

# Department of Computer Science, University of Otago

UNIVERSITY  
of  
OTAGO



*Te Whare Wānanga o Ōtāgo*

---

Technical Report OUCS-2013-09

## **A neural network model of visual attention and object classification: technical details**

Authors:

**Hayden Walles, Anthony Robins and Alistair Knott**

Department of Computer Science, University of Otago, New Zealand



Department of Computer Science,  
University of Otago, PO Box 56, Dunedin, Otago, New Zealand

<http://www.cs.otago.ac.nz/research/techreports.php>

# A neural network model of visual attention and object classification: technical details

Hayden Walles, Anthony Robins and Alistair Knott  
Department of Computer Science, University of Otago, New Zealand

September 12, 2013

## Abstract

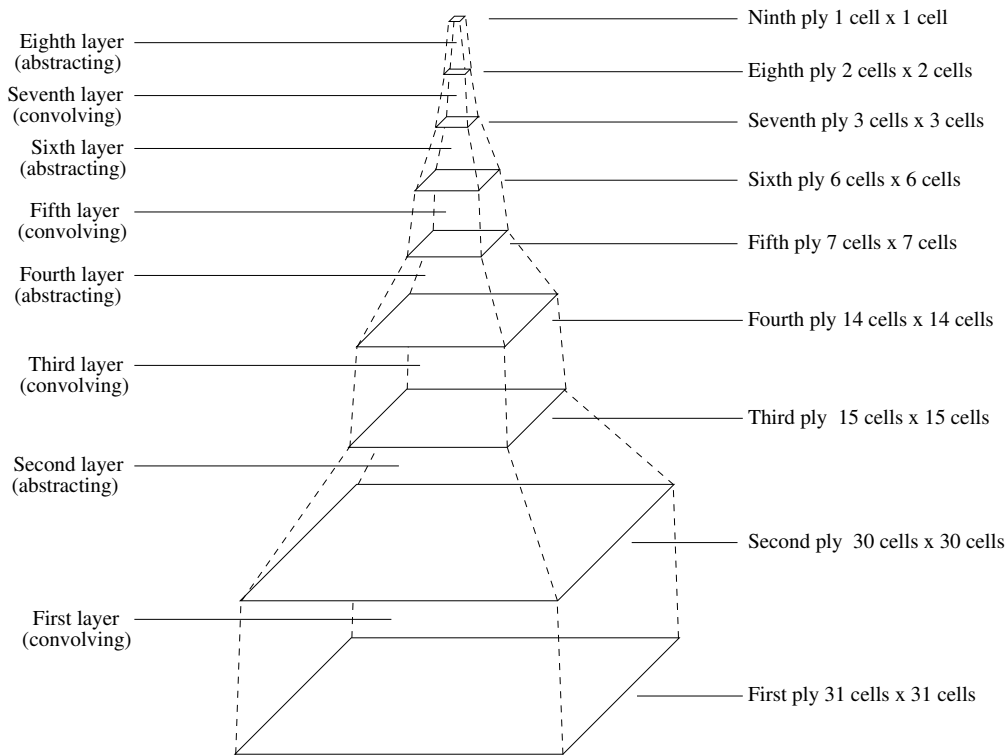
This document provides technical details of our neural network model of visual object classification and attention, and of experiments using the system in a visual search task.

## 1 Overview

The model of the classification and attentional subsystems can be thought of as a collection of retinotopic **map** representations. We implement a map as a matrix. The input map  $\mathbf{I}$  is a greyscale image measuring  $128 \times 128$  pixels, with element values in the range 0 to 255. The input is read directly from bitmap image files. Other maps are computed from this using a variety of operations. Most of these maps also measure  $128 \times 128$  pixels with the exception of some employed by the classifier, as noted in Section §2. Except where noted it is safe to assume that the output of a map operation has the same dimensions as its inputs. Because of this we sometimes refer to pixels in maps other than  $\mathbf{I}$  even though they don't, strictly speaking, form an image.

The map operations used are convolution (matrix convolution, denoted  $*$  with pixels lying beyond the map edge assumed to be white unless noted otherwise), addition, subtraction and scalar multiplication (computed as for their matrix equivalents), modulus (denoted  $|\mathbf{X}|$ , computed by taking the modulus of each element) and some more complicated operations which will be defined where they occur. The matrix or map element at the  $i$ th row and  $j$ th column of  $\mathbf{X}$  is denoted  $\mathbf{X}_{i,j}$ .

Some special maps have additional information associated with them, such as regions. Regions are sets of contiguous pixels in a map and we implemented these as either maps with characteristic pixel values for each region or as sets of maps, one per region, depending on which was more convenient. The map itself can still be considered just a matrix, with this extra information represented separately and bound to the map.



**Figure 1** The general structure of our convolutional neural network. Plies of cells are connected by layers of weights. Each cell contains one unit for every feature represented by that ply.

## 2 The classifier

### 2.1 Classifier structure

The classifier used was a convolutional neural network which takes a set of input maps and activates a set of output category units via a series of layered plies which alternately combine visual features from the ply below into more complex features and abstract over the spatial location of visual features. The CNN was mostly as described in Wallis *et al.* (2008), except that the number of features used in each ply was different and there was some additional input preprocessing.

Figure 1 illustrates the overall structure of our CNN.

The **units** of the network are arranged in a series of **plies**, with units in each ply connected to units in the one above by a **layer of weights**. Our network used had nine plies and eight layers. Units within each ply are clustered into **cells**, which are arranged retinotopically. Every cell in a particular ply contains the same number of units, one for each feature that the ply represents. Each unit in a cell represents the strength of its associated feature at the cell's location, and so each cell in a ply represents in parallel the presence of a set of features at the corresponding location in the input field. The successive plies of our network (going from input to output, measured in terms of cells)  $31 \times 31$ ,  $30 \times 30$ ,  $15 \times 15$ ,  $14 \times 14$ ,  $7 \times 7$ ,  $6 \times 6$ ,  $3 \times 3$ ,  $2 \times 2$  and  $1 \times 1$ .

The features in the first (input) ply were divided into two groups. One feature was provided

for high-frequency input which represented luminance directly. Four features represented low-frequency input: these were obtained with  $9 \times 9$  convolution filters tuned to horizontal, vertical and diagonal black-on-white lines.

Units receive input from a small square region of the ply beneath, the **integration window**, meaning they are connected locally and can only make use of local features. we used a window measuring  $2 \times 2$  cells in all layers. All units in a cell have the same window, which can thus also be called the cell's window. The region of the retina that contributes to a unit's input is its **receptive field**. In addition the weights for corresponding units in different cells of a ply are constrained to be the same, effectively sharing the weights. This means that the response to activity inside a cell's window will be the same irrespective of where in a ply the cell is located.

Successive plies divide the visual field more and more coarsely, so contain fewer cells than their predecessors, each of which has a wider receptive field than those in earlier plies. However later plies generally represent more features than earlier plies, and therefore contain more units per cell.

The function and structure of the weight layers alternates throughout the network between convolution and abstraction. Convolving layers compute combinations of features in the previous ply with little change in the number of cells between plies, while abstracting layers reduce the number of cells of the input ply without interaction between different features.

In convolving layers, an output unit receives input from every unit within its  $2 \times 2$  integration window. A unit receiving input from a ply representing  $n$  features will have  $4n + 1$  inputs (including a bias).

Weights in abstracting layers are simpler. Input and output plies contain the same number of features and there is no interaction between features. A unit receiving input from its  $2 \times 2$  window will have 5 inputs (including a bias). The window of a cell in the output ply precisely abuts but does not overlap with the windows of neighbouring cells. The effect is that the integration windows of cells in the output ply tile the input ply. Weights are shared even further within abstracting layers, with all weights for a feature constrained to be identical. This means that each abstracting layer really has only two variable parameters per feature: one weight shared among all the inputs units, and the bias.

Apart from the varying structure of the layers, unit activation is computed in the same way throughout the network. For a unit with  $n$  inputs  $p_1 \dots p_n$  (excluding the bias) and  $n + 1$  weights (including the bias)  $w_1 \dots w_{n+1}$  the unit's **activation**, a weighted sum,  $\sigma$  is computed:

$$\sigma = \sum_{i=1}^n p_i w_i + w_{n+1}$$

which for an abstracting unit can be simplified further to:

$$\sigma = w_{ply} \sum_{i=1}^n p_i + w_{bias}$$

because of weight sharing.

The output of the unit is then computed via the logistic function:

$$f = \frac{1}{1 + e^{-\sigma}}$$

This is conventional for feed-forward networks.

Going from the input ply to the output ply the number of features in each ply were 5, 25, 25, 32, 32, 32, 32, 7 and 7.

Although inputs to the system as a whole measure  $128 \times 128$  pixels, inputs to the classifier always measure  $31 \times 31$  pixels as in our original design. This is a practical limitation of the classifier to allow training in reasonable time and the disparity is resolved by always centring the attended region in the classifier’s input for classification purposes. This ensures that the bounding rectangle of the attended region is centred in the classifier’s input.

## 2.2 Training regime





The network was trained using the RPROP algorithm (Riedmiller, 1994). This is a variation of the BACKPROP algorithm (Rumelhart *et al.*, 1986). The training algorithm is described in more detail in Walles *et al.* (2008).

We trained with small (high-frequency) shapes, each presented at a randomly chosen third of possible retinal locations. We also trained with large (low-frequency) shapes each at a random third of all possible retinal locations for each of four densities. These included solid shapes as well as large shapes with pixels randomly ablated to the background colour with probabilities  $\frac{1}{6}$ ,  $\frac{1}{3}$  or  $\frac{1}{2}$ . Thus for the low-frequency training total spatial coverage was likely. The small shapes were presented at the high-frequency inputs only and the large shapes at the low-frequency inputs only. During operation only one of the sets of inputs is used at a time, the other being suppressed entirely. There were 1566 high-frequency training examples and 2152 low-frequency training examples. As in Walles *et al.* (2008) these included 371 noise examples which were each fed to the low- and high-frequency inputs in turn. New noise examples were generated on each cycle of training. In other respects the architecture and training of the CNN was as described in Walles *et al.* (2008).

## 3 Parallel attention component: saliency analysis

Saliency analysis is based on the model presented by Itti and Koch (2000) and Walther and Koch (2006), modified to fit the size constraints of the classifier and support scale-based attention in the selection mechanism.

The saliency analysis module parses the visual field and produces a saliency representation: a saliency map, identifying a number of salient regions. Each region contains either a single shape, or a set of shapes which are treated as a single item (i.e. grouped) due to their proximity and/or similarity. Salient regions are identified by computing partial saliency maps from the input. Two **local contrast maps** are computed, using Laplacian of Gaussian (LoG) filters tuned to two different spatial frequencies. A single **texture homogeneity map** at the higher spatial frequency

Visual input	□×□×	□ □ × ×	□ × □ ×	× × × ×
Salient regions identified				
	(a)	(b)	(c)	(d)

**Figure 2** Salient regions identified by the system for four sample visual inputs. Different shades of grey indicate different salient regions.

is computed using a statistical histogram-based method Liu and Wang (2000). The two local contrast maps and the homogeneity map are combined to produce the saliency representation.

Prima facie, there may seem to be a conflict between local contrast and homogeneity as indicators of saliency. Saliency computed from contrast, and saliency computed from homogeneity (which implies *no* contrast) seem inconsistent with each other. However, while the principles may be in conflict at a single spatial scale, we suggest that they are complementary at different spatial scales. In our model, maximally salient stimuli are those which contrast from their background at a spatial frequency commensurate with their size, and which are *composed* of uniform texture elements. Thus uniformity is required at a higher spatial frequency than contrast.

Some illustrations of the regions found by the saliency analysis module are given in Figure 2. These demonstrate the module’s ability to identify salient regions of different sizes: the input stimuli in Figure 2(a) are grouped into a single large region, those in Figure 2(b) are grouped into two medium-sized regions and those in Figures 2(c) and 2(d) are identified as four small regions. They also show how the module reconciles conflicting local contrast and homogeneity cues to salience. If items are close enough (2(a)) then grouping can occur even among heterogeneous items. At an intermediate separation, grouping is determined by homogeneity: homogeneous stimuli are grouped (2(b)) and heterogeneous stimuli are not (2(c)). Finally, if items are separated widely enough, they are not grouped even if they are homogeneous (2(d)).

The contributions of the local contrast and homogeneity maps to overall salience are determined by the weights of two parameters, whose relative value determines the separation at which homogeneous stimuli are grouped. These parameter settings can be related to individual variations in grouping behaviour found in experiments on human subjects. Quinlan and Wilton Quinlan and Wilton (1998) explored the interaction of the Gestalt properties of similarity and proximity in humans. They found that proximity always dominates similarity if stimuli are sufficiently close, but that the distance at which this happens varies from subject to subject. The ratio between contrast and homogeneity weights in the computation of salience directly models this parameter of variation between subjects.

### 3.1 Local contrast

Local contrast computation begins by taking the input image (a luminance image) and stretching the values into the range  $-128$  (black) to  $127$  (white).

$$\mathbf{I}' = \text{stretch}(\mathbf{I}, -128, 127), \text{ where} \quad (1)$$

$$\text{stretch}(\mathbf{X}, L, U)_{i,j} = \frac{(\mathbf{X}_{i,j} - \min(\mathbf{X}))(U - L)}{\max(\mathbf{X}) - \min(\mathbf{X})} + L \quad (2)$$

and  $\min$  and  $\max$  are functions that produce the minimum and maximum element values, respectively, of a matrix or map. Local contrast is then computed by convolving with two normalised Laplacian of Gaussian filters, one for each spatial frequency ( $\sigma = 1$  and  $\sigma = 15$ , chosen by trial and error to produce strong response to shapes of the relevant scale while trying to minimise response to shapes at the other scale). The absolute value of these results is then taken. Given

$$\text{LoG}(\sigma)_{i,j} = \left(1 - \frac{r^2}{\sigma^2}\right)e^{-\frac{r^2}{\sigma^2}}, \text{ where} \quad (3)$$

$$0 \leq i, j < 5\sigma$$

$$o = \lfloor \frac{5\sigma}{2} \rfloor$$

$$r^2 = (i - o)^2 + (j - o)^2 \quad (4)$$

and normalisation was achieved using

$$\text{norm}(\mathbf{X})_{i,j} = \frac{\mathbf{X}_{i,j}}{s}, \text{ where} \quad (5)$$

$$s = \sum_i \sum_j |X_{i,j}|$$

we compute the high-frequency local contrast  $\mathbf{C}_{\text{hi}}$  and low-frequency local contrast  $\mathbf{C}_{\text{lo}}$  using

$$\mathbf{C}_{\text{hi}} = |\text{norm}(\text{LoG}(\sigma = 1)) * \mathbf{I}'| \quad (6)$$

$$\mathbf{C}_{\text{lo}} = |\text{norm}(\text{LoG}(\sigma = 15)) * \mathbf{I}'| \quad (7)$$

We use LoG filters here rather than the orientation-specific filters used in the classifier for two reasons. First, the orientation-specific filters used in the classifier grew out of the existing orientation-specific features used by Mozer and Sitton (1998), which our classifier is based on. Second, while one of the purposes of filtering the classifier inputs is to provide directed information (orientation) to aid classification, here we are only interested in contrast of suitably-sized shapes whatever their orientation. Having said that, it would be desirable in future to find a way to use the classifier's filters to produce these contrast maps instead of the LoG.

### 3.2 Homogeneity

The similarity measure is computed by the procedure described in Liu and Wang (2000). This procedure samples a small  $7 \times 7$  pixel region around each pixel in the input image, computing its spectral histogram which can be thought of as a high dimensional feature vector, and finally finds the closest match to this histogram among those belonging to a set of texture templates derived from images of the small shapes used in the experiment both closely packed and sparsely scattered.

The spectral histogram is constructed by first convolving the  $7 \times 7$  window with each of seven normalised filter matrices. The first three are the Kronecker  $\delta$  filter, which constitutes an identity operation in this instance and the  $D_{xx}$  and  $D_{yy}$  filters:

$$\delta = [ 1 ] \quad (8)$$

$$D_{xx} = [ -1 \quad 2 \quad -1 ] \quad (9)$$

$$D_{yy} = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} \quad (10)$$

There are also two Laplacian of Gaussian filters (see Equation (3))  $LoG(\sigma = 1)$  and  $LoG(\sigma = 2)$ .

Finally there are three Gabor filters  $G(\sigma = 2, \theta = \frac{\pi}{6})$ ,  $G(\sigma = 2, \theta = \frac{\pi}{2})$  and  $G(\sigma = 2, \theta = \frac{5\pi}{6})$  where

$$\begin{aligned} G(\sigma, \theta)_{i,j} &= e^{\frac{-1}{2\sigma^2 r}} \cos\left(\frac{-2\pi}{\sigma}((j-o)\cos\theta \right. \\ &\quad \left. +(i-o)\sin\theta)\right) \\ 0 &\leq i, j < w = \lfloor \frac{8\sigma}{\sqrt{2}} \rfloor \\ o &= \lfloor \frac{w}{2} \rfloor \\ r &= ((j-o)\cos\theta + (i-o)\sin\theta)^2 \\ &\quad + ((o-j)\sin\theta + (i-o)\cos\theta)^2 \end{aligned} \quad (11)$$

The filter matrices are each normalised with the *norm* function given in Equation (5). Their choice is justified by Liu and Wang (2000). The window is convolved with each normalised filter with pixels at the edge of the map replicated to infinity to ensure a result for every pixel in the input. The histograms of the resulting maps (with unit-sized bins) are concatenated to produce the spectral histogram. Spectral histograms are compared using the  $\chi^2$  value. If  $H_1$  and  $H_2$  are two spectral histograms, and  $H(i)$  is the  $i$ th element of the histogram  $H$  then this is computed as follows.

$$\chi^2 = \sum_i \frac{(H_1(i) - H_2(i))^2}{H_1(i) + H_2(i)} \quad (12)$$



For each pixel’s associated histogram, the template histogram which has the lowest  $\chi^2$  value relative to it determines the category assigned to the pixel.

Once each pixel is assigned a category (square, ell, etc. or background), boundaries are determined by comparing each pixel with its four-neighbours. The four-neighbours of a pixel at coordinates  $(i, j)$  are the pixels at coordinates  $(i - 1, j)$ ,  $(i + 1, j)$ ,  $(i, j - 1)$  and  $(i, j + 1)$ . Whenever a pair of pixels differs in category, the pixel that was least certainly classified (measured by the  $\chi^2$  of its histogram relative to its category’s template) is marked as a texture boundary. In the resulting boundary map  $\mathbf{B}$  homogeneous regions are marked with zero, boundaries with one.

For the experiments presented here we wanted some stimuli to be considered similar enough for saliency analysis to group them even though they were distinct. To this end we defined that boundaries between ells and squares, crosses and arrows, arrows and arms, arrows and triangles and triangles and arms would not be marked in the boundary map.

This confusion of types was based on the confusion patterns of the CNN but is effectively arbitrary. It is intended to model the Gestalt principle of similarity between types. We would have preferred to model this confusion using comparisons between the histograms of neighbouring pixels directly but the small size of the retina made this impractical (we consider this to be just an implementation detail).

### 3.3 Partial saliency maps

The boundary map  $\mathbf{B}$  is combined with the low-frequency local contrast map  $\mathbf{C}_{lo}$  by a weighted sum and thresholded to produce the low-frequency saliency map  $\mathbf{S}_{lo}$ :

$$\mathbf{S}_{lo} = H(\tau_{lo}; \alpha \mathbf{C}_{lo} - \beta \mathbf{B}) \quad (13)$$

where

$$H(\tau; \mathbf{X})_{i,j} = \begin{cases} 1 & \text{if } \mathbf{X}_{i,j} > \tau \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

$$\tau_{lo} = 0.060 \quad (15)$$

$$\alpha = 15.5 \quad (16)$$

$$\beta = 1.45 \quad (17)$$

These scalings were chosen by trial and error so that contrast and homogeneity would interact without one dominating all the time. The high-frequency saliency map  $\mathbf{S}_{hi}$  is just the same as high-frequency local contrast map, thresholded.

$$\mathbf{S}_{hi} = H(\tau_{hi}; \mathbf{C}_{hi}) \quad (18)$$

where

$$\tau_{hi} = 0.4 \quad (19)$$

The threshold values were chosen by trial and error so that regions of both frequencies at a reasonable contrast would become salient.

Contrast weight ( $\alpha$ )	Homogeneity weight ( $\beta$ )	Max. separation between grouped hetero stimuli	Min. separation between separate homogen. stimuli
15.5	0.8	3	4
15.5	1.45	1	3
15.5	2.5	1	1

**Table 1** Effect of changing contrast and homogeneity weights in grouping behaviour.

The ratio between contrast and homogeneity weights ( $\alpha$  and  $\beta$  in Equation (13)) determines the relative contributions of contrast and homogeneity to overall salience. Table 1 shows the effect on grouping behaviour of changing  $\beta$  while keeping  $\alpha$  constant. Column 3 shows the maximum separation between heterogeneous stimuli for which they are grouped together, and Column 4 shows the minimum separation between homogeneous stimuli for which they are treated as separate regions, for a range of different weight ratios. Distances are measured in pixels. The second column corresponds to the parameter values used in the experiments in the current paper.

### 3.4 Combination of partial saliency maps

Regions which are four-neighbour contiguous are next identified and labelled by region merging (Gonzalez and Woods, 1992, see Section §3.2 for a definition of four-neighbouring pixels). Any labelled region in the low-frequency map containing fewer than 55 pixels is discarded, yielding a new low-frequency saliency map  $S'_{lo}$  which is used for further operations:

$$S'_{lo} = F(S_{lo}) \quad (20)$$

where  $F$  is a function that just sets pixels belonging to such regions in the input to zero.

This was done to remove high-frequency objects strong enough to stimulate the low-frequency saliency map as well as occasional artifacts between objects, both of which we consider to be noise. It acts as a kind of low-pass filter, removing regions too small to be of interest to the low-frequency map. The point-wise sum of these maps yields the master saliency map in which contiguous regions are also identified and labelled.

$$S = S'_{lo} + S_{hi} \quad (21)$$

### 3.5 Inhibition and suppression

The preceding operations have been all bottom-up, but further computation relies on some top-down influence in the form of *inhibition*. A map is inhibited by combining a top-down *inhibition map* with its bottom-up activation. It can be thought of as an additional factor in the computation of the map. If  $\mathbf{X}$  is a map and  $\mathbf{X}^I$  its corresponding inhibition map then the inhibited version of the map  $\mathbf{X}^I$  (its effective value, used by operations which depend on the map) is given by

$$\mathbf{X}'_{i,j} = \begin{cases} \mathbf{X}_{i,j} & \text{if } \mathbf{X}_{i,j}^I = 0 \\ 0 & \text{otherwise} \end{cases} \quad (22)$$

It is also possible to inhibit an entire map at once, equivalent to inhibiting with a map containing no zero elements.

In this paper we use the term **suppression** where inhibition is only temporary as part of a computation. Where applicable inhibition and suppression are governed by independent inhibition maps associated with the primary map.

### 3.6 Computation of salient regions

The final stage of saliency analysis is the extraction of a well-defined set of salient regions, each tagged with a default classification scale based on its size. In our implementation these are represented in a series of maps, one for each salient region — though a single map with an appropriate coding could be used because the regions do not overlap. First, any of the most strongly activated pixels in the master saliency map is chosen (we used the left- and top-most such point but this is arbitrary). If there is a low-frequency salient region at that point, the low frequency is selected as the salient scale, otherwise the high frequency is selected. Standard morphological dilation (Gonzalez and Woods, 1992) is then applied to the corresponding region (radius 2 pixels for high frequency, 4 pixels for low frequency). Finally, pixels are removed from the region if they overlap salient regions that have already been computed, other active pixels in the master saliency map or pixels inhibited by attention operations (for which the associated inhibition map will be active).

The resulting region is added to the set of salient regions, tagged with its associated scale. The region is suppressed in the corresponding scale saliency map and the master saliency map and any overlapping regions in the non-selected scale saliency map are also suppressed. The above process is repeated until all activity in the master saliency map has been suppressed.

Once the set of salient regions is computed, one is chosen at random by the selection mechanism and the associated region and scale become the subjects of attention. We do not select salient regions by decreasing order of saliency, as is typically done, because our stimuli are very simple and the standard measure of “degree of saliency” doesn’t really apply. The randomisation of selection can be viewed as the addition of noise to simulate the variation of saliency found in real-world stimuli.

After the winner is selected, suppression of the saliency maps introduced during computation of salient regions is then removed. Salient regions are recomputed whenever there is a change to the maps that the computation depends on, which happens when the selection mechanism inhibits the saliency maps.

## 4 Serial attention component: the selection mechanism

The saliency representation just described provides input to the serial attention component of our model, whose role is to selectively deliver information from the retina to the classification

subsystem. Selection occurs in two different attentional media. One is spatial location: the classifier can be restricted to receive input only from a particular region of the visual field. The other is classification scale: the classifier can be restricted to receive input from visual features at a particular spatial frequency (high or low).

Processing in the serial attention component takes the form of a sequence of **attentional operations**. There are two operations. One is the selection of a new salient region to attend to. When this happens, an appropriate classification scale is also automatically selected, namely the default classification scale for the selected region. The other is the selection of a new classification scale, *without* a change in the currently attended region. Each attentional operation has lasting side-effects on the master saliency representation. Selection of a new salient region involves inhibiting the currently selected salient region (if there is one), a process analogous to the spatial inhibition-of-return found in humans Posner (1980). Selection of a new classification scale also involves inhibition, namely inhibition of the currently selected classification scale.

When a display is presented to the system, the first attentional operation is the selection of a random salient region in the master saliency representation, and the classification of this region at its default classification scale. The region’s default classification scale is a function of its size: if it is large, the coarse-grained (low-frequency) classification scale is the default; if it is small, the fine-grained (high-frequency) scale is the default. If the region is large, it is then re-analysed at the fine-grained classification scale, after which a new salient region is selected, by inhibiting the currently selected region and picking a new one. If the region is small, it is not re-analysed at a finer classification scale, since the model only features two spatial scales; instead, a new salient region is selected immediately. This cycle continues until all the salient regions in the original stimulus have been selected and inhibited.

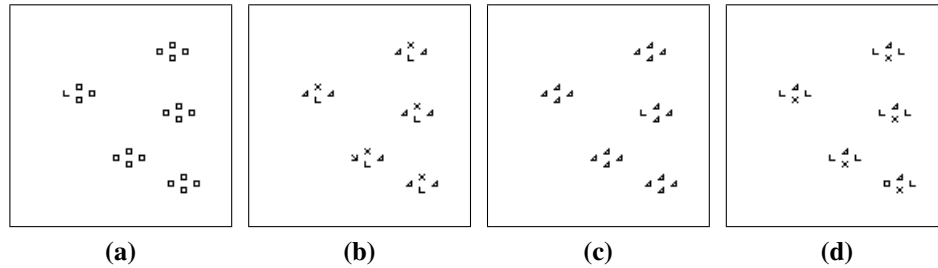
The reanalysis of a large salient region at a finer classification scale allows the classification of a group of objects occupying this region, if it is homogeneous. For instance in Figure 2(b), two large salient regions are selected, each containing a homogeneous group of two objects. When these groups are classified at the default scale, the classifier will not return a result, because there is no global form to be detected. But when they are classified at a finer-than-default scale, the classifier will return their local form: ‘square’ for the left-hand region, and ‘cross’ for the right-hand one. By using homogeneity as a cue to the formation of salient regions, and by allowing salient regions to be reanalysed at a finer classification scale, the attentional system naturally exploits the classifier’s ability to operate on homogeneous groups.

## 4.1 Attentional gating operations

Serial attentional operations are implemented as **inhibition** and **gating** operations. We have already described inhibition in Section §3.5. In this section we describe the gating operation.

Where inhibition inhibits a map in place, gating inhibits the elements of a map as they feed into another operation. A map  $\mathbf{Y}$  gates another map  $\mathbf{X}$  with the result given by the *gate* function:

$$gate(\mathbf{X}, \mathbf{Y})_{i,j} = \begin{cases} \mathbf{X}_{i,j} & \text{if } \mathbf{Y}_{i,j} \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (23)$$



**Figure 3** Example displays for each search condition. (a) t-d similar, d-d similar (target ell); (b) t-d similar, d-d different (target arrow); (c) t-d different, d-d similar (target ell); (d) t-d different, d-d different (target square).

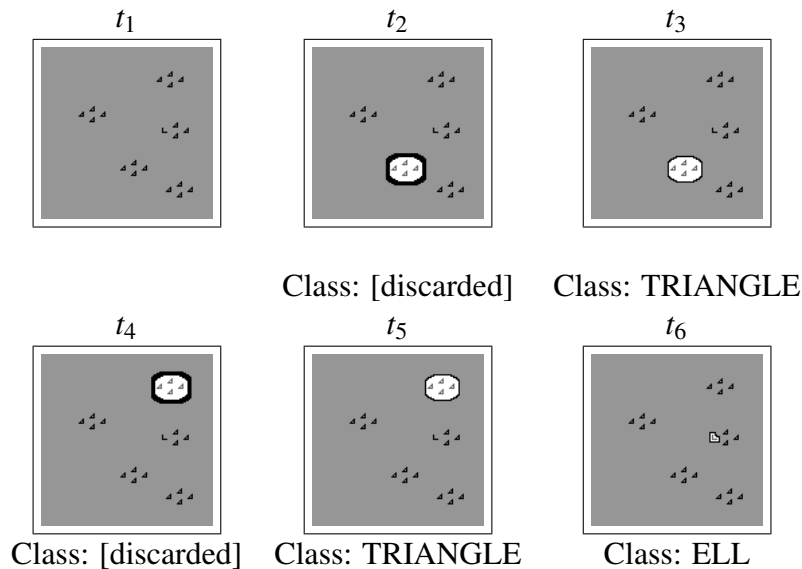
In addition to this spatial gating there is scale gating which is achieved by entirely gating a scale-specific set of classifier input maps (equivalent to gating the maps with a map containing only zero elements). When the low frequency is selected, the high frequency maps are entirely gated off and vice-versa.

## 5 Performance of the system in a visual search task

### 5.1 Defining four search tasks

To test the search performance of our model, we created four different search tasks, defined by varying two independent binary parameters based on those used by Duncan and Humphreys: target-distractor similarity (with values ‘t-d similar’ and ‘td-different’) and distractor-distractor similarity (with values ‘d-d similar’ and ‘d-d different’).

Our system was required to find a single target in displays of varying size (the target was always present). The displays were carefully set up to produce the desired condition, while controlling for other variables, similar to the strategy used by Duncan and Humphreys. Our displays are less controlled than those used with humans because some of the variables that must be controlled in human subjects need not be in our model. We do not have to guard against inter-trial priming, for example. Figure 3 shows example displays for the four different search conditions. In t-d similar conditions ((a) and (b)), distractors adjacent to the target are similar to it, and tend to be grouped with it; in t-d different conditions ((c) and (d)), the distractors adjacent to the target are different to it, and tend not to be grouped with it. In d-d similar conditions ((a) and (d)), adjacent distractors are similar to another, and tend to be grouped; in d-d different conditions ((b) and (c)), adjacent distractors are different from one another, and tend not to be grouped. For the purposes of perceptual grouping, similarity is arbitrarily defined in the saliency analysis phase. Ells are similar to squares, crosses to arrows, arrows to arms, arrows to triangles and triangles to arms. As shown in Figure 3, search displays were all based on the same spatial configuration: twenty items were arranged in five groups of four, in locations appearing at fixed positions in the visual field. In each condition a full display with twenty items was created. One group was chosen randomly and the target placed at the left side of it. Next fifteen search trials



**Figure 4** An example sequence of operations during simple search. At  $t_1$  the input is presented and at subsequent time steps attention is directed as shown until the target (ell) is found. Thick borders around a region indicate attention to the low spatial frequency, thin borders attention to the high spatial frequency.

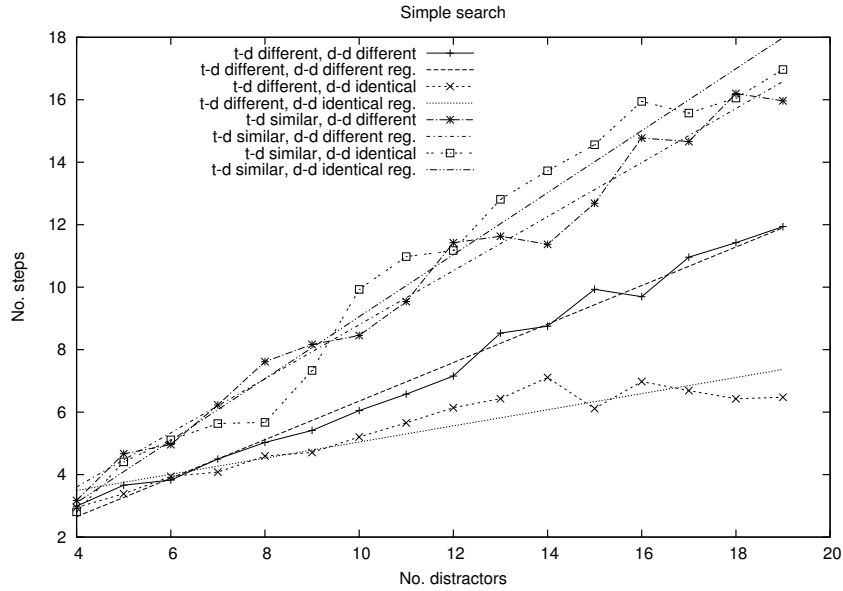
were run. First the original display was run. Then one item was removed randomly with the constraints that at least one item remained in each group (the leftmost item) and the items in each group remained contiguous. Now this case was run and the process repeated until only one item remained in each group.

Each trial display was passed to the system, which was configured for search as follows. The attention system directs the sequence of operations on the salient regions. At each step attention is directed at the selected region. If the region is small, it is classified at the fine-grained classification scale. If the region is large, it is first classified at the coarse-grained classification scale and then at the fine-grained scale, but the former result is discarded, as the search is for small shapes. If the classifier returns a category at the fine-grained classification scale, and this matches the search target, search is complete and the attention system halts. Otherwise the region is inhibited and selection begins again. If the system runs out of candidates for selection then all inhibition is removed, the low-frequency saliency map is entirely suppressed and selection begins anew: in this second pass, each small shape is identified as a separate salient region.

For each trial display the number of selection steps needed to locate the target was recorded. Figure 4 shows the steps taken by the system during one trial.

## 5.2 Results

For each of the four search conditions the mean number of steps taken by the system to find the target was plotted against the number of distractor items present, to create four search graphs, and the slope of each graph was determined by linear regression (see Figure 5). The slope of



**Figure 5** Search slopes for each search condition. Regression lines are also shown.

each graph was significantly different to zero ( $p < 3.2 \times 10^{-12}$ ).  $t$ -tests were used to compare the slopes of each pair of graphs; in each case the null hypothesis that slopes are equal was rejected, with  $p$ -values ranging from 0.0039 to 0. This result shows that different search conditions have a range of different slopes: in our model there is no dichotomy between flat pop-out search and slow serial search. Instead, there are a range of gradients for different search conditions. Our simulation reproduces Duncan and Humphreys' main experimental results: when targets are dissimilar to distractors but distractors are similar to one another the search slope is close to flat, and when targets are similar to distractors the slopes are highest.

There are some results which we do not reproduce. Duncan and Humphreys (1989) found no effect of d-d similarity when targets are very different from distractors and only a small effect of t-d similarity when distractors are all very similar. We found some effect of d-d similarity when targets are very different from distractors. We also found a large effect of t-d similarity when distractors are all identical. However, in these cases there is also considerable variability within human search experiments: much depends on details of the particular stimuli used, and on the way 'similarity' between stimuli is defined Pashler (1998). Like other modellers Wolfe (2007) we claim only qualitative consistency with Duncan and Humphreys' main results. But our reproduction of these results is still interesting, because it is achieved through a novel mechanism.

## 6 Visual search refined

In the t-d similar condition, targets can be grouped with adjacent distractors, creating heterogeneous groups, whose local form the classifier typically cannot identify. In the system described

above, if the system attends to a heterogeneous group that cannot be group classified, the group is assumed not to contain the target; if all salient regions are visited without finding the target, a second pass search is conducted, with each shape identified as a separate salient region. This is an expensive policy. An alternative is to examine the contents of an unclassifiable group as soon as it is identified.

With this in mind we extended the model to allow a new kind of attention shift, allowing the system to treat any selected region whose local form it cannot classify as a saliency map in its own right, within which sub-regions can be attended to and classified individually. This recursive ‘search within a search’ is somewhat similar to Treisman and Gormican’s group scanning theory Treisman and Gormican (1988). Technical details are again given in Walles *et al.* (2013). Search slopes were recomputed for each condition. In this extended model, the t-d similar conditions both have shallower slopes than the original: as might be expected, search is now more efficient if the target is missed in a group of distractors. But the main findings of Duncan and Humphreys are still reproduced: search slopes are still highest in these t-d similar conditions, and they are lowest when targets are dissimilar to distractors and distractors are homogeneous.

## 7 Comparison to existing models of visual search

Our model of visual search is closely related to several other models in which a map of salient locations functions to bias processing in a separate object classification pathway. The closest model is probably that of Walther and Koch Walther and Koch (2006), which uses a modified saliency map for the attentional processing stream and a variety of convolutional neural network for the classification stream. But this model does not perform any kind of grouping or group classification.

There are also several models of visual search which posit that attentional processing occurs within the system which performs object categorisation, rather than in a separate visual stream. One of these models has a number of similarities to ours—the ‘search by recursive rejection’ (SERR) model of Humphreys and Müller Humphreys and Müller (1993). Their model also reproduces the distractor similarity effects in visual search found by Duncan and Humphreys. The way our system performs visual search is in fact very similar to SERR: homogeneous groups of distractors are selected and classified in parallel. However, in SERR, grouping of similar items happens within the categorisation network. This network features a set of **match maps**: each region of the retina is associated with several specialised maps, one for each object shape which can be recognised. Each match map accumulates evidence for nearby objects of one particular shape, feeding activation to a localist ‘shape’ unit. Locations within each match map activate one another, so each map is most strongly activated by a homogeneous group of objects of the appropriate shape. This architecture allows the model to reproduce distractor similarity effects in visual search: homogeneous groups of distractors are selected and classified as groups, while heterogeneous groups must be selected and classified individually. But while SERR’s behaviour in visual search is similar to that of our model, it is achieved very differently. In SERR, selection and classification are tightly integrated: the maps which select individual items or homogeneous groups are connected one-to-one with shape units. As has been noted before



Heinke and Humphreys (2005), it is unrealistic to scale this model up to work with large numbers of complex types. In our system scaling is easier, because grouping is performed on the basis of simple visual features: once a region of homogeneous features is selected, classification of the objects possessing this feature is performed by a separate more powerful system, whose resources are directed at this region. This system is still able to classify homogeneous groups in parallel, but the classifier does not need to be replicated all over the retina.

## References

- Duncan, J. and Humphreys, G. W. (1989). Visual search and stimulus similarity. *Psychological Review*, **91**(3), 433–458.
- Gonzalez, R. C. and Woods, R. C. (1992). *Digital Image Processing*. Addison-Wesley, Reading, Mass.
- Heinke, D. and Humphreys, G. (2005). Computational models of visual selective attention: A review. In G. Houghton, editor, *Connectionist Models in Psychology*, pages 273–312. Psychology Press, Hove, UK.
- Humphreys, G. and Müller, H. (1993). SEArch via Recursive Rejection (SERR): A connectionist model of visual search. *Cognitive Psychology*, **25**, 43–110.
- Itti, L. and Koch, C. (2000). A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, **40**, 1489–1506.
- Liu, X. and Wang, D. (2000). Texture classification using spectral histograms. Technical Report TR17, Department of Computer and Information Science, The Ohio State University.
- Mozer, M. C. and Sitton, M. (1998). Computational modeling of spatial attention. In H. E. Pashler, editor, *Attention*, pages 341–393. Psychology Press, Hove.
- Pashler, H., editor (1998). *Attention*. MIT Press, Cambridge, MA.
- Posner, M. I. (1980). Orienting of attention. *QJEP*, **32**(1), 3–25.
- Quinlan, P. T. and Wilton, R. N. (1998). Grouping by proximity or similarity? competition between the gestalt principles in vision. *Perception*, **27**, 417–430.
- Riedmiller, M. (1994). Rprop - description and implementation details. Technical report, Institut für Logik, Komplexität und Deduktionssysteme, University of Karlsruhe.
- Rumelhart, D., Hinton, G., and Williams, R. (1986). Learning internal representations by error propagation. In *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*, volume 1: Foundations, chapter 8, pages 318–362. MIT Press, Cambridge, MA.

- Treisman, A. and Gormican, S. (1988). Feature analysis in early vision: Evidence from search asymmetries. *Psychological Review*, **95**(1), 15–48.
- Walles, H., Knott, A., and Robins, A. (2008). A model of cardinality blindness in inferotemporal cortex. *Biological Cybernetics*, **98**(5), 427–437.
- Walles, H., Robins, A., and Knott, A. (2013). A neural network model of visual attention and object classification: technical details. Technical Report OUCS-2013-09, Dept of Computer Science, University of Otago.
- Walther, D. and Koch, C. (2006). Modeling attention to salient proto-objects. *Neural Networks*, **19**(9), 1395 – 1407. Brain and Attention, Brain and Attention.
- Wolfe, J. M. (2007). Guided search 4.0: Current progress with a model of visual search. In W. Graw, editor, *Integrated Models of Cognitive Systems*, pages 99–119. Oxford University Press.