A Canonical Microfunction For Learning Perceptual Invariances

James V Stone*
Biological Sciences/Cognitive and

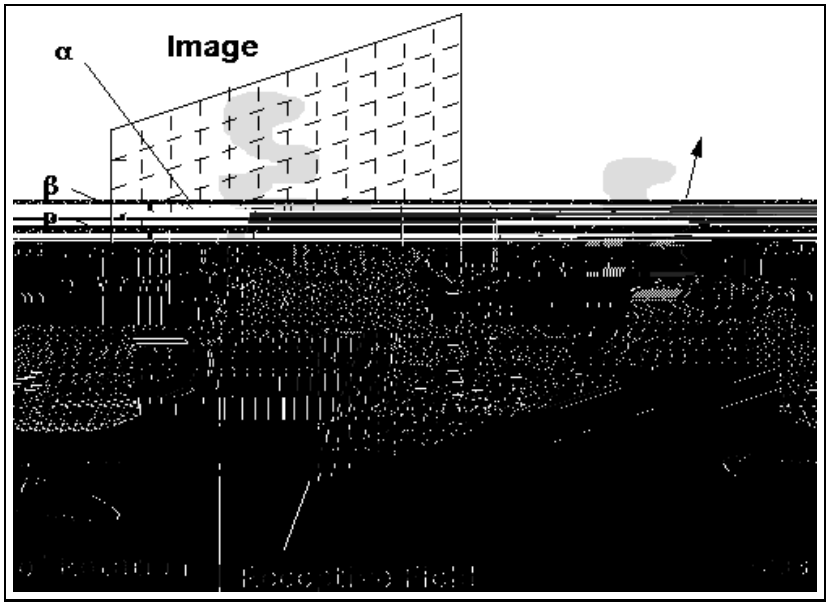


Figure 1: A small amount of object rotation can have a dramatic effect on the intensities (e.g. α, β and γ) of individual image pixels. The intensities of pixels in a 'receptive field' change as the object rotates. In this example, three pixel intensities define an image vector, (α, β, γ), as depicted in Figure 2a.

The approach adopted here is consistent with that of (Douglas et al., 1989) and others, but it is more general because it allows us to concentrate on computational aspects of perceptual learning in a modular neuronal model.

Learning Invariances Using Spatio-Temporal Constraints

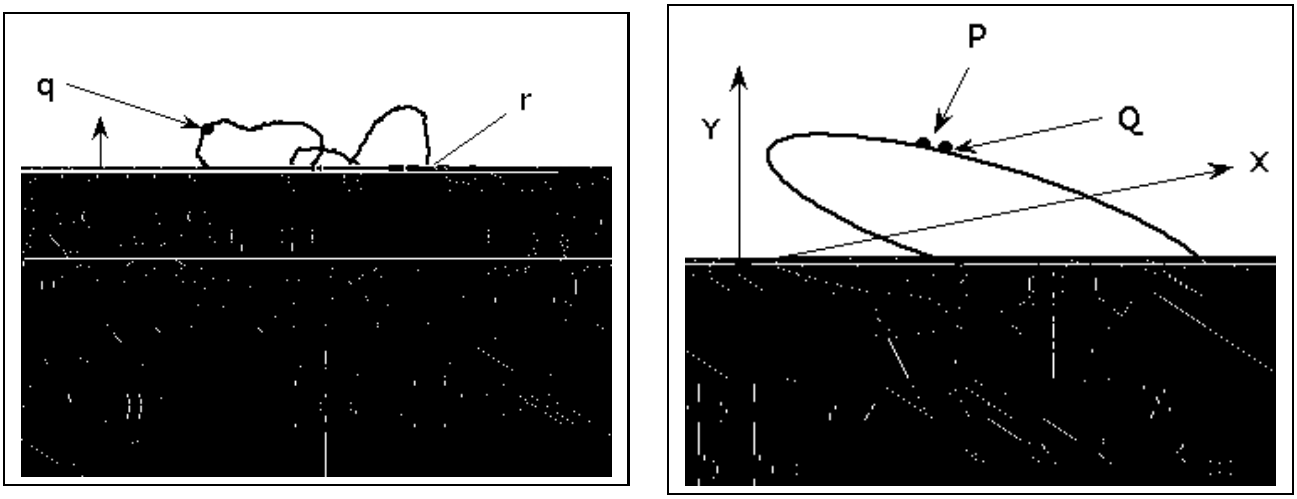


Figure 2: Diagram of (a) image vector and (b) parameter vector. Three physical parameters (rotation of an object around the X, Y and Z axes) defines a 3D parameter vector which changes over time (b). The corresponding changes in three image pixels (α, β and γ in Figure 1) are represented as the change in position of a 3D image vector over time (a).

image, and also to perceptually salient physical parameters other than rotation (e.g. depth, motion and curvature).

Figure 1 shows a rotating textured object and its image I. Clearly, a small change in the orientation of the object can give rise to a relatively large change in the intensity of individual pixels as the texture moves across the image plane. In general, a perceptually salient parameter such as rotation is characterised by variability over time, but the rate of change of the parameter is usually small, relative to that of the intensities of individual image pixels.

The intensities α, β and γ of three of the n pixels in I are plotted along perpendicular axes in Figure 2a. Any point, such as p, defines a value for the intensities of these pixels at one time. The intensities of pixels change rapidly but continuously as the object rotates, and the image vector[1] (α, β, γ) sweeps out a trajectory as shown in Figure 2a. Just as (α, β, γ) defines a 3D image vector, so, the n pixels in I define an nD image vector. A similar type of trajectory is swept out by this nD image vector, but we would need n axes to display it.

In Figure 2b, the three components of a *parameter vector* are shown. Each component of this vector specifies the rotation of an object around a 'world' axis (X, Y or Z). The closed loop of the parameter vector in Figure 2b implies complete rotation of the imaged object, so that the object begins and ends in the same orientation. This, in turn, implies that the initial and final images are identical. Therefore, the sequence of image vectors also defines a closed loop, as shown in Figure 2a.

For display purposes, the object in this example rotates at a constant rate, so that the temporal proximity of two parameter vectors is proportional to their proximity along the curve in Figure 2b (e.g. P and Q). Given a sequence of image vectors derived from a rotating object, can the corresponding sequence of parameter vectors be recovered?

It is tempting to assume that if two image vectors are near to each other then they were derived from a single object at similar orientations. Unfortunately, a rotating object can generate quite different image vectors, even though the amount of object rotation which separates these image vectors is small. Therefore, the similarity between successive

intervals are likely to be derived from different physical scenarios.

It is noteworthy that conventional unsupervised learning techniques (e.g. Kohonen maps (Kohonen, 1984), Hebbian learning (Oja, 1982)) which cluster input vectors according to their Euclidean distance would not, in general, be capable of clustering together images which were generated by similar physical scenarios. In contrast, the method presented here takes advantage of the temporal proximity of (often dissimilar) input vectors to discover which invariances they share.

So far, the general characteristics of how perceptually salient parameters change over time have been described. These observations can be used to constrain the outputs of a model microcircuit such that its outputs come to reflect these general characteristics. This can be achieved without specifying the desired output (target) value for any input to the microcircuit. An 'economical' way for a microcircuit to generate such a set of outputs is to adapt its connection weights so that the outputs specify some invariance which is implicit in the microcircuit's inputs.

The Learning Method

A model which uses a type of *temporal smoothness* constraint can be made to learn visual invariances. The degree of smoothness of the output or *state* of a model unit can be measured in terms of the 'temporally local', or *short term*, variance associated with a sequence of output values. A sequence of states defines a curve which is maximally smooth if the variance of this curve is minimal (the straighter the curve, the smoother the output). However, minimising only the short term variance has a trivial solution. This consists of setting all model weights to zero, generating a horizontal output curve. This is consistent with one characteristic, smoothness, of perceptually salient invariances, but it does not conform to the other characteristic, variability over time. The output can be made to reflect both smoothness *and* variability by forcing it to have a small short-term variance, and a large *long-term* variance. Thus the variance of the output over small intervals should be small, relative to its variance over longer intervals.

The general strategy just described can be implemented using a multi-layer model.

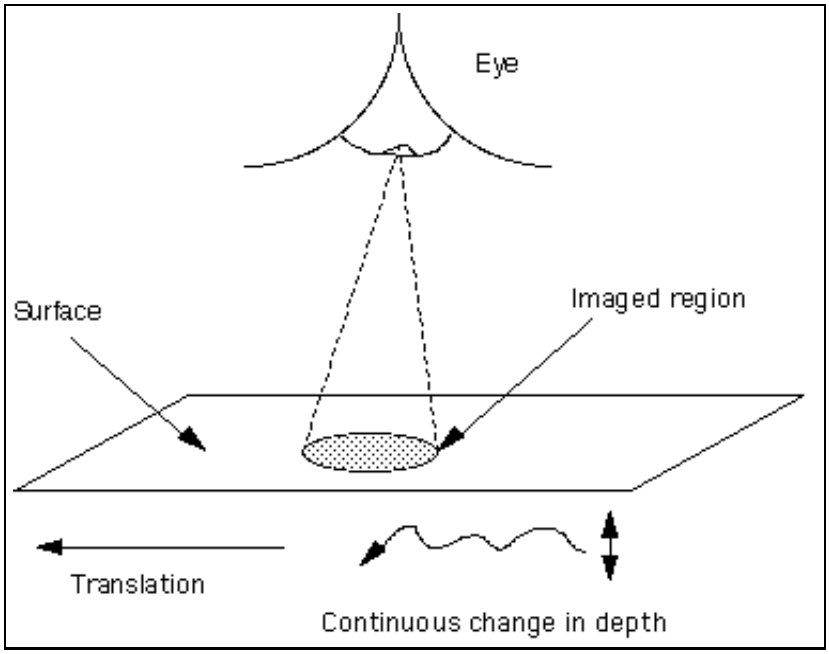
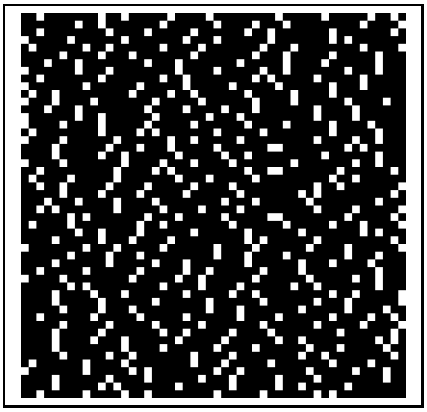
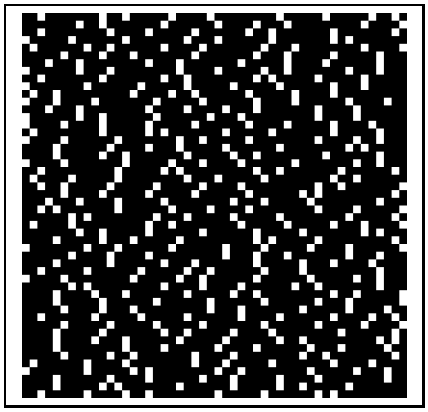


Figure 3: Variation of surface depth over time.

(For hidden unit weights, additional terms resulting from the $tanh$ hidden unit activation function are required (see Appendix)).





any input.

An analysis of the structure of the weight vectors of hidden units is beyond the scope of this paper. However, a more detailed analysis of the behaviour of this type of microcircuit is given in (Stone, 1995b, 1995a).

No Hidden Units: As expected for a system which attempts to discover an input/output mapping which is not linearly separable, learning without a hidden layer of units failed to compute disparity. With the output unit connected directly to the input layer, correlations of $|r| > 0.01$ were not exceeded over 10 different simulations with different initial random weights.

Generalisation: If the model has learned disparity (and not some spurious correlate of disparity) then it should generalise to new stereo data sets, *without any learning of these new sets*. Accordingly, the model was tested on two new data sets. For both sets, generalisation was tested using the microcircuit after 800 epochs.

First, a smoothed randomly varying disparity profile with disparities between ± 1 pixels was used to generate a sequence of 1D random dot stereograms. For this sequence, the correlation between microcircuit output and stereo disparity was $r = 0.921$.

Second, test data was derived from a smoothed (Gaussed) version of the stereo pair shown in Figure 5. As with the learning data, the standard deviation of the Gaussian was one dot width, the dot density was 0.167, and each Gaussed array was normalised to have zero mean and unit variance. The right array was identical to the left array, except for a square region which was shifted by a constant disparity d. For display purposes, d in Figure 5 is equal to one dot width, but $d = 0.5$ of a dot width for the microcircuit inputs. This sub-pixel disparity was achieved by using linear interpolation over grey-levels in the array. The stereo pair used to test the microcircuit was therefore identical to that shown in Figure 5, except that the former had a disparity of half a dot width. Each input pair was obtained by reading data from a Gaussed version of the left and right arrays of this stereo pair into the 5-unit upper and lower input rows, respectively, of the microcircuit.

The microcircuit input was scanned across the Gaussed version of the stereo pair, and an output value was obtained for each position. This simulates the action of a single microcircuit at every location in the visual field. Microcircuit outputs from the border were discarded[3], and the resultant 40×40 array of microcircuit outputs were plotted in Figure 5. The correlation between disparity and microcircuit outputs over the resultant 1600 1D input pairs was $r = 0.91$.

Discussion

Dis arity and Tem oral Smoothness

The model discovers a perceptually salient visual invariance by unsupervised learning from a sequence of images. The stereo disparity task learned is a hyper-acuity task. That is, the amount of disparity is smaller than the width of any single receptor (pixel). This is consistent with psychophysical studies which demonstrate th9.3(p)-10002999.4(a)-5te tl

requires only that discontinuities in parameter values are *rare*, relative to gradual changes over time. In the example presented in (Stone, 1995b), four discontinuities every 1000 time steps did not disrupt the learning process. Thus, the model requires not that all parameters change smoothly at all times, but, more realistically, that parameters change smoothly *most* of the time. To paraphrase Marr (quoted in the introduction to this paper), "disparity varies smoothly almost every*when*".

Canonical Microfunctions

In terms of the evolution of the neocortex, it makes sense to have a single type of canonical microcircuit which can be used to learn any perceptual invariance. The approach adopted here seeks to delineate functional characteristics of such a circuit in terms of learning perceptual invariances. This microfunctional approach is useful because it permits computational aspects of neocortical microcircuits to be considered, even though the detailed neuroanatomy of such circuits is not known. This represents a compromise between high-level functional models associated with artificial intelligence, and detailed biophysical models.

Note that the approach does *not* require the assumption that all neocortical microcircuits are the same. It requires an assumption that all microcircuits are pre-disposed to learning about particular *types* of properties in their inputs. The neocortex has evolved in a world which has changed little in terms of what constitutes perceptually salient physical entities. Therefore, an economical strategy would be to evolve a canonical microcircuit that uses a single strategy to learn about invariances which remain perceptually important over long periods of evolutionary time. This single strategy can be formalised in terms of a *canonical microfunction* (of which F may be an example).

Important aspects of the microfunctional framework are that it requires models which are implemented as artificial neuronal networks, that these networks learn without the aid of a teacher, and that they are (broadly speaking) functionally consistent with the capabilities of neuronal systems. The term "functionally consistent" is not intended to refer to the particular learning method used, but rather, to the unsupervised nature of the learning method, and to the final input/output mapping learned by the model.

It is possible to decouple a given putative canonical strategy (e.g. maximise temporal smoothness) from the microfunction used to implement that strategy; for every strategy, there are many microfunctions which can be used to implement it. Similarly, it is possible to decouple a given microfunction from the learning method used to maximise that function. Thus, for every learning algorithm which learns a given mapping[4], there exist many others which can learn this mapping in a manner more consistent with the known neurophysiology. An example of this was given by a pair of papers (Zipser & Anderson, 1988; Mazzoni, Anderson, & Jordan, 1991). The receptive field properties of parietal neurons were simulated using the "biologically implausible" backpropagation method (Zipser & Anderson, 1988). Later, similar results were obtained using a more biologically plausible learning method (Mazzoni et al., 1991).

The learning method described above is consistent with, but is not determined by, the 'temporal smoothness' strategy. It is intended that this microfunctional approach will yield other strategies which are more general in application than that described here. At the very least, it is intended that this approach will yield a series of microfunctions which embody the 'temporal smoothness' assumption in a manner which is increasingly consistent with the known function *and* structure of neocortical microcircuits.

Conclusion

Conventional low-level computer vision techniques rely upon the assumption that a parameter value is invariant over some region of *space*(see (Stone, 1992)). The model described in this paper assumes that perceptually salient parameters vary slowly over *time*. When presented with a sequence of images, the model discovered precisely those parameters which describe the behaviour of the imaged surface through time.

Temporal smoothness is a fundamental property of the perceptual world. Given a sequence of inputs, any learning system that did not take advantage of the temporal smoothness of perceptual invariances implicit in that sequence would be discarding a powerful and general heuristic for discovering perceptually salient properties of the physical world.

Acknowledgements: Thanks to Nikki Hunkin and Julian Budd for comments on this paper, and to Alistair Bray for useful discussions. Thanks also to Raymond Lister, David Willshaw and Tom Collett for discussions on the learning method presented here. This research was supported by a Joint Council Initiative grant awarded to Jim Stone, Tom Collett and David Willshaw.

Appendix: The Learning Algorithm

The learning algorithm relies upon batch update of a weight vector $\mathbf{w}$, which contains all weights in the network. At each time step t, a stereo pair is presented at the input layer, and the derivative of F with respect to every weight in the network is computed and added to a cumulative weight derivative vector $\nabla F_{\mathbf{w}}$. This derivative vector is used to update $\mathbf{w}$ only after all $T = 1000$ stereo pairs have been presented at the input units. The same set of stereo pairs is repeatedly presented in the same order during learning. Storage requirements are minimal because all quantities required for learning can be computed incrementally.

Notation: Units in the input, hidden and output layers are indexed by subscripts i, j and k, respectively. For example, a weight which connects hidden unit u_j to output unit u_k is denoted w_{jk}. The state z_{kt} of u_k at time t is:

$$z_{kt} = \sum_j w_{jk}\, z_{jt} \tag{5}$$

Where z_{jt} is the state of u_j. Input and output layer units have linear activation functions, whereas hidden units have non-linear ($tanh$) activation functions. For such a unit u_j, its output z_j is the hyperbolic tangent of its input:

$$z_{jt} = tanh\left(\sum_i w_{ij}\, z_{it}\right) \tag{6}$$

Where w_{ij} is a weight connecting input unit u_i to hidden unit u_j, and z_{it} is the state of u_i at time t. Weights connecting input to hidden units, and hidden to output units, are referred to as *lower* and *upper* weights, respectively.

The function to be maximised is $F = log\ V/U$, where:

$$\begin{aligned} U &= 1/2 \sum_{t=1}^{T} (\tilde{z}_{kt} - z_{kt})^2 \\ V &= 1/2 \sum_{t=1}^{T} (\overline{z}_{kt} - z_{kt})^2 \end{aligned}$$

V is the long-term variance of z_k, U is the short-term variance of z_k, and T is the period over which they are defined. Both V and U are defined in terms of exponentially weighted means of z_k. The weighted means $\tilde{z}_k$ and $\overline{z}_k$ differ only in terms of their respective exponential rates of decay:

$$\tilde{z}_{kt} = \lambda_S\, \tilde{z}_{k(t-1)} + (1 - \lambda_S)\, z_{k(t-1)} \quad : 0 \leq \lambda_S \leq 1 \tag{7}$$

$$\overline{z}_{kt} = \lambda_L\, \overline{z}_{k(t-1)} + (1 - \lambda_L)\, z_{k(t-1)} \quad : 0 \leq \lambda_L \leq 1 \tag{8}$$

Where λ_S and λ_L have half-lives of h_S and h_L, respectively, with $h_L \gg h_S$. The formula obtaining a value of λ for

$$\frac{\partial U(t)}{\partial w} = \frac{\partial U(t-1)}{\partial w} + (\tilde{z}_{kt} - z_{kt}) \left(\frac{\partial \tilde{z}_{kt}}{\partial w} - \frac{\partial z_{kt}}{\partial w} \right) \tag{11}$$

Where, from (7):

$$\frac{\partial \tilde{z}_{kt}}{\partial w} = \lambda_S \frac{\partial \tilde{z}_{kt-1}}{\partial w} + (1 - \lambda_S) \frac{\partial z_{kt-1}}{\partial w} \tag{12}$$

Thus, the incremental computation of (11) depends upon evaluation of $\partial z_{kt}/\partial w$ in (11) and $\partial z_{k(t$

Reference

Barlow, H. (1985). Cerebral cortex as a model builder. In Rose, D., & Dobson, V. (Eds.), *Models of the Visual Cortex*, pp. 37–46. John Wiley, New York.

Becker, S. (1992). Learning to categorize objects using temporal coherence. *Neural Information Processing Systems*, 361–368.

Becker, S., & Hinton, G. (1992). Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, *335*, 161–163.

Bienstock, E., Cooper, L., & Munro, P. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *Journal of Neuroscience*, *2*, 32–48.

Creutzfeldt, O. D. (1978). The neocortical link: Thoughts on the generality of structureand function of the neocortex. In Brazier, M., & Petsche, H. (Eds.), *Architectonics of the Cerebral Cortex*. Raven Press, New York.

Douglas, J., Martin, K., & Nelson, J. (1993). The neurobiology of primate vision. *Bailliere's Clinical neurology*, *2*(2), 191–225.

Douglas, R., Martin, K., & Whitteridge, D. (1989). A canonical microcircuit for neocortex. *Neural Computation*, *1*, 480–488.

Ebdon, M. (1993). Is the cerebral neocortex a uniform cognitive architecture?. *Mind and Language*, *8*(3), 369–403.

Stone, J. V. (1995a). Hierarchical learning of visual invariances via spatio-temporal constraints. In *Int. Conference on Neural Networks, Cambridge*, pp. 110–115.

Stone, J. V. (1995b). Learning perceptually salient visual parameters through spatio-temporal smoothness constraints. *Neural Computation, (Accepted).*

Stone, J. V., & Bray, A. (1995). A learning rule for extracting spatio-temporal invariances. *Network, 6*(3), 1–8.

Szenátgoathai, J. (1978). The neuron network of the cerebral cortex: a functional approach. *Proc Royal Society London (B), 201*, 219–248.

Westheimer, G. (1994). The ferrier lecture, 1992. seeing depth with two eyes: Stereopsis. *Proc Royal Soc London, B, 257*, 205–214.

Williams, P. (1991). A Marquardt algorithm for choosing the step-size in backpropagation learning with conjugate gradients. Cognitive science research paper CSRP 229, University of Sussex.

Zemel, R., & Hinton, G. (1991). Discovering viewpoint invariant relationships that characterize objects. *Technical Report, Dept. of Computer Science, University of Toronto, Toronto, ONT MS5 1A4.*

Zipser, D., & Anderson, A. (1988). A back-propagation network that simulates response properties of a subset of posterior parietal neurons. *Nature, 331*, 679–684.