INSTITUTE OF PHYSICS PUBLISHING NETWORK: COMPUTATION IN NEURAL SYSTEMS

Network: Comput. Neural Syst. **14** (2003) 233–247 PII: S0954-898X(03)58143-X

Nonlinear ideal observation and recurrent preprocessing in perceptual learning

L Zhaoping[1], Michael H Herzog[2] and Peter Dayan[3]

[1] Department of Psychology, University College, London WC1E 6BT, UK
[2] Human Neurobiology, Argonnenstrasse 3, 28211 Bremen, Germany
[3] Gatsby Computational Neuroscience Unit, University College, London WC1N 3AR, UK

Received 24 April 2002, in final form 21 November 2002
Published 7 February 2003
Online at stacks.iop.org/Network/14/233

Abstract

Residual micro-saccades, tremor and fixation errors imply that, on different trials in visual tasks, stimulus arrays are inevitably presented at different positions on the retina. Positional variation is likely to be specially important for tasks involving visual hyperacuity, because of the severe demands that these tasks impose on spatial resolution. In this paper, we show that small positional variations lead to a structural change in the nature of the ideal observer's solution to a hyperacuity-like visual discrimination task such that the optimal discriminator depends quadratically rather than linearly on noisy neural activities. Motivated by recurrent models of early visual processing, we show how a recurrent preprocessor of the noisy activities can produce outputs which, when passed through a linear discriminator, lead to better discrimination even when the positional variations are much larger than the threshold acuity of the task. Since, psychophysically, hyperacuity typically improves greatly over the course of perceptual learning, we discuss our model in the light of results on the speed and nature of learning.

1. Introduction

At first blush, our ability to solve high precision visual discrimination tasks seems quite amazing. However, computational studies have shown that such optimal discrimination can be achieved in a straightforward manner, using a *linear* feedforward network acting on the noisy outputs of neural population codes (e.g. Snippe and Koenderink 1992, Seung and Sompolinsky 1993, see Dayan and Abbot 2001 for a review). These linear algorithms make a crucial assumption, that the position of the stimulus array on the retina is known or fixed. This assumption is hard to justify in the light of small fixation errors, eye tremor and also micro-saccades, which happen continually and involuntarily, even when the subjects are instructed to fixate (Alpern 1972). Furthermore, these fixation errors are often much larger than the hyperacuity spatial discriminations that subjects can achieve in the tasks concerned.

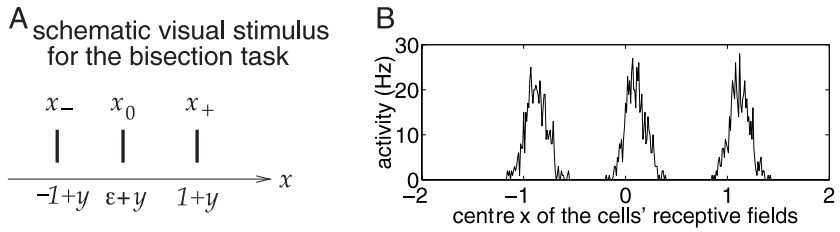

Figure 1. The bisection task. (A) Three bars are presented at x_-, x_0 and x_+. The task is to report to which of the outer bars the central bar is closer. y represents the variable placement of the stimulus array dependent, for example on eye tremor or micro-saccades. (B) Population activities in cortical cells evoked by the stimulus bars—the activities (firing rates a_i) are plotted against the preferred locations x_i of the cells. Here, the spatial tuning curves are Gaussian, with peak firing rate 20 Hz and a width of $\tau = 0.1$. The activities are corrupted by Poisson noise. There are 81 units whose preferred values are placed at regular intervals of $\Delta x = 0.05$ between $x = -2$ and 2.

We therefore set out to consider discrimination in the light of positional variation. Using a bisection task as an example (although the results are more general), we first consider in section 3 the effect of this variation on the structure and quality of the optimal estimator. We show analytically that optimal discrimination under small positional variations is actually quadratic rather than linear in the activities of the neural population that codes the input stimuli. We show the good performance of the quadratic discriminator for moderate positional variations, by comparison with a correctly trained linear discriminator. In appendix A, we consider the impact of prior information on the discrimination. In section 4, we further consider how the good performance of this estimator can be approached by a non-linear network model of recurrent interactions, possibly located in V1, acting as a preprocessor of the input population activities. This network actually generalizes to a substantially greater range of positional variance than the quadratic discriminator.

In psychophysical practice, good performance in problems like the bisection task is only achieved after a period of perceptual learning. We thus end by discussing the recurrent network in the light of psychological results on the time course and nature of perceptual learning.

2. The bisection task

Figure 1(A) shows one form of the bisection task. Three bars are presented at horizontal positions $x_0 = \epsilon + y$, $x_- = -1 + y$ and $x_+ = 1 + y$, where $-1 \ll \epsilon \ll 1$. The subject has to report whether the central bar is closer to the left outer bar ($\epsilon < 0$) or the right outer bar ($\epsilon > 0$). Here, we have included the possibility that the whole stimulus array might be presented at different points on the retina, through the variable y. This is a *nuisance* parameter in that, since all the bars are affected by it, it has no impact on the answer to the bisection task. However, inference algorithms have to take its effects into account, lest they be led astray.

We consider the case where the bars create a population-coded representation in those cells possibly located in primary visual cortex that are tuned for vertical bars (for simplicity, we restrict ourselves to this class, and also consider only a horizontal slice of the visual input). In figure 1(B), we show the activity of, or neural response from, cell i (called a_i) as a function of the position x_i of the centre of its receptive field. Note three activity bumps in the plot, each evoked by one of the bars.

We assume an underlying additive model for cortical responses. That is, the mean response of cell i is

$$\bar{a}_i(\epsilon, y) = f(x_i - x_0) + f(x_i - x_-) + f(x_i - x_+). \tag{1}$$

We often drop the dependence on ϵ, y and write $\bar{a}_i$, or, for all the components, $\bar{\boldsymbol{a}}$. Here, f is, for concreteness, a Gaussian tuning curve with height k and tuning width τ

$$f(x) = k\mathrm{e}^{-x^2/2\tau^2}. \tag{2}$$

Usually, we have $\tau \ll 1$. Further, we make the standard assumption that the cells are sufficiently dense that the total mean activity $\sum_i \bar{a}_i(\epsilon, y)$ is a constant, independent of ϵ and y.

The actual activity is corrupted by noise

$$a_i = \bar{a}_i + n_i \tag{3}$$

where we assume that the noise terms n_i are independent across units, and such that a_i comes from a Poisson distribution of mean $\bar{a}_i$. Further, we assume that y and ϵ have mean zero. In some cases, we will consider posterior distributions for ϵ and/or y given the recorded activities. We will do this using (improper) prior distributions for these quantities, assuming that we have no prior knowledge other than that the means are zero.

The task is to report whether ϵ is greater or less than zero on the basis of the activities $\boldsymbol{a} = \{a_i\}$. One result of this paper is that the optimal solution to this task, for infinitesimal ϵ, turns out to have a different character in the case where y is known (we will take it to be zero) versus the case where y is unknown.

2.1. Fixed position stimulus array

When the stimulus array is in a fixed position $y = 0$, analysis is very similar to that carried out by Seung and Sompolinsky (1993). We proceed by calculating the probability density $P[\epsilon|\boldsymbol{a}]$ of the shift ϵ of the central bar given the activities $\boldsymbol{a}$, and reporting by maximum likelihood (ML) that $\epsilon > 0$ if $\int_{\epsilon>0} \mathrm{d}\epsilon\, P[\epsilon|\boldsymbol{a}] > 0.5$. By Bayes' rule,

$$P[\epsilon|\boldsymbol{a}] = \frac{P[\boldsymbol{a}|\epsilon]P[\epsilon]}{P(\boldsymbol{a})} \tag{4}$$

where $P[\epsilon]$ is the prior probability of ϵ, $P(\boldsymbol{a})$ is the prior probability of neural activities $\boldsymbol{a}$ and $P[\boldsymbol{a}|\epsilon]$ is the conditional probability of neural responses $\boldsymbol{a}$ given ϵ. Without prior information about ϵ (we treat the case of prior information in appendix A), we have

$$P[\epsilon|\boldsymbol{a}] \propto P[\boldsymbol{a}|\epsilon] \tag{5}$$

since the prior $P(\boldsymbol{a})$ does not depend on ϵ. Let us approximate $P[\boldsymbol{a}|\epsilon]$ by Taylor expanding it about $\epsilon = 0$ to second order in ϵ,

$$\log P[\boldsymbol{a}|\epsilon] \sim \text{constant} + \epsilon \frac{\partial}{\partial \epsilon} \log P[\boldsymbol{a}|\epsilon]\bigg|_{\epsilon=0} + \frac{\epsilon^2}{2} \frac{\partial^2}{\partial \epsilon^2} \log P[\boldsymbol{a}|\epsilon]\bigg|_{\epsilon=0}. \tag{6}$$

Provided that the last term is negative (which it indeed is, almost surely), we derive an approximately Gaussian distribution

$$P[\epsilon|\boldsymbol{a}] \propto \exp[-(\epsilon - \bar{\epsilon})^2/(2\sigma_\epsilon^2)] \tag{7}$$

with variance $\sigma_\epsilon^2 \equiv \left[-\frac{\partial^2}{\partial \epsilon^2} \log P[\boldsymbol{a}|\epsilon]\right]^{-1}$ and mean $\bar{\epsilon} \equiv \sigma_\epsilon^2 \frac{\partial}{\partial \epsilon} \log P[\boldsymbol{a}|\epsilon]$. Thus the subject should report that $\epsilon > 0$ or $\epsilon < 0$ if the *test*

$$t_F(\boldsymbol{a}) = \frac{\partial}{\partial \epsilon} \log P[\boldsymbol{a}|\epsilon] \tag{8}$$

which is proportional to the posterior mean of ϵ, is greater or less than zero respectively. Assuming Poisson noise $n_i = a_i - \bar{a}_i$, such that the probability of neural activity a_i given

a mean (expectation) response $\bar{a}_i$ is $P(a_i) = \mathrm{e}^{-\bar{a}_i}(\bar{a}_i)^{a_i}$, and assuming that noise in different neurons is independent, we have

$$P[\boldsymbol{a}|\epsilon] \propto \prod_i P[a_i|\epsilon] = \prod_i \mathrm{e}^{-\bar{a}_i(\epsilon)}(\bar{a}_i(\epsilon))^{a_i}. \tag{9}$$

Hence, $\log P[\boldsymbol{a}|\epsilon] = \text{constant} + \sum_i a_i \log \bar{a}_i(\epsilon)$ since $\sum_i \bar{a}_i(\epsilon)$ is a constant, independent of ϵ. Thus,

$$t_F(\boldsymbol{a}) = \sum_i a_i \frac{\partial}{\partial \epsilon} \log \bar{a}_i(\epsilon). \tag{10}$$

Therefore, ML discrimination can be implemented by a linear feedforward network mapping inputs a_i through feedforward weights $w_i = \frac{\partial}{\partial \epsilon} \log \bar{a}_i$ to calculate the output $t_F(\boldsymbol{a}) = \sum_i w_i a_i$. The task therefore has an essentially *linear character*. Note that if the noise corrupting the activities is Gaussian, the weights should instead be $w_i = \frac{\partial}{\partial \epsilon}\bar{a}_i$.

The quality of discrimination in this case is determined by the Fisher information

$$\mathcal{I}_\epsilon = \left\langle -\frac{\partial^2}{\partial \epsilon^2} \log P[\boldsymbol{a}|\epsilon] \right\rangle_{\boldsymbol{a}} = \langle 1/\sigma_\epsilon^2 \rangle_{\boldsymbol{a}}, \tag{11}$$

averaging over the random activities. Thus, for instance, the usual psychophysical discriminability measure is

$$d'_\epsilon = 2\epsilon\sqrt{\mathcal{I}_\epsilon} \tag{12}$$

(the factor of two coming from testing $-\epsilon$ against $+\epsilon$), and the probability of error is

$$p_\epsilon = \frac{1}{2}\,\mathrm{erfc}\left(\frac{d'_\epsilon}{2\sqrt{2}}\right) \tag{13}$$

where erfc is the standard complementary error function.

Alternatively, instead of going through the route of finding the posterior distribution for ϵ to the optimal test, it is known that the likelihood ratio test for testing ϵ against the value zero is the score function, namely $\frac{\mathrm{d}}{\mathrm{d}\epsilon} \log P[\boldsymbol{a}|\epsilon]$, which is exactly the test $t_F(\boldsymbol{a})$ (see Dayan and Abbott 2001).

Figure 2(A) shows the optimal discrimination weights as a function of the preferred positions of the cells. They are essentially *linear* in x_i (i.e., $w_i \sim x_i$) for the central neural activity bump when $-0.5 < x_i < 0.5$, and are 0 otherwise. This is because

$$\begin{aligned} w_i = \frac{\partial}{\partial \epsilon} \log \bar{a}_i(\epsilon) = \frac{1}{a_i}\frac{\partial a_i}{\partial \epsilon} &= \frac{-f'(x_i)}{f(x_i) + f(x_i+1) + f(x_i-1)} \\ &\propto \frac{x_i f(x_i)}{f(x_i) + f(x_i+1) + f(x_i-1)} \\ &\approx \begin{cases} x_i & \text{if } |x_i| < |x_i - x_-| \text{ and } |x_i| < |x_i - x_+| \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

If the noise were Gaussian with a fixed variance rather than Poisson, then $w_i = \frac{\partial a_i}{\partial \epsilon} \propto x_i f(x_i)$. This is also near zero outside the central activity bump, but lacks the linear structure of the case of Poisson noise. The nature of the neural noise, whether Poisson or Gaussian, does not affect our general conclusions throughout this paper (see appendix A).

The lower solid curve in figure 2(C) shows performance as a function of ϵ. The error rate drops precipitately from 50% for very small (and thus difficult) ϵ to almost zero, long before ϵ approaches the tuning width τ. The performance predicted from the Fisher information is shown as the circles, which lie almost exactly on the curve. Note that the Taylor expansion

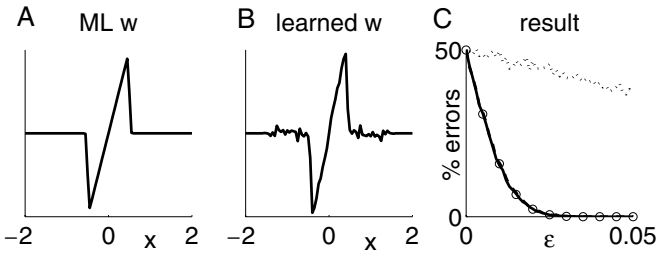

Figure 2. (A) The ML optimal discrimination weights $\boldsymbol{w} = \frac{\partial}{\partial \epsilon} \log \bar{\boldsymbol{a}}$ (plotted as w_i versus x_i) for deciding whether $\epsilon > 0$ when $y = 0$. (B) The learned discrimination weights $\boldsymbol{w}$ for the same decision. During on-line learning, random examples were selected with $\epsilon \in [-2\tau, 2\tau]$ uniformly, $\tau = 0.1$, and the weights were adjusted online to maximize the log probability of generating the correct discrimination under a model in which the probability of declaring that $\epsilon > 0$ is $1/(1 + \exp(-\sum_i w_i a_i))$. (C) Performance of the networks with ML (lower solid curve) and learned (lower dashed curve; almost indistinguishable) weights as a function of ϵ. Performance is measured by drawing $\boldsymbol{a}$ randomly given ϵ and y, and assessing the percentage of trials the answer is incorrect. The circles show points from the theoretical probability of error based on equation (13). The upper dotted curve shows the effect of drawing $y \in [-0.2, 0.2]$ uniformly, yet using the ML weights in (B) that assume $y = 0$.

(and equivalently the Fisher information) depends on the assumption that ϵ is small. That the inference is as good as expected shows that this assumption holds.

It is also possible to learn weights in a variety of ways (e.g. Poggio *et al* 1992, Weiss *et al* 1993, Fahle *et al* 1995). Figure 2(B) shows discrimination weights learned using a version of the perceptron learning rule (Rosenblatt 1958). These are almost the same as the optimal weights and lead to performance that is essentially optimal (the lower dashed curve in figure 2(C)).

3. Moveable stimulus array

If the stimulus array can move around, i.e., if y is not necessarily zero, then the discrimination task gets considerably harder. We consider the case where the array is at a different position from trial to trial. Tremor will induce shifts at an even finer timescale. The upper dotted curve in figure 2(C) shows the (rather unfair) test of using the learned weights in figure 2(B) when $y \in [-0.2, 0.2]$ varies uniformly in a range much larger than the discrimination threshold for ϵ. Clearly this has a highly detrimental effect on the quality of discrimination. This is obvious from the weight structure in figures 2(A) and (B)—the central bump provides information only about $\epsilon + y$, which is not enough to solve the problem. The outer neural activity bumps provide information about the value of y independent of the value of ϵ, but the optimal and learned weight patterns are such that no notice is taken of these bumps.

Since y is now unknown, the uncertainty associated with it should make discrimination worse. One conceptual way to think about discrimination in this case is indicated in figure 3. This shows how one might apply the tests that would pertain for each individual values of y, say $t_y(\boldsymbol{a})$, and then integrate the results according to the posterior probability of the values of y. We can derive a simple approximation to the result of this that shows what happens in the true case. Looking at figure 2(A), we can approximate test

$$t_y(\boldsymbol{a}) = \sum_i a_i w_i(y) \sim \sum_i^0 a_i(x_i - y)$$

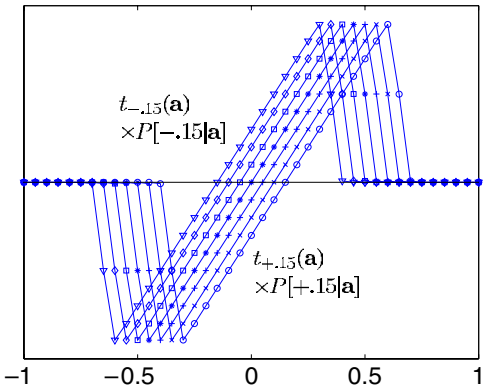

Figure 3. Sliding template. The test allowing for unknown y can be thought of as a weighted sum of the tests for known y (the forms from figure 2(A) shifted according to y), where the weights are the posterior probabilities $P[y|\boldsymbol{a}]$. This is reminiscent of a neocognitron-like scheme (Fukushima 1980, see also the MAX model of Riesenhuber and Poggio 1999) with a linear stage (the template) and a non-linear integration stage (here involving multiplication).

(This figure is in colour only in the electronic version)

where $w_i(y)$ is the weight that is learned or derived at a fixed value y, and $\sum^0$ (and similarly $\sum^-$ and $\sum^+$) indicates that the sum is only over the central (outer) activity bump. Provided that the tests are equally weighted (i.e. the scale of their answers does not favour one value of y over another), this makes the final test

$$\int \mathrm{d}y\, P[y|\boldsymbol{a}] \sum_i^0 a_i(x_i - y) = \sum_i^0 a_i x_i - \sum_i^0 a_i E[y|\boldsymbol{a}].$$

Now, if we make the further approximation of using only the outer two bumps to work out the value of y, ignoring the information coming from the middle bump, then, remembering that $x_+ - 1$ and $x_- + 1$ are both estimates of y,

$$E[y|\boldsymbol{a}] = \frac{\sum_i^+ a_i(x_i - 1) + \sum_i^- a_i(x_i + 1)}{\sum_i^+ a_i + \sum_i^- a_i}$$

making the final test

$$\sum_i^0 a_i x_i - \left(\sum_i^0 a_i\right) \frac{\sum_i^+ a_i(x_i - 1) + \sum_i^- a_i(x_i + 1)}{\sum_i^+ a_i + \sum_i^- a_i}$$

which has the same sign as, and so is equivalent to

$$\tilde{t}'(\boldsymbol{a}) = \left(\sum_i^0 a_i x_i\right)\left(\sum_i^+ a_i + \sum_i^- a_i\right) - \left(\sum_i^+ a_i(x_i - 1) + \sum_i^- a_i(x_i + 1)\right)\left(\sum^0 a_i\right) \qquad (14)$$

which is a quadratic function of $\boldsymbol{a}$ and takes a form that we will recognize below.

The more formal derivation of the test comes from expanding the log probability in the Taylor series as in equation (6), but without assuming *a priori* that $y = 0$ (again, a fuller treatment of prior information is presented in appendix A). This gives, with all the derivatives taken at $\epsilon, y = 0$,

$$\log P[\epsilon, y|\boldsymbol{a}] \sim \text{constant} + \left(\epsilon\frac{\partial}{\partial\epsilon} + y\frac{\partial}{\partial y} + \frac{\epsilon^2}{2}\frac{\partial^2}{\partial\epsilon^2} + \frac{y^2}{2}\frac{\partial^2}{\partial y^2} + \epsilon y\frac{\partial^2}{\partial y\partial\epsilon}\right)\log P[\boldsymbol{a}|\epsilon, y].$$

Thus, to second order, a Gaussian distribution can still approximate the joint distribution $P[\epsilon, y|\boldsymbol{a}]$. Figure 4(A) shows the high quality of this approximation when y is not too large. Here, as we mentioned, ϵ and y are anti-correlated given activities $\boldsymbol{a}$, because the information from the centre stimulus bar only constrains their sum $\epsilon + y$. Of interest is the probability $P[\epsilon|\boldsymbol{a}] = \int \mathrm{d}y\, P[\epsilon, y|\boldsymbol{a}]$, which, using a quadratic approximation to its logarithm, is approximately Gaussian with mean $\beta\rho_\epsilon^2$ and variance ρ_ϵ^2, where

$$\beta = \frac{\partial}{\partial\epsilon}\log P[\boldsymbol{a}|\epsilon, y] + \left(\frac{\partial}{\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right) \times \left(\frac{\partial^2}{\partial\epsilon\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right)\Big/\left(-\frac{\partial^2}{\partial y^2}\log P[\boldsymbol{a}|\epsilon, y]\right) \tag{15}$$

$$\rho_\epsilon^{-2} = -\frac{\partial^2}{\partial\epsilon^2}\log P[\boldsymbol{a}|\epsilon, y] - \left(\frac{\partial^2}{\partial\epsilon\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right)^2\Big/\left(-\frac{\partial^2}{\partial y^2}\log P[\boldsymbol{a}|\epsilon, y]\right). \tag{16}$$

The inverse variance of the Gaussian distribution of y that we integrated out is $-\frac{\partial^2}{\partial y^2}\log P[\boldsymbol{a}|\epsilon, y]$, and is almost surely positive. This makes the appropriate test for the sign of ϵ

$$t_M(\boldsymbol{a}) = \left(\frac{\partial}{\partial\epsilon}\log P[\boldsymbol{a}|\epsilon, y]\right)\left(-\frac{\partial^2}{\partial y^2}\log P[\boldsymbol{a}|\epsilon, y]\right) - \left(\frac{\partial}{\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right)\left(-\frac{\partial^2}{\partial\epsilon\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right). \tag{17}$$

In the case of Poisson noise, this reduces to

$$t_M(\boldsymbol{a}) = \left[\left(\boldsymbol{a}\cdot\frac{\partial\log\bar{\boldsymbol{a}}}{\partial\epsilon}\right)\left(-\boldsymbol{a}\cdot\frac{\partial^2\log\bar{\boldsymbol{a}}}{\partial y^2}\right) - \left(\boldsymbol{a}\cdot\frac{\partial\log\bar{\boldsymbol{a}}}{\partial y}\right)\left(-\boldsymbol{a}\cdot\frac{\partial^2\log\bar{\boldsymbol{a}}}{\partial y\partial\epsilon}\right)\right] \tag{18}$$

which is the full version of the approximate form in equation (14), matched term by term. If $t_M(\boldsymbol{a}) > 0$ then we should report $\epsilon > 0$, and conversely. In fact, $t_M(\boldsymbol{a})$ is a very simple quadratic form

$$t_M(\boldsymbol{a}) = \boldsymbol{a}\cdot\boldsymbol{Q}\cdot\boldsymbol{a} \equiv \sum_{ij} a_i a_j\left[\left(\frac{\partial^2\log\bar{a}_i}{\partial y\partial\epsilon}\right)\left(\frac{\partial\log\bar{a}_j}{\partial y}\right) - \left(\frac{\partial\log\bar{a}_i}{\partial\epsilon}\right)\left(\frac{\partial^2\log\bar{a}_j}{\partial y^2}\right)\right]. \tag{19}$$

Therefore, the discrimination problem in the face of positional variance has a quantifiable *non-linear* character. The quadratic test $t_M(\boldsymbol{a})$ cannot be implemented by a linear feedforward architecture only, since the optimal boundary $t_M(\boldsymbol{a}) = 0$ to separate the state space $\boldsymbol{a}$ for a decision is now curved. Writing $t_M(\boldsymbol{a}) = \boldsymbol{a}\cdot\boldsymbol{Q}\cdot\boldsymbol{a}$ where the symmetric form $Q_{ij} = (Q'_{ij} + Q'_{ji})/2$, we find $\boldsymbol{Q}$ only has four non-zero eigenvalues, for the four-dimensional sub-space spanned by four vectors (where the derivatives are taken at the point $\epsilon, y = 0$) $\frac{\partial^2\log\bar{\boldsymbol{a}}}{\partial y\partial\epsilon}$, $\frac{\partial\log\bar{\boldsymbol{a}}}{\partial y}$, $\frac{\partial\log\bar{\boldsymbol{a}}}{\partial\epsilon}$ and $\frac{\partial^2\log\bar{\boldsymbol{a}}}{\partial y^2}$. The form of $\boldsymbol{Q}$ is shown in figure 4(B). Note that if Gaussian rather than Poisson noise is used for $n_i = a_i - \bar{a}_i$, the test $t_M(\boldsymbol{a})$ will still be quadratic.

In appendix A, we consider the effect of imposing a Gaussian prior distribution for y, with variance $\tilde{\sigma}_y^2$. In this case, the correct test turns out to be

$$t_B(\boldsymbol{a}) = t_F(\boldsymbol{a})/\tilde{\sigma}_y^2 + t_M(\boldsymbol{a}) \tag{20}$$

which weights the linear test for fixed eye position ($t_F(\boldsymbol{a})$) according to the prior certainty that y is zero. For eye movements of a size that is the same order of magnitude as that of a receptive field, the poor discrimination shown by the dotted line of figure 2(C) indicates that the linear test is significantly outweighed by the quadratic test.

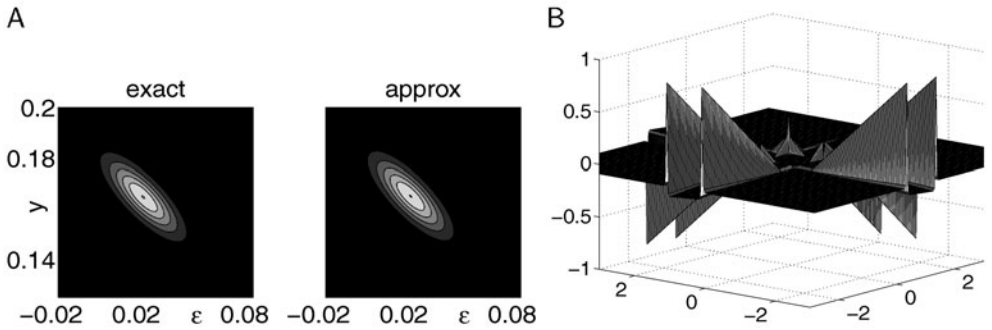

Figure 4. Varying y. (A) Posterior distribution $P[\epsilon, y|\boldsymbol{a}]$. Exact (left) $P[\epsilon, y|\boldsymbol{a}]$ for a particular $\boldsymbol{a}$ with true values $\epsilon = 0.2\tau$, $y = 1.5\tau$ (with $\tau = 0.1$) and its bivariate Gaussian approximation (right). Only the relevant region of (ϵ, y) space is shown—outside this, the probability mass is essentially zero (and the contour values are the same). (B) The quadratic form $\boldsymbol{Q}$, Q_{ij} versus x_i and x_j. Note that $\boldsymbol{Q}$ links activity values a_i evoked by one stimulus bar to those of a_j evoked by another. The absolute scale of $\boldsymbol{Q}$ is arbitrary.

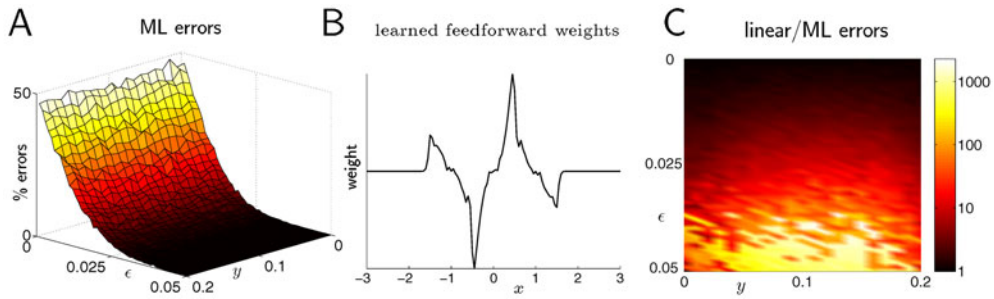

Figure 5. $y \neq 0$. (A) Performance of the approximate inference based on the quadratic form of figure 4(B) in terms of percentage error as a function of $|y|$ and $|\epsilon|$ ($\tau = 0.1$). (B) Feedforward weights, w_i versus x_i, learned using the same procedure as in figure 2(B), but with $y \in [-0.2, 0.2]$ chosen uniformly at random. (C) Ratio of error rates for the linear (weights from (B)) to the quadratic discrimination.

Of course, $t_M(\boldsymbol{a})$, like $t_F(\boldsymbol{a})$, is based on an expansion, assuming that y and ϵ are small. Figure 5(A) shows that using $t_M(\boldsymbol{a})$ to infer ϵ is sound for y up to two standard deviations (τ) of the tuning curve $f(x)$ away from zero. By comparison, a feedforward network, of weights shown in figure 5(B) which were learned using the same error-correcting learning procedure as above, only based on values of y drawn uniformly at random from $[-0.2, 0.2]$, performs substantially worse. Nevertheless, this trained linear network is substantially better than the feedforward net of figures 2(A) and (B), for which training was done using fixed $y = 0$. Figure 5(C) shows the *ratio* of the error rates for the linear to the quadratic decisions. The linear network is often dramatically worse, because it fails to take proper account of y. The pattern of weights in figure 5(B) can be understood as comprising a similar, nearly linear, central component (with a contribution $\sum_i^0 w_i a_i$ to the test), coupled with flanks for the outer bumps that have a net *positive* effect (i.e., $\sum_i^{\pm} w_i a_i > 0$) on the test if $y < 0$ and a net *negative* effect if $y > 0$. This flanking effect is an attempt to compensate for the way that the central bump also shifts with y.

4. Recurrent preprocessing

In the previous section, we suggested using a simple quadratic non-linearity to perform ML discrimination, at least over a small range of ϵ and y. Such a nonlinearity could be implemented by one of a number of biophysical mechanisms (Koch 1999). However, there are reasons to believe (Crist *et al* 2001, Gilbert *et al* 2001) that recurrent network processing, based on horizontal long range connections in the primary visual cortex (Rockland and Lund 1983, Gilbert and Wiesel 1983), can play a significant role.

There has been substantial interest in the statistical processing power of recurrent non-linear networks, acting as pre-processors for noisy population activity. They can give rise to regularized activities which are suitable inputs for further feedforward stages of processing (Pouget *et al* 1998, Deneve *et al* 1999, 2001). The idea has been applied most forcefully to recurrent processing within hypercolumns to arrive at a noise insensitive coding of the receptive field features such as orientation of the input. In the bisection task, however, this way of looking at recurrent processing cannot be the whole story, since the main task for regularization here is actually eliminating the effect of y, rather than cleaning up the noise in the activities. We know, from the previous sections, that if $y = 0$, then the optimal test, even in the face of noise, is still linear. Thus, merely cleaning up the bumps, whilst leaving them where they are, would not make linear feedforward inference that much more accurate. Instead, computations linking neural activities in separate hypercolumns are needed, as is apparent in our test $t_M(\boldsymbol{a})$, for which activities a_i and a_j from different (separate) bars, one central and one outer, have to be associated (multiplied) with each other.

Figure 6 demonstrates the plausibility of this idea. Input activity (as in figure 1(B)) is used to initialize the state $\boldsymbol{u}$ at time $t = 0$ of a recurrent network. The recurrent weights are symmetric, as shown in figure 6(B). The network firing rates evolve according to

$$\mathrm{d}u_i/\mathrm{d}t = -u_i + \sum_j J_{ij} g(u_j) + a_i \tag{21}$$

where J_{ij} is the recurrent weight from unit j to i, and function g is a threshold nonlinearity ($g(u) = u$ if $u > 0$ and $g(u) = 0$ for $u \leqslant 0$). The network evolves according to $u_i \to u_i + 0.5\,\mathrm{d}u_i/\mathrm{d}t$ for 150 steps, and the resulting near-equilibrium activities are fed into the linear, feedforward, decision-making stage.

Starting from noisy input $\boldsymbol{a}$, the activities of the neurons settle to an equilibrium $\boldsymbol{u}(t \to \infty)$ (figure 6(C), note that $u_i(t \to \infty) = a_i$ when $J = 0$). The activity values $\boldsymbol{u}(t \to \infty)$ at this equilibrium are fed through feedforward weights $\boldsymbol{w}$, that are trained for this recurrent network just as they were for the pure feedforward case, to reach a decision $\sum_i w_i u_i(t \to \infty)$. The recurrent weight matrix (figure 6(B)) emerges from three influences:

(1) a short range interaction J_{ij} for $|x_i - x_j| \lesssim \tau$ to stabilize activities a_i induced by a single bar in the input;
(2) a longer range interaction J_{ij} for $|x_i - x_j| \sim 1$ to mediate interaction between neighbouring stimulus bars, amplifying the effects of the displacement signal ϵ, and
(3) a slight local interaction J_{ij} for $|x_i|, |x_j| \lesssim \tau$.

The first two interaction components are translation invariant in the spatial range of $x_i, x_j \in [-2, 2]$ where the stimulus array appears, in order to accommodate the positional variance in y. The last component is not translation invariant and counters variations in y. The quantitative details of the recurrent weight matrix are given in the appendix.

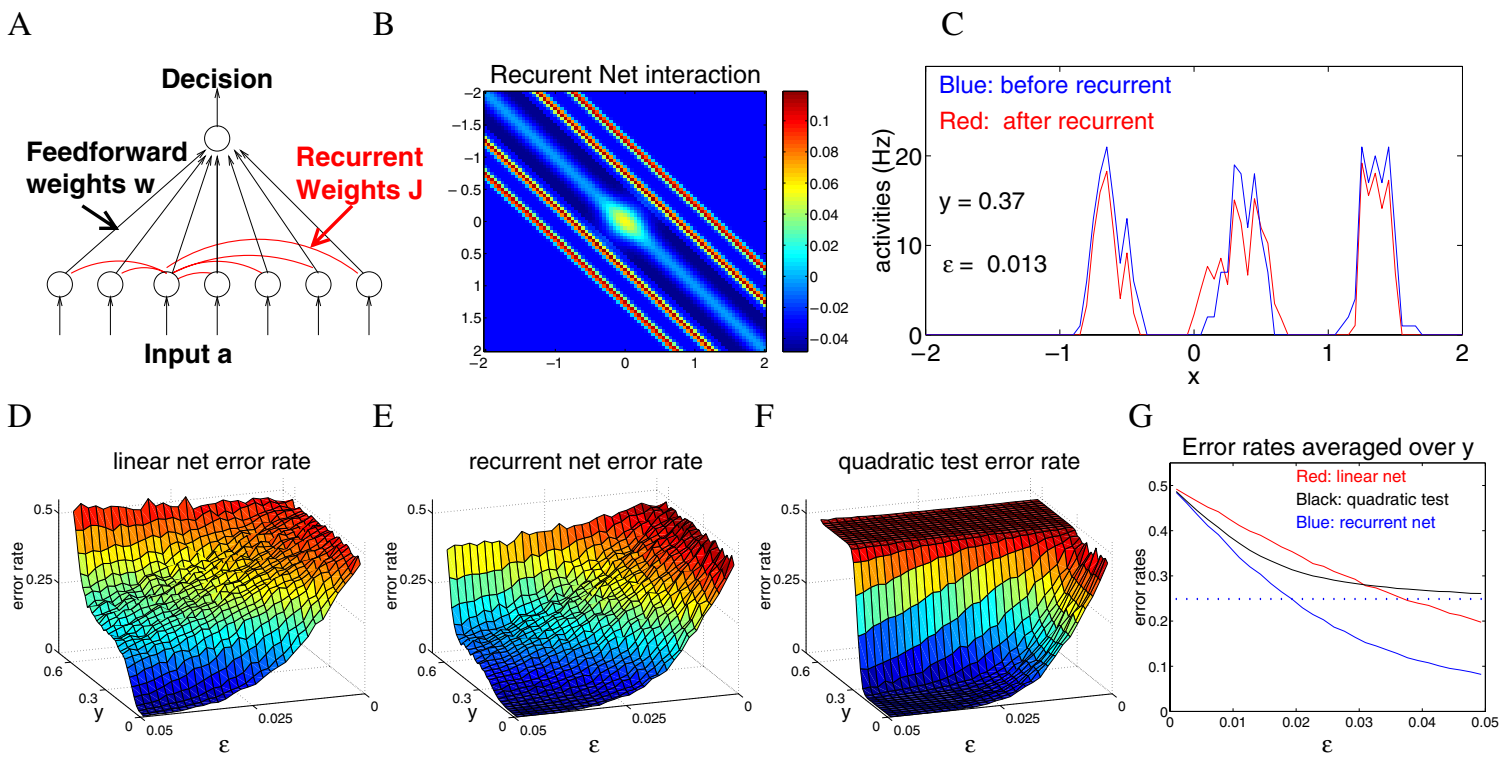

Figure 6. (A) The recurrent network (with weights J) is treated as a non-linear preprocessor of the population activity. (B) The weight matrix, whose quantitative details are presented in the appendix. (C) The effect of the recurrent processing on the neural activities for an example when $y = 0.37$ and $\epsilon = -0.013$. The recurrent interaction shifts the central bump of the neural activities to the left, towards $x = 0$, without shifting the left-hand bump left, and at the same time shifts the centre of the right-hand bump right. This corrects for $y \neq 0$ and amplifies the effect of $\epsilon \neq 0$. With increasing y, the discrimination performance deteriorates significantly for the linear net, shown in (D), but deteriorates much more slowly with y for the nonlinear net, shown in (E). Feedforward weights in both nets are trained with trials of $\epsilon \in (-0.05, 0.05)$ and $y \in (-0.6, 0.6)$. (F) The quadratic test performs very well for small y but fails disastrously for larger y. (G) The observed error rates in a hypothetical experiment in which the experimenter does not explicitly observe y, but only measures error rates as a function of ϵ. Note that the threshold acuity in the recurrent net, measured as the value of ϵ when error rate $= 25\%$, is half that of the linear net.

Figure 6(E) shows that this network gives results that are less susceptible to variations of y than those of the linear network (figure 6(D)). Meanwhile, the quadratic test (figure 6(F)) is the most invariant with respect to y for small y among the three alternatives, but fails catastrophically for larger $y > 2\tau$, when the Taylor approximation is no longer valid. In a typical psychophysical experiment, the variation in y is not measured, and the experimenter simply measures the error rate in the performance for a given ϵ. This experimental error rate is the error rate averaged over y. Figure 6(G) shows that the nonlinear net has by far the best performance, having a threshold acuity of about half that of the linear net.

5. Discussion

The problem of position invariant discrimination is common to many high precision visual tasks, including hyperacuity tasks such as the standard line vernier, three-dot vernier, curvature vernier and orientation vernier tasks (e.g. Westheimer and McKee 1977, McKee and Westheimer 1978, Watt *et al* 1983, Levi *et al* 1985, Fahle *et al* 1995, Fahle 1997).

Formally, the essential problem that we have discussed is discriminating a stimulus variable ϵ that depends only on the relative positions between the stimulus features, while the absolute position y of the whole stimulus array can vary between trials by an amount that is much larger than the discrimination threshold (or acuity) on ϵ. The positional variable y may not have to correspond to the absolute position of the stimulus array, but merely to the error in the estimation of the absolute position of the stimulus by other neural areas. When $y = 0$ is fixed, the discrimination is easy and soluble by a linear, feedforward network, whose weights can be learnt in a straightforward manner. However, our study shows that when y is not fixed, but is known to be small, optimal discrimination of ϵ is based on an quadratic function of the input activities, and cannot be implemented using a linear feedforward net. Note that since the essential parts of our mathematical derivation do not depend on the specificities of the bisection task, our conclusions hold for general hyperacuity tasks for which positional variation is larger than performance acuity.

We also showed that a non-linear recurrent network can perform much better than a pure feedforward network on the bisection task in the face of position variance. In fact, it performs almost as well as the quadratic form within the range of y where the latter works well, and continues to perform well for substantially greater positional variation than the quadratic form is able to handle. The precise recurrent interactions in our network are very specific to the task and its parameters. In particular, the range of the interactions is completely determined by the scale of spacing between stimulus bars; and the distance-dependent excitation and inhibition in the recurrent weights is determined by the nature of the bisection task. This may be why there is little transfer of learning between tasks, when the nature and the spatial scale of the task change, even if the same input units are involved (e.g. Fahle *et al* 1995, Fahle 1994, 1997). The model predicts that negative transfer of performance will occur for small changes of the distance between the outer bars since the weight matrix depends heavily on this distance, and the strength of the weight varies quickly from being positive to negative as the distance between the linked neurons varies (figure 6(B)).

To clarify the respective relevances of our analytical results and our recurrent network model, and the relationships between them, we note the following. A perfect ideal observer should solve the perceptual task by carrying out the ML decision on ϵ, a task specific stimulus variable, given neural responses $\boldsymbol{a}$, based on the conditional probability $P[\epsilon|\boldsymbol{a}]$ of ϵ given $\boldsymbol{a}$, for the bisection task or any generic task of such nature. Our analytical results show that, in the presence of significant eye movements, this ML decision has to be a nonlinear function of the neural responses $\boldsymbol{a}$ (even before the final simple thresholding). The quadratic

decision rule was only an approximation of the ML decision rule for sufficiently small eye movements and ϵ. These analytical results do not address how the non-linear decision rules, whether general or approximate (quadratic), should be implemented. Meanwhile, our recurrent network model uses an example to show that it is feasible to achieve better performance in a nonlinear network than a best possible linear network (which by definition performs a best possible linear computation for the task). Our recurrent network is not an implementation of the quadratic decision rule, but could be viewed as a first and rough attempt to implement the general ML decision rule.

It is important to ask which connections might comprise this recurrent network. The specificity of learning to such things as the eye of origin and spatial location strongly suggests that lower visual areas such as V1 are directly involved in learning. Indeed, it is becoming clear that V1 is a visual processor of quite some computational power (performing tasks such as segmentation, contour integration, pop-out and noise removal) rather than being just a feedforward, linear, processing stage (e.g. Deneve *et al* 1999, 2001, Li 1999, 2002). Further, Crist *et al* (2001), Gilbert *et al* (2001) have collected specific evidence that recurrent processing in V1 changes over the course of perceptual learning in cases such as the bisection task. The recurrent connections are designed to counter the variabilities of the eye positions and are not dedicated to processing higher order aspects such as attention or error feedback computation as used in hierarchical network architectures (e.g. Ahissar and Hochstein 1993, 1997, Herzog and Fahle 1998). The recurrent connections in our model do not serve either to better encode the stimulus; they merely serve to modify the neural responses in a task specific manner that is unlikely to improve the coding or representation of the stimulus for other tasks. The learning of the feedforward weights is responsible for the learning of decision making for the task, even though the recurrent weights help to improve the quality of the decisions by modifying the neural responses to the stimulus and thus the resulting feedforward weights.

One interesting facet of perceptual learning is its time-course. A core of psychophysical results is that there are two stages of learning, one fast, which happens over the first few trials, and another slow, which happens over multiple sessions and can last for months or even years (Fahle 1994, Karni and Sagi 1993, Fahle *et al* 1995). Part of the fast component may come from factors like accommodation and other non-specific aspects of learning. We suggest another part comes from learning the feedforward weights. Although we have yet to construct an appropriate learning rule, we suggest that learning the weights to implement the nonlinear transform corresponds to the slow component. This learning rule is delicate, because of the nonlinearity of the transform and the readily destabilized recurrent dynamics. Further, the feedforward weights need to be adjusted further as the recurrent weights change the activities on which they operate.

Acknowledgments

Funding is from the Gatsby Charitable Foundation (LZ, PD) and the German Research Council (DFG) Sonderforschungsbereich 517 'Neurocognition' (MHH). We are very grateful to Shimon Edelman and Maneesh Sahani for discussions. This paper is partly based on Li and Dayan (2001).

Appendix A. Derivation of $P(\epsilon|a)$ and the tests

Let $\bar{\boldsymbol{a}}(\epsilon, y)$ be the mean neural activities induced by offset ϵ and eye position y, and the actual neural activities $\boldsymbol{a} = \bar{\boldsymbol{a}} + \boldsymbol{n}$ include noise $\boldsymbol{n}$. Bayes' theorem (the extension of equation (4))

implies

$$P[\epsilon, y|\boldsymbol{a}] = \frac{P[\boldsymbol{a}|\epsilon, y]P[\epsilon]P[y]}{P(\boldsymbol{a})} \propto P[\boldsymbol{a}|\epsilon, y]P[\epsilon]P[y] \tag{22}$$

since $P(\boldsymbol{a})$ does not depend on ϵ. Here $P[\epsilon]$ and $P[y]$ are the prior probabilities of ϵ and y. For explicit illustration, we take these to be Gaussians $P[\epsilon] \propto \mathrm{e}^{-\epsilon^2/(2\tilde{\sigma}_\epsilon^2)}$ and $P[y] \propto \mathrm{e}^{-y^2/(2\tilde{\sigma}_y^2)}$ centred at zero and with variances $\tilde{\sigma}_\epsilon^2$ and $\tilde{\sigma}_y^2$ respectively. Thus we have

$$\log P[\epsilon, y|\boldsymbol{a}] = \text{constant} - \epsilon^2/(2\tilde{\sigma}_\epsilon^2) - y^2/(2\tilde{\sigma}_y^2) + \log P[\boldsymbol{a}|\epsilon, y]. \tag{23}$$

Taylor expanding $\log P[\boldsymbol{a}|\epsilon, y]$ to second order about $\epsilon = 0$; $y = 0$:

$$\log P[\boldsymbol{a}|\epsilon, y] \sim \text{constant} + \left(\epsilon\frac{\partial}{\partial\epsilon} + y\frac{\partial}{\partial y} + \frac{\epsilon^2}{2}\frac{\partial^2}{\partial\epsilon^2} + \frac{y^2}{2}\frac{\partial^2}{\partial y^2} + \epsilon y\frac{\partial^2}{\partial y\partial\epsilon}\right)\log P[\boldsymbol{a}|\epsilon, y]|_{\epsilon,y=0}. \tag{24}$$

This leads to a Gaussian approximation for $P[\epsilon, y|\boldsymbol{a}]$ whose logarithm is approximated as a second order polynomial in ϵ and y (from here on we omit $|_{\epsilon,y=0}$ to avoid clutter in the expressions):

$$\begin{aligned}\log P[\epsilon, y|\boldsymbol{a}] \approx \text{constant} &- \frac{\epsilon^2}{2\tilde{\sigma}_\epsilon^2} - \frac{y^2}{2\tilde{\sigma}_y^2} \\ &+ \left(\epsilon\frac{\partial}{\partial\epsilon} + y\frac{\partial}{\partial y} + \frac{\epsilon^2}{2}\frac{\partial^2}{\partial\epsilon^2} + \frac{y^2}{2}\frac{\partial^2}{\partial y^2} + \epsilon y\frac{\partial^2}{\partial y\partial\epsilon}\right)\log P[\boldsymbol{a}|\epsilon, y].\end{aligned} \tag{25}$$

The distribution of ϵ, $P[\epsilon|\boldsymbol{a}]$, comes from marginalizing out y, $P[\epsilon|\boldsymbol{a}] = \int \mathrm{d}y\, P[\epsilon, y|\boldsymbol{a}]$. This involves an integration $\int \mathrm{d}y\, \exp(-\frac{y^2}{2}\mathcal{A} + y\mathcal{B}) \propto \exp(\frac{\mathcal{B}^2}{2\mathcal{A}})$ where

$$\mathcal{A} = \frac{1}{\tilde{\sigma}_y^2} - \frac{\partial^2}{\partial y^2}\log P[\boldsymbol{a}|\epsilon, y] \tag{26}$$

$$\mathcal{B} = \left(\epsilon\frac{\partial^2}{\partial y\partial\epsilon} + \frac{\partial}{\partial y}\right)\log P[\boldsymbol{a}|\epsilon, y]. \tag{27}$$

Hence,

$$P[\epsilon|\boldsymbol{a}] \propto \exp\left(-\frac{\epsilon^2}{2\rho_\epsilon^2} + \beta\epsilon\right) \propto \exp\left(-\frac{(\epsilon - \beta\rho_\epsilon^2)^2}{2\rho_\epsilon^2}\right) \tag{28}$$

is approximately a Gaussian with a variance ρ_ϵ^2 and mean $\beta\rho_\epsilon^2$, where

$$\begin{aligned}\beta = \frac{\partial}{\partial\epsilon}\log P[\boldsymbol{a}|\epsilon, y] + &\left(\frac{\partial}{\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right) \\ &\times \left(\frac{\partial^2}{\partial\epsilon\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right)\Big/\left(\frac{1}{\tilde{\sigma}_y^2} - \frac{\partial^2}{\partial y^2}\log P[\boldsymbol{a}|\epsilon, y]\right)\end{aligned} \tag{29}$$

$$\rho_\epsilon^{-2} = \frac{1}{\tilde{\sigma}_\epsilon^2} - \frac{\partial^2}{\partial\epsilon^2}\log P[\boldsymbol{a}|\epsilon, y] - \left(\frac{\partial^2}{\partial\epsilon\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right)^2\Big/\left(\frac{1}{\tilde{\sigma}_y^2} - \frac{\partial^2}{\partial y^2}\log P[\boldsymbol{a}|\epsilon, y]\right). \tag{30}$$

Consequently, the test is simply $t_B(\boldsymbol{a}) \propto \beta$ or

$$\begin{aligned}t_B(\boldsymbol{a}) = &\left(\frac{\partial}{\partial\epsilon}\log P[\boldsymbol{a}|\epsilon, y]\right)\left(\frac{1}{\tilde{\sigma}_y^2} - \frac{\partial^2}{\partial y^2}\log P[\boldsymbol{a}|\epsilon, y]\right) \\ &+ \left(\frac{\partial}{\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right)\left(\frac{\partial^2}{\partial\epsilon\partial y}\log P[\boldsymbol{a}|\epsilon, y]\right)\end{aligned} \tag{31}$$

$$= t_F(\boldsymbol{a})/\tilde{\sigma}_y^2 + t_M(\boldsymbol{a}) \tag{32}$$

using the forms of the individual tests in equations (8) and (17).

For Poisson noise, $P[\boldsymbol{a}|\epsilon, y] = P[\boldsymbol{a}|\bar{\boldsymbol{a}}] = \Pi_i \mathrm{e}^{-\bar{a}_i(\epsilon,y)}(\bar{a}_i(\epsilon, y))^{a_i}$. Then

$$\log P[\boldsymbol{a}|\epsilon, y] = \text{constant} - \sum_i \bar{a}_i(\epsilon, y) + \sum_i a_i \log \bar{a}_i(\epsilon, y) \approx \text{constant} + \sum_i a_i \log \bar{a}_i(\epsilon, y) \tag{33}$$

since $\sum_i \bar{a}_i(\epsilon, y)$ is roughly independent of ϵ and y, and

$$\left(\frac{\partial}{\partial\epsilon}, \frac{\partial}{\partial y}, \frac{\partial^2}{\partial\epsilon^2}, \frac{\partial^2}{\partial y^2}, \frac{\partial^2}{\partial y\partial\epsilon}\right) \log P[\boldsymbol{a}|\epsilon, y] = \boldsymbol{a} \cdot \left(\frac{\partial}{\partial\epsilon}, \frac{\partial}{\partial y}, \frac{\partial^2}{\partial\epsilon^2}, \frac{\partial^2}{\partial y^2}, \frac{\partial^2}{\partial y\partial\epsilon}\right)\bar{\boldsymbol{a}}(\epsilon, y) \tag{34}$$

in which case the decomposition in equation (32) involves the explicit linear and quadratic forms of the tests $t_F(\boldsymbol{a})$ and $t_M(\boldsymbol{a})$ in equations (10) and (19), respectively.

This decomposition acutely indicates the importance of the linear and quadratic terms. If the prior standard deviation of y is small, then the linear term dominates; as $\tilde{\sigma}_y$ gets bigger, so does the importance of the quadratic term. Usually, the extent of eye movements is much greater than the vernier positional discrimination threshold. Comparing with equation (11), we note that $\frac{\partial^2}{\partial y^2} \log P[\boldsymbol{a}|\epsilon, y]$ is roughly the Fisher information in a hypothetical task when one discriminates the stimulus position y from $y = 0$ when $\epsilon = 0$. Such discrimination performance should have a threshold no larger than that for discriminating ϵ given $y = 0$ shown in figure 2(C). This threshold is apparently much smaller than a single receptive field size ($\tau = 0.1$), and thus much smaller than $\tilde{\sigma}_y$. Hence, we conclude that $\frac{1}{\tilde{\sigma}_y^2} \ll \frac{\partial^2}{\partial y^2} \log P[\boldsymbol{a}|\epsilon, y]$, and hence, from equations (31) and (32), the quadratic term in the test should dominate.

If the neural noise $\boldsymbol{n}$ has a Gaussian rather than Poisson distribution, with fixed variance (and no correlations), then the same analysis leads to similar conclusions. The equivalent of test $t_M(\boldsymbol{a})$ involves both linear and quadratic terms, but again, taking proper account of the latter is critical in the face of movement of the stimulus array. Hence, our general conclusion regarding the nonlinear nature of the computation given positional variance does not depend on whether the neural noise is Poisson or Gaussian.

Appendix B. Recurrent weights

The recurrent weights J are shown in figure 6(B), and include translation invariant components associated with the average distance between the bumps (the four sub- and super-diagonal parallel sidelobes in the weight matrix), and translation invariant and non-invariant regularization terms.

Formally, define the distance functions

$$d_{ij} = |x_i - x_j| \qquad e_{ij} = d_{ij} - 0.75 \qquad f_{ij} = d_{ij} - 1.25$$

and the step function

$$H(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{otherwise.} \end{cases}$$

Then,

$$J_{ij} = -0.03 + 0.06\mathrm{e}^{-(x_i^2+x_j^2)} + H(5\tau - d_{ij})\left[0.06\mathrm{e}^{-d_{ij}^2/6\tau^2} - 0.035\mathrm{e}^{-d_{ij}^2/40\tau^2}\right] + 0.15 \times \left[\mathrm{e}^{-e_{ij}^2/2d^2} H(5\tau - e_{ij}) + \mathrm{e}^{-f_{ij}^2/2d^2} H(5\tau - f_{ij})\right]$$

where $d = 0.5\tau$.

References

Ahissar M and Hochstein S 1993 Attentional control of early perceptual learning *Proc. Natl Acad. Sci. USA* **90** 5718–22

Ahissar M and Hochstein S 1997 Task difficulty and the specificity of perceptual learning *Nature* **387** 401–5

Alpern M 1972 Eye movements *Handbook of Sensory Physiology* vol VII/4, ed D Jameson and L M Hurvich (Berlin: Springer)

Crist R E, Li W and Gilbert C D 2001 Learning to see: experience and attention in primary visual cortex *Nature Neurosci.* **4** 519–25

Dayan P and Abbott L F 2001 *Theoretical Neuroscience* (Cambridge, MA: MIT Press)

Deneve S, Latham P and Pouget A 1999 Reading population codes: a neural implementation of ideal observers *Nature Neurosci.* **2** 740–5

Deneve S, Latham P and Pouget A 2001 Efficient computation and cue integration with noisy population codes *Nature Neurosci.* **4** 826–31

Fahle M 1994 Human pattern recognition: parallel processing and perceptual learning *Perception* **23** 411–27

Fahle M 1997 Specificity of learning curvature, orientation, and vernier discriminations *Vis. Res.* **37** 1885–95

Fahle M, Edelman S and Poggio T 1995 Fast perceptual learning in hyperacuity *Vis. Res.* **35** 3003–13

Fukushima K 1980 Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position *Biol. Cybern.* **36** 193–202

Gilbert C D, Sigman M and Crist R E 2001 The neural basis of perceptual learning *Neuron* **31** 681–97

Gilbert C D and Wiesel T N 1983 Clustered intrinsic connections in cat visual cortex *J. Neurosci.* **3** 1116–33

Herzog M H and Fahle M 1998 Modelling perceptual learning: difficulties and how they can be overcome *Biol. Cybern.* **78** 107–17

Karni A and Sagi D 1993 The time course of learning a visual skill *Nature* **365** 250–2

Koch C 1999 *Biophysics of Computation* (Oxford: Oxford University Press)

Levi D M, Klein S A and Aitsebaomo A P 1985 Vernier acuity, crowding and cortical magnification *Vis. Res.* **25** 963–77

Li Z 1999 Visual segmentation by contextual influences via intracortical interactions in primary visual cortex *Network: Comput. Neural Syst.* **10** 187–212

Li Z 2002 A saliency map in primary visual cortex *Trends Cogn. Sci.* **6** 9–16

Li Z and Dayan P 2001 Position variance, recurrence and perceptual learning *NIPS 2000* ed T K Leen, T G Dietterich and V Tresp pp 31–7

McKee S P and Westheimer G 1978 Improvement in vernier acuity with practice *Percept. Psychophys.* **24** 258–62

Poggio T, Fahle M and Edelman S 1992 Fast perceptual learning in visual hyperacuity *Science* **256** 1018–21

Pouget A, Zhang K, Deneve S and Latham P E 1998 Statistically efficient estimation using population coding *Neural Comput.* **10** 373–401

Riesenhuber M and Poggio T 1999 Hierarchical models of object recognition in cortex *Nat. Neurosci.* **2** 1019–25

Rockland K S and Lund J S 1983 Intrinsic laminar lattice connections in primate visual cortex *J. Comp. Neurol.* **216** 303–18

Rosenblatt F 1958 The perceptron: a probabilistic model for information storage and organization in the brain *Psychol. Rev.* **65** 386–408

Seung H S and Sompolinsky H 1993 Simple models for reading neuronal population codes *Proc. Natl Acad. Sci. USA* **90** 10749–53

Snippe H P and Koenderink J J 1992 Discrimination thresholds for channel-coded systems *Biol. Cybern.* **66** 543–51

Watt R J, Morgan M J and Ward R M 1983 The use of different cues in vernier acuity *Vis. Res.* **23** 991–5

Weiss Y, Edelman S and Fahle M 1993 Models of perceptual learning in vernier hyperacuity *Neural Comput.* **5** 695–718

Westheimer G and McKee S 1977 Spatial configurations for visual hyperacuity *Vis. Res.* **17** 941–7