A Model for Analyzing Dependencies between Two ICA Features in Natural Images

Mika Inki

Neural Networks Research Centre
Helsinki University of Technology
P.O. Box 5400, FIN-02015 HUT, Finland

Abstract. In this paper we examine how the activation of one independent component analysis (ICA) feature changes first and second order statistics of other independent components in image patches. Essential for observing these dependencies is normalizing patch statistics, and selecting patches according to activation. We then estimate a model predicting the conditional statistics of a component using the properties of the corresponding feature as well as those of the conditioning feature.

1 Introduction

Independent component analysis has been used successfully in analyzing image data, even though the model is fundamentally insufficient for describing images. In ICA the observed data is expressed as a linear transformation of latent variables that are nongaussian and independent. We can express the model as

$$\mathbf{x} = \mathbf{A}\mathbf{s} = \sum_i \mathbf{a}_i s_i, \tag{1}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_m)$ is the vector of observed random variables, $\mathbf{s} = (s_1, s_2, \ldots, s_n)$ is the vector of latent variables called the independent components (ICs) or source signals, and $\mathbf{A}$ is an unknown constant matrix, called the mixing matrix. The columns of $\mathbf{A}$ are often called features or basis vectors. Exact conditions for the identifiability of the model were given in [2], and several methods for estimation of the classic ICA model have been proposed in the literature, see [5] for a review.

The assumption of independence is fundamental in ICA. Most types of natural data (e.g. image data) do not, however, have such independent (linear) features that ICA attempts to find. It is important to know about the data structures not captured by the ICA model. It is possible to use these structures to extend and improve the ICA model of image data. The usefulness of this approach can be motivated by the link between ICA features and cortical simple cell receptive fields, see [7]. Our approach here has similarities to some analyses made with Gabor functions [1], or with higher-order ICA models [4] and with our earlier paper [6], but the approach of using a parametric model to predict the changes in the statistics of one IC due to the activity of another has not, to our knowledge, been used earlier. We chose our model so that its properties are easily analyzable, yet is able to capture most of the structure.

2 Analysis of conditional dependencies

We will investigate how the statistics of image patches (image windows, data samples) change, when we know one specific IC is highly active. By activity we mean that the absolute value of the estimated IC ($|\mathbf{y}_i|, \quad \mathbf{y}_i = \mathbf{w}_i^T \mathbf{x}$) exceeds some threshold α. In these cases the component can be considered to describe something essential appearing in the patch, i.e. part of an edge or line. We denote the indexes for which IC i exceeds α by

$$I_{\alpha,i} = \{t \quad | \quad |y_i(t)| > \alpha\}, \tag{2}$$

and the subset of the whole data associated with $I_{\alpha,i}$ by $\mathbf{X}_{\alpha,i}$.

There are two essential steps of preprocessing we do here that are important for observing the statistics. The first is normalizing variances of individual patches or patch norms (these differ by an irrelevant scaling factor). With this, and with the reduction of the mean and whitening of the data, contrast variations between the patches are mostly eliminated. With this normalization, speaking of activation is more meaningful because a certain level of activation means that the feature contributes a specified portion of the content (variance) of the patch.

Note that we demand that the patch variances equal unity and that the data is white simultaneously. The requirement for whitening can be written as

$$\mathrm{cov}(\mathbf{x}) = \mathbf{I} - \frac{1}{n}\mathbf{1}, \tag{3}$$

where the latter term on the right side results from eliminating the patch means. For fixed t (i.e. for each patch) we require for the mean, variance and norm:

$$\mathrm{mean}_i(x_i(t)) = 0, \qquad \mathrm{var}_i(x_i(t)) = 1 \quad \Longleftrightarrow \quad \|\mathbf{x}(t)\| = \sqrt{n}. \tag{4}$$

We will later discuss how we enforce all these requirements simultaneously.

The second step of preprocessing before examining the dependencies is normalization by the sign of the active component. There is no inherent sign attached to an image patch as components can be positive or negative mostly regardless of each other. But, for example two collinear edge detectors may very well exhibit consistently the same signs when either one is highly active as this corresponds to having an edge in the patches to which both react. We will denote the sign-normalized data associated with independent component i and threshold α as $\mathbf{Z}_{\alpha,i}$:

$$\mathbf{Z}_{\alpha,i} = \{\mathbf{z}(t) \quad | \quad \mathbf{z}(t) = \mathbf{x}(t)\mathrm{sign}(y_i(t)), \quad t \in I_{\alpha,i}\}. \tag{5}$$

We will denote the vectors in $\mathbf{Z}_{\alpha,i}$ as $\mathbf{z}_{\alpha,i}$.

3 Data Selection and Preprocessing

As data we used 24 images of landscapes, plants and animals. The images were taken with a digital camera (Canon Ixus 400), converted to grayscale, and block averaged in four by four pixel blocks (and downscaled by the same factor). An area of 512 by 384 pixels was then selected of each of the downscaled images. The downscaling should pretty much negate artifacts brought by color interpolation, noise reduction and even compression. The images were saved in a 16-bit grayscale TIFF-format (after the histograms were stretched to cover the 16-bit range). The original images, and the 16-bit TIFFs can be found at the web address **http://www.cis.hut.fi/inki/images/**.

Next we sampled 200000 12 by 12 pixel patches from these images. The patch mean was subtracted from each patch. Sometimes, due to the fact that usually the upper parts of the images are the brightest (e.g. parts of the sky are visible), even though the mean value has been subtracted from each patch, the pixels still do not have zero mean. Therefore we randomly assigned a new sign to each of the patches.

We then whitened the data and normalized the patch variances. As patch variance normalization affects the covariance of the data, whitening and patch normalization were repeated (alternately) a total of ten times. After this, the largest (nonzero) eigenvalue of the covariance matrix was less than a millionth larger than the smallest. Note that this whole process can be described as whitening and patch normalization, as it corresponds to multiplying the data with a matrix (product of all the whitening matrices) and then normalizing the patches.

FastICA [3] in symmetric mode using the hyperbolic tangent nonlinearity was then used to perform ICA on the data. The basis vectors we found can be seen on the left side of Figure 1. These are presented here in the original, not whitened space.

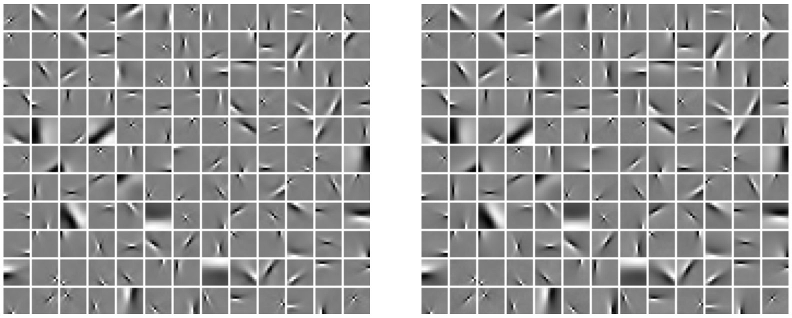

Fig. 1. Left: ICA basis found in normalized data. Right: Mean values of the patches when normalizing by the sign of the active component and using threshold $\alpha = 3$.

We will have to limit our analysis of the statistics to a single value of α. As a compromise between the number of samples, and size of the dependencies, we selected $\alpha = 3$ as the baseline value. In this case 87.7 percent of all patches are included in at least one of the active sets $I_{3,i}$ (Eq. 2), average size of which is 2650 patches, and the active feature contributes at least 6.3 % of patch variance.

4 Dependencies in the ICA basis

In order to analyze the qualities of the dependencies, we decided to estimate a model where the properties of the features predict the change in the statistics for a given value of α. So, the inputs are calculated from the properties of the conditioning and conditioned feature, and the output should be the statistic, e.g. variance of the conditioned component.

4.1 A model for analyzing the dependencies

In order to estimate the properties of the features, we fitted Gabor functions to the ICA features on the left side of Figure 1. As Gabor functions we used real-valued two-dimensional functions:

$$g(\mathbf{r}) \propto \exp(-\sum_{i=1}^{2} \frac{r_i^2}{2b_i^2}) \cos(2\pi\omega r_1 + \theta). \tag{6}$$

Here $\mathbf{r}$ is the two dimensional position vector, b_i:s are the widths in corresponding dimensions of $\mathbf{r}$, θ is the phase and ω the frequency. Any Gabor function can now be obtained with a rotation, translation, and scaling of $g(\mathbf{r})$. Let us denote the angle of this rotation as β.

Our model was of the type

$$R'_{i,j} = f_l(\prod_{k=1}^{l-1} f_k(G_{i,j}(k))), \tag{7}$$

where $R'_{i,j}$ is the estimate of the dependency $R_{i,j}$ between the conditioning IC i and the conditioned IC j, $G_{i,j}(k)$ is the k:th value measured from the features corresponding to ICs i and j. As f_k, $k < l$ we used functions consisting of evenly spaced five points (the smallest of which is at the smallest value of corresponding G_i, the largest at the largest value) that were interpolated with piecewise cubic Hermite interpolation as implemented in Matlab version 6.5.

The function f_l consisted of eleven unevenly spaced points. The first point is at the smallest value of $\prod_{j=1}^{l-1} f_i(G_i)$, i.e. the zero percent mark, the second at the five percent mark (where five percent of $\prod_{j=1}^{l-1} f_i(G_i)$ are smaller), third at the ten percent mark, then 20%, 35%, 50%, 65%, 80%, 90% 95% and the final one at the 100% mark. Additionally, f_l was required to be monotonically increasing and positive. We used $l = 5$ here, and the model had therefore a total of 31 free parameters. Of these 31 parameters, four are actually redundant, as the scaling in f_5 can offset the scaling in any (and all) of the other functions.

As $R_{i,j}$ we used the variances of the conditioned components, as well as the absolute values of the mean values of the conditioned components. Variances highlight large dependencies better than standard deviations, which is partly why we chose to model them. Additionally, we achieved best fits with these choices. We fitted Gabor functions to the ICA features and, in order to have sensible results, excluded the smallest features that cannot be so well described as Gabor functions from the analysis. We picked 98 of the best fitting features, so there were a total of 9506 ($= 98^2 - 98$) examples of $R_{i,j}$ for further analysis.

We used four variables as $G_{i,j}(k)$:s. The first was $G_{i,j}(1) = \log(b_i/b_j)$, where b_i and b_j are the widths of the fitted Gabors for the conditioning and conditioned components respectively. The second was a measure of the difference between the orientations of the components, $G_{i,j}(2) = |\sin(\beta_i - \beta_j)|$. The third was a collinearity measure: new features are obtained by ignoring the attenuation of the Gabors along the edge, i.e. along r_2, these new features are normalized (w.r.t. inner product with themselves), and $G_{i,j}(3)$ is the absolute value of the inner product of these new features for the conditioning and conditioned component. The final variable $G_{i,j}(4)$ is obtained by discarding the cosine part of the Gabors, normalizing, and taking the inner product of these new features, i.e. it was an overlap measure of the functions, which depends on distance and size difference.

Note that of these only the first variable G_1 can capture nonsymmetric properties of the dependency, i.e. if the places of the conditioning and conditioned component are exchanged, the dependency can change.

4.2 Results

We fitted the model in equation (7) to the statistics for $\alpha = 3$. The minimum of $\sum_{i,j}(R_{i,j} - R'_{i,j})^2$ was searched by Matlab's fminsearch -function which uses the Nelder-Mead method not requiring derivatives. The error we ended up with was 0.2656 of the variance of $R_{i,j}$ for second order statistics, and 0.4787 of the variance of $R_{i,j}$ for first order statistics.

We have plots of the individual f_k:s, $k \leq 4$, in Figure 2. The functions for second order statistics have been plotted with solid lines. The first function f_1 shows a maximum at zero, i.e. when the functions are of the same size, and slightly unsymmetric behaviour (so the model is slightly unsymmetric). The second variable has a maximum at zero, i.e. when the orientations of the components are identical (or differ by π). The third component shows a maximum when the modified features overlap the most, i.e. are in a sense collinear. The fourth shows a maximum when the overlap of the features is the greatest. These functions are plotted in (natural) logarithmic scale, and their precise values do not matter, for which reason their maximum values have been normalized. The differences between their ranges does matter, and one can see that this is the biggest in the case of the fourth function, and smallest in the case of the first function. This would strongly suggest that the fourth component is the most important for the fit, and the first is the least important.

We have also plotted the individual f_k:s for first order statistics with dashed lines in Figure 2. As one can see, the f_k:s have similar shapes to the corresponding functions for second order statistics, yet there are differences. Again, judging by the ranges of the functions, the fourth variable (overlap) appears to be the most important for the fit, but now the importance of the second variable (difference of orientation) seems to be smaller, and the third variable (collinearity) appears to be more important.

We have scatterplots in Figure 3, where on the x-axis are the values obtained by multiplying the $f_k(G_{i,j}(k))$:s, $k \leq 4$, and on the y-axis the observed values $R_{i,j}$. Both axes are in logarithmic scale. On the left side we have the scatterplot for the first order statistics, and on the right side the scatterplot for second order statistics. Also plotted in the figure (with a solid line) is the function f_l. The interpolation of function f_l was done on logarithmic scale.

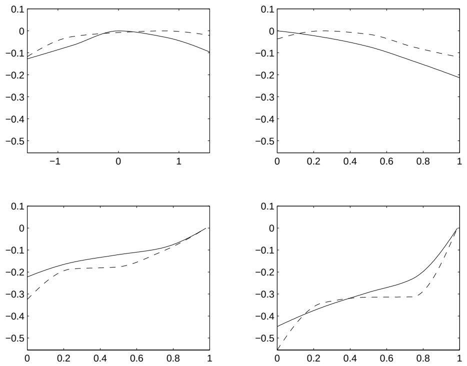

Fig. 2. Logarithms of the f_k:s, $k \leq 4$. Top left: Logarithmic difference in feature width. Top right: Difference in angle. Bottom left: Collinearity measure. Bottom right: Overlap measure. First and second order statistics with dashed and solid lines, respectively.

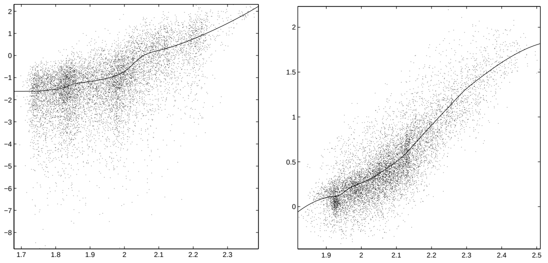

Fig. 3. Logarithm of the observed statistics on the y-axis, $\log(\prod_{k=1}^{l-1} f_k(G_{i,j}(k)))$ on the x-axis. Left side: First order statistics. Right side: Second order statistics. Also plotted in both figures is the correspoding f_l.

It appears that for second order statistics, the function f_l has a somewhat sigmoidal shape. This is sensible, as very low values indicate that all the functions f_k give a low value, but already for example insignificant overlap of the features is enough to make the them virtually independent (and the significance of the other factors should be reduced). Similar argument can be made of very high values. It is harder to say anything of the shape of f_l for first order statistics.

Another way of exploring how important the different variables are for the fit is by excluding one variable from the analysis, and fitting the model again. For the second order statistics, excluding the first variable produced an error of 0.2869, excluding the second 0.3668, excluding the third 0.3084, and excluding the fourth 0.4775. This supports our earlier conjecture that the fourth variable is the most important, and the first the least important in the fit.

For the first order statistics, the errors were 0.4923, 0.5128, 0.5846, and 0.6844, for excluding the first, second, third, and fourth variable respectively. This supports our earlier reasoning that the second variable is not so important for the fit as the third, which makes sense as collinearity of two features means they basically describe the same edge at different positions. As can be seen in Figure 1, the mean values for high activation essentially express how the feature on average continues (extends to orthogonal dimensions). The mean value features are longer than the original feature and extend further from the zero crossing. Orientation is not so important for first order statistics because similar orientation without collinearity does not produce a consistent edge.

So, one can say that the most important factor for the size of the dependency (first or second order) is the overlap of the features. Note that the way in which we measure overlap depends on distance between the features and on size difference. However, one can't say that overlap is the only factor to be considered. We also attempted to use additional parameters ($G_{i,j}(k)$, $k > 4$) in the model, but could not achieve essentially better fits.

4.3 Assessing the validity of the model

In order to test the validity of our model, we also fitted a multilayer perceptron (MLP) network to the same variables. An MLP should be able to fit into the dependencies between the parameters, whereas in our model the parameters are essentially independent w.r.t. their contribution to the dependency, barring for the effect of f_l. We used Matlab's Neural Network Toolbox for creating and training the MLP. The input and target variables were the same as earlier.

For second order statistics, with five hidden layer neurons (and as many parameters as in our model), we obtained a very similar error measure: 0.2625. But with an MLP it is harder to interpret the properties of the fitted model. With fifty hidden layer neurons, i.e. a total of 301 parameters, we obtained an error of 0.200. For first order statistics, with five hidden layer neurons, we obtained an error measure of 0.4580. With fifty hidden layer neurons, the error was 0.3492. These values are sufficiently close to the errors we obtained with our model with less free parameters (and less chance of overfitting) for us to say that most of the information available in the four varibles is captured by our model.

Note also that we can estimate a lower bound for the error in the fit (without overfitting). We made a new version of the data, where by construction the value of the conditioning (active) IC does not affect other components. We call this $\mathbf{U}_{\alpha,i}$. For each component i and every patch $\mathbf{z}_{\alpha,i}(t)$ (Equation 5), we keep the active component, and select randomly a patch $\mathbf{x}(t_2)$ from which we take the other components. We multiply these other components so that the variance of the new patch is normalized. That is:

$$\mathbf{u}_{\alpha,i}(t) \leftarrow \mathbf{P}_i\mathbf{z}_{\alpha,i}(t) + (\mathbf{I} - \mathbf{P}_i)\mathbf{x}(t_2)\frac{\sqrt{n - \|\mathbf{P}_i\mathbf{z}_{\alpha,i}\|^2}}{\|(\mathbf{I} - \mathbf{P}_i)\mathbf{x}(t_2)\|}, \tag{8}$$

where $\mathbf{P}_i$ projects the data into a subspace spanned by component i. We calculated the variance of $R_{i,j}$ from this control data, and it was as low as 0.0061 for second order statistics, even though our best fit with MLP was only 0.200. When fitting our model and the MLP network, we have the added difficulty of Gabor parameter estimation and choosing Gabor parameters for further use. This is significant, especially as the ICA features are not perfectly Gabor functions. We can assume that this is for a large part responsible for the difference between the best error and our noise estimate. For first order statistics this lower bound for the fit was 0.0468, which is still significantly lower than our best fits.

5 Conclusions

Here we studied the residual dependencies in the ICA model for image data by looking at what effect the activation of one feature has for the first and second order statistics of other features. Changes in these statistics of a conditioned IC are linked to the changes in the probability of its activity. We showed how these changes can be largely explained by a model with a few basic properties of the features as parameters, including their overlap, collinearity and orientation.

The results obtained here can offer some useful information for image analysis, or processing, or may offer a small insight into the workings of biological visual systems.

References

1. R.W. Buccigrossi and E.P. Simoncelli. Image compression via joint statistical characterization in the wavelet domain. *IEEE Transactions on Image Processing*, 8(12):1688–1701, 1999.
2. P. Comon. Independent component analysis—a new concept? *Signal Processing*, 36:287–314, 1994.
3. A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 10(3):626–634, 1999.
4. A. Hyvärinen, P. O. Hoyer, and M. Inki. Topographic independent component analysis. *Neural Computation*, 13(7), 2001.
5. A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis.* Wiley Interscience, 2001.
6. M. Inki. Examining the dependencies between ICA features of image data. In *Proc. of ICANN/ICONIP 2003*, Istanbul, Turkey, 2003.
7. B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.