Properties of Independent Components of Self-Motion Optical Flow

Marwan A. Jabri†, Ki-Young Park‡, Soo-Young Lee‡ and Terrence J. Sejnowski§

†*Oregon Graduate Institute of Science and Technology, Oregon, USA, and*
Computer Engineering Laboratory, School of Electrical and Information Engineering
The University of Sydney, Australia
marwan@ece.ogi.edu
‡*Brain Science Research Center and Department of Electrical Engineering*
Korea Advanced Institute of Science and Technology, Taejeon, Korea
§*Computational Neurobiology Laboratory, The Salk Institute, California, USA*

Abstract

In this paper we describe the properties of independent components of optical flow of moving objects. Video sequences of objects seen by an observer moving at various angles, directions and distances are used to produce optical flow maps. These maps are then processed using independent component analysis, which yields filters that resemble the receptive fields of dorsal medial superior temporal cells of the primate brain. Contraction, expansion, rotation and translation receptive fields have been identified. Our results support Barlow's sensory coding theory and are in-line with other work on independent components of image and video intensities.

1. Introduction

About fourty years ago, Barlow proposed that the brain could represent sensory information using factorial code [1]. More recently researchers have reported the emergence of independent components from natural images [2] and video sequences [3] when entropy maximization techniques are used on their intensities. The independent components of natural images have properties similar to the localized edge receptive fields of simple cells in the primary visual cortex of mammals. The independent components of video sequences resemble localized spatiotemporal receptive fields – moving edge filters [4].

It is known that complex visual motion processing is performed by the middle temporal (MT) and medial superior temporal (MST) areas in the brain of primates. In particular, the dorsal region of MST (MSTd) has attracted a great deal of neurophysiological interest because of its role in processing complex visual motion patterns. Cells in this area have large receptive fields and respond selectively to the expansion, rotation, and spiral motion stimuli that are generated when the observer moves.

Area MSTd receives its primary input from the MT area. MT cells produce highly selective responses to directional motion and speed in relatively small regions of the visual field. Hence it is considered that their role is representing optical flow information, though representation aspects are not well understood. MT cells also respond to motion disparity.

If we extend the factorial representation/coding hypothesis to MT and MST, the question is, what would be the independent components of complex motions and how well would they fit to the properties of the receptive fields of MSTd cells?

Zemel and Sejnowski [5] hypothesized that complex optical flows produced by the combination of observer motion with other independently moving objects. According to this hypothesis the optical flow is composed of multiple regular patterns, to which MSTd cells had been found to be selectively tuned. The authors suggested that what the functional role for the MST area: to encode the ensemble of motion causes that generates the complex flow field. They proposed an MST model based upon an auto-encoder neural network. An auto-encoder is a neural network trained using supervised learning techniques to duplicate, at its outputs, the same patterns applied to its inputs. The input-output mapping of an auto-encoder network is constrained by a non-linear transformation through the middle layers of the neural network. The distance measure between the target output and the actual output can be the cross-entropy measure. After training, the filters (receptive fields of neurons) of the hidden layer of the auto-encoder were found to selectively tuned to specific motion like rotation, contraction and translation.

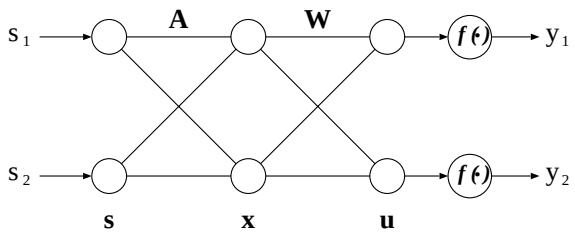


Figure 1. The linear mixing/unmixing model.

In this paper, we describe simulation experiments aimed at exploring the properties of independent components extracted from optical flow of complex object motion. We have used a ray tracing system to generate video sequences of objects seen by an observer moving at various distances, angles and directions. The optical flow of the sequences was computed and independent components were extracted using an independent component analysis algorithm. Our hypothesis was that filters produced using the analysis would have similarities with the receptive fields of MSTd cells. An analysis of the independent components revealed this similarity and filters tuned to contraction, rotation, and translation were present. Our results support the theory of sensory coding proposed by Barlow and elaborated by others [6, 7]

In Section 2 we review independent component analysis, associated learning algorithms, and its applications in signal separation and feature extraction. In Section 3 we review optical flow computation and describe briefly the algorith we have used in our experiments. Section 4 describes our experiments and presents our results. Finally in Section 5 we discuss our results and presents directions for future work.

2. Independent Componen t Analysis

Independent component analysis (ICA) is an information maximization method for extracting the causes or sources from multidimensional observations. ICA has been applied to blind source separation and feature extraction problems.

Blind separation techniques can be used in any domain where an array of N sensors receive linear mixtures of N source signals. Examples of such blind separation of such mixtures include speech separation ('cocktail party problem'), processing of arrays of radar or sonar signals and processing of arrays of multi-sensor biomedical recordings. The term *blind* indicates that both the source signals and the mixing process of the signals are unknown. Figure 1 shows the basic network for the blind signal separation in the case when two sources are mixed by an unknown mixing matrix $\mathbf{A}$. The objective of the ICA algorithm is the following: given a set of observation vectors, where each vector $\mathbf{x}$ represents one observation, find the vector of underlying sources $\mathbf{s}$. That is, find the *unmixing* matrix $\mathbf{W}$ which is the inverse matrix of the mixing matrix $\mathbf{A}$. The assumption is that the underlying sources are statistically independent from each other.

Parallel to blind source separation studies, unsupervised learning rules based on information theory were proposed by Linsker [8]. Here the goal is to maximize the mutual information between the inputs and outputs of a neural network. This approach is related to the principle of redundancy reduction suggested by Barlow [1] as a coding strategy by neurons in the brain. According to this strategy, each neuron would encode features that are as statistically independent as possible from other neurons over a natural ensemble of inputs.

2.1. Derivation of Learning Rule

In the linear mixing model the observation vector can be written as

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{1}$$

where the independent sources, $\mathbf{s}$, the components of an observation vector $\mathbf{x}$ are no longer independent and their mutual information is defined as

$$I(\mathbf{x}) = \int p(\mathbf{x}) \log \frac{p(\mathbf{x})}{\prod_{i=1}^{N} p_i(x_i)} d\mathbf{x} \tag{2}$$

The mutual information is always positive and is zero if and only if the components are independent [9]. The goal of the ICA algorithm is to find the linear transformation $\mathbf{W}$ of the dependent observations that makes the output $\mathbf{u}$ as independent as possible.

$$\mathbf{u} = \mathbf{W}\mathbf{x} \tag{3}$$

Nadal and Parga showed that in the low-noise case, the maximum of the mutual information between the inputs $\mathbf{x}$ and outputs $\mathbf{y}$ of a neural network implied that the output distributions were factorial where $\mathbf{y}$ was the nonlinearly transformed version of the $\mathbf{u}$: $\mathbf{y} = f(\mathbf{u})$ [10]. In other words, maximizing the information transfer in a nonlinear neural network minimizes the mutual information among the outputs when optimization is done over both the synaptic weights and the nonlinear transfer function.

Bell and Sejnowski proposed a simple learning rule for a feedforward neural network that blindly separates linear mixtures $\mathbf{x}$ of independent sources $\mathbf{s}$ using information maximization. They showed that maximizing

the joint entropy $H(\mathbf{y})$ of the output of a neural network can approximately minimize the mutual information among the output components. The joint entropy at the outputs of a neural network is

$$H(\mathbf{y}) = H(y_1) + \cdots + H(y_N) - I(\mathbf{y}) \quad (4)$$

where $H(y_i)$ are the marginal entropies of the outputs and $I(y_1, \ldots y_N)$ is their mutual information. Each marginal entropy can be written as

$$H(y_i) = -E[\log p(y_i)]. \quad (5)$$

The probability density of the output y_i can be described using the probability density of the estimated independent sources, u_i.

$$p(y_i) = \frac{p(u_i)}{|\frac{\partial y_i}{\partial u_i}|} \quad (6)$$

Then the Eq. 4 can be written as

$$H(\mathbf{y}) = -E[\log \frac{p(u_1)}{|\frac{\partial y_1}{\partial u_1}|}] + \cdots - E[\log \frac{p(u_1)}{|\frac{\partial y_1}{\partial u_1}|}] - I(\mathbf{y}) \quad (7)$$

If we know the probability distribution of the $\mathbf{u}$ which is the estimation of the underlying source, $\mathbf{s}$, we can eliminate all the marginal entropy terms to zero by setting $\frac{\partial y_i}{\partial u_i} = p(u_i)$, which gives

$$H(\mathbf{y}) = -I(\mathbf{y}). \quad (8)$$

Now the direct maximization of the joint entropy between output components implies the minimization of the mutual information which makes the output components independents. To maximize the joint entropy, an iterative gradient ascent algorithm is used by calculating,

$$\frac{\partial H(\mathbf{y})}{\partial \mathbf{W}} = (\mathbf{W}^T)^{-1} + \left(\frac{\frac{p(\mathbf{u})}{\partial u}}{p(\mathbf{u})} \right) \mathbf{x}^T \quad (9)$$

A complete derivation of the algorithm can be found in [11]. Amari suggested the natural gradient learning rule to speed up the convergence which rescales the entropy gradient [12]:

$$\Delta \mathbf{W} \propto \frac{\partial H(\mathbf{y})}{\partial \mathbf{W}} \mathbf{W}^T \mathbf{W} = \left[\mathbf{I} + \left(\frac{\frac{p(\mathbf{u})}{\partial u}}{p(\mathbf{u})} \right) \mathbf{u}^T \right] \mathbf{W} \quad (10)$$

If we define the score function $\phi(\cdot)$ as

$$\phi(\mathbf{u}) = -\frac{\frac{p(\mathbf{u})}{\partial u}}{p(\mathbf{u})}, \quad (11)$$

Eq. 10 leads to

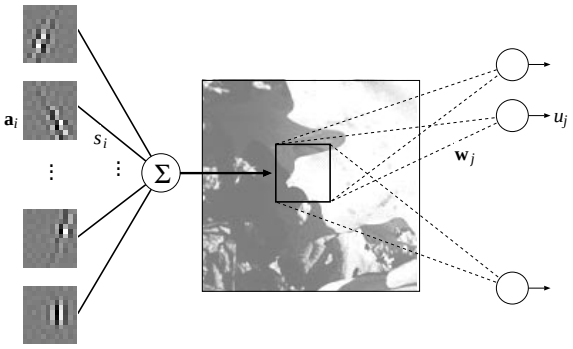


Figure 2. Linear image synthesis model.

$$\Delta \mathbf{W} \propto \left[\mathbf{I} - \phi(\mathbf{u}) \mathbf{u}^T \right] \mathbf{W} \quad (12)$$

The maximum likelihood estimation approach to independent component analysis, which assumes the factorial code of output neurons, gives the same learning rule as the information maximization approach.

2.2. ICA and Feature Extraction

In feature extraction studies using sparse coding [13] or ICA [14], the observation vectors like natural images are assumed to be linear mixture of several underlying basis vectors, $\mathbf{a}_i$ which constitute the columns of a matrix $\mathbf{A}$ corresponding to the mixing matrix of the blind signal separation problem. The amount of contribution each basis vector makes to compose an observation is represented by the vector $\mathbf{s}$. Each element of vector $\mathbf{s}$ has its own associated basis function, and represents an underlying "causes" of the image or any observation vector. Hence the linear image synthesis model is described by:

$$\begin{aligned} \mathbf{x} &= \sum_i s_i \mathbf{a}_i \\ &= \mathbf{A}\mathbf{s} \end{aligned} \quad (13)$$

where $\mathbf{x}$ is an observation vector. Figure 2 shows the linear image synthesis model and the basis functions extracted using the ICA on natural images.

The underlying causes can be extracted by corresponding independent component filters $\mathbf{w}_i$ which constitutes the rows of $\mathbf{W}$.

$$u_i = \mathbf{w}_i \cdot \mathbf{x} \quad (14)$$

where $\cdot$ operation denotes the inner product and u_i is an element of the recovered underlying cause $\mathbf{u}$, which responds to a specific feature of observation $\mathbf{x}$ captured by the related filter $\mathbf{w}_i$.

The problem is to find $\mathbf{W}$, if it exists, with the assumption that underlying causes are statistically independent. This can be done by ICA algorithms. In the simulations we report below, we have used the ICA learning rule, which maximizes the entropy of output $\mathbf{y}$, the nonlinear transformed version of $\mathbf{u}$ as reported by Bell and Sejnowski [2]. ICA on natural images resulted in the filters shown in figure 2 – localized edge filters – and similar work on the video sequences yielded the spatiotemporal edge filters that are sequences of edge filters shown in figure 2 but moving in temporal domain.

3. Optical Flow

Optical flow is an approximation to the 2-d motion field of spatiotemporal patterns of image intensity [15] – a projection of the 3-d velocities of surface points onto the imaging surface. Sequences of time-ordered images allow the estimation of two-dimensional image motion as either instantaneous image velocities or discrete image displacements. In other words each element in an optical flow map represents the instantaneous velocity of the object or the point in that element location. Formally if $I(\mathbf{x}, t)$ is the image intensity function, then

$$I(\mathbf{x}, t) \approx I(\mathbf{x} + \delta\mathbf{x}, t + \delta t) \tag{15}$$

where $\delta\mathbf{x}$ is the displacement of the local image region at $(\mathbf{x}, t)$ after time δt. Expanding the left-hand side of this equation in a Taylor series yields

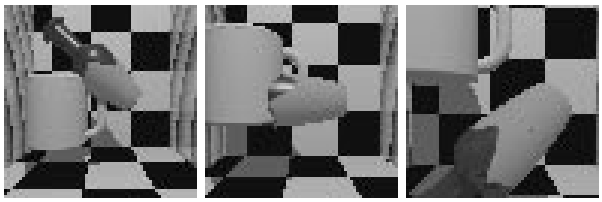
$$I(\mathbf{x}, t) = I(\mathbf{x}, t) + \nabla I \cdot \delta\mathbf{x} + \delta t I_t + O^2 \tag{16}$$

where $\nabla I = (I_x, I_y)$ and I_t are the 1st order partial derivatives of $I(\mathbf{x}, t)$ and O^2, the 2nd and higher order terms which are assumed negligible. Subtracting $I(\mathbf{x}, t)$ and dividing by δt reduces Eq. 16 to the following *optical flow constraint equation*,

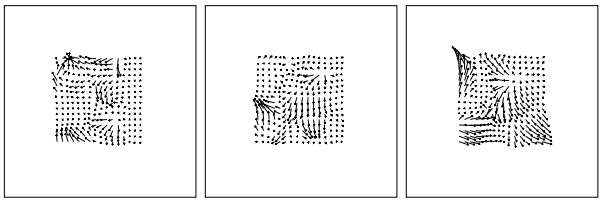
$$\nabla I \cdot \mathbf{v} + I_t = 0 \tag{17}$$

where $\mathbf{x} = (u, v)$ is the image velocity. Because there is only one constraint to solve two unknown parameters, finding exact solution of the Eq. 17 is ill-posed problem and only at the image locations where there is sufficient intensity structure can the motion be fully estimated with the use of the optical flow constraint equation.

There have been many studies on optical flow computation, which suggest auxiliary constraints to the optical flow equations including differential methods, frequency-based methods, correlation-based methods, multiple motion methods and temporal refinement methods. Beauchemin and Barron reviewed several methods and compared their performances [16, 15].



(a) video frame



(b) optical flows

Figure 3. Extract of an example video sequence and its associated optical flow

While deriving the optical flow constraint equations, the smooth changes of intensity over both spatial and temporal axis are assumed, but the occlusions of the objects obviously violate this assumption. In order to deal with object occlusions, Nagel proposed second-order derivatives to measure optical flow and suggested oriented-smoothness constraints where smoothness is not imposed across steep intensity gradients (edges). This prevents smoothing over intensity discontinuities [17, 15] most likely to represent object boundaries. Since our video sequences have many occlusions, and because occlusions represent important information for motion segmentation, we have computed the optical flow in our experiments using the Nagel algorithm.

In contrast to image intensities, each pixel in an optical flow map has two components (called x and y here) and thus a coding scheme is needed to apply independent component analysis. The representation we have used is the simplest. For each optical flow observation, we concatenated all x components followed by all y components to form a single optical flow observation.

4. Experiments

4.1. Methods

All video sequences used in our simulations were synthetic scenes created by a computer program (Persistence of Vision Ray Tracer, which is publicly available at http://www.povray.org). This program allows sim-

ulation of dynamic scenes including various stationary or moving objects, backgrounds, and observer motions. A scene contains up to three objects from a choice of seven types of objects – ball, cup, table, chair, cube, vase and table lamp – and various backgrounds (textures) which can be composed of planes of 5 different patterns. Figure 3 shows a snapshot of video sequence and the computed optical flow.

For each video sequence, a virtual space with a background was defined and the number of objects was set, together with an initial and final positions of an observer. Each object was placed in space at random. The sequence was generated as follows:

1. The field of view was set to 60×60 deg.
2. The observer was stationary with a probability of 1/3; the observer moved in the x-direction or the z direction with probabilities of 1/3 and 2/3 respectively; the observer could rotate to track a point or an object.
3. An object could move in all directions independently with a probability 1/5 in each of the x, y and z directions. Hence about half of the objects were in motion and any (complete or partial) occlusion of objects was allowed. In addition to translation, an object could rotate independently in the x-y plane.

For each sequence, the generation parameters were determined and a script was produced to generate a video sequence. A total of 30 frames was produced by specifying 3-D positions of the camera, and each object in the image and then updating the position of the camera and each object, based on the motion parameters. The script contained the description of the content of an image and made use of a set of graphics programs supplied in the ray-tracing package to render each image. The script also used a library contained in the package that included descriptions of the seven object shapes and backgrounds.

A total of 15,000 movies were created. From these, 45,000 optical flow maps were extracted for training. To produce an optical flow map, 15 frames were used for Gaussian smoothing. The size of each frame was 64×64 which corresponds to a 60×60 deg visual field and due to the spatial smoothing and a 2-to-1 subsampling, the size of the resulting optical flow map was 16×16. Hence, the training data consisted of 45,000 vectors of length 512 (optical flow map is 16×16, and each element has 2 dimensions). These vectors were zero-meaned and sphered (whitened by multiplying the square root matrix of the covariance matrix of the input vectors) before being processed by the ICA algorithm. This preprocessing removes the first and second order statistics of input vectors and therefore enhance the speed of the convergence.

Our experiments made use of the Matlab ICA toolbox developed by Makeig and his colleagues, which implements the information maximization algorithm of Bell and Sejnowski with the natural gradient feature of Amari, Cichocki and Yang [18, 2, 12].

The hyperbolic tangent function was used as the nonlinearity between the estimated underlying source, $\mathbf{u}$ and the output $\mathbf{y}$ and no additive bias term in the linear synthesis model was considered. We have used principal component analysis (PCA) preprocessing for dimension reduction prior to applying ICA. PCA is typically used to reduce dimension of input data based on *second* order statistics.

4.2. Results

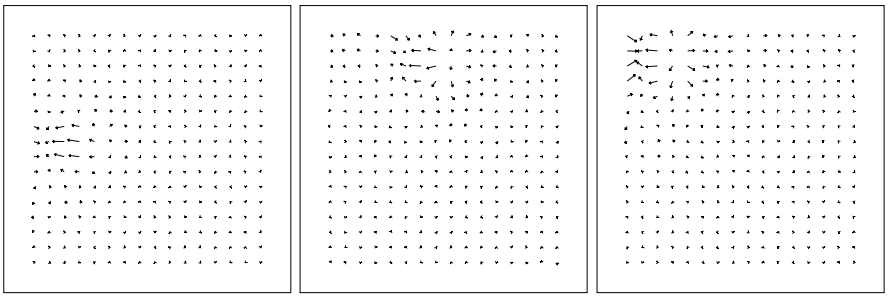
Figure 4 shows some of the filters extracted by ICA. We have experimented with both - with and without PCA as a preprocessing step. In cases where PCA was performed before ICA, the yielded filters were meaningful. However without PCA, the filters had too small receptive regions and showed no regular patterns. This means that the number of underlying basis functions to produce optical flow of dynamic scenes was quite small compared to the dimensions of input data (512 here). Using PCA, we reduced the dimensions of the input data to 50, 70, 100 and 200. More than 200 principal components seemed to be too many and less than 50 to be too few. The number of principal components between 50 and 150 resulted in qualitatively similar filters.

The ICA filters shown in figure 4 were obtained with a PCA preprocessing step that reduced the dimensions down to 100. The filters in the top row are receptive fields that respond to contraction and expansion, the middle row correspond to rotation and the third row corresponds to translation.

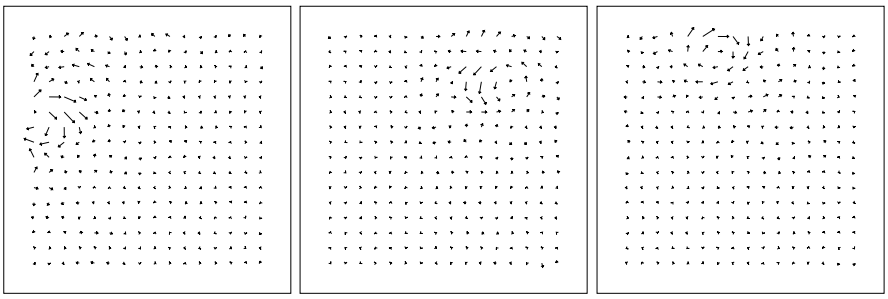
Although detailed quantitative measure have not yet been computed for the selectivity of these units, each filter can be predicted to be spatially localized and responds to a specific type of motion selectively.

The filters shown in the bottom row of figure 4 can be considered to respond to the combination of more than two types of movements. This could be similar to the response of MSTd cells that have been observed in many studies and that Duffy and Wurtz call double- and triple-component neurons [19].

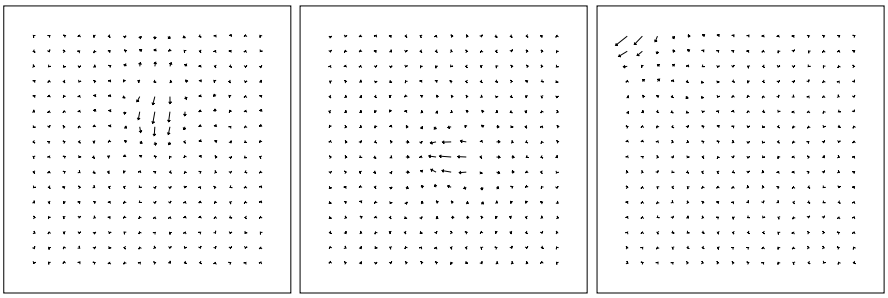
To test the response patterns of each filter, 15,000 optical flows were generated from 5,000 video sequences which were produced using the same procedures as the



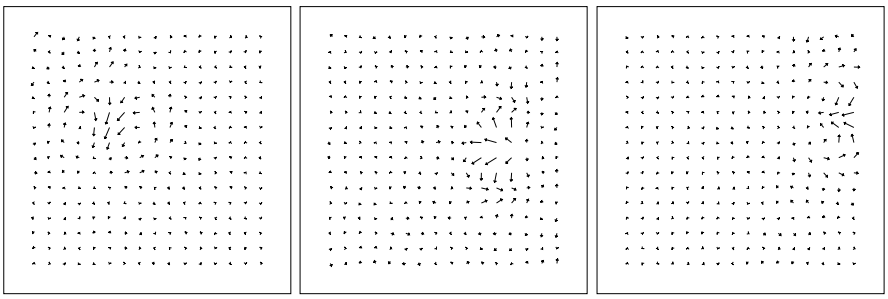
(a) Expansion/Contraction



(b) Rotation

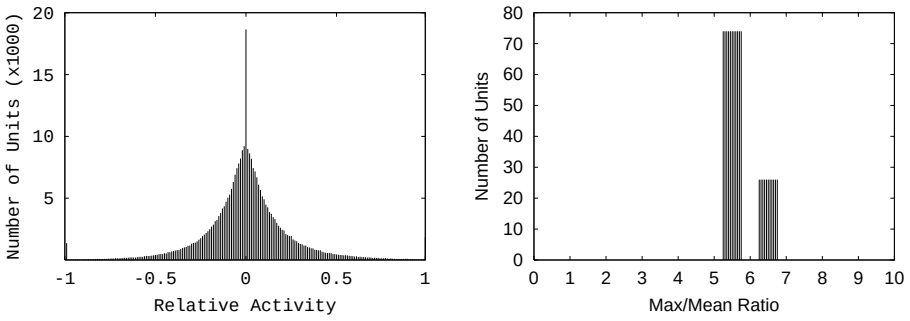


(c) Translation



(d) Combination

Figure 4. Examples of filters extracted by ICA



(a) Sparseness (b) Selectivity

Figure 5. Distribution of output amplitude for test patterns, which shows the sparseness of output activation.

training set. The response pattern of each filter for these test data are shown in figure 5. Figure 5(a) shows the number of output units (filters), which give the output activation of each range. Outputs of all units were linearly normalized to have the same maximum value over all input data and trivial input stimulus that contains little movement were discarded. This normalization was used to prevent redundant input which made all output units inactive. As shown in the figure, due to the learning constraints of ICA and the assumed distribution of underlying sources, the response patterns show sparseness of filters – only a small number of units respond for the given input pattern.

Figure 5(b) shows the selectivity of the filters. Since a filter responds only to preferred types of input and is inactive for all others patterns, the ratio between the maximum activity of an output unit and its mean over the testing set is quite large.

5. Discussion and Future W ork

In this study, independent component analysis was performed on the optical flow of complex motion and the resulting filters show characteristics that resemble those of MSTd cells in visual cortex.

The resulting filters were tuned to specific motion patterns, moderate in size, and localized in their positions. Filters selective to translations, rotations, contractions, and expansions have been observed.

However, some important properties of MSTd cells could not be observed explicitly in our present results. First, some studies on MSTd reported cells with some levels of position invariance. The linear filter model described in this paper could not show any invariance. By introducing interactions between these linear filters, it is possible that position invariance can be achieved to some extent [20, 5].

Secondly, more elaborate computational modeling of MT cells is required. Many units were selective to vertical or horizontal translational movements and some were selective to translation at arbitrary angles. This is probably due to the fact that, in the coordinate system we adopted, any movement could be represented as linear combination of vertical and horizontal movements. A more elaborate model of MT cells similar to that of Nowlan and Sejnowski [21] capable of representing motion components by population coding of many direction selective units, may be able to produce translation selective units at various angles. An alternative is to use a log-Polar coordinate system similar to that used by Grossberg and colleagues [22].

6. Acknowledgments

We wish to acknowledge valuable comments from the reviewers. This research is supported by grants from the Australian Research Council,the University of Sydney, and the Brain Science and Engineering Research Program, the Ministry of Science and Technology, Korea.

7. References

[1] H. B. Barlow. Possible principles underlying the transformation of sensory messages. In W.A. Rosenbluth, editor, *Sensory Communication*, pages 371–394. MIT Press, 1961.

[2] A. J. Bell and T. J. Sejnowski. An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[3] J. H. van Hateren and A. van der Schaaf. Independent component filters of natural images compared wiht simple cells in primary visual cortex. *Proceedings of Royal Society of London B*, 265:2315–2320, 1998.

[4] J. H. van Hateren and D. L. Ruderman. Independent component analysis of natural image sequences yield spatio-temporal filters similar to simple cells in primary visual cortex. *Proceedings of Royal Society of London B*, 265:2315–2320, 1998.

[5] R. S. Zemel and T. J. Sejnowski. A model for encoding multiple object motions and self-motion in area MST of primate visual cortex. *The Journal of Neuroscience*, 18(1):531–547, 1998.

[6] D. J. Field. What is the goal of the sensory coding? *Neural Computation*, 6:559–601, 1994.

[7] D. J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of Optical Society America A*, 4:2379–2394, 1987.

[8] R. Linsker. Local synaptic learning rules suffices to maximize mutual information in a linear network. *Neural Computation*, 4:691–702, 1992.

[9] T. Cover and J. Thomas. *Elements of Information Theory*. John Wiley and Sons, New York, 1991.

[10] J. P. Nadal and N. Parga. Non linear neurons in the low noise limit: a factorial code maximizes information transfer. *Network*, 5:565–581, 1994.

[11] Te-Won Lee. *Independent Component Analysis - Theory and Applications*. Kluwer Academic Publisher, Boston, 1998.

[12] S. Amari, A. Cichocki, and H. Yang. A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing System 8*, pages 757–763, 1996.

[13] B. A. Olshausen and D. J. Field. Natural image statistics and efficient coding. *Network: Computation in Neural Systems*, 7(2):333–339, 1996.

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

[15] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[16] S. S. Beauchemin and J. L. Barron. The computation of optical flow. *Neural Computing Survey*, 1995.

[17] H.-H. Nagel. On the estimation of optical flow: Relations between different approaches and some new results. *Artificial Intelligence*, 33:299–324, 1987.

[18] S. Makeig. ICA toolbox for psychophysiological research (version 3.4). WWW Site, Computational Neurobiology Laboratory, The Salk Institute for Biological Studies `<www.cnl.salk.edu/~ica.html>` [World Wide Web Publication], 1999.

[19] C. J. Duffy and R. H. Wurtz. Sensitivity of MST neurons to optic flow stimuli. I. A continuum of response selecitivity to large-field stimuli. *Journal of Neurophysiology*, 65(6):1329–1345, 1991.

[20] K. Zhang, M. I. Sereno, and M. E. Sereno. Emergence of position-independent detectors of sense of rotation and dilation with Hebbian learning: An analysis. *Neural Computation*, 5(4):597–612, 1993.

[21] S. J. Nowlan and T. J. Sejnowski. A selection model for motion processing in area MT of primates. *The Journal of Neuroscience*, 15(2):1195–1214, 1995.

[22] S. Grossberg, E. Mingolla, and C. Pack. A neural model of motion processing and visual navigation by cortical area MST. *Cerebral Cortex*, 9:878–895, 1999.