

Learning Canonical Correlations

M. Borga, H. Knutsson, T. Landelius
 Computer Vision Laboratory
 Department of Electrical Engineering
 Linköping University
 S-581 83 Linköping, Sweden

Abstract

This paper presents a novel learning algorithm that finds the linear combination of one set of multi-dimensional variates that is the best predictor, and at the same time finds the linear combination of another set which is the most predictable. This relation is known as the *canonical correlation* and has the property of being invariant with respect to affine transformations of the two sets of variates. The algorithm successively finds all the canonical correlations beginning with the largest one. It is shown that canonical correlations can be used in computer vision to find feature detectors by giving examples of the desired features. When used on the pixel level, the method finds quadrature filters and when used on a higher level, the method finds combinations of filter output that are less sensitive to noise compared to vector averaging.

1 Introduction

A common problem in neural networks and learning, incapacitating many theoretically promising algorithms, is the high dimensionality of the input-output space. As an example, typical dimensionalities for systems having visual inputs far exceed acceptable limits. For this reason, a priori restrictions must be invoked. A common restriction is to use only locally linear models. To obtain efficient systems, the dimensionalities of the models should be as low as possible. The use of locally low-dimensional linear models will in most cases be adequate if the subdivision of the input and output spaces are made adaptively [3, 11].

An important problem is to find the best directions in the input- and output spaces for the local models. Algorithms like the Kohonen self organizing feature maps [10] and others that work with principal component analysis will find directions where the signal variances are high. This is, however, of little use in a response generating system. Such a system should find directions that efficiently represents signals that are *important* rather than signals that have large energy.

In general the input to a system comes from a set of different sensors and it is evident that the range of the signal values from a given sensor is unrelated to the importance of the received information. The same line of reasoning holds for the output which may consist of signals to a set of different effectuators. For this reason the *correlation* between input and output signals is interesting since this measure of input-output relation is independent of the signal variances. However, correlation alone is not necessarily meaningful. Only input-output pairs that are regarded as relevant should be entered in the correlation analysis.

Relating only the projections of the input, \mathbf{x} , and output, \mathbf{y} , on two vectors, \mathbf{w}_x and \mathbf{w}_y , establishes a one-dimensional linear relation between the input and output. We wish to find the vectors that maximizes $\text{corr}(\mathbf{x}^T \mathbf{w}_x, \mathbf{y}^T \mathbf{w}_y)$, i.e. the correlation between the projections. This relation is known as *canonical correlation* [6]. It is a statistical method of finding the linear combination of one set of variables that is the best predictor, and at the same time finding the linear combination of an other set which is the most predictable.

It has been shown [7] that finding the canonical correlations is equivalent to maximizing the *mutual information* between the sets \mathcal{X} and \mathcal{Y} if \mathbf{x} and \mathbf{y} come from elliptical symmetric random distributions.

In section 2, a brief review of the theory of canonical correlation is given. In section 3 we present an iterative learning rule, equation 7, that finds the directions and magnitudes of the canonical correlations. To illustrate the algorithm behaviour, some experiments are presented and discussed in section 4. Finally, in section 5, we discuss how the concept of canonical correlation can be used for finding representations of local features in computer vision.

2 Canonical Correlation

Consider two random variables, \mathbf{x} and \mathbf{y} , from a multi-normal distribution:

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \sim N \left(\begin{bmatrix} \mathbf{x}_0 \\ \mathbf{y}_0 \end{bmatrix}, \begin{bmatrix} \mathbf{C}_{xx} & \mathbf{C}_{xy} \\ \mathbf{C}_{yx} & \mathbf{C}_{yy} \end{bmatrix} \right), \quad (1)$$

where $\mathbf{C} = \begin{bmatrix} \mathbf{C}_{xx} & \mathbf{C}_{xy} \\ \mathbf{C}_{yx} & \mathbf{C}_{yy} \end{bmatrix}$ is the covariance matrix. \mathbf{C}_{xx} and \mathbf{C}_{yy} are nonsingular matrices and $\mathbf{C}_{xy} = \mathbf{C}_{yx}^T$. Consider the linear combinations, $x = \mathbf{w}_x^T(\mathbf{x} - \mathbf{x}_0)$ and $y = \mathbf{w}_y^T(\mathbf{y} - \mathbf{y}_0)$, of the two variables respectively. The correlation between x and y is given by equation 2, see for example [2]:

$$\rho = \frac{\mathbf{w}_x^T \mathbf{C}_{xy} \mathbf{w}_y}{\sqrt{\mathbf{w}_x^T \mathbf{C}_{xx} \mathbf{w}_x \mathbf{w}_y^T \mathbf{C}_{yy} \mathbf{w}_y}}. \quad (2)$$

The directions of the partial derivatives of ρ with respect to \mathbf{w}_x and \mathbf{w}_y are given by:

$$\begin{cases} \frac{\partial \rho}{\partial \mathbf{w}_x} \overset{\rightarrow}{=} & \mathbf{C}_{xy} \hat{\mathbf{w}}_y - \frac{\hat{\mathbf{w}}_x^T \mathbf{C}_{xy} \hat{\mathbf{w}}_y}{\hat{\mathbf{w}}_x^T \mathbf{C}_{xx} \hat{\mathbf{w}}_x} \mathbf{C}_{xx} \hat{\mathbf{w}}_x \\ \frac{\partial \rho}{\partial \mathbf{w}_y} \overset{\rightarrow}{=} & \mathbf{C}_{yx} \hat{\mathbf{w}}_x - \frac{\hat{\mathbf{w}}_y^T \mathbf{C}_{yx} \hat{\mathbf{w}}_x}{\hat{\mathbf{w}}_y^T \mathbf{C}_{yy} \hat{\mathbf{w}}_y} \mathbf{C}_{yy} \hat{\mathbf{w}}_y \end{cases} \quad (3)$$

where ' $\hat{\cdot}$ ' indicates unit length and ' $\overset{\rightarrow}{=}$ ' means that the vectors, left and right, have the same directions. A complete description of the canonical correlations is given by:

$$\begin{bmatrix} \mathbf{C}_{xx} & [0] \\ [0] & \mathbf{C}_{yy} \end{bmatrix}^{-1} \begin{bmatrix} [0] & \mathbf{C}_{xy} \\ \mathbf{C}_{yx} & [0] \end{bmatrix} \begin{pmatrix} \hat{\mathbf{w}}_x \\ \hat{\mathbf{w}}_y \end{pmatrix} = \rho \begin{pmatrix} \lambda_x \hat{\mathbf{w}}_x \\ \lambda_y \hat{\mathbf{w}}_y \end{pmatrix} \quad (4)$$

where: $\rho, \lambda_x, \lambda_y > 0$ and $\lambda_x \lambda_y = 1$. Equation 4 can be rewritten as:

$$\begin{cases} \mathbf{C}_{xx}^{-1} \mathbf{C}_{xy} \hat{\mathbf{w}}_y = \rho \lambda_x \hat{\mathbf{w}}_x \\ \mathbf{C}_{yy}^{-1} \mathbf{C}_{yx} \hat{\mathbf{w}}_x = \rho \lambda_y \hat{\mathbf{w}}_y \end{cases} \quad (5)$$

Solving equation 5 gives N solutions $\{\rho_n, \hat{\mathbf{w}}_{xn}, \hat{\mathbf{w}}_{yn}\}$, $n = \{1..N\}$. N is the minimum of the input dimensionality and the output dimensionality. The linear combinations, $x_n = \hat{\mathbf{w}}_{xn}^T \mathbf{x}$ and $y_n = \hat{\mathbf{w}}_{yn}^T \mathbf{y}$, are termed *canonical variates* and the correlations, ρ_n , between these variates are termed the *canonical correlations* [6]. An important aspect in this context is that the canonical correlations are *invariant to affine transformations* of \mathbf{x} and \mathbf{y} . Also note that the canonical variates corresponding to the different roots of equation 5 are uncorrelated, implying that:

$$\begin{cases} \mathbf{w}_{xn}^T \mathbf{C}_{xx} \mathbf{w}_{xm} = 0 \\ \mathbf{w}_{yn}^T \mathbf{C}_{yy} \mathbf{w}_{ym} = 0 \end{cases} \quad \text{if } n \neq m \quad (6)$$

It should be noted that equation 4 is a special case of the *generalized eigenproblem* [4]:

$$\mathbf{A} \mathbf{w} = \lambda \mathbf{B} \mathbf{w}.$$

The solution to this problem can be found by finding the vectors \mathbf{w} that maximizes the *Rayleigh quotient*:

$$r = \frac{\mathbf{w}^T \mathbf{A} \mathbf{w}}{\mathbf{w}^T \mathbf{B} \mathbf{w}}.$$

3 Learning Canonical Correlations

We have developed a novel learning algorithm that finds the canonical correlations and the corresponding canonical variates by an iterative method. The update rule for the vectors \mathbf{w}_x and \mathbf{w}_y is given by:

$$\begin{cases} \mathbf{w}_x \leftarrow \mathbf{w}_x + \alpha_x \mathbf{x} (\mathbf{y}^T \hat{\mathbf{w}}_y - \mathbf{x}^T \mathbf{w}_x) \\ \mathbf{w}_y \leftarrow \mathbf{w}_y + \alpha_y \mathbf{y} (\mathbf{x}^T \hat{\mathbf{w}}_x - \mathbf{y}^T \mathbf{w}_y) \end{cases} \quad (7)$$

where \mathbf{x} and \mathbf{y} both have the mean $\mathbf{0}$. To see that this rule finds the directions of the canonical correlation we look at the expected change, in one iteration, of the vectors, \mathbf{w}_x and \mathbf{w}_y :

$$\begin{cases} E\{\Delta \mathbf{w}_x\} = \alpha_x E\{\mathbf{x}\mathbf{y}^T \hat{\mathbf{w}}_y - \mathbf{x}\mathbf{x}^T \mathbf{w}_x\} = \alpha_x (\mathbf{C}_{xy} \hat{\mathbf{w}}_y - \|\mathbf{w}_x\| \mathbf{C}_{xx} \hat{\mathbf{w}}_x) \\ E\{\Delta \mathbf{w}_y\} = \alpha_y E\{\mathbf{y}\mathbf{x}^T \hat{\mathbf{w}}_x - \mathbf{y}\mathbf{y}^T \mathbf{w}_y\} = \alpha_y (\mathbf{C}_{yx} \hat{\mathbf{w}}_x - \|\mathbf{w}_y\| \mathbf{C}_{yy} \hat{\mathbf{w}}_y) \end{cases}$$

Identifying with equation 3 gives:

$$E\{\Delta \mathbf{w}_x\} \cong \frac{\partial \rho}{\partial \mathbf{w}_x} \quad \text{and} \quad E\{\Delta \mathbf{w}_y\} \cong \frac{\partial \rho}{\partial \mathbf{w}_y} \quad (8)$$

with

$$\|\mathbf{w}_x\| = \frac{\hat{\mathbf{w}}_x^T \mathbf{C}_{xy} \hat{\mathbf{w}}_y}{\hat{\mathbf{w}}_x^T \mathbf{C}_{xx} \hat{\mathbf{w}}_x} \quad \text{and} \quad \|\mathbf{w}_y\| = \frac{\hat{\mathbf{w}}_y^T \mathbf{C}_{yx} \hat{\mathbf{w}}_x}{\hat{\mathbf{w}}_y^T \mathbf{C}_{yy} \hat{\mathbf{w}}_y}$$

This shows that the expected changes of the vectors \mathbf{w}_x and \mathbf{w}_y are in the same directions as the gradient of the canonical correlation, ρ , which means that the learning rules in equation 7 on average is a gradient search on ρ . λ_x and λ_y are found as:

$$\rho = \sqrt{\|\mathbf{w}_x\| \|\mathbf{w}_y\|}; \quad \lambda_x = \lambda_y^{-1} = \sqrt{\frac{\|\mathbf{w}_x\|}{\|\mathbf{w}_y\|}}. \quad (9)$$

3.1 Learning of successive canonical correlations

The learning rule maximizes the correlation and finds the directions, $\hat{\mathbf{w}}_{x1}$ and $\hat{\mathbf{w}}_{y1}$, corresponding to the largest correlation, ρ_1 . To find the second largest canonical correlation and the corresponding canonical variates of equation 5 we use the modified learning rule

$$\begin{cases} \mathbf{w}_x \Leftarrow \mathbf{w}_x + \alpha_x \mathbf{x} ((\mathbf{y} - \mathbf{y}_1)^T \hat{\mathbf{w}}_y - \mathbf{x}^T \mathbf{w}_x) \\ \mathbf{w}_y \Leftarrow \mathbf{w}_y + \alpha_y \mathbf{y} ((\mathbf{x} - \mathbf{x}_1)^T \hat{\mathbf{w}}_x - \mathbf{y}^T \mathbf{w}_y) \end{cases} \quad (10)$$

where

$$\mathbf{x}_1 = \frac{\mathbf{x}^T \hat{\mathbf{w}}_{x1} \mathbf{v}_{x1}}{\hat{\mathbf{w}}_{x1}^T \mathbf{v}_{x1}} \quad \text{and} \quad \mathbf{y}_1 = \frac{\mathbf{y}^T \hat{\mathbf{w}}_{y1} \mathbf{v}_{y1}}{\hat{\mathbf{w}}_{y1}^T \mathbf{v}_{y1}}.$$

\mathbf{v}_{x1} and \mathbf{v}_{y1} are estimates of $\mathbf{C}_{xx} \hat{\mathbf{w}}_{x1}$ and $\mathbf{C}_{yy} \hat{\mathbf{w}}_{y1}$ respectively and are estimated using the iterative rule:

$$\begin{cases} \mathbf{v}_{x1} \Leftarrow \mathbf{v}_{x1} + \beta (\mathbf{x} \mathbf{x}^T \hat{\mathbf{w}}_{x1} - \mathbf{v}_{x1}) \\ \mathbf{v}_{y1} \Leftarrow \mathbf{v}_{y1} + \beta (\mathbf{y} \mathbf{y}^T \hat{\mathbf{w}}_{y1} - \mathbf{v}_{y1}) \end{cases} \quad (11)$$

The expected change of \mathbf{w}_x and \mathbf{w}_y is then given by

$$\begin{cases} E\{\Delta \mathbf{w}_x\} = \alpha_x \left(\mathbf{C}_{xy} \left[\hat{\mathbf{w}}_y - \hat{\mathbf{w}}_{y1} \frac{\hat{\mathbf{w}}_{y1}^T \mathbf{C}_{yy} \hat{\mathbf{w}}_y}{\hat{\mathbf{w}}_{y1}^T \mathbf{C}_{yy} \hat{\mathbf{w}}_{y1}} \right] - \|\mathbf{w}_x\| \mathbf{C}_{xx} \hat{\mathbf{w}}_x \right) \\ E\{\Delta \mathbf{w}_y\} = \alpha_y \left(\mathbf{C}_{yx} \left[\hat{\mathbf{w}}_x - \hat{\mathbf{w}}_{x1} \frac{\hat{\mathbf{w}}_{x1}^T \mathbf{C}_{xx} \hat{\mathbf{w}}_x}{\hat{\mathbf{w}}_{x1}^T \mathbf{C}_{xx} \hat{\mathbf{w}}_{x1}} \right] - \|\mathbf{w}_y\| \mathbf{C}_{yy} \hat{\mathbf{w}}_y \right) \end{cases} \quad (12)$$

It can be seen that the parts of \mathbf{w}_x and \mathbf{w}_y parallel to $\mathbf{C}_{xx} \hat{\mathbf{w}}_{x1}$ and $\mathbf{C}_{yy} \hat{\mathbf{w}}_{y1}$ respectively will vanish ($\Delta \mathbf{w}_x^T \mathbf{w}_{x1} \leq 0 \quad \forall \mathbf{x}$ and $\Delta \mathbf{w}_y^T \mathbf{w}_{y1} \leq 0 \quad \forall \mathbf{y}$ in equation 10). In the subspaces orthogonal to $\mathbf{C}_{xx} \hat{\mathbf{w}}_{x1}$ and $\mathbf{C}_{yy} \hat{\mathbf{w}}_{y1}$ the learning rule will be equivalent to that given by equation 7. In this way the parts of the signals correlated with $\mathbf{w}_{x1}^T \mathbf{x}$ (and $\mathbf{w}_{y1}^T \mathbf{y}$) are disregarded leaving the rest unchanged. Consequently the algorithm finds the second largest correlation ρ_2 and the corresponding vectors \mathbf{w}_{x2} and \mathbf{w}_{y2} . Successive canonical correlations can be found by repeating the procedure.

4 Performance

In this section, two different experiments are presented to illustrate the efficiency and performance of the proposed algorithm.

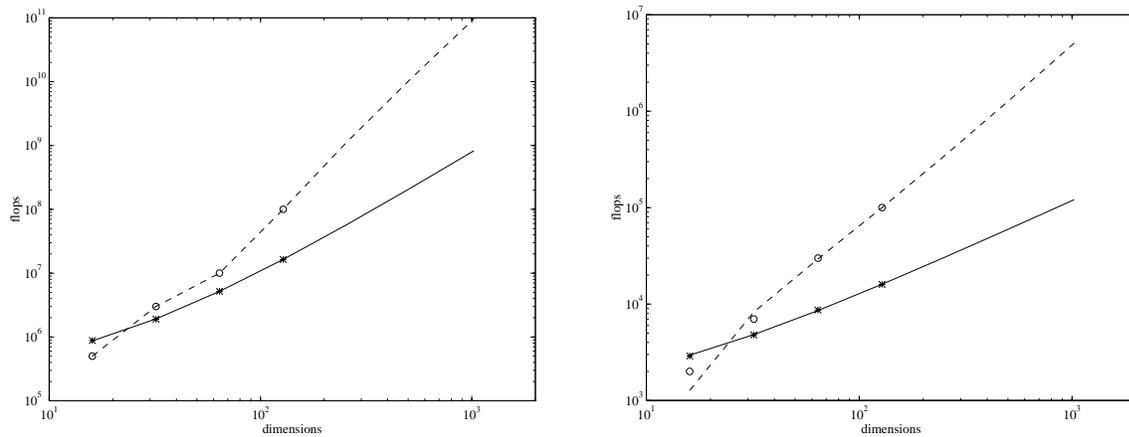


Figure 1: **Left:** Number of flops until convergence for RLS (dashed line) and for our algorithm (solid line). **Right:** Number of flops per iteration for RLS (dashed line) and for our algorithm (solid line).

4.1 $\mathcal{O}(n)$ speedup

In the first experiment, we will demonstrate, the advantage of using our canonical correlation algorithm to find the proper subspace for a low-dimensional linear model in a high dimensional space. A set of training data was generated with n -dimensional input vectors \mathbf{x} and 8-dimensional output vectors \mathbf{y} . There were two linear relations between \mathbf{x} and \mathbf{y} and, hence, the proper subspaces should be two-dimensional in the input and output spaces respectively.

In the experiment, the canonical correlation algorithm was run until it found the proper subspace. This can be determined by the algorithm since it, besides the directions of canonical correlation, gives an estimate of each canonical correlation. After convergence, a standard recursive least square (RLS) algorithm was applied to find the linear relations between the two-dimensional subspaces. This algorithm was iterated until convergence, i.e. until the error ϵ was below a certain threshold. The error was defined as

$$\epsilon = \|\mathbf{B}\mathbf{C}^T\mathbf{x} - \mathbf{y}\|^2$$

where \mathbf{B} is the (small) matrix found by the RLS algorithm and \mathbf{C} is the matrix found by the canonical correlation algorithm extracting the relevant subspace.

As a comparison, the standard RLS algorithm was used directly on the data, iterated until convergence with the same threshold as in the first experiment. The error in this case was defined as

$$\epsilon = \|\mathbf{A}\mathbf{x} - \mathbf{y}\|^2$$

where \mathbf{A} is the (large) matrix found by the RLS algorithm. The results are plotted in figure 1.

In both experiments the dimensionalities n of the input space was 16, 32, 64 and 128 (computational problems with the standard RLS method set the upper limit). The complexity was measured by counting the number of floating point operations (flops) until convergence (left) and per iteration (right) for the standard RLS method (marked with rings) and for our method (marked with stars). The lines show the estimated number of flops for larger dimensionalities. They were calculated by fitting polynomials to the data. For our method, a second and a first order polynomial was sufficient for the data in the left and right figures respectively. For the standard RLS method, a third and a second order polynomial respectively had to be used, in accordance with theory. Note the logarithmic scale.

The results show that our algorithm is of order $\mathcal{O}(n^2)$ when run until convergence ($\mathcal{O}(n)$ per iteration) while the standard RLS method is of order $\mathcal{O}(n^3)$ ($\mathcal{O}(n^2)$ per iteration).

The dimensionality of the linear relation can, of course, in general not be known in advance. This is, however, not a problem since the canonical correlation algorithm first finds the largest correlation and then proceeds by finding the second and so on until all existing correlations are found.

4.2 High dimensional spaces

The second experiment illustrates the algorithm's ability to handle high-dimensional spaces. The dimensionality of \mathbf{x} is 800 and the dimensionality of \mathbf{y} is 200, so the total dimensionality of the signal space is 1000.

Rather than tuning parameters to produce a nice result for a specific distribution, we have used adaptive update factors and parameters producing similar behaviour for different distributions and different number of dimensions. Also note that the adaptability allows a system without a pre-specified time dependent update rate decay. The coefficients α_x and α_y were in the experiments calculated according to equation 13:

$$\begin{cases} \alpha_x = a\lambda_x E_x^{-1} \\ \alpha_y = a\lambda_y E_y^{-1} \end{cases} \quad \text{where} \quad \begin{cases} E_x \leftarrow E_x + b (\|\mathbf{x}\mathbf{x}^T \mathbf{w}_x\| - E_x) \\ E_y \leftarrow E_y + b (\|\mathbf{y}\mathbf{y}^T \mathbf{w}_y\| - E_y) \end{cases} \quad (13)$$

To get a smooth and yet fast behaviour, an adaptively time averaged set of vectors, \mathbf{w}_a was calculated. The update speed was made dependent on the consistency in the change of the original vectors \mathbf{w} according to equation 14.

$$\begin{cases} \mathbf{w}_{ax} \leftarrow \mathbf{w}_{ax} + c \|\Delta_x\| \|\mathbf{w}_x\|^{-1} (\mathbf{w}_x - \mathbf{w}_{ax}) \\ \mathbf{w}_{ay} \leftarrow \mathbf{w}_{ay} + c \|\Delta_y\| \|\mathbf{w}_y\|^{-1} (\mathbf{w}_y - \mathbf{w}_{ay}) \end{cases} \quad \text{where} \quad \begin{cases} \Delta_x \leftarrow \Delta_x + d (\Delta \mathbf{w}_x - \Delta_x) \\ \Delta_y \leftarrow \Delta_y + d (\Delta \mathbf{w}_y - \Delta_y) \end{cases} \quad (14)$$

This process we call *adaptive smoothing*.

The experiment have been carried out using a randomly chosen distribution of a 800-dimensional \mathbf{x} variable and a 200-dimensional \mathbf{y} variable. Two \mathbf{x} and two \mathbf{y} dimensions were partly correlated. The variances in the 1000 dimensions are in the same order of magnitude

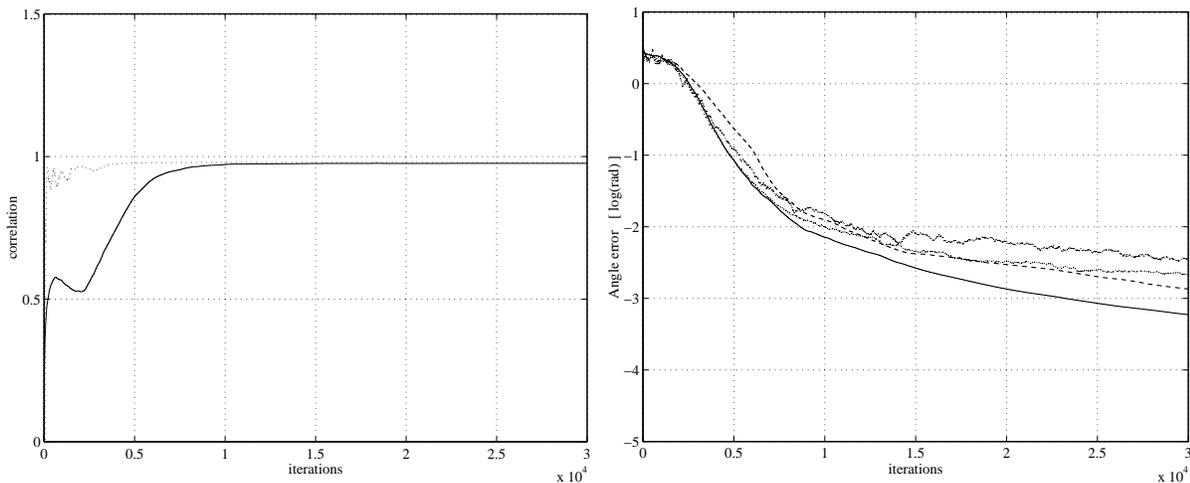


Figure 2: **Left:** Figure showing the estimated first canonical correlation (solid line) as a function of number of actual events and the true correlation in the current directions found by the algorithm (dotted line). The dimensionality of one set of variables is 800 and of the second set 200. **Right:** Figure showing the log of the angular error as a function of number of actual events on a logarithmic scale.

To the left in figure 2, the estimated first canonical correlation as a function of number of actual events (solid line) and the true correlation in the current directions found by the algorithm (dotted line) is shown.

To the right in the same figure, the effect of the adaptive smoothing is shown. The angle errors of the smoothed estimates (solid and dashed curves) are much more stable and decrease more rapidly than the 'raw' estimates. The errors after 3×10^4 samples is in the order of a few degrees. (It should be noted that this is an extreme precision as, with a resolution of 3 degrees, a low estimate of the number of different orientations in a 1000-dimensional space is 10^{1500} .) The angular errors were calculated as the angle between the vectors \mathbf{w}_a and the exact solutions, $\hat{\mathbf{e}}$ (known from the $\mathbf{x} \mathbf{y}$ sample distribution), i.e. $\arccos(\hat{\mathbf{w}}_a^T \hat{\mathbf{e}})$. Note the logarithmic scale.

5 Applications in computer vision

At the Computer Vision Laboratory in Linköping, we have developed a tensor representation of image features (see e.g. [8, 9]) that has received attention in the computer vision society. A possible extension of the tensor concept towards more robust estimations, representation of certainty, representation of higher order features, etc. involves higher order filter combinations.

As an example, consider a three-dimensional filtering with a $7 \times 7 \times 7$ neighborhood on three scales. This gives approximately 1000 filter responses. A complete second order function of this filtering involves about 10^6 signals.

The selection among all different possible filter combinations to design a tensor representation is very difficult and a learning system is called for. A standard optimization method, based on mean square error is, however, not very useful since it is the *shape* of the tensor that is of interest rather than size. If we, for example, want to have a tensor representing the orientation of a signal, we want the tensor to carry as much information as possible about the *orientation* and not the *magnitude* of the signal.

For this reason, the canonical correlation algorithm is a suitable method, since it is based on mutual information maximization rather than mean square error [1]. It can also handle high-dimensional signal spaces which is essential in a further development of the tensor concept in image processing.

Experiments show that the canonical correlation algorithm can be used to find filters that describe a particular feature in an image invariant with respect to other features. The features to be described are learned by giving examples that are presented in pairs to the algorithm in such a way that the desired feature, e.g. orientation, is equal for each pair while other features, e.g. phase, are presented in an unordered way.

5.1 Learning low level operations

In the first experiment, we show that quadrature filters are found by this method when products of pixel data are presented to the algorithm. Quadrature filters can be used to describe lower order features, e.g. local orientation.

Let \mathbf{I}_x and \mathbf{I}_y be a pair of 5×5 images. Each image consists of a sine wave pattern and additive Gaussian noise. A sequence of such image pairs is constructed so that, for each pair, the orientation is equal in the two images while the phase differs in a random way. The images have independent noise. Each image pair is described by vectors \mathbf{i}_x and \mathbf{i}_y .

Let \mathbf{X} and \mathbf{Y} be the outer products of the image vectors, i.e. $\mathbf{X} = \mathbf{i}_x \mathbf{i}_x^T$ and $\mathbf{Y} = \mathbf{i}_y \mathbf{i}_y^T$ and rearrange the matrices \mathbf{X} and \mathbf{Y} into vectors \mathbf{x} and \mathbf{y} respectively. Now, we have a sequence of pairs of 625-dimensional vectors describing the products of pixel data from the images.

The sequence consist of 6500 examples, i.e. 20 examples per degree of freedom. (The outer product matrices are symmetric and, hence, the number of free parameters is $\frac{n^2+n}{2}$ where n is the dimensionality of the image vector.) For a signal to noise ratio (SNR) of 0 dB, there were 6 significant¹ canonical correlations and for an SNR of 10 dB there were 10 significant canonical correlations. The two most significant correlations for the 0 dB case were both 0.7 which corresponds to an SNR² of 3.7 dB. For the 10 dB case, the two highest correlations were both 0.989, corresponding to an SNR of 19.5 dB.

The projections of image signals \mathbf{x} for orientations between 0 and π onto the 10 first canonical correlation vectors \mathbf{w}_x from the 10 dB case are shown to the left in figure 3. The signals were generated with random phase and without noise. As seen in the figure, the first two canonical correlations are sensitive to the double angle of the orientation of the signal and invariant with respect to phase. The two curves are 90° out of phase and, hence, form a quadrature pair [5]. The following curves show the lower correlations which are sensitive to the fourth, sixth, eighth, and tenth multiples of the orientation and they also form quadrature pairs.

To be able to interpret the canonical correlation vectors, \mathbf{w}_x , we can write the vectors as 25×25 matrices, \mathbf{W}_x , and then do an eigenvalue decomposition, i.e. $\mathbf{W}_x = \sum \lambda_i \mathbf{e}_i \mathbf{e}_i^T$. (Note that the data was generated as outer products, resulting in positive semi definite symmetric matrices.) The eigenvectors, \mathbf{e}_i , can be seen as linear filters acting on the image. If the eigenvectors are rearranged into 5×5 matrices, "eigenimages", they can be interpreted in terms of image features. To the right in figure 3, the two most significant eigenimages are shown for the first (top) and second (bottom) canonical correlations respectively. We see that these eigenimages form two quadrature filters sensitive to two perpendicular orientations.

¹By significant, we mean that they differ from the random correlations caused by the limited set of samples. The random correlations, in the case of 20 samples per degree of freedom, is approximately 0.2 (given by experiments).

²The relation between correlation and SNR in this case is defined by the correlation between two signals with the same SNR, i.e. $\text{corr}(s + \eta_1, s + \eta_2)$.

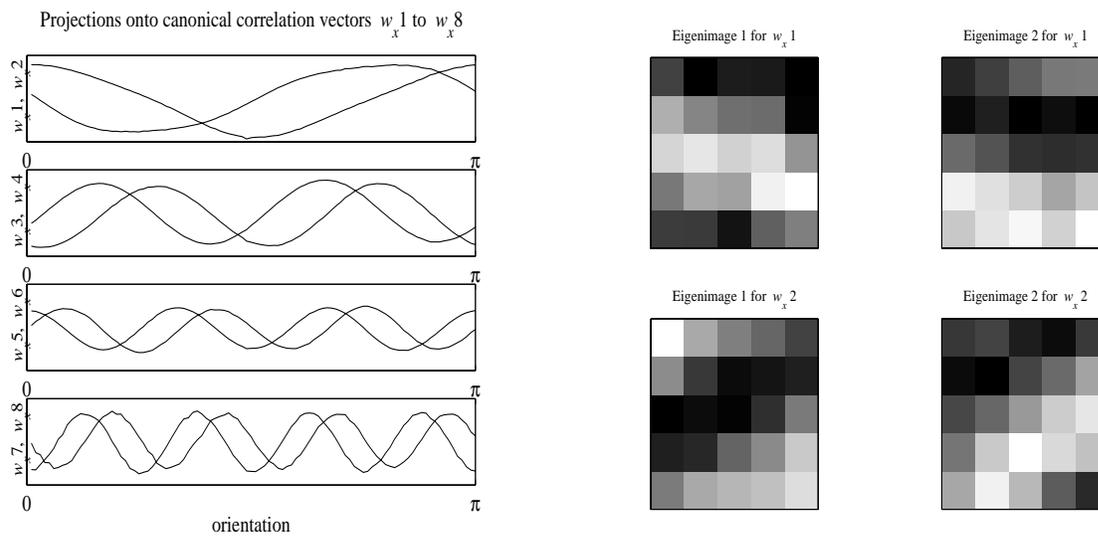


Figure 3: **Left:** Projections of outer product vectors \mathbf{x} onto the 10 first canonical correlation vectors. **Right:** Eigenvectors of the canonical correlation vectors viewed as images, “eigenimages”.

5.2 Learning higher level operations

In this experiment, we use the output from neighboring sets of quadrature filters rather than pixel values as input to the algorithm. This can be justified by the fact that we have seen that quadrature filters can be developed on the lowest (pixel) level using this algorithm. We will show that canonical correlation can find a way of combining filter output from a local neighborhood to get orientation estimates that is less sensitive to noise than the standard vector averaging method [5].

Let \mathbf{q}_x and \mathbf{q}_y be 16-dimensional complex vectors of filter responses from four quadrature filters from each position in a 2×2 neighborhood. (The content in each position could be calculated using the method in the previous experiment.) Let \mathbf{X} and \mathbf{Y} be the real parts³ of the outer products of \mathbf{q}_x and \mathbf{q}_y with themselves respectively and rearrange \mathbf{X} and \mathbf{Y} into 256-dimensional vectors \mathbf{x} and \mathbf{y} . For each pair of vectors, the local orientation was equal while the phase and noise differed randomly. The SNR was 0 dB. The two largest canonical correlations were both 0.8. The corresponding vectors detected the double angle of the orientation invariant with respect to phase.

New data were generated using a rotating sine-wave pattern with an SNR of 0 dB and projected onto the two first canonical correlation vectors. The orientation estimates are shown to the left in figure 4 together with estimates using vector averaging on the same data. In the right figure, the angular error is shown for both methods. The mean absolute angular error was 16° for the canonical correlation method and 22° for the vector averaging method, i.e. an improvement by 27%. Note that the neighborhood is very small (2×2), but preliminary tests indicate that the attainable improvement in noise reduction increase rapidly with neighborhood size.

References

- [1] S. Becker and G. E. Hinton. Learning mixture models of spatial coherence. *Neural Computation*, 5(2):267–277, March 1993.
- [2] R. D. Bock. *Multivariate Statistical Methods in Behavioral Research*. McGraw-Hill series in psychology. McGraw-Hill, 1975.
- [3] M. Borga. Hierarchical Reinforcement Learning. In S. Gielen and B. Kappen, editors, *ICANN'93*, Amsterdam, September 1993. Springer-Verlag.

³It turns out that no correlations are found in the imaginary parts which, hence, can be omitted.

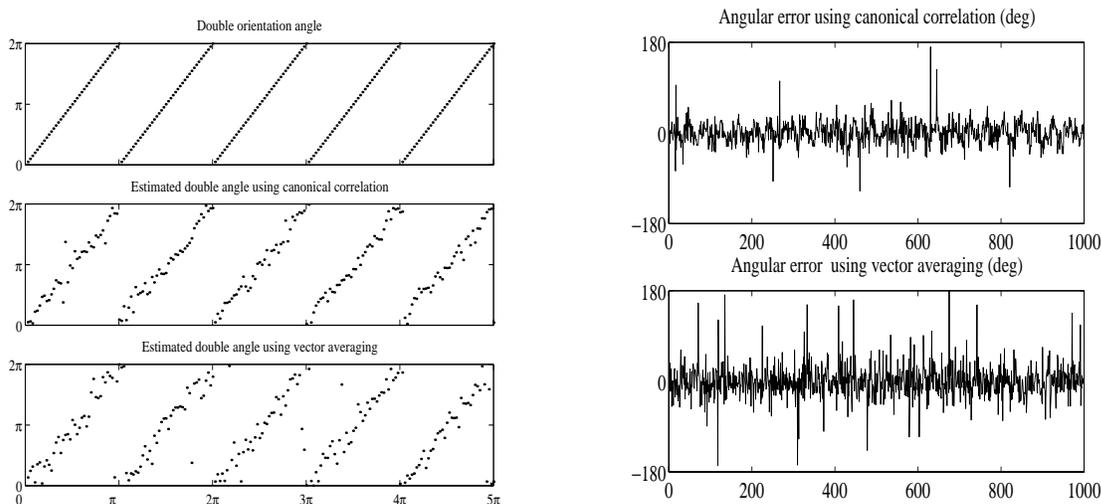


Figure 4: **Left:** Orientation estimates for a rotating sine-wave pattern on a 2×2 neighborhood with an SNR of 0 dB, using filter combinations found by canonical correlations (middle) and vector averaging (bottom). The correct double angle is shown as reference (top). **Right:** Angular errors for 1000 different samples using canonical correlations (middle) and vector averaging (bottom).

- [4] M. Borga. Reinforcement Learning Using Local Adaptive Models, August 1995. Thesis No. 507, ISBN 91-7871-590-3.
- [5] G. H. Granlund and H. Knutsson. *Signal Processing for Computer Vision*. Kluwer Academic Publishers, 1995. ISBN 0-7923-9530-1.
- [6] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28:321-377, 1936.
- [7] J. Kay. Feature discovery under contextual supervision using mutual information. In *International Joint Conference on Neural Networks*, volume 4, pages 79-84. IEEE, 1992.
- [8] H. Knutsson. Representing local structure using tensors. In *The 6th Scandinavian Conference on Image Analysis*, pages 244-251, Oulu, Finland, June 1989. Report LiTH-ISY-I-1019, Computer Vision Laboratory, Linköping University, Sweden, 1989.
- [9] H. Knutsson. Tensor Based Spatio-temporal Signal Analysis. In J.L. Crowley H.I. Christensen, editor, *Vision as Process*. Springer, 1995. Basic Research Series.
- [10] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59-69, 1982.
- [11] T. Landelius and H. Knutsson. The Learning Tree, A New Concept in Learning. In *Proceedings of the 2:nd Int. Conf. on Adaptive and Learning Systems*. SPIE, April 1993.