IEEE Transactions on Image Processing, 22(4):1631–43, 2013.

Image Segmentation Using a Sparse Coding Model of Cortical Area V1

M. W. Spratling
King's College London, Department of Informatics, London. UK.

Abstract

Algorithms that encode images using a sparse set of basis functions have previously been shown to explain aspects of the physiology of primary visual cortex (V1), and have been used for applications such as image compression, restoration, and classification. Here, a sparse coding algorithm, that has previously been used to account of the response properties of orientation tuned cells in primary visual cortex, is applied to the task of perceptually salient boundary detection. The proposed algorithm is currently limited to using only intensity information at a single scale. However, it is shown to out-perform the current state-of-the-art image segmentation method (Pb) when this method is also restricted to using the same information.

Keywords: Image Segmentation; Edge detection; Neural Networks; Predictive Coding; Sparse Coding; Primary Visual Cortex

1 Introduction

Discontinuities in image intensity (or in other image features such as colour or texture), often correspond to important changes in the properties of the world, such as changes in material or the boundaries of surfaces or of objects. Finding discontinuities that correspond to boundaries is thus the aim of a large class of image segmentation methods (Zhanga et al., 2008), which in turn underlie many algorithms for high-level vision, such as those used for object detection, tracking, recognition and classification. Given the fundamental importance of image segmentation for much of computer vision, boundary detection has been an active area of research for many decades. Despite these extensive efforts, current boundary detection methods fall short of human-level performance.

Many boundary detection methods employ linear filtering as an initial step to locate image discontinuities. Many different types of filter have been developed for this purpose. For example, the Sobel filter, the Roberts filter, the Prewitt filter, the Difference of Gaussian (DoG) filter, the Laplacian of Gaussian (LoG) filter (as used by the Marr-Hildreth algorithm (Marr and Hildreth, 1980)), the first derivative of Gaussian filter (as used by the Canny edge detector (Canny, 1986)), higher-order Gaussian derivative filters, and Gabor filters. Many of these filtering methods are analogous to, or have been directly inspired by, early stages of visual processing performed in the brain. For example, DoG or LoG filtering is similar to the processing performed by cells with circular-symmetric centre-surround receptive fields (RFs) in retina and lateral geniculate nucleus (LGN) (Derrington and Lennie, 1984; Marr and Hildreth, 1980; Rodieck, 1965), while derivative of Gaussian filters and Gabor filters have similar shapes to the RFs of orientation-tuned cells in primary visual cortex (V1) (Daugman, 1980; Jones and Palmer, 1987; Marcelja, 1980; Young and Lesperance, 2001).

When a filter is applied to an image, the output shows a response at every location where there is a partial match, or partial overlap, between the filter and an image feature that the filter "detects". This results in a dense representation, in which the same image feature is represented by multiple, redundant, filter responses. For example, consider using a Gabor filter bank to locate edges in a simple, artificial, image which contains a single non-zero pixel, as shown in the first column of the top-row of Fig. 1[1]. The maximum response across all filters at each pixel location is shown in the second column of Fig. 1. This is just the point-spread function of the filters. Active filters have a range of orientation preferences, centred at a range of different locations around the location of the non-zero pixel in the original image. The full range of responses can be visualised by using the filter outputs to reconstruct the edges they represent, *i.e.*, the strength of each filter response at each location is used to determine the strength with which a short oriented line segment is added to the reconstructed image. This is shown in the third column of Fig. 1. Similar results for a range of different, artificial, images and for a patch taken from a natural image are shown in rows 2 to 5 of Fig. 1. In each case linear filtering generates a large range of redundant responses.

Most boundary detection algorithms employ post-processing techniques to 1) remove the unwanted, redundant, responses generated by linear filtering, and 2) enhance weak, but wanted, responses. An example of a method from

[1]Results are produced here using quadrature-pairs of Gabor filters at eight, equally spaced, orientations, with the output of the quadrature-pair combined using the standard energy-model (Adelson and Bergen, 1985; Morrone and Burr, 1988). However, the same arguments apply to any shape of filter.

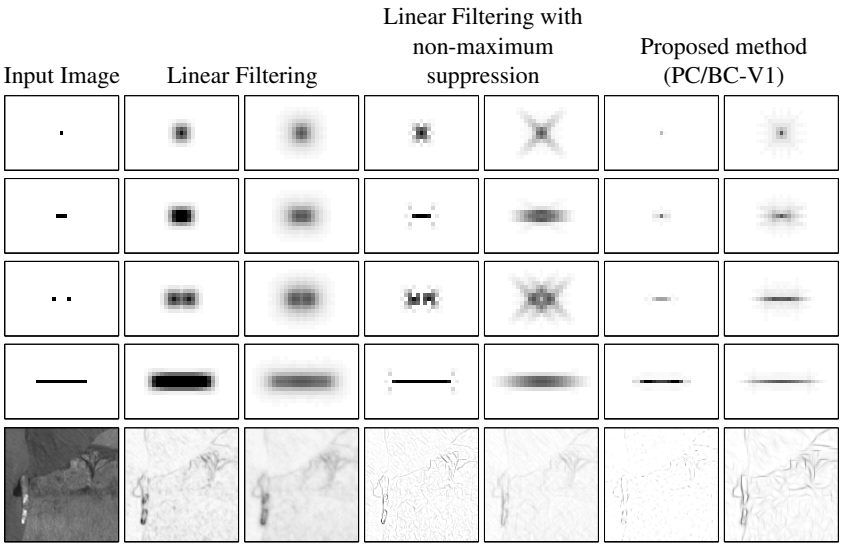

Figure 1: Comparison of linear filtering with the proposed method when applied to edge detection. Column 1 shows the original image, the first four rows use simple artificial images, while the last row shows a patch taken from a natural, greyscale, image (the full image appears in Fig. 5). Column 2 shows the maximum output at each pixel location of a Gabor filter bank. Column 3 shows the edge reconstruction generated from all the Gabor filter responses. Column 4 is that same as column 2 but with non-maximum suppression applied to the filter outputs. Column 5 shows the edge reconstruction generated by taking only the maximum Gabor filter responses at each pixel. Column 6 is the maximum output at each pixel location of the proposed method. Column 7 shows the edge reconstruction generated from all responses of the proposed method.

the first category is non-maximum suppression (as used in the Canny edge detector (Canny, 1986)). This technique thins edges by removing responses at locations perpendicular to the dominant edge orientation that are less than the local maximum response. The effects of non-maximum suppression are shown in the fourth column of Fig. 1. Steerable filters provide an efficient method of calculating the response of a filter at an arbitary orientation and phase (Freeman and Adelson, 1991). This is often used to find the single orientation with the maximum response at each image location, and hence, to implicitly ignore weaker responses at all other orientations at that location. The effect on the reconstructed image of using only the orientation with the strongest response at each location is shown in the fifth column of Fig. 1. An example of a post-processing method in the second category is hysteresis thresholding or edge linking (as used in the Canny edge detector (Canny, 1986)). This technique allows a filter that is weakly activated by the image content, but which is neighboured by a strongly activated filter, to be labelled as an edge. Other methods include edge relaxation (Geman and Geman, 1984; Hancock and Kittler, 1990; Parent and Zucker, 1989; Rosenfeld et al., 1976) and the "curvilinear continuity" extension to the Probability of Boundary (Pb) algorithm (Ren et al., 2008). In both of these methods, the probability of a location being classified as an edge is increased if it is consistent with a continuous boundary linking it to other edge locations in the neighbourhood.

The human visual system is far superior at performing boundary detection than any current machine vision system. For this reason, numerous neural network approaches have been proposed that are inspired by the physiology of the primary visual cortex (Ben-Shahar and Zucker, 2004; Grigorescu et al., 2003, 2004; Hansen and Neumann, 2008; Huang et al., 2009; Li, 1998; Mundhenk and Itti, 2005; Papari et al., 2007; Papari and Petkov, 2011; Petkov and Westenberg, 2003; Tang et al., 2007a,b; Ursino and La Cara, 2004; Vonikakis et al., 2006; Zeng et al., 2011a,b). A neuron in such a model has a classical receptive field (cRF), often defined using a Gabor function, that receives input from the image. The weights of the cRF are multiplied by the corresponding pixel values such that the initial response of the neuron is identical to the output of a linear filter. However, this response is subsequently modified by inputs to the non-classical receptive field (ncRF) which originate from neighbouring neurons. Typically ncRF connections are excitatory from neurons in the local neighbourhood which have a location and an orientation preference that is consistent with a smooth contour linking the two neuron's cRFs, and ncRF connections are inhibitory from neurons in the local neighbourhood which are not consistent with such smooth contours. The inhibitory and excitatory ncRF connections thus perform operations analogous to the two types of post-processing typically applied to linear filtering methods (as discussed above): suppressing unwanted, redundant, responses, while enhancing weak, but wanted, responses by interpolating between edge elements to

find extended contours. Current biologically-inspired edge-detection methods are therefore not radically different from traditional image processing techniques which employ linear filtering followed by post-processing operations.

These post-processing methods (whether biologically-inspired or not) are only partially successful in suppressing unwanted filter responses while retaining filter responses that correspond to true boundaries. Hence, rather than attempting to tidy up the output of a linear filtering process, many recent boundary-based image segmentation algorithms have abandoned such filtering entirely (*e.g.*, Arbelaez et al., 2011; Dollar et al., 2006; Maire et al., 2008; Martin et al., 2004; Ren, 2008; Ruzon and Tomasi, 2001; Smith and Brady, 1995). These methods typically extract a range of image features from the local patch of image on each side of a potential boundary, and make a decision as to whether or not there is a boundary at that location and orientation by comparing the distributions of those image features. The current work takes a similar stance in proposing that boundary-detection methods based on linear filtering (whether biologically-inspired or not) have short-comings. However, the current work takes an alternative approach in overcoming these limitations: it proposes replacing the linear filtering stage, which produces a dense, redundant, representation of the image with an algorithm that produces a sparse, efficient, representation.

The proposed algorithm is an extension to the predictive coding/biased competition (PC/BC) model of V1. This model has been shown to explain a very wide range of V1 response properties (Spratling, 2010, 2011, 2012a,b). Indeed, despite the simplicity of the PC/BC model compared to many other models of V1 physiology, it is the most comprehensive model of V1 response properties so far developed. The success of the PC/BC model of V1 in explaining cortical function is due to the responses of neurons not being independent of the responses of other neurons. Instead, these neurons interact in a mutually suppressive and nonlinear manner to determine the most efficient, sparse, representation of the input image (see Section 2). The method used to determine the sparse subset of neurons that best explain the underlying causes of the sensory input is a form of perceptual inference called predictive coding. The model proposes that the diverse range of response properties observed in V1 are a consequence of this particular perceptual inference process (Lochmann et al., 2012; Spratling, 2010).

For easy comparison with the linear filtering results, the output of the PC/BC model is illustrated in the rightmost two columns of Fig. 1. When the orientation of the input is ambiguous (top-row) multiple neurons representing the range of possible orientations are active, but each is only weakly active. As the orientation of the input becomes less ambiguous (second row), or when contextual information makes certain orientations more probable (third row), so the range of active neurons reduces. When the orientation of the input is not ambiguous (fourth row), neurons representing a single orientation are strongly active. Responses are sparser than with linear filtering and the edge is much better localised in orientation and space. Specifically, using Hoyer's measure of sparsity (Hoyer, 2004, see Section 2), the sparsity of the response of the linear filters is between 0.83 and 0.88 for the artificial images in Fig. 1, and 0.51 for the natural image. In comparison, the sparsity of the response of the proposed method is between 0.98 and 0.99 for the artificial images and 0.92 for the natural image. Two extensions to the PC/BC algorithm are proposed (see Section 2), which are analogous to the post-processing methods used in many previous edge-detection algorithms. However, here they are used to influence the on-going, iterative, process that determines the activity of each neuron that is required to explain the underlying causes of the sensory input. The BSDS300 benchmark (Martin et al., 2001) is used to assess the performance of the extended algorithm (see Section 3).

2 Methods

2.1 The LGN Model

The input to the PC/BC model of V1, described below, is an input image (I) pre-processed by convolution with a Laplacian-of-Gaussian (LoG) filter (l) with standard deviation equal to σ_{LGN} pixels. The output from this filter is subject to a multiplicative gain (the strength of which is determined by parameter κ_{LGN}) followed by a saturating non-linearity, such that:

$$\mathbf{X} = \tanh\left\{\kappa_{LGN}(I * l)\right\}$$

The positive and rectified negative responses are separated into two images $\mathbf{X}_{ON}$ and $\mathbf{X}_{OFF}$ simulating the outputs of cells in retina and LGN with circular-symmetric on-centre/off-surround and off-centre/on-surround RFs respectively. This pre-processing is illustrated on the left of Figure 2. To avoid strong edges being detected along the border of the input image, the $\mathbf{X}_{ON}$ and $\mathbf{X}_{OFF}$ values within a distance $2.5\sigma_{LGN}$ pixels from the border were set to zero.

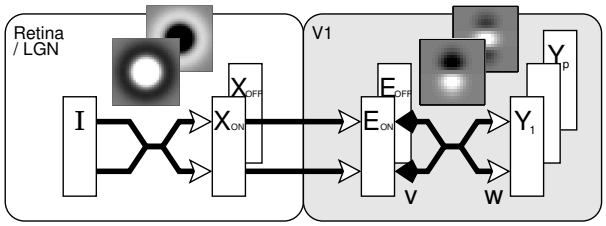

Figure 2: The PC/BC model of V1. Neural populations are represented by the white rectangles, and interconnection are shown using arrows. Open arrows signify excitatory connections, filled arrows indicate inhibitory connections, crossed connections signify a many-to-many connectivity pattern between the neurons in two populations, parallel connections indicate a one-to-one mapping between the neurons in two populations. The boxes adjacent to the connections illustrate the type of weight kernels used (circular-symmetric LoG kernels in the LGN model, and first and second derivatives of Gaussians in the V1 model). Feedforward and feedback connections have the same weights, but may be scaled differently.

2.2 The V1 Model

The PC/BC model of V1 is illustrated on the right of Figure 2 and described by the following equations:

$$\mathbf{E}_o = \mathsf{G}\left(\mathbf{X}_o\right) \oslash \left(\epsilon_2 + \sum_{k=1}^{p} \left(\mathbf{v}_{ok} * \mathbf{Y}_k\right)\right) \tag{1}$$

$$\mathbf{Y}_k \leftarrow \left(\epsilon_1 + \mathbf{Y}_k\right) \otimes \sum_{o} \left(\mathbf{w}_{ok} \star \mathbf{E}_o\right) \tag{2}$$

Where $o \in \{ON, OFF\}$; $\mathbf{X}_o$ is a 2-dimensional array, equal in size to the input image, that represents the input to the model of V1; $\mathbf{E}_o$ is a 2-dimensional array, equal in size to the input image, that represents the error-detecting neuron responses; $\mathbf{Y}_k$ is a 2-dimensional array, equal in size to the input image, that represent the prediction neuron responses; $\mathbf{w}_{ok}$ is a 2-dimensional kernel representing the synaptic weights for a particular class (k) of neuron normalised so that sum of all the weights is equal to one; $\mathbf{v}_{ok}$ is a 2-dimensional kernel representing the same synaptic weights as $\mathbf{w}_{ok}$ but normalised so that the maximum weight is equal to one; p is the total number of kernels; ϵ_1 and ϵ_2 are parameters; $\oslash$ and $\otimes$ indicate element-wise division and multiplication respectively (addition and subtraction operations involving scalars and matrices/vectors are also performed element-wise); $\star$ represents cross-correlation (which is equivalent to convolution without the kernel being rotated 180^o); and $*$ represents convolution (which is equivalent to cross-correlation with a kernel rotated by 180^o). Convolution is used in equation 1 while cross-correlation is used in equation 2 to ensure that the feedback weights between a prediction neuron and an error-detecting neuron correspond to the feedforward weights between the same two neurons, *i.e.*, this arrangement results in the feedforward weight between two neurons being identical to the feedback weight between the same two neurons (up to the different scaling applied to $\mathbf{w}_{ok}$ and $\mathbf{v}_{ok}$). G is a function that clips values at 1 (*i.e.*, $\mathsf{G}(x) = x \; \forall x \leq 1; \mathsf{G}(x) = 1 \; \forall x > 1$). This is used to prevent runaway increases in the $\mathbf{Y}$ values caused by positive feedback effects in a reciprocally connected network.

Initially the values of $\mathbf{Y}$ are all set to zero, although random initialisation of the prediction node activations can also be used with little influence on the results. Equations 1 and 2 are then updated for a number of iterations with the new values of $\mathbf{Y}$ calculated by equation 2 substituted into equations 1 and 2 to recursively calculate the changing neural activations at each iteration. Equation 2 describes the updating of the prediction neuron activations. The response of each prediction neuron is a function of its activation at the previous iteration and a weighted sum of afferent inputs from the error-detecting neurons. Equation 1 describes the calculation of the neural activity for each population of error-detecting neurons. These values are a function of the activity of the input to V1 divisively modulated by a weighted sum of the outputs of the prediction neurons in V1. The values of $\mathbf{Y}$ represent predictions of the causes underlying the inputs to the V1 model. The values of $\mathbf{E}$ represent the residual error between the input reconstructed from these predictions and the actual input to PC/BC. The iterative process described above changes the values of $\mathbf{Y}$ so as to minimise the error between actual inputs and the predicted inputs. This is achieved because if an element of $\mathbf{E}$ is greater than one (respectively less than one) prediction neurons that receive input from that error-detecting neuron will increase (decrease) their response (via equation 2). The output of the prediction neurons will then more strongly (weakly) represent the predicted causes of the input, which will in turn reduce (increase) the residual error encoded by that element of $\mathbf{E}$ (via equation 1).

The weights, $\mathbf{w}_{ok}$ and $\mathbf{v}_{ok}$, are defined using the first and second derivatives of a Gaussian. These functions were used as they closely match the pattern of response generated by the LoG filters (as used in the model of LGN)

when applied to idealised step and ridge edges. The first and second derivatives of a Gaussian also resemble the RFs of a simple cells in primary visual cortex (Young and Lesperance, 2001). The weights used were the first derivative of a Gaussian at orientations in the range $[0^o, 360^o)$ in steps of σ_C, and the positive and negative of the second derivative of a Gaussian at orientations in the range $[0^o, 180^o)$ in steps of σ_C. Typically, a value of $\sigma_C = 22.5^o$ was used resulting in a family of 32 weight kernels (in which case k ranged from 1 to 32).

The Gaussian used to define the derivative of Gaussian filters had a length (the extent perpendicular to the axis along which the derivative was taken) defined by σ_{V1} and a width in the perpendicular direction defined by σ_{LGN}. The positive and rectified negative values of the Gaussian derivatives were separated and used to define the weights to the ON and OFF channels of the input. The cross-correlation and convolution performed in Equations 1 and 2 mean that neurons with these RFs are reproduced at every pixel location in the image, and consequently, that the size of the population of V1 cells simulated varies with image size. Specifically, for $\sigma_C = 22.5^o$ the number of prediction neurons will be $32\times$ the number of image pixels.

The PC/BC algorithm is closely related to a number of other methods and can be interpreted in many different ways (Spratling, 2012b). The most useful interpretations in the context of the current work are as follows.

- A generative model. The weights of all the prediction neurons constitute an overcomplete dictionary of non-negative "basis vectors", "codebook vectors", or "elementary components". The activation dynamics determine a non-negative feature vector, $\mathbf{Y}$, that defines a linear combination of non-negative basis functions that most accurately represent the sensory-driven input (Achler and Bettencourt, 2011; Solbakken and Junge, 2011; Spratling et al., 2009).
- A method of probabilistic inference. The prediction neuron activations, $\mathbf{Y}$, represent hypotheses about the underlying causes of the sensory-driven evidence $\mathbf{X}$. In ambiguous situations, where multiple hypotheses can explain the evidence, multiple prediction nodes representing those hypotheses are concurrently active. However, in unambiguous situations the activation dynamics choose the most likely hypothesis, and the evidence is 'explained away' (Kersten et al., 2004; Lochmann and Deneve, 2011) suppressing the responses of prediction neurons representing competing hypotheses. This suppression of alternative explanations leads to a sparse response.

When the PC/BC model is used to process an image, a sparse subset of the prediction neurons whose RFs best explain the underlying causes of the sensory input are selected, via the activation dynamics, to be active. The strength of activation reflects the strength with which each image component is required to be present in order to accurately reconstruct the input. This strength of response also reflects the probability with which that image component is believed to be present, taking into account the evidence provided by the image and the full range of alternative explanations encoded in the RFs of the neurons. Given that the RFs in the current model have been chosen to represent intensity discontinuities, we can interpret the prediction neuron responses as representing the location and orientation of edges, and hence, use the responses of the prediction neurons to linearly reconstruct the edges that have been detected in the image, such that:

$$pb = \sum_{k=1}^{p} (\mathbf{r}_{ok} * \mathbf{Y}_k)$$

where $\mathbf{r}_{ok}$ are kernels representing straight edge elements. Such edge reconstructions using the basic PC/BC model described above are shown in the last column of Fig. 1 and the second column of Fig. 5.

Although the PC/BC algorithm does not explicitly minimise the reconstruction error using a global objective function, the activity of the prediction neurons can be used to reconstruct the input with high fidelity (Spratling, 2012b; Spratling et al., 2009), images are thus represented accurately. Furthermore, although PC/BC does not explicitly enforce sparseness, the activity of the prediction neurons in response to natural image patches is highly sparse and statistically independent (Spratling, 2010, 2012b), images are therefore represented efficiently. In order to demonstrate that PC/BC generates sparse codes in the current application, the sparsity of the response to each image was measured using the metric proposed by Hoyer (2004):

$$SI_H = \frac{\sqrt{n} - \frac{\|\mathbf{Y}\|_1}{\|\mathbf{Y}\|_2}}{\sqrt{n} - 1}$$

where n is the total number of coefficients (*i.e.*, the size of $\mathbf{Y}$). SI_H ranges from 0 to 1, with higher values corresponding to sparser distributions.

2.3 Recurrent Lateral Excitation

The standard PC/BC model of V1 described above does a reasonably good job of locating intensity discontinuities in images without the redundant responses generated by linear filtering. However, in some cases weak intensity

discontinuities are not represented. To solve this problem an extension of the PC/BC model of V1 is proposed that employs lateral excitatory connections. These connections take a similar form as those proposed in many previous biologically-inspired models. Specifically, neurons are linked by lateral excitatory connections if they represent edge elements that have locations and orientations consistent with a smooth, co-circular, contour. Such connections are consistent with natural image statistics (Geisler et al., 2001; Sigman et al., 2001) and are believed to be encoded by long-range horizontal connections in V1 (Field et al., 1993; Fitzpatrick, 2000; Hess et al., 2003; Hunt et al., 2011).

Excitatory lateral connectivity has been used in many previous algorithms for boundary detection with limited success. However, these previous algorithms employed linear filtering. The redundant responses are then just as likely to be enhanced by lateral excitation as are the true responses. In contrast, PC/BC produces a sparse representation which enables lateral excitation to be much more selective in enhancing the responses to those edges that are most probable.

The method of implementing lateral excitatory connections in the PC/BC model is illustrated in Fig. 3. Recurrent connections are used to take the outputs of the prediction neurons and use them as additional inputs to the V1 model, *i.e.*,:

$$\mathbf{X}_k = \mathbf{Y}_k$$

Where $\mathbf{X}_k$ is the a 2-dimensional array, equal in size to the input image, which is substituted into equation 1. When lateral connections are present, the subscripts for $\mathbf{X}$ in equation 1 thus range over $\{ON, OFF\}$ and $[1, p]$ (where p is the total number of receptive field types, see equation 1). $\mathbf{X}_k$ has been omitted from Fig. 3 as these values do not need to be represented in a separate neural population. Using recurrent inputs to implement lateral connectivity is, from an implementational perspective, very elegant as it does not require the introduction of any new mechanisms into the model. As with the feedforward weights, the lateral input to each prediction neuron is calculated by convolving the recurrent inputs with weight kernels. The convolution process effectively reproduces the same pattern of lateral connectivity between prediction neurons with equivalent RFs at all locations in the model. The strength of the connection between prediction neuron a of type i and prediction neuron b of type j is given by:

$$\mathbf{w}_{ij}^{ab} = S \exp\left\{ -\frac{(d - 2\sigma_D)^2}{2\sigma_D^2} - \frac{\theta^2}{2\sigma_C^2} - \frac{\psi^2}{2\sigma_A^2} \right\} \tag{3}$$

where d is the euclidean distance between the coordinates of points a and b; θ is the difference in angle between the orientation of filters of type i and j and the tangent to a circle passing through points a and b; ψ is the curvature of the circle passing through points a and b; the σ values are parameters that control the extent and specificity of the lateral connections, and S controls the strength of the lateral connections. This leads to strong connections between prediction neurons the have roughly co-circular RFs. No lateral connections are defined between first and second derivative filters. The lateral weights between the first-derivative filters are illustrated on the top-left of Fig. 4. The feedback weights ($\mathbf{v}_{ij}$) are defined to be identical to the lateral weights, but are normalised so that the maximum weight is equal to one.

Another advantage of implementing lateral connectivity as recurrent inputs to the model can be seen if we consider another interpretation of the operation of the PC/BC model. Each input can be interpreted as providing a fixed amount for activation which is distributed among the active prediction neurons to which it connects (Achler and Amir, 2008; Reggia, 1985). Hence, if an input connects to many active prediction neurons it will cause very little additional activation in those neurons. However, if the number of active predictions neurons is small, the input is distributed less thinly, and has a greater effect. In the limit, if there is only one active prediction node this will receive all the input, and hence, its response can be strongly influenced. Hence, the lateral connections have very little effect when there are many possible contours as the recurrent input gets distributed among all the possible contours. However, when there are few possible contours linking the active prediction neurons in the local neighbourhood, the lateral inputs can noticeably enhance these responses. When this occurs, the lateral input is "explained away" and does not produce redundant responses from other prediction neurons that lie on other, less probable, contours.

The effects of the lateral connections can be seen by comparing the third column with the second column of Fig. 5. Lateral connections serve to enhance the representation of extended contours. However, in areas of the image where there are lots of edges generated by texture, coincidental alignments of different texture elements are also enhanced by the lateral connections.

2.4 Suppressing Texture

For most edge detection tasks, where the aim is to locate the boundaries of surfaces and objects, the edges of texture elements are unwanted. The PC/BC algorithm attempts to generate an accurate reconstruction of the input.

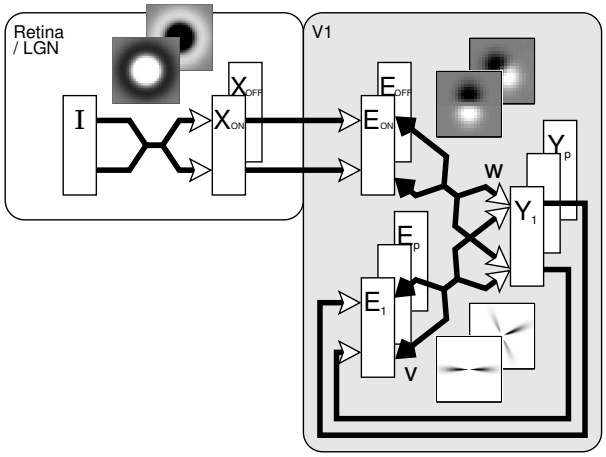

Figure 3: The PC/BC model of V1 with recurrent inputs providing lateral connectivity between the prediction neurons. The format of the diagram is identical to, and described in the caption of, Fig. 2.

It is therefore not possible to simply suppress the response of prediction neurons that are responding to texture. However, it is possible to ignore texture by introducing another set of prediction neurons that will represent the edges of the texture elements. The prediction neurons representing texture edges are defined to have feedforward weights to the ON and OFF channels of the input that are identical to the feedforward weights of the prediction neurons representing boundary edges. To reduce the computational complexity of the model, only first derivative of a Gaussian are used to define these RFs: second derivative kernels are no longer used. This also means that the total number of prediction neurons in the model remains the same.

For each pair of prediction neurons with identical RFs, one neuron will be used to respond to texture edges the other to non-texture edges. Given that the RFs are the same, it is purely differences in the lateral connectivity that differentiates prediction neurons that operate as texture edge detectors from those that act as boundary edge detectors. This is similar to some models of cortical area V2 which employ neurons representing boundaries that come in pairs to represent different configurations of border-ownership (Craft et al., 2007; Vecera and O'Reilly, 1998; Zhaoping, 2005). There is no empirical evidence that V1 contains separate sub-populations of neurons responding to texture edges and boundary edges. However, if such neurons did exist they would have similar RFs, and hence, similar response properties and would be unlikely to be classified separately unless such distinctions were being actively sought.

In this extension to the PC/BC model of V1, there are four possible types of lateral connections: those in which the pre- and post-synaptic prediction neurons both represent boundaries; those in which the pre- and post-synaptic prediction neurons both represent texture; those in which the pre-synapic prediction neurons represent boundaries and the post-synapic connections represent texture; and those in which the pre-synapic prediction neurons represent texture and the post-synaptic connections represent boundaries. The boundary-to-boundary connections are as described in the previous section and are defined using equation 3. Texture-to-boundary connections are also definined using equation 3, but with θ off-set by 90^o, and the resulting filter rotated by 90^o, so that prediction neurons selective for boundary edges are excited by texture elements that are roughly perpendicular. Also, the values of θ are calculated modulo 180^o, rather than 360^o as the "phase" of the perpendicular texture is not relevant. S is also halved for texture-to-boundary connections compared to boundary-to-boundary connections, to give more weight to connections between prediction neurons representing the same class of feature.

Lateral connections to prediction neurons representing texture edges are definined using the following equation:

$$\mathbf{w}_{ij}^{ab} = S \exp\left\{-\frac{(d-2\sigma_D)^2}{2\sigma_D^2} - \frac{\phi^2}{2\sigma_C^2} - \frac{\omega^2}{2\sigma_A^2}\right\} \tag{4}$$

Where d is the euclidean distance between the coordinates of points a and b; ϕ is the difference in angle between the line subtended by points a and b and a line perpendicular to the post-synaptic filter; ω is the difference in angle between the orientation of filters of type i and filters of type j; the σ values are parameters that control the extent and specificity of the lateral connections, and S controls the strength of the lateral connections. For texture-to-texture connections this leads to strong connections between prediction neurons the have roughly parallel RFs. For boundary-to-texture connections ω is off-set by 90^o, and the resulting filter rotated by 90^o, so that prediction neurons selective for texture edges are excited by boundary elements that are roughly perpendicular. Also, the values of ψ are calculated modulo 180^o, rather than 360^o as the "phase" of the perpendicular boundary is not

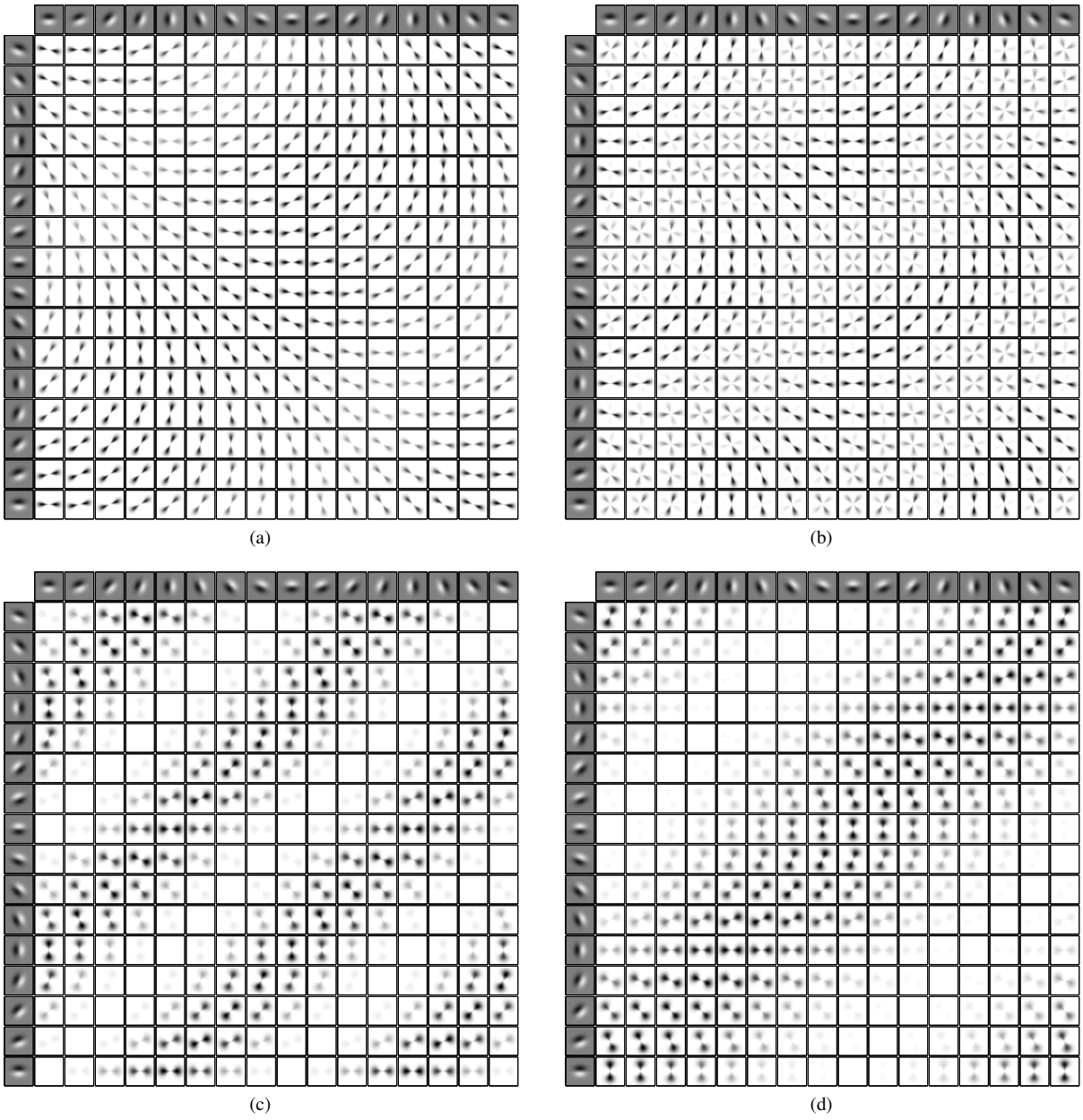

Figure 4: Weights of the lateral connections. In each of the four sub-figures, the left hand column shows the RF (feedforward weights) of the post-synaptic neuron and the top row shows the RF of the pre-synaptic neurons (light pixels indicate strong weights to the OFF channel of the input from the LGN, and dark pixels indicate strong weights to the ON channel). Each other element within the four sub-figures shows the spatial distribution of lateral weights between the corresponding pre-synaptic and post-synaptic neuron. (a) Shows synaptic weights between prediction neurons that represent boundary edges. These connections link neurons with co-circular RFs. (b) Shows synaptic weights from prediction neurons that represent texture edges to prediction neurons that represent boundary edges. These connections preferentially link neurons with perpendicular RFs. (c) Shows synaptic weights from prediction neurons that represent boundary edges to prediction neurons that represent texture edges. These connections preferentially link neurons with perpendicular RFs. (d) Shows synaptic weights between prediction neurons that represent texture edges. These connections preferentially link neurons with parallel RFs. The RFs of the feedforward and lateral weights are shown at different spatial scales, with each feedforward RF covering 21×21 pixels while the lateral weights have an extent of 55×55 pixels.

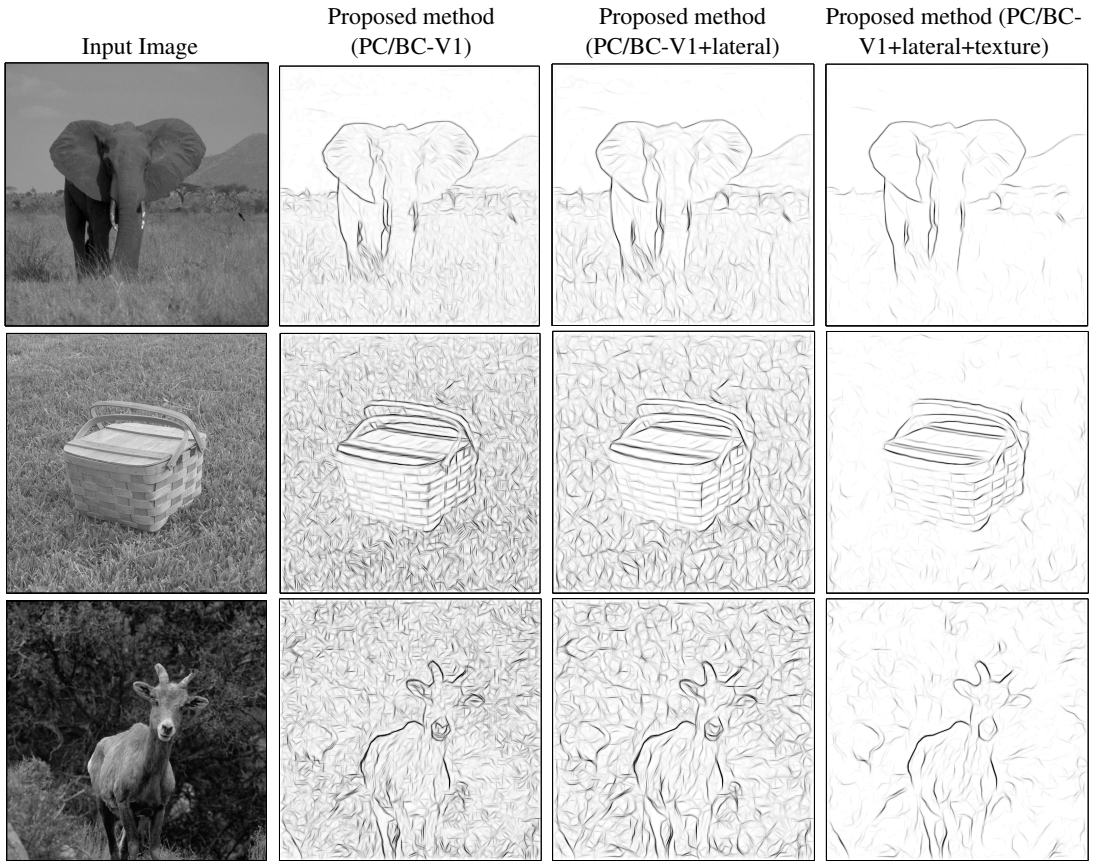

Figure 5: Results for three images from the RuG dataset (Grigorescu et al., 2004).

relevant. S is also halved for boundary-to-texture connections compared to texture-to-texture connections, to give more weight to connections between prediction neurons representing the same class of feature.

While the lateral connections between prediction neurons representing boundaries are consistent with co-circularity, the lateral weights to and from prediction neurons representing texture elements are anti-correlated with co-circularity, both these types of connections are seen in cortex (Hunt et al., 2011). All four types of lateral connection are illustrated in Fig. 4. In all four cases the corresponding feedback weights ($\mathbf{v}_{ij}$) are defined to be identical to the lateral weights, but are normalised so that the maximum weight for all the feedback weights for each prediction neuron is equal to one.

The right-hand column of Fig. 5 shows the results of reconstructing edges using only the responses of the boundary selective prediction neurons. It can be seen that the algorithm is very effective at finding the boundaries of objects and surfaces while disregarding texture. The parameter values used to generate the results shown in Fig. 5, and reported in the next section, were: $\kappa_{LGN} = 2\pi$, $\sigma_{LGN} = 2$, $\sigma_{V1} = 3$, $\sigma_C = 22.5$, $\sigma_D = 6$, $\sigma_A = 60$, $S = 0.5$, $\epsilon_1 = 1 \times 10^{-5}$, $\epsilon_2 = 1 \times 10^{-3}$, and 30 iterations of the PC/BC algorithm were preformed. Clearly, the optimal parameters may differ between images, and even between different parts of the same image. However, these parameters prove to be suitable for all the test images the algorithm has been applied to (see section 3).

2.5 Code

Software, written in MATLAB, which implements the algorithm described in the preceding sections is available at `http://www.corinet.org/mike/Code/edge_detection.zip`.

3 Results

The results shown in Fig. 5 use three images from the RuG dataset (Grigorescu et al., 2003). This dataset includes a total of 40 512-by-512 pixel images of animals and man-made objects in natural scenes. It has been used extensively for testing other biologically-inspired boundary detection algorithms, and the results shown in the last column of Fig. 5 can be compared directly to: Fig.15a in Papari and Petkov (2008); Fig.8 in Papari and Petkov

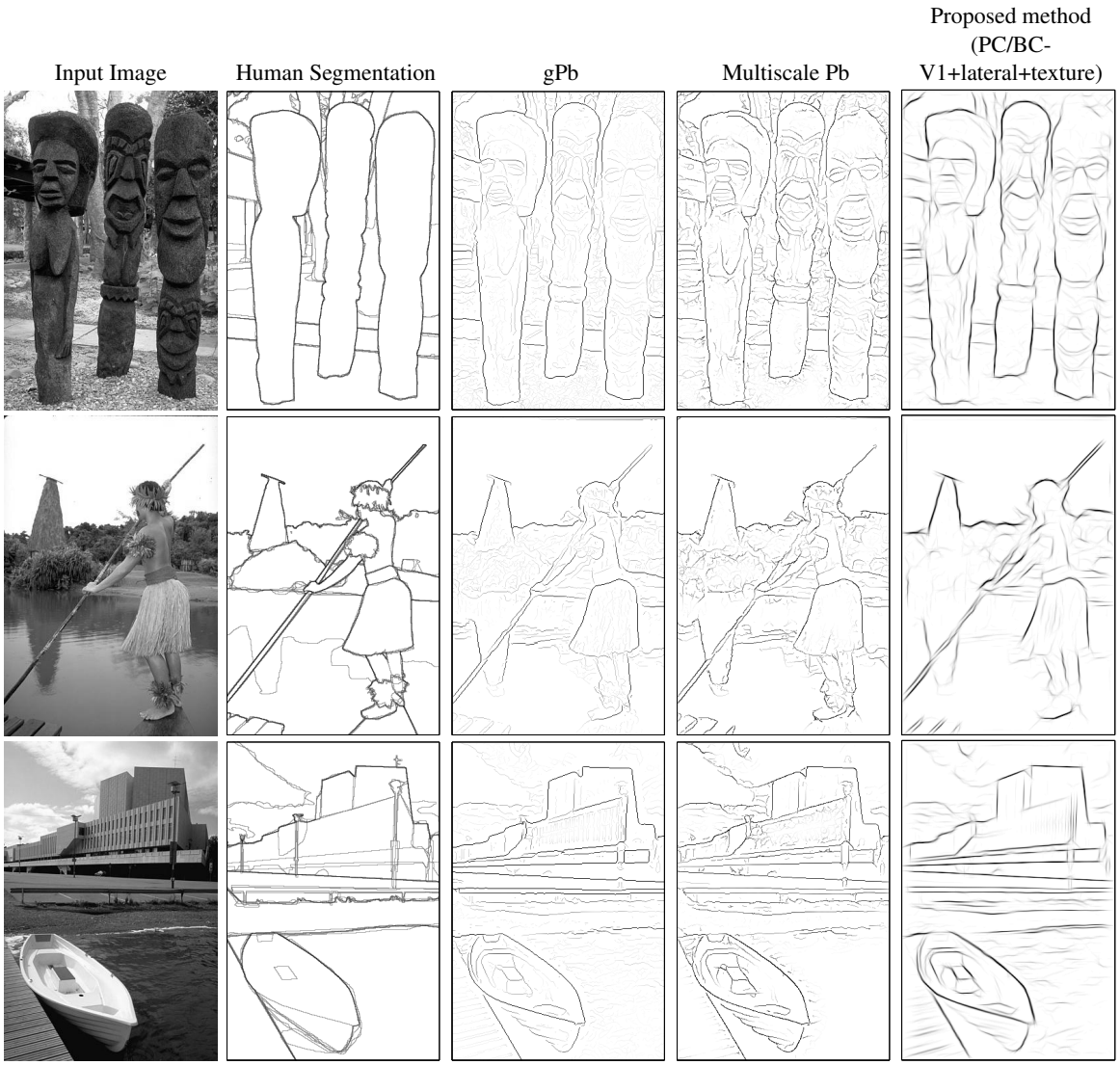

Figure 6: Results for three images from the BSDS300 dataset (Martin et al., 2001).

(2011); Fig.5 in Zeng et al. (2011b); Fig.2 in Zeng et al. (2011a); Fig.7 in Grigorescu et al. (2003); and Fig.9 in Tang et al. (2007b). The sparsity of the representations found by PC/BC are not strongly influenced by the proposed extensions to this algorithm: the inclusion lateral connections, and subsequently the inclusion of two sets of prediction neurons to represent boundary edges and texture edges. For each of the three images in Fig. 5, the sparsity of the PC/BC prediction neuron population was measured using Hoyer's sparsity index (Hoyer, 2004, see section 2.2), for all three variations of the model. For the PC/BC model of V1 without lateral connections, the sparsity varied between 0.91 and 0.93. For the PC/BC model of V1 with lateral connections, the sparsity also varied between 0.91 and 0.93. For the model with lateral connections and texture selective prediction neurons, the sparsity varied between 0.89 and 0.93.

The results shown in Fig. 6 are for three images taken from the BSDS300 dataset (Martin et al., 2001). This dataset contains a test set of 100 481-by-321 pixel images of animals, people, buildings, man-made objects, and natural scenes. It has also been used extensively for testing various image segmentation algorithms. Two algorithms that have produced the best results on greyscale images from this dataset are Global Probability of Boundary (gPb) (Maire et al., 2008) and Multi-scale Pb (Ren, 2008). Results for these two algorithms are shown in the second and third columns of Fig. 6 for easy comparison.

In addition to qualitative comparisons of results, the BSDS300 dataset also provides a method to quantitatively compare the performance of image segmentation algorithms against segmentations performed by humans (Martin et al., 2001). Performance is measured using the F score which compares the boundaries found by the algorithm with those identified by human subjects. In practice, for grayscale images F varies from 0.41 (which is the performance of an algorithm that assigns each pixel a probability of being a boundary at random) to 0.79 (which is

	Algorithm	F-score
Multi-feature/scale algorithms:	gPb-owt-ucm (Arbelaez et al., 2011)	0.71 (Arbelaez et al., 2011)
	gPb (Maire et al., 2008)	0.68 (Maire et al., 2008)
	Multi-scale Pb (Ren, 2008)	0.66 (Ren, 2008)
	BEL (Dollar et al., 2006)	0.64 (Maire et al., 2008)
	Pb (brighness+texture) (Martin et al., 2004)	0.63 (Martin et al., 2004)
Single-feature/scale algorithms:	**PC/BC-V1+lateral+texture**	0.61
	Pb (brighness) (Martin et al., 2004)	0.60 (Martin et al., 2004)
	Canny (Canny, 1986)	0.58 (Maire et al., 2008)

Table 1: Quantitative comparison of algorithms on the BSDS300 benchmark (Martin et al., 2001) with greyscale images.

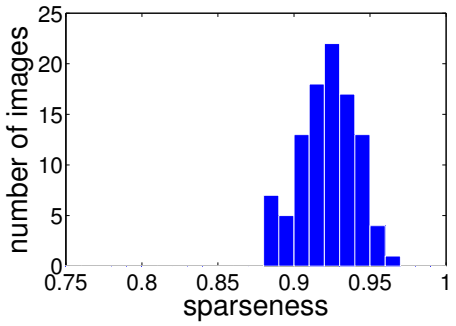

Figure 7: Sparsity of the response of the proposed algorithm PC/BC-V1+lateral+texture. The distribution of Hoyer's sparsity index for all of the 100 test images in the BSDS300 dataset.

the performance of a human observer when compared to other human observers[2]). On this benchmark the PC/BC algorithm, with both proposed extensions, scores 0.61. This score is below that of the state-of-the-art (see Table 1). However, gPb, Multi-scale Pb, Boosted Edge Learning (BEL), and Pb all employ multiple image features and/or image scales. In contrast, PC/BC uses only image intensity at a single scale. The performance of PC/BC is an improvement on that of other algorithms that also only use intensity information (see Table 1).

For each of the 100 images in the BSDS300 test set the sparsity of the representation formed by the PC/BC algorithm, with both proposed extensions, was measured using Hoyer's sparsity index (Hoyer, 2004, see section 2.2). As can be seen from the distribution of sparsity indices plotted in Fig. 7, for every image the neural code is highly sparse, and the sparsity index is never less than 0.88. Although the PC/BC algorithm does not explicitly control the sparsity of the prediction neuron responses, the competition between different prediction neurons to most accurately represent the image elements results in responses to natural images that are highly sparse.

To obtain an estimate of the parameter sensitivity of the proposed algorithm, benchmarking with the BSDS300 dataset was repeated with different parameter settings. In each case one parameter was altered at a time while all other parameters were kept fixed at their default values. The results are shown in Table 2. It can be seen that out of the 20 parameter combinations tested, 12 produced F-scores greater than or equal to 60, the F-score achieved by Pb (brighness). Segmentation performance is little affected by significant changes in several parameter values, *i.e.*, κ_{LGN}, σ_{V1}, σ_C, ϵ_2. However, changing other parameter values does have a significant affect on the results. Example results for the parameter settings that achieve the poorest performance are provided in Fig. 8. These results can be compared directly to the last row of Fig. 6.

Increasing σ_{LGN} causes small scale intensity discontinuities to be smoothed out of the input to the PC/BC algorithm. Changing the value of σ_D changes the range of the lateral connectivity in the model. For low values of σ_D, lateral connections are short, and hence, only very local contextual information is taken into account by the model. There is therefore little enhancement of co-circular elements forming boundaries, and insufficient information for boundary edges and texture edges to be reliably distinguished. On the other hand, increasing σ_D results in a very large region of contextual influence, resulting in only large-scale boundaries being detected. The results are also sensitive to changes in the values of σ_A. Reducing σ_A causes the lateral connections to be more selective for straighter boundary edges and perpendicular texture edges. However, a side-effect of changing this parameter is to change the relative overall strength of the lateral connections targeting boundary selective

[2]this is less than one as human observers are not consistent in their decisions about what is and is not a true boundary.

Parameter	Description	Standard Value	New Value	F-score
κ_{LGN}	LGN response gain	2π	π	0.61
			4π	0.61
σ_{LGN}	LGN RF size	2	1	0.60
			4	0.55
σ_{V1}	V1 RF length	3	1.5	0.60
			6	0.61
σ_C	Step between RF orientations, and lateral connections degree of preference for co-circular/perpendicular pre-synaptic RFs	22.5	15	0.61
			30	0.61
σ_D	Spatial extent of lateral connections	6	3	0.57
			12	0.55
σ_A	Lateral connections degree of preference for co-linear/parallel pre-synaptic RFs	60	30	0.58
			90	0.57
S	Strength of lateral connections	0.5	0.25	0.60
			0.75	0.53
ϵ_1	Controls the baseline response and rate of increase in the response of prediction neurons	1×10^{-5}	1×10^{-6}	0.61
			1×10^{-4}	0.57
ϵ_2	Prevents division by zero errors and determines minimum significant input activity	1×10^{-3}	1×10^{-4}	0.60
			1×10^{-2}	0.61
iterations	Number of iterations of the PC/BC algorithm	30	15	0.59
			60	0.61

Table 2: Evaluation of the sensitivity of the proposed algorithm PC/BC-V1+lateral+texture to the parameter values. Performance was evaluated using BSDS300 benchmark (Martin et al., 2001) with greyscale images for a variety of parameter values.

prediction neurons and texture selective prediction neurons. Reducing σ_A increases the relative strength of the lateral connections to boundary selective prediction neurons, resulting in more intensity discontinuities being classified as boundary edges. Increasing σ_A has the opposite effect, resulting in boundary edges being misclassified as texture edges. The overall strength of the lateral weights is also important. When S is high lateral influences become more dominant, resulting in very strong responses to the most salient boundaries, but relatively weak responses to less salient boundaries. The number of iterations performed by the PC/BC algorithm also affects performance. When few iterations are performed, the model does not have time to converge to a steady-state response. The response is also much less sparse: at 15 iterations Hoyer's sparsity index across all 100 images is in the range [0.69, 0.93], compared to [0.88, 0.96] at 30 iterations. However, it can also be seen that once sufficient iterations have been performed to reach a steady-state, that the representation produced by the model is stable, and the F-score remains constant.

Intensity discontinuities occur at different scales depending on the physical characteristics of the edge, and the extrinsic and intrinsic parameters of the imaging system. In order to detect image features across multiple scales the traditional approach is to employ a multiscale image pyramid (Adelson et al., 1984). Feature detection is then performed at multiple scales and the outputs are combined to provide scale-invariant results. However, this method is biologically-implausible. It is well known that RF size, in retina and V1, increases with eccentricity from the fovea. However, at any particular location V1 RFs have a small range of spatial frequency preferences (Foster et al., 1985; Gattass and Rosa, 1987; Van Essen et al., 1984), and hence, at each image location processing occurs at a small range of scales. A biologically-plausible multiscale extension to the PC/BC model of V1 is left for future work. However, to allow comparison of the current model with multiscale methods, the PC/BC algorithm, with both proposed extensions, was applied to the original image and four down-sampled images, forming a five-level image pyramid. The boundaries detected at each scale were resized to the size of the original image and added together. This produced an F-score of 62 on the BSDS300 benchmark. While there may be scope for improving this result by using more sophisticated methods of combining results across scales (Papari et al., 2007; Ren, 2008) this results suggest that the proposed algorithm is not limited significantly by sensitivity to scale, and that incorporating other cues to segmentation, such as texture and colour, may be more effective at

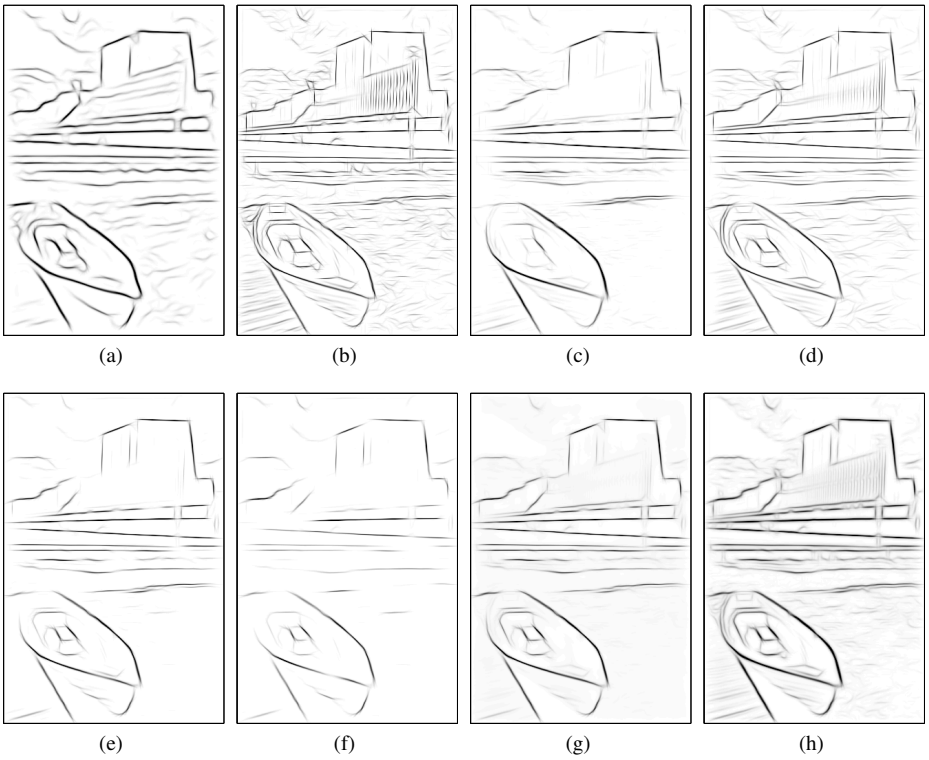

Figure 8: Worst results for one images from the BSDS300 dataset (Martin et al., 2001) with varying parameter values. Results were obtained using the default parameter setting for the proposed algorithm PC/BC-V1+lateral+texture, except (a) $\sigma_{LGN} = 4$, (b) $\sigma_D = 3$, (c) $\sigma_D = 12$, (d) $\sigma_A = 30$, (e) $\sigma_A = 90$, (f) $S = 0.75$, (g) $\epsilon_1 = 1e - 4$, (h) iterations=15.

improving performance of the model.

4 Discussion

It is widely believed that primary visual cortex is involved in edge detection (Lee, 2003; Ng et al., 2007). It is also widely believed that V1 encodes images using a sparse set of basis vectors selected from an over-complete dictionary (Field, 1994; Olshausen and Field, 1997). When the PC/BC algorithm (with appropriate learning rules) is trained on natural images, it learns a dictionary of basis vectors (*i.e.*, synaptic weights) that resemble the RFs of V1 cells (Spratling, 2012b). Many other algorithms, when trained on natural images, have also been shown to be able to learn basis sets that resemble the RFs of cells in primary visual cortex (*e.g.*, Bell and Sejnowski, 1997; Falconbridge et al., 2006; Hamker and Wiltschut, 2007; Harpur, 1997; Hoyer, 2003, 2004; Hoyer and Hyvärinen, 2000; Jehee and Ballard, 2009; Lücke, 2009; Olshausen and Field, 1996; Perrinet, 2010; Ranzato et al., 2007; Rehn and Sommer, 2007; Van Hateren and van der Schaaf, 1998; Weber and Triesch, 2008; Wiltschut and Hamker, 2009). A common feature of all these algorithms is that the learnt representation is sparse. Sparsity is either imposed by explicitly limiting or penalising the number of basis vectors that can be used to encode an image or is the implicit outcome of a form of competition between different basis vectors (or specifically nodes in a neural network with synaptic weights representing those basis vectors) for the right to be involved in representing an image. While many boundary detection algorithms have employed filters that are inspired by, or analogous to, the shapes of V1 RFs, these filters have been applied in very different way than seems to be the case in V1. Specifically, in traditional approaches to boundary detection, filters have been applied using linear filtering to generate dense, redundant, representations. Other work has demonstrated that sparse coding can be beneficial for image compression (*e.g.*, Fischer et al., 2006, 2007; Murray and Kreutz-Delgado, 2006; Pece and Petkov, 2000), image restoration (*e.g.*, Elad and Aharon, 2006; Hyvarinen et al., 1998; Mairal et al., 2009), and classification (*e.g.*, Bociu and Pitas, 2004; Mairal et al., 2008; Ramirez et al., 2010; Wright et al., 2009). Here, it has been shown that sparse coding can also be used for boundary detection.

The PC/BC model of V1 has previously been shown to provide a comprehensive account of neurophysiological

data recorded in primary visual cortex (Spratling, 2010, 2011, 2012a,b). Here, two extensions are proposed to enable this model of biological vision to be applied to the real-world task of image segmentation. The first extension adds recurrent lateral connections to the model, such that the responses of prediction neurons can provide input to, and hence influence the response of, neighbouring prediction neurons. The second extension proposes that prediction neurons come in pairs with identical RFs but different lateral connectivity patterns, so that one prediction neuron will represent boundaries while the other responds to texture. These two extensions perform operations analogous to the post-processing techniques employed by many existing boundary detection algorithms and by many previous biologically-inspired neural approaches to boundary detection. The main innovation in the current algorithm is therefore the use of the PC/BC algorithm in place of linear filtering. While linear filtering produces a dense, redundant, representation of the discontinuities in an image, PC/BC produces a sparse, efficient, representation where the strength of response also reflects the probability with which that edge element is believed to be present, taking into account the evidence provided by the image and the full range of alternative explanations encoded in the RFs of the neurons (both those in the local neighbourhood with overlapping feedforward RFs, and due to long-rang recurrent connections those at greater distances). By using a sparse representation, the current model has been shown to produce good image segmentation performance on the BSDS300 benchmark (Martin et al., 2001), out-performing the current state-of-the-art image segmentation method, Pb (Arbelaez et al., 2011; Maire et al., 2008; Martin et al., 2004), when that algorithm is also restricted to using only intensity information at a single scale, as is the case for the PC/BC algorithm.

Obvious extensions to the current work would expand the model to make use of the same range of information sources as are used in the full version of the Pb algorithm, gPb (Arbelaez et al., 2011; Maire et al., 2008). Colour could be incorporated by modifying the LGN stage to include colour-opponent centre-surround neurons, and including prediction neurons in the V1 model with RFs selective for these additional stimulus properties. Texture could be incorporated by using, rather than ignoring, the outputs of the texture selective prediction neurons in order to locate borders between textured and non-textured regions, or between regions in which there is orientation contrast between textures. Incorporating information across multiple spatial scales could be achieved by providing additional channels of input to V1 (extracted in the model of LGN using LoG filters with multiple scales) and equipping V1 prediction neurons with RFs that take information from these multiple LGN channels and/or by using lateral connections with a range of spatial extents.

References

Achler, T. and Amir, E. (2008). Input feedback networks: Classification and inference based on network structure. In Wang, P., Goertzel, B., and Franklin, S., editors, *Artificial General Intelligence*, pages 15–26.

Achler, T. and Bettencourt, L. (2011). Evaluating the contribution of top-down feedback and post-learning reconstruction. In Samsonovich, A. V. and Jóhannsdóttir, K. R., editors, *Biologically Inspired Cognitive Architectures*, pages 9–15.

Adelson, E. and Bergen, J. (1985). Spatiotemporal energy models for the perception of motion. *J. Opt. Soc. Am. A Opt. Image Sci. Vis.*, 2:284–99.

Adelson, E. H., Anderson, C. H., Bergen, J. R., Burt, P. J., and Ogden, J. M. (1984). Pyramid methods in image processing. *RCA Engineer*, 29(6):33–41.

Arbelaez, P., Maire, M., Fowlkes, C., and Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916.

Bell, A. J. and Sejnowski, T. J. (1997). The 'independent components' of natural scenes are edge filters. *Vision Res.*, 37(23):3327–38.

Ben-Shahar, O. and Zucker, S. W. (2004). Geometrical computations explain projection patterns of long range horizontal connections in visual cortex. *Neural Comput.*, 16(3):445–76.

Bociu, I. and Pitas, I. (2004). A new sparse image representation algorithm applied to facial expression recognition. In *Proceedings of the IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, pages 539–48.

Canny, J. F. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8:679–98.

Craft, E., Schütze, H., Niebur, E., and von der Heydt, R. (2007). A neural model of figure-ground organization. *J. Neurophysiol.*, 97:4310–326.

Daugman, J. G. (1980). Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Res.*, 20:847–56.

Derrington, A. M. and Lennie, P. (1984). Spatial and temporal contrast sensitivities of neurons in the lateral geniculat nucleus of the macaque. *J. Physiol. (Lond.)*, 357:219–40.

Dollar, P., Tu, Z., and Belongie, S. (2006). Supervised learning of edges and object boundaries. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*

Elad, M. and Aharon, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Proc.*, 15(12):3736–45.

Falconbridge, M. S., Stamps, R. L., and Badcock, D. R. (2006). A simple Hebbian/anti-Hebbian network learns the sparse, independent components of natural images. *Neural Comput.*, 18(2):415–29.

Field, D., Hayes, A., and Hess, R. (1993). Contour integration in the human visual system: evidence for a local 'association' field. *Vision Res.*, 33:173–93.

Field, D. J. (1994). What is the goal of sensory coding? *Neural Comput.*, 6(4):559–601.

Fischer, S., Cristóbal, G., and Redondo, R. (2006). Sparse overcomplete gabor wavelet representation based on local competition. *IEEE Trans. Image Proc.*, 15(2):265–72.

Fischer, S., Redondo, R., Perrinet, L., and Cristóbal, G. (2007). Sparse approximation of images inspired from the functional architecture of the primary visual areas. *EURASIP Journal on Advances in Signal Processing*, 2007:90727.

Fitzpatrick, D. (2000). Seeing beyond the receptive field in primary visual cortex. *Curr. Opin. Neurobiol.*, 10:438–43.

Foster, K. H., Gaska, J. P., Nagler, M., and Pollen, D. A. (1985). Spatial and temporal frequency selectivity of neurones in visual cortical areas V1 and V2 of the macaque monkey. *J. Physiol. (Lond.)*, 365:331–63.

Freeman, W. T. and Adelson, E. H. (1991). The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(9):891–906.

Gattass, R. nad Sousa, A. P. B. and Rosa, M. G. P. (1987). Visual topography of V1 in the cebus monkey. *J. Comp. Neurol.*, 259:529–48.

Geisler, W. S., Perry, J. S., Super, B. J., and Gallogly, D. P. (2001). Edge co-occurrence in natural images predicts contour grouping performance. *Vision Res.*, 41(6):711–24.

Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 6:721–41.

Grigorescu, C., Petkov, N., and Westenberg, M. A. (2003). Contour detection based on nonclassical receptive field inhibition. *IEEE Trans. Image Proc.*, 12(7):729–39.

Grigorescu, C., Petkov, N., and Westenberg, M. A. (2004). Contour and boundary detection improved by surround suppression of texture edges. *Image Vis. Comput.*, 22(8):609–22.

Hamker, F. H. and Wiltschut, J. (2007). Hebbian learning in a model with dynamic rate-coded neurons: an alternative to the generative model approach for learning receptive fields from natural scenes. *Network*, 18:249–66.

Hancock, E. R. and Kittler, J. (1990). Edge-labeling using dictionary-based relaxation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(2):165–81.

Hansen, T. and Neumann, H. (2008). A recurrent model of contour integration in primary visual cortex. *J. Vis.*, 8(8).

Harpur, G. F. (1997). *Low Entropy Coding with Unsupervised Neural Networks*. PhD thesis, Department of Engineering, University of Cambridge.

Hess, R., Hayes, A., and Field, D. J. (2003). Contour integration and cortical processing. *J. Physiol. Paris*, 97:105–19.

Hoyer, P. O. (2003). Modeling receptive fields with non-negative sparse coding. *Neurocomputing*, 52–54:547–52.

Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *J. Mach. Learn. Res.*, 5:1457–69.

Hoyer, P. O. and Hyvärinen, A. (2000). Independent component analysis applied to feature extraction from colour and stereo images. *Network: Computation in Neural Systems*, 11(3):191–210.

Huang, W., Jiao, L., Jia, J., and Yu, H. (2009). A neural contextual model for detecting peceptually salient contours. *Pattern Recognit. Let.*, 30:985–93.

Hunt, J., Bosking, W., and Goodhill, G. (2011). Statistical structure of lateral connections in the primary visual cortex. *Neural Systems and Circuits*, 1(1):3.

Hyvarinen, A., Hoyer, P., and Oja, E. (1998). Sparse code shrinkage for image denoising. In *Proc. Int. Joint Conf. on Neural Netw.*, volume 2, pages 859–64.

Jehee, J. F. M. and Ballard, D. H. (2009). Predictive feedback can account for biphasic responses in the lateral geniculate nucleus. *PLoS Comput. Biol.*, 5(5):e1000373.

Jones, J. P. and Palmer, L. A. (1987). An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. *J. Neurophysiol.*, 58(6):1233–58.

Kersten, D., Mamassian, P., and Yuille, A. (2004). Object perception as Bayesian inference. *Annu. Rev. Psychol.*, 55(1):271–304.

Lee, T. S. (2003). Computations in the early visual cortex. *J. Physiol. Paris*, 97(2–3):121–39.

Li, Z. (1998). A neural model of contour integration in the primary visual cortex. *Neural Comput.*, 10:903–40.

Lochmann, T. and Deneve, S. (2011). Neural processing as causal inference. *Curr. Opin. Neurobiol.*, 21(5):774–81.

Lochmann, T., Ernst, U. A., and Denève, S. (2012). Perceptual inference predicts contextual modulations of sensory responses. *J. Neurosci.*, 32(12):4179–95.

Lücke, J. (2009). Receptive field self-organization in a model of the fine structure in V1 cortical columns. *Neural Comput.*, 21(10):2805–45.

Mairal, J., Bach, F., Ponce, J., Sapiro, G., and Zisserman, A. (2008). Discriminative learned dictionaries for local image analysis. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 1–8.

Mairal, J., Bach, F., Ponce, J., Sapiro, G., and Zisserman, A. (2009). Non-local sparse models for image restoration. In *Proc. Int. Conf. Comput. Vision*, pages 2272–9.

Maire, M., Arbelaez, P., Fowlkes, C., and Malik, J. (2008). Using contours to detect and localize junctions in natural images. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*

Marcelja, S. (1980). Mathematical description of the responses of simple cortical cells. *J. Opt. Soc. Am. A Opt. Image Sci. Vis.*, 70:1297–1300.

Marr, D. and Hildreth, E. (1980). theory of edge detection. *Proc. R. Soc. Lond., B, Biol. Sci.*, 207(1167):187–217.

Martin, D., Fowlkes, C., Tal, D., and Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. Int. Conf. Comput. Vision*, volume 2, pages 416–23.

Martin, D. R., Fowlkes, C. C., and Malik, J. (2004). Learning to detect natural image boundaries using local brightness, color and tex- ture cues. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(5):530–49.

Morrone, M. C. and Burr, D. C. (1988). Feature detection in human vision: a phase-dependent energy model. *Proc. R. Soc. Lond., B, Biol. Sci.*, 235(1280):221–45.

Mundhenk, T. N. and Itti, L. (2005). Computational modeling and exploration of contour integration for visual saliency. *Biol. Cybern.*, 93:188–212.

Murray, J. F. and Kreutz-Delgado, K. (2006). Learning sparse overcomplete codes for images. *Journal of VLSI Signal Processing*, 45:97–110.

Ng, J., Bharath, A. A., and Zhaoping, L. (2007). A survey of architecture and function of the primary visual cortex (V1). *EURASIP Journal on Advances in Signal Processing*, 2007:97961.

Olshausen, B. A. and Field, D. J. (1996). Emergence of simple-cell receptive properties by learning sparse code for natural images. *Nature*, 381:607–9.

Olshausen, B. A. and Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Res.*, 37(23):3311–25.

Papari, G., Campisi, P., Petkov, N., and Neri, A. (2007). A biologically motivated multiresolution approach to contour detection. *EURASIP Journal on Advanced Signal Processing*, 2007:71828.

Papari, G. and Petkov, N. (2008). Adaptive pseudo dilation for gestalt edge grouping and contour detection. *IEEE Trans. Image Proc.*, 17(10):1950–62.

Papari, G. and Petkov, N. (2011). An improved model for surround suppression by steerable filters aand multilevel inhibition with application to contour detection. *Pattern Recognit.*, 44:1999–2007.

Parent, P. and Zucker, S. (1989). Trace inference, curvature consistency, and curve detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(8):823–39.

Pece, A. E. C. and Petkov, N. (2000). Fast atomic decomposition by the inhibition method. In *Proc. Int. Conf. Pattern Recognit.*, pages 215–8.

Perrinet, L. U. (2010). Role of homeostasis in learning sparse representations. *Neural Comput.*, 22(7):1812–36.

Petkov, N. and Westenberg, M. A. (2003). Suppression of contour perception by band-limited noise and its relation to non-classical receptive field inhibition. *Biol. Cybern.*, 88:236–46.

Ramirez, I., Sprechmann, P., and Sapiro, G. (2010). Classification and clustering via dictionary learning with structured incoherence and shared features. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 3501–8.

Ranzato, M., Poultney, C., Chopra, S., and LeCun, Y. (2007). Efficient learning of sparse representations with an energy-based model. In Schölkopf, B., Platt, J., and Hoffman, T., editors, *Advs. Neural Info. Proc. Sys.*, volume 19, pages 1137–44, Cambridge, MA. MIT Press.

Reggia, J. A. (1985). Virtual lateral inhibition in parallel activation models of associative memory. In *Proc. Int. Joint Conf. on Artif. Intell.*, volume 1, pages 244–8.

Rehn, M. and Sommer, F. T. (2007). A network that uses few active neurons to code visual input predicts the diverse shapes of cortical receptive fields. *J. Comput. Neurosci.*, 22:135–46.

Ren, X. (2008). Multi-scale improves boundary detection in natural images. In *Proc. European Conf. Comput. Vision*.

Ren, X., Fowlkes, C. C., and Malik, J. (2008). Learning probabilistic models for contour completion in natural images. *Int. J. Comput. Vis.*, 77:47–63.

Rodieck, R. W. (1965). Quantitative analysis of cat retinal ganglion cell response to visula stimulation. *Vision Res.*, 5:583–601.

Rosenfeld, A., Hummel, R. A., and Zucker, S. W. (1976). Scene labeling by relaxation operations. *IEEE Trans.Sys. Man Cybern.*, 6(6):420–33.

Ruzon, M. A. and Tomasi, C. (2001). Edge, junction, and corner detection using color distributions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11):1281–95.

Sigman, M., Cecchi, G. A., Gilbert, C. D., and Magnasco, M. O. (2001). On a common circle: natural scenes and gestalt rules. *Proc. Natl. Acad. Sci. U.S.A.*, 98(4):1935–40.

Smith, S. M. and Brady, J. M. (1995). Susan: a new approach to low-level image-processing. *Int. J. Comput. Vis.*, 23:45–78.

Solbakken, L. and Junge, S. (2011). Online parts-based feature discovery using competitive activation neural networks. In *Proc. Int. Joint Conf. on Neural Netw.*, pages 1466–73.

Spratling, M. W. (2010). Predictive coding as a model of response properties in cortical area V1. *J. Neurosci.*, 30(9):3531–43.

Spratling, M. W. (2011). A single functional model accounts for the distinct properties of suppression in cortical area V1. *Vision Res.*, 51(6):563–76.

Spratling, M. W. (2012a). Predictive coding accounts for V1 response properties recorded using reverse correlation. *Biol. Cybern.*, 106(1):37–49.

Spratling, M. W. (2012b). Unsupervised learning of generative and discriminative weights encoding elementary image components in a predictive coding model of cortical function. *Neural Comput.*, 24(1):60–103.

Spratling, M. W., De Meyer, K., and Kompass, R. (2009). Unsupervised learning of overlapping image components using divisive input modulation. *Comput. Intell. Neurosci.*, 2009(381457):1–19.

Tang, Q., Sang, N., and Zhang, T. (2007a). Contour detection based on contextual influences. *Image Vis. Comput.*, 40(11):3100–09.

Tang, Q., Sang, N., and Zhang, T. (2007b). Extraction of salient contours from cluttered scenes. *Pattern Recognit.*, 40(11):3100–09.

Ursino, M. and La Cara, G. E. (2004). A model of contextual interactions and contour detection in primary visual cortex. *Neural Netw.*, 17(5–6):719–35.

Van Essen, D., Newsome, W. T., and Maunsell, J. H. (1984). The visual field representation in striate cortex of the macaque monkey: asymmetries, anisotropies, and individual variability. *Vision Res.*, 24(5):429–48.

Van Hateren, J. H. and van der Schaaf, A. (1998). Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc. R. Soc. B*, 265:359–66.

Vecera, S. P. and O'Reilly, R. C. (1998). Figure-ground organization and object recognition processes: An interactive account. *J. of Expt. Psyc.: Human Percept. & Perform.*, 24:441–62.

Vonikakis, V., Gasteratos, A., and Andreadis, I. (2006). Enhancement of perceptually salient contours using a parallel artificial cortical network. *Biol. Cybern.*, 94(3):192–214.

Weber, C. and Triesch, J. (2008). A sparse generative model of V1 simple cells with intrinsic plasticity. *Neural Comput.*, 20:1261–84.

Wiltschut, J. and Hamker, F. H. (2009). Efficient coding correlates with spatial frequency tuning in a model of V1 receptive field organization. *Vis. Neurosci.*, 26:21–34.

Wright, J., Ma, Y., Mairal, J., Sapiro, G., Huang, T., and Yan, S. (2009). Sparse representation for computer vision and pattern recognition. *Proc. IEEE*, 98(6):1031–44.

Young, R. A. and Lesperance, R. M. (2001). The gaussian derivative model for spatial-temporal vision: II cortical data. *Spatial Vis.*, 14:321–89.

Zeng, C., Li, Y., , Yang, K., and Li, C. (2011a). Contour detection based on non-classical receptive field model with butterfly-shaped inhibition subregions. *Neurocomputing*, 74:1527–34.

Zeng, C., Li, Y., and Li, C. (2011b). Center-surround interations with adaptive inhibition: a computational model for contour detection. *NeuroImage*, 55:49–66.

Zhanga, H., Fritts, J. E., and Goldman, S. A. (2008). Image segmentation evaluation: A survey of unsupervised methods. *Comput. Vis. Image Underst.*, 110(2):260–80.

Zhaoping, L. (2005). Border ownership from intracortical interactions in visual area V2. *Neuron*, 47:143–53.