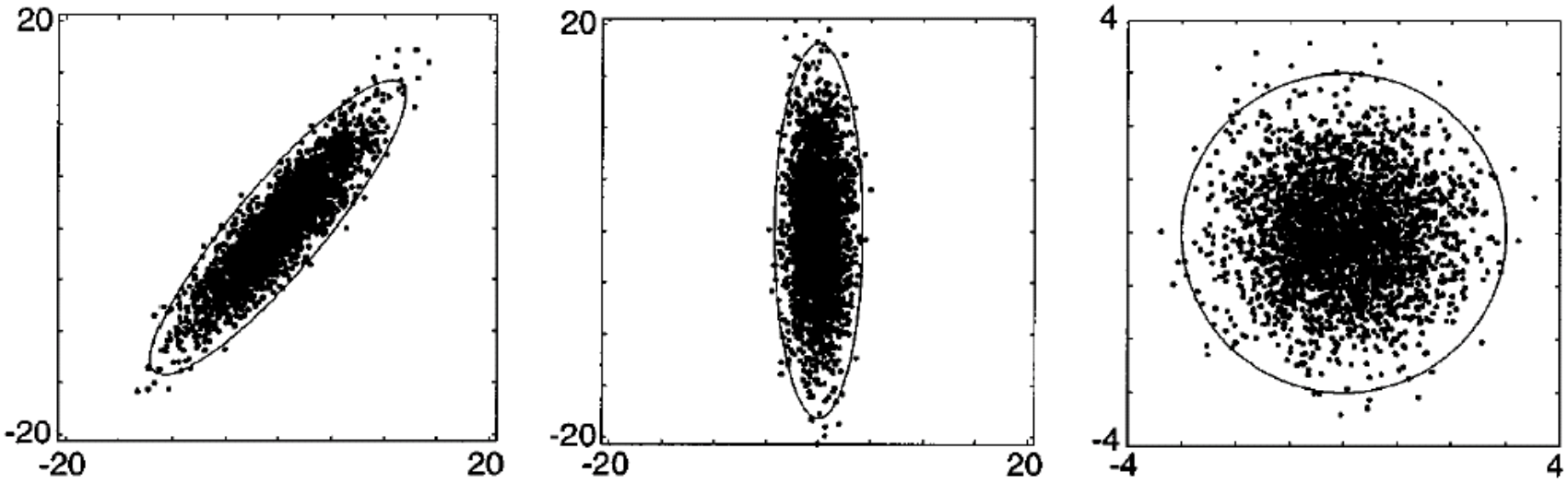
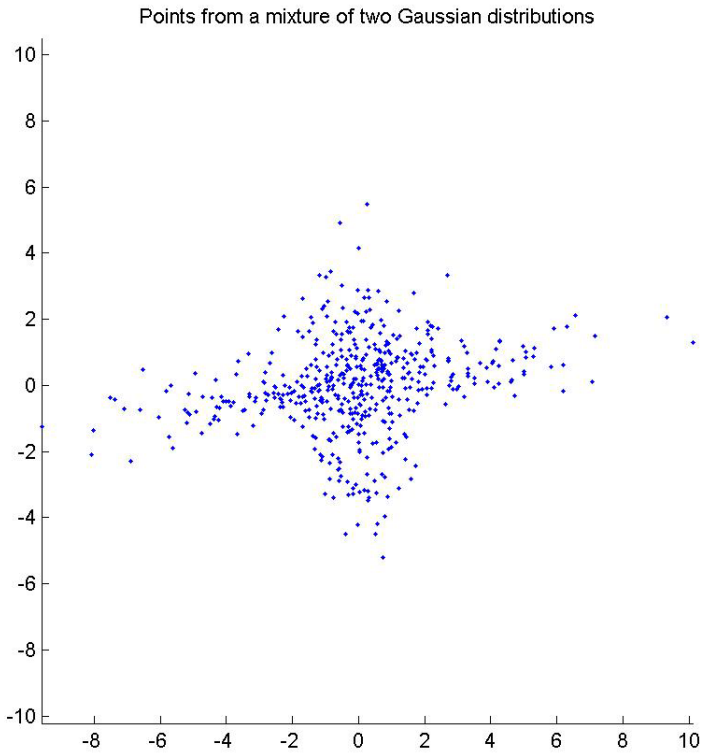
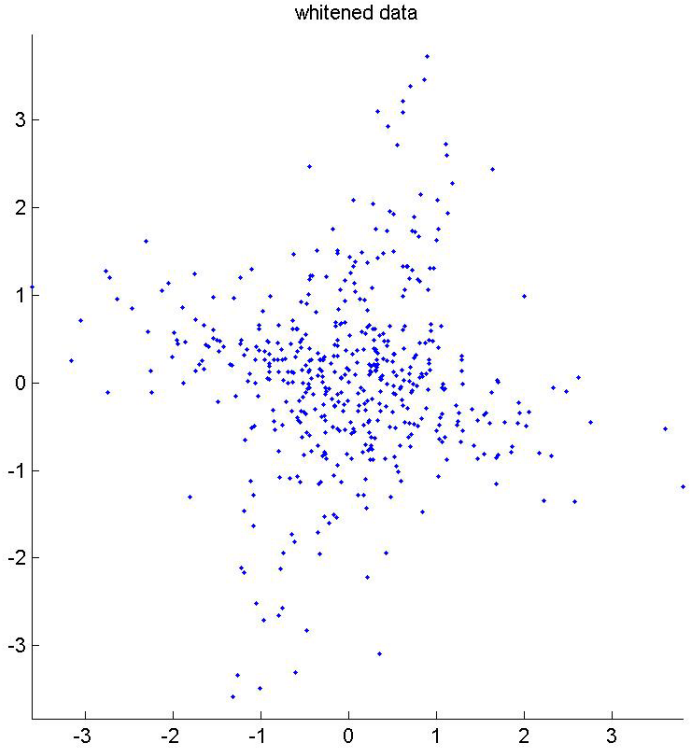


Figure 1: Illustration of principal component analysis on Gaussian-distributed data in two dimensions. (*a*) Original data. Each point corresponds to a sample of data drawn from the source distribution (i.e. a two-pixel image). The ellipse is three standard deviations from the mean in each direction. (*b*) Data rotated to principal component coordinate system. Note that the ellipse is now aligned with the axes of the space. (*c*) Whitened data. When the measurements are represented in this new coordinate system, their components are distributed as uncorrelated (and thus independent) univariate Gaussians.



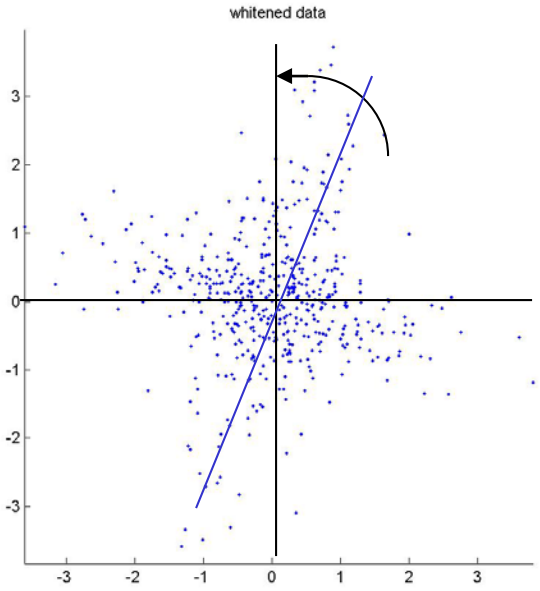


On the left is a set of data taken from two different Gaussian distributions. On the right is the "whitened" set of data, obtained by computing the principal components of the left data set, projecting the data onto these principal components and scaling the axes so as to make the covariance (of the whitened data) equal to the identity matrix.

Notice that the lobes in the data scatterplot do not line up with the principal component axes.

Even though the *x* and *y* components of the data are uncorrelated after the whitening process, they are not independent.

To make the components independent, we have to do an additional rotation of the coordinate system to make the lobes of the scatterplot line up with the coordinate axes.



But how can we know by how much to rotate? Principal component analysis won't help. It has done all it can.

In order to find the rotation of the axes that will result in independent components, we have to look at higher order statistics of the data.

As we have seen, Principal Component Analysis aims to find the axes of projection which *maximize the variance* of the projected data.
Thus, PCA only deals with the ***second order statistics*** of the data.

Note: there may not exist a rotation which will result in independent components. Even if there is, a particular algorithm might not be able to find it.

We can measure the dependency of random variables by evaluating a distance measure between the two distributions - $p(\vec{x})$ and $\prod_{i=1}^{n} p(x_i)$.

Comon suggested using the *Kullback - Leibler Divergence* as a measure of distance between the two distributions:

$$d = \int p(\vec{x}) \log \left(\frac{p(\vec{x})}{\prod_{i=1}^{n} p(x_i)} \right) d\vec{x}$$

This distance will be zero when the two distributions are equal, i.e. when the components of $\vec{x}$ are independent.

P. COMON, ``Independent Component Analysis, a new concept?," *Signal Processing, Elsevier*, **36**(3):287--314, April 1994, Special issue on Higher-Order Statistics.

The Kullback-Leibler divergence is related to the mutual information:

$$d(\vec{x}) = -H(\vec{x}) + \sum_{i=1}^{n} H(x_i)$$

where $H(\vec{x})$ is the entropy of $\vec{x}$ and $H(x_i)$ is the marginal entropy of x_i.

So, we want to find a linear transformation W of $\vec{x}$ which minimizes $d(W\vec{x})$:

$$W = \arg\min_{W} \left(-H(W\vec{x}) + \sum_{i=1}^{n} H(Wx_i) \right)$$

The process of finding the optimal W is referred to as *Independent Components Analysis* (a term coined by Comon).

To determine the optimal W a gradient descent type algorithm is usually used. Many different algorithms have been proposed.

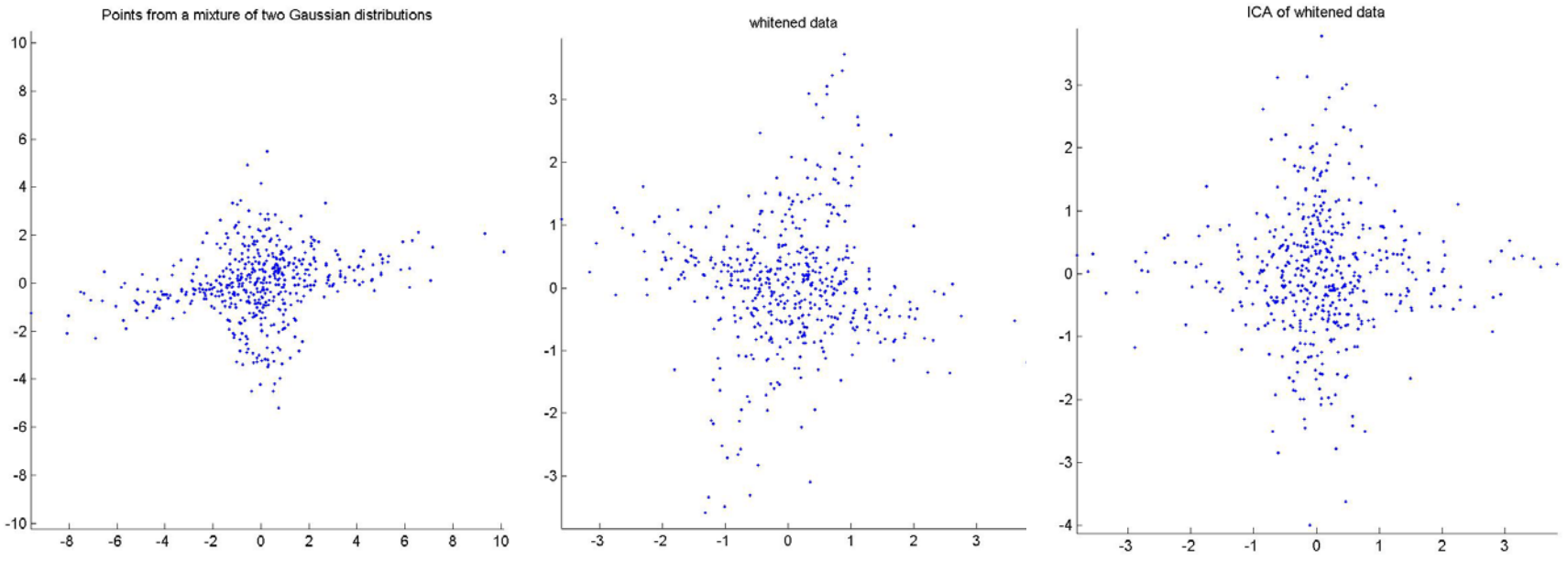
One effective algorithm was proposed by Amari *et al* based on the original method of Comon. It uses an approximation to the marginal entropy that models the probability distribution in terms of a power-series expansion in its higher-order cumulants.

This leads to an iterative solution with the following form of update:

$$\frac{dW}{dt} = \eta(t)\{I - f(W\vec{x})(W\vec{x})^T\}W$$

where $f()$ is an odd nonlinearity and $\eta(t)$ is a time-varying learning rate.

Amari S., Cichocki A. and Yang H.H. 1996. A new learning algorithm for blind signal separation, *Advances in Neural Information Processing Systems 8*, MIT press



On the left is a set of data taken from two different Gaussian distributions. In the midde is the "whitened" set of data. On the right is the whitened data after the Independent Components Analysis.

The lobes in the data scatterplot now line up with the axes. The x and y components of the data are now both uncorrelated and independent.

```
% FastICA for Matlab 5.x          http://www.cis.hut.fi/projects/ica/fastica/
% Version 2.1, January 15 2001
% Copyright (c) Hugo Gävert, Jarmo Hurri, Jaakko Särelä, and Aapo Hyvärinen.
```

ICA is commonly expressed in terms of a "*blind source separation*" problem. In this view we model a signal (e.g. an image or a audio waveform) as coming from a linear combination of sources that are independent (i.e. that have nothing to do with each other, such as a person standing in front of a wall). That is, the k^{th} element in a set of data vectors is generated by:

$$\vec{x}_k = A\vec{s}_k$$

The matrix A is called the mixing matrix.

In the mixing model the k^{th} element in a set of data vectors is generated by:

$$\vec{x}_k = A\vec{s}_k$$

The matrix A is called a mixing matrix. ICA is a way to find the source components $\vec{s}_k$ from the measured data components $\vec{x}_k$ under the assumption that the sources are independent.

Thus, ICA determines a linear transformation, W, which acts on $\vec{x}_k$ to give independent components, which are a model for the $\vec{s}_k$.

$$\vec{s}_k \approx \vec{y}_k = W\vec{x}_k$$

W should be $\approx$ the inverse of A.

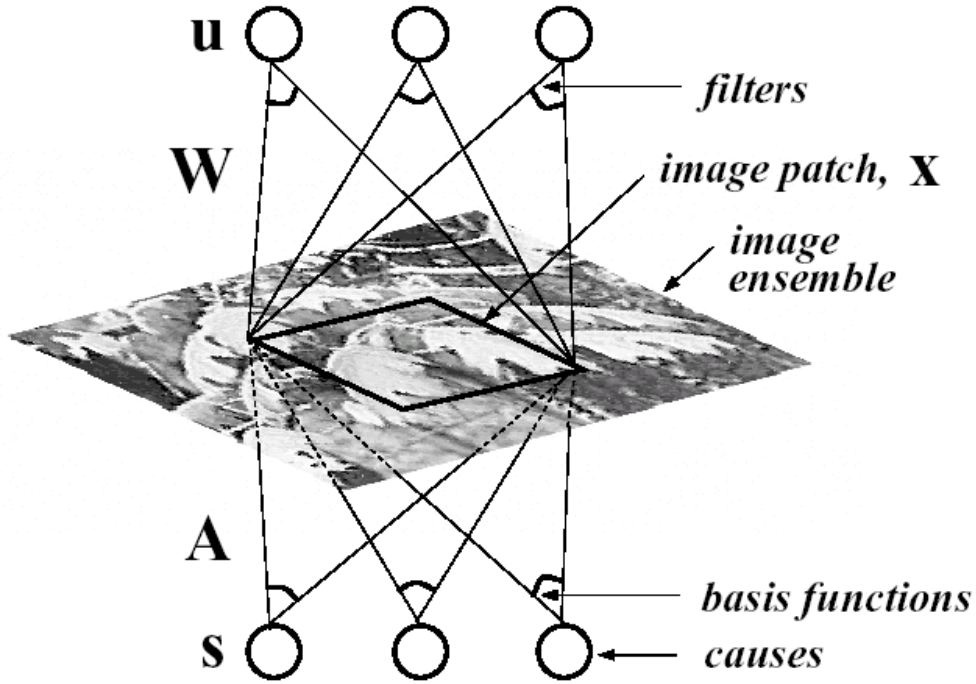


Figure 1: The Blind Linear Image Synthesis model (Olshausen & Field, 1996). Each patch, **x**, of an image is viewed as a linear combination of several (here three) underlying basis functions, given by the matrix **A**, each associated with an element of an underlying vector of 'causes', **s**. In this paper, causes are viewed as statistically independent 'image sources'. The causes are recovered (in a vector **u**) by a matrix of filters, **W**, more loosely 'receptive fields', which attempt to invert the unknown mixing of unknown basis functions constituting image formation.

Bell A.J. and Sejnowski T.J. 1996. Edges are the `independent components'of natural scenes, *Advances in Neural Information Processing Systems 9*, MIT press

Assumptions

- Source signals are statistically independent
  - Knowing the value of one of the components does not give any information about the others
- ICs have nongaussian distributions
  - Initial distributions unknown
  - At most one Gaussian source
- Recovered sources can be permutated and scaled

More on ICA

- In reality, due to the global orthogonality transformation, principal components tend to capture global features (e.g. eigenfaces for face recognition, brightness if normalization is not performed beforehand.)
- In contrast, ICA is usually local (e.g. noses, eyes, hairs for face analysis, edges for natural scenes). Why?

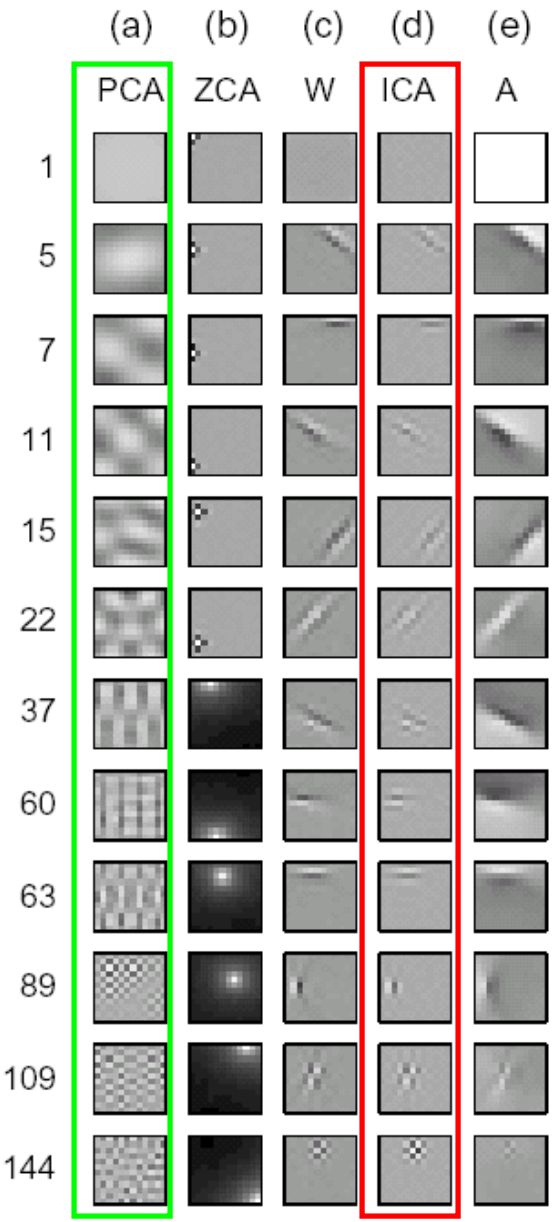


Figure 3: Selected decorrelating filters and their basis functions extracted from the natural scene data. Each type of decorrelating filter yielded 144 12x12 filters, of which we only display a subset here. Each column contains filters or basis functions of a particular type, and each of the rows has a number relating to which row of the filter or basis function matrix is displayed. (a) PCA ($\mathbf{W}_P$): The 1st, 5th, 7th etc Principal Components, calculated from Eq.(7), showing increasing spatial frequency. There is no need to show basis functions and filters separately here since for PCA, they are the same thing. (b) ZCA ($\mathbf{W}_Z$): The first 6 entries in this column show the one-pixel wide centre-surround filter which whitens while preserving the phase spectrum. All are identical, but shifted. The lower 6 entries (37, 60 show the basis functions instead, which are the columns of the inverse of the $\mathbf{W}_Z$ matrix. (c) $\mathbf{W}$: the weights learnt by the ICA network trained on $\mathbf{W}_Z$-whitened data, showing (in descending order) the DC filter, localised oriented filters, and localised checkerboard filters. (d) $\mathbf{W}_I$: The corresponding ICA filters, calculated according to $\mathbf{W}_I = \mathbf{W}\mathbf{W}_Z$, looking like whitened versions of the $\mathbf{W}$-filters. (e) $\mathbf{A}$: the corresponding basis functions, or columns of $\mathbf{W}_I^{-1}$. These are the patterns which optimally stimulate their corresponding ICA filters, while not stimulating any other ICA filter, so that $\mathbf{W}_I\mathbf{A} = \mathbf{I}$.

Note that the independent components are localized in space, in contrast with the principal components, which are distributed in space.

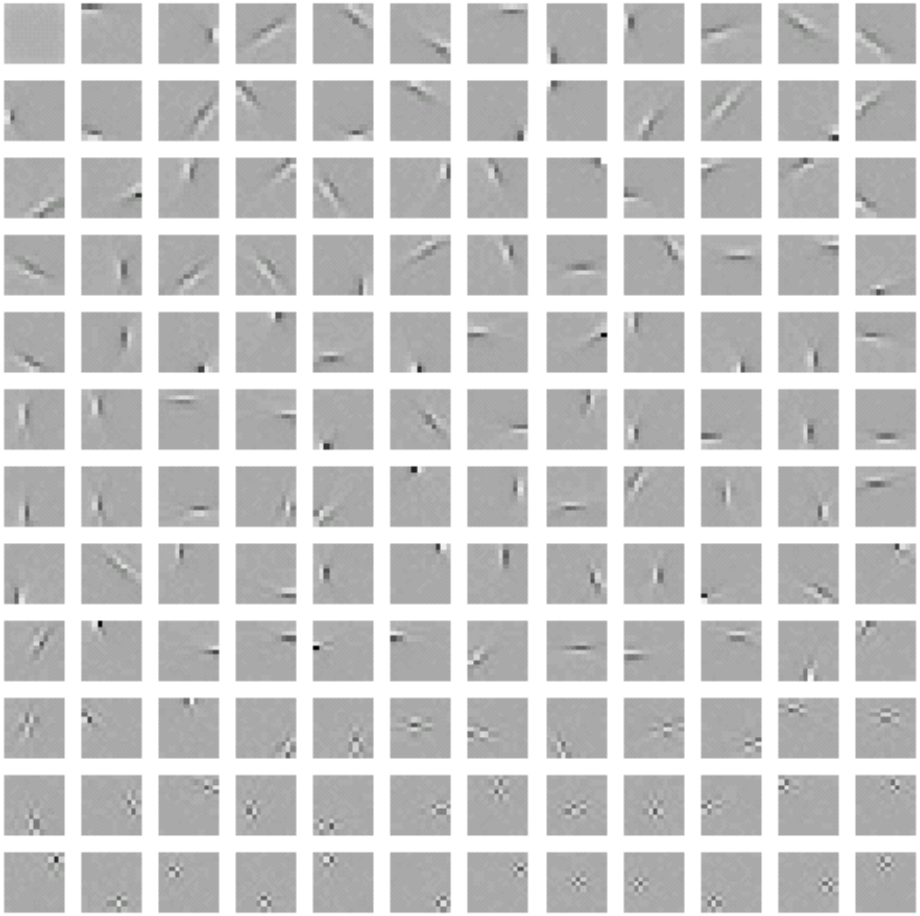
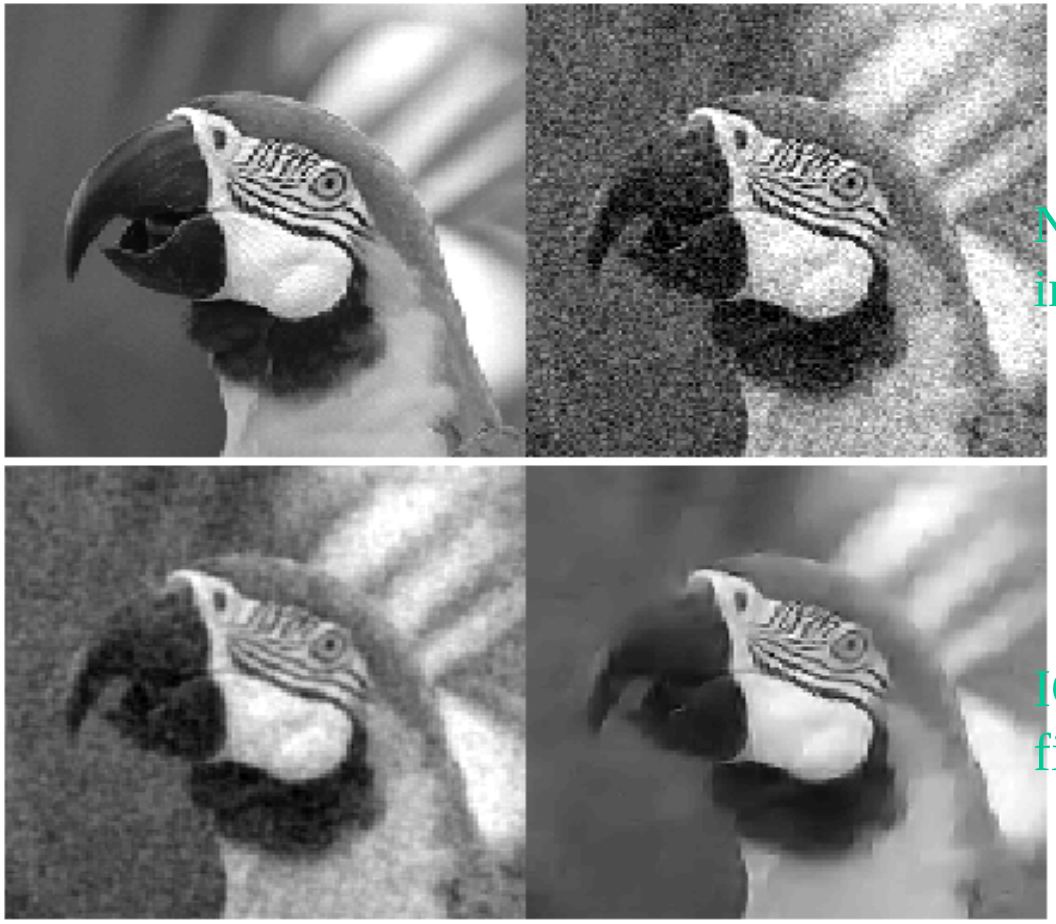


Figure 4: The matrix of 144 filters obtained by training on ZCA-whitened natural images. Each filter is a row of the matrix $\mathbf{W}$, and they are ordered left-to-right, top-to-bottom in reverse order of the length of the filter vectors. In a rough characterisation, and more-or-less in order of appearance, the filters consist of one DC filter (top left), 106 oriented filters (of which 35 were diagonal, 37 were vertical and 34 horizontal), and 37 localised checkerboard patterns. The diagonal filters are longer than the vertical and horizontal due to the bias induced by having square, rather than circular, receptive fields.

ICA for Image denoising

Original image



Noisy image

Wiener filtering

ICA filtering

Nathan Intrator

ICA for Removing Artifacts from EEG

- EEG ~ *Neural cocktail party*
- Severe ***contamination*** of EEG activity by
  - eye movements
  - blinks
  - muscle
  - heart, ECG artifact
  - vessel pulse
  - electrode noise
  - line noise, alternating current (60 Hz)

- ICA can improve signal
  - effectively ***detect, separate and remove*** activity in EEG records from a wide variety of artifactual sources.
    (Jung, Makeig, Bell, and Sejnowski)

- ICA weights (mixing matrix) help find **location** of sources

Póczos & Singh

Motion Style Components

- Method for analysis and synthesis of human motion from motion captured data
- Provides perceptually meaningful "style" components
- 109 markers, (327dim data)
- Motion capture $\Rightarrow$ data matrix

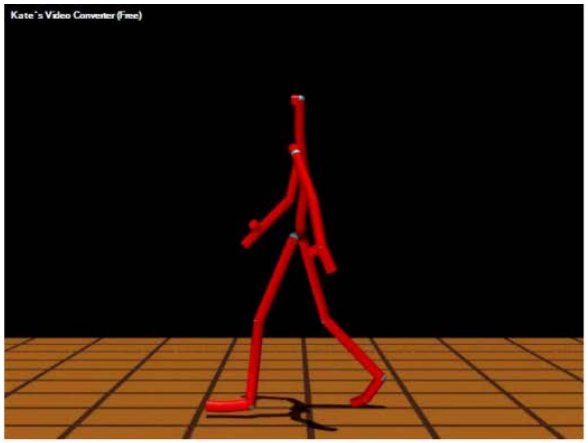
Goal: Find motion style components.

ICA $\Rightarrow$ 6 independent components (emotion, content,...)

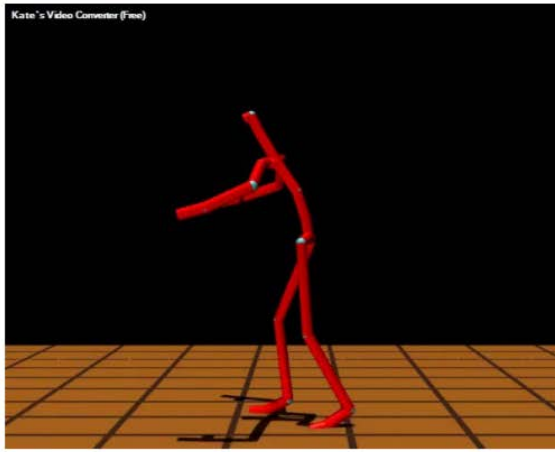
(Mori & Hoshino 2002, Shapiro et al 2006, Cao et al 2003)

Póczos & Singh

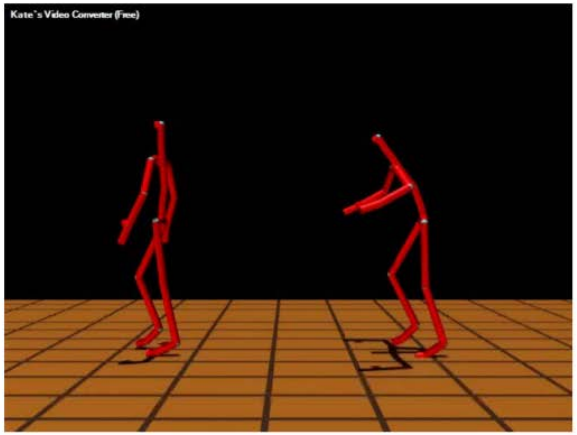
Motion Style Components



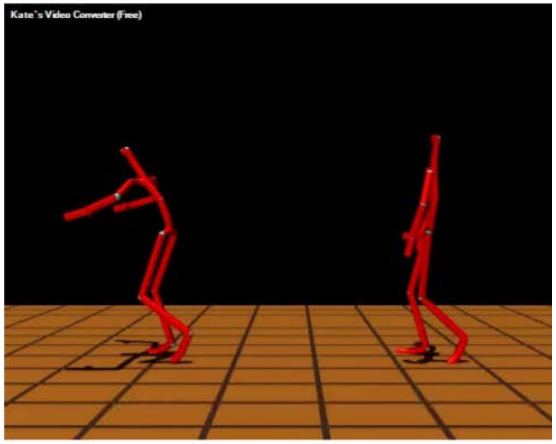
walk



sneaky

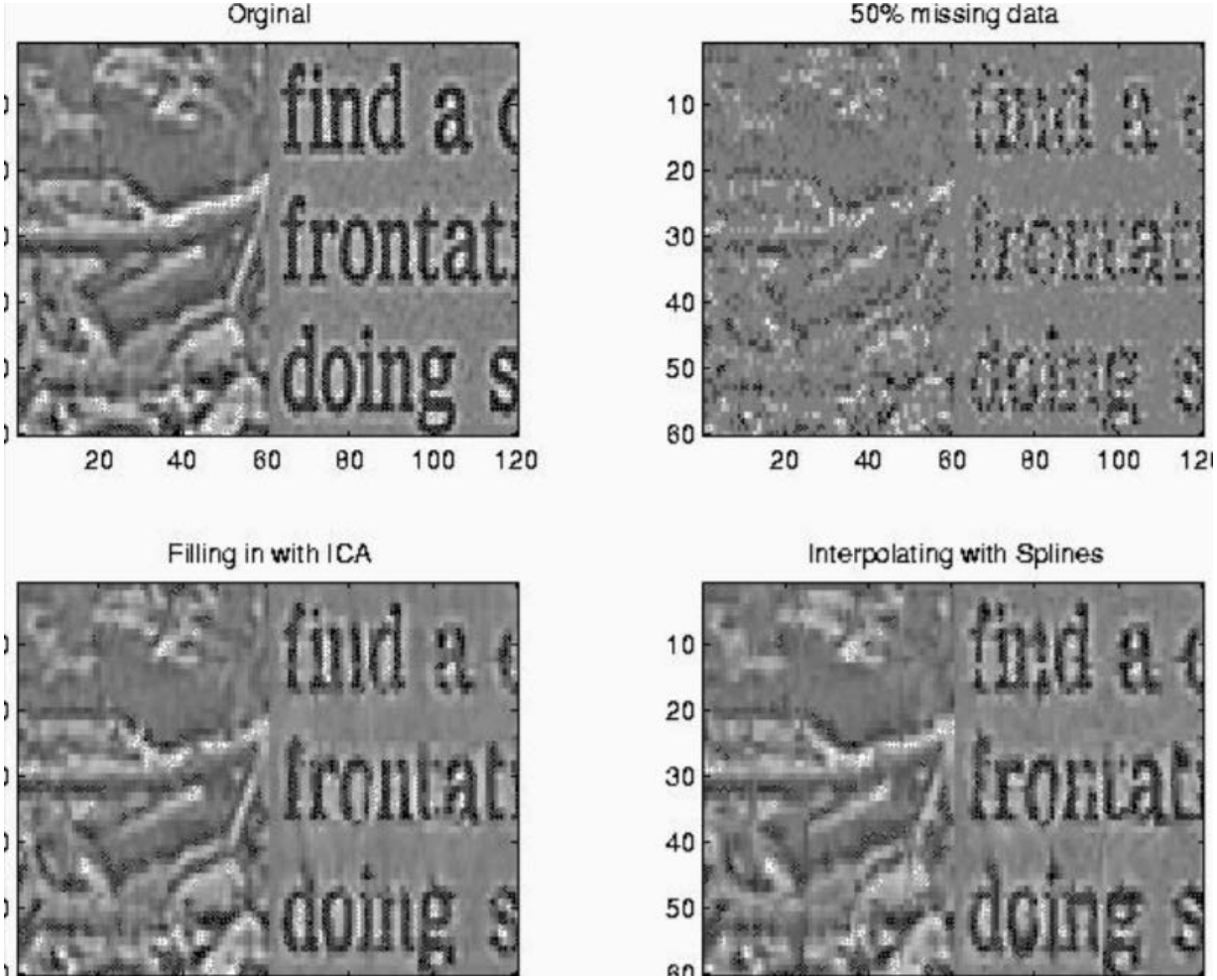


walk with sneaky



sneaky with walk

ICA for Filling in Missing Data



Jung 2006