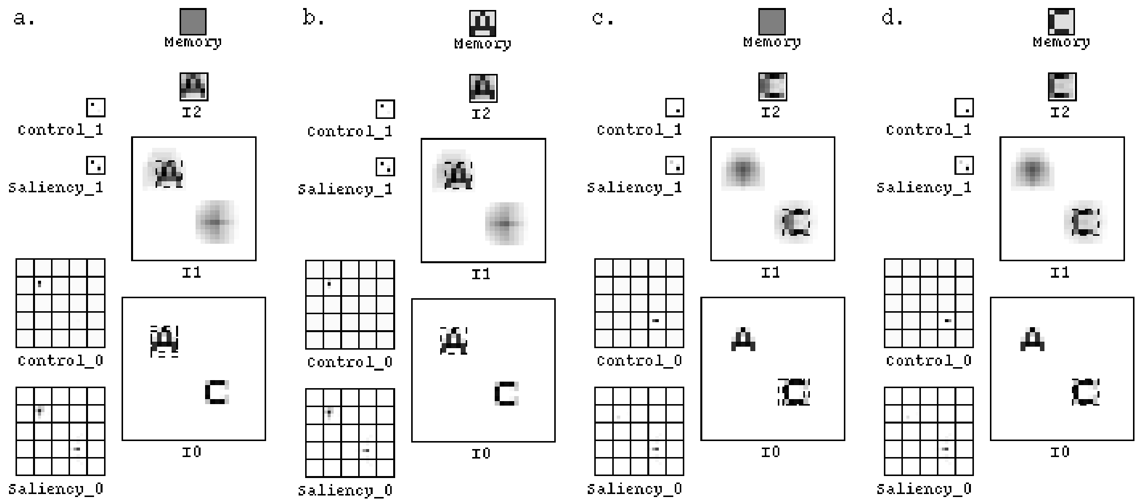

Figure 2.22: **Shifting attention in the multistage routing circuit.** The circuit is the same as previously. In *a*, the circuit has converged on the 'A' since it has the greatest overall brightness. *b*, After allowing enough time for recognition, the first stage saliency map is inhibited in the vicinity corresponding to the currently active control neuron of that stage. *c*, The circuit is then switched back into blob search mode, allowing the next most salient object, 'C', to be attended and subsequently recognized (*d*).

Competition among scales

We now revisit the multiscale stack circuit in which there are three different routing streams corresponding to different spatial scales (Fig. 2.8). Here, there are three control neurons at the top stage for gating the output of the routing circuit for each scale into a final output layer, I^3, as shown in Figure 2.23.

Following the same steps as before, the input term to c_s^2 in blob search mode is given by

$$-\frac{\partial E_{blob}}{\partial c_s^2} = \sum_i \sum_j G_i \Gamma^2_{ijs} I_j^{2,s} \,, \tag{2.31}$$

where $I^{l,s}$ denotes the nodes of layer l of scale s. Equation 2.31 states that c_s^2 will have a Gaussian receptive field in the top layer of scale s (Fig. 2.23). Or, in terms of the saliency map, c_s^2 is driven by the sum of activity in the top-stage saliency map for scale s.

The constraint matrix is set so that control neurons of different scales compete

$$U^2_{st} = \begin{cases} -1 & s \neq t \\ 0 & s = t \,. \end{cases} \tag{2.32}$$

Thus, extending the previous scheme, competition will begin at the top stage (stage 2); control neurons here will compete to select the scale with the most salience, and the winning control neuron will enable control neurons below to compete for the most salient module within that scale, and finally the most salient position within that module.

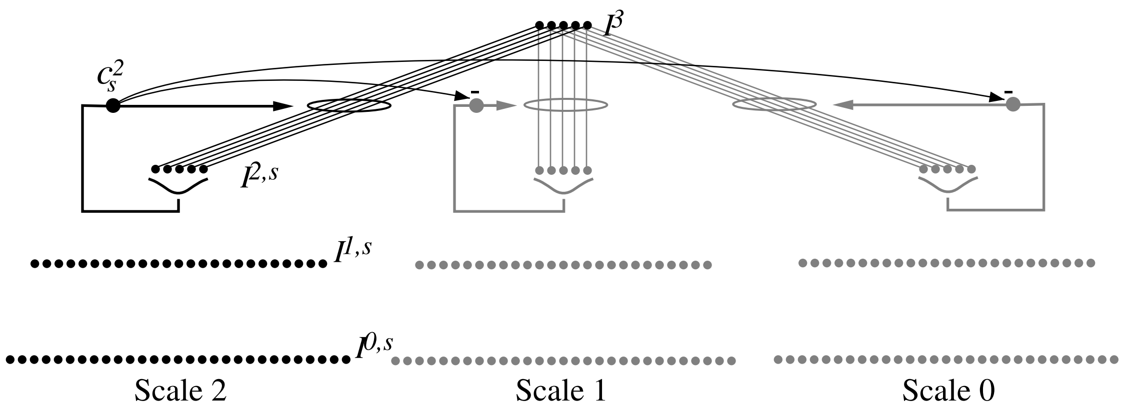

Figure 2.23: **Autonomous control for the multiscale stack routing circuit.** Each control neuron of stage 2 has a Gaussian receptive field in the top layer of the scale corresponding to the control neuron. These control neurons then compete among either other to decide on the scale to attend to.

In order to make the comparison between saliencies at different scales meaningful, the shape of the saliency function, **G**, will need to be changed so that it is selective for a particular scale. As it stands, the saliency nodes for the smallest scale will respond equally well or better to part of a large object as compared to a small object alone. The actual objective we seek during blob search is to just fill the window of attention with a blob that is confined within the bounds of the window. This objective can be expressed by adding an inhibitory surround to G, as in Figure 2.24. This way, a small object that stands alone in the high resolution array will be registered with higher salience than a high luminance region that is part of a larger object. It may also be desirable to build-in a precedence for global (low-resolution) over local (high-resolution) information by providing the low-resolution circuits with faster time constants. This would have the effect of "canceling out" the larger objects before attending to the small objects.

Figures 2.25-2.28 illustrate some examples from a computer simulation of an autonomous, 2D version of the model of Figure 2.8. The saliency function used here was $+1$ within a 7x7 window (with a gaussian taper) and -2 within a two-pixel wide perimeter. The method of computing the saliency for stages 1 and 2 was changed slightly from that described above, because simply summing the total saliency in a module can lead to deceivingly high saliencies in the higher stages. For example, there may be a single strong node within one salience module, but if another salience module has many weak nodes the total may actually be greater than that in the module with a single strong node. This effect was ameliorated somewhat by squaring the values of the saliency nodes, which has the effect of attenuating the low salience nodes. In Figures 2.25 and 2.26, the circuit is shown first attending to the 'A' at the lowest resolution level of the stack, and then to the 'C' at the highest resolution level several fixations later. Figures 2.27 and 2.28 demonstrate that the circuit is capable of discerning both local and global structure. In all figures, the "warper" window

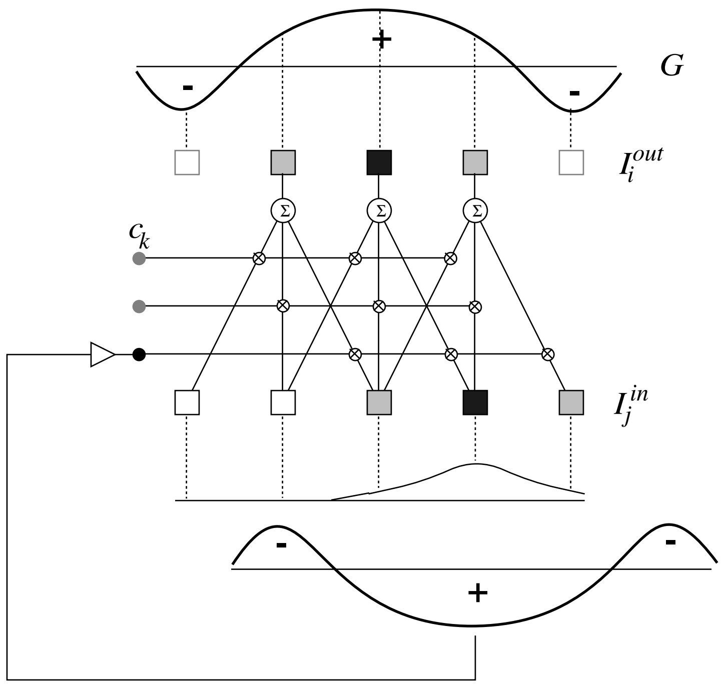

Figure 2.24: **Modified saliency function for scale selectivity.**
We can imagine having two flanking nodes on the output array which we desire to be inactive, since we want an attended object to fall within the bounds of the attentional window. This now requires that we add an inhibitory surround to the saliency function, **G**.

displays the state of the attended $9 \rightarrow 5$ module in the first stage, where scaling (and warping) less than a scale factor of two is performed. (This is just the same circuit as demonstrated in Figure 2.16.) The circuit as a whole is capable of continuously scaling the window over a factor of eight (from 5x5 to 40x40 nodes).

2.5 Summary of the model

In this chapter we developed a model circuit that adheres to the important neurobiological constraints of fan-in, fan-out, multiscale representation, and logarithmic spatial sampling. In order to specify how the control neurons are driven autonomously, we assumed that a useful strategy would be to focus attention on interesting regions within a scene and then attempt to recognize whatever is there. From this basic assumption, we derived equations for governing the dynamics of the control neurons in both "pre-attentive" (blob search) and "attentive" (recognition) modes. Breaking the routing circuit into multiple stages results in a hierarchical control circuit that is capable of setting the position and size of the window of attention globally with only local interactions among the control neurons. Although these circuits have been greatly scaled-down for the purpose of illustration and simulation, the basic principles can be extended to larger, scaled-up routing circuits. In the next chapter we show how such circuits may be implemented in the brain.

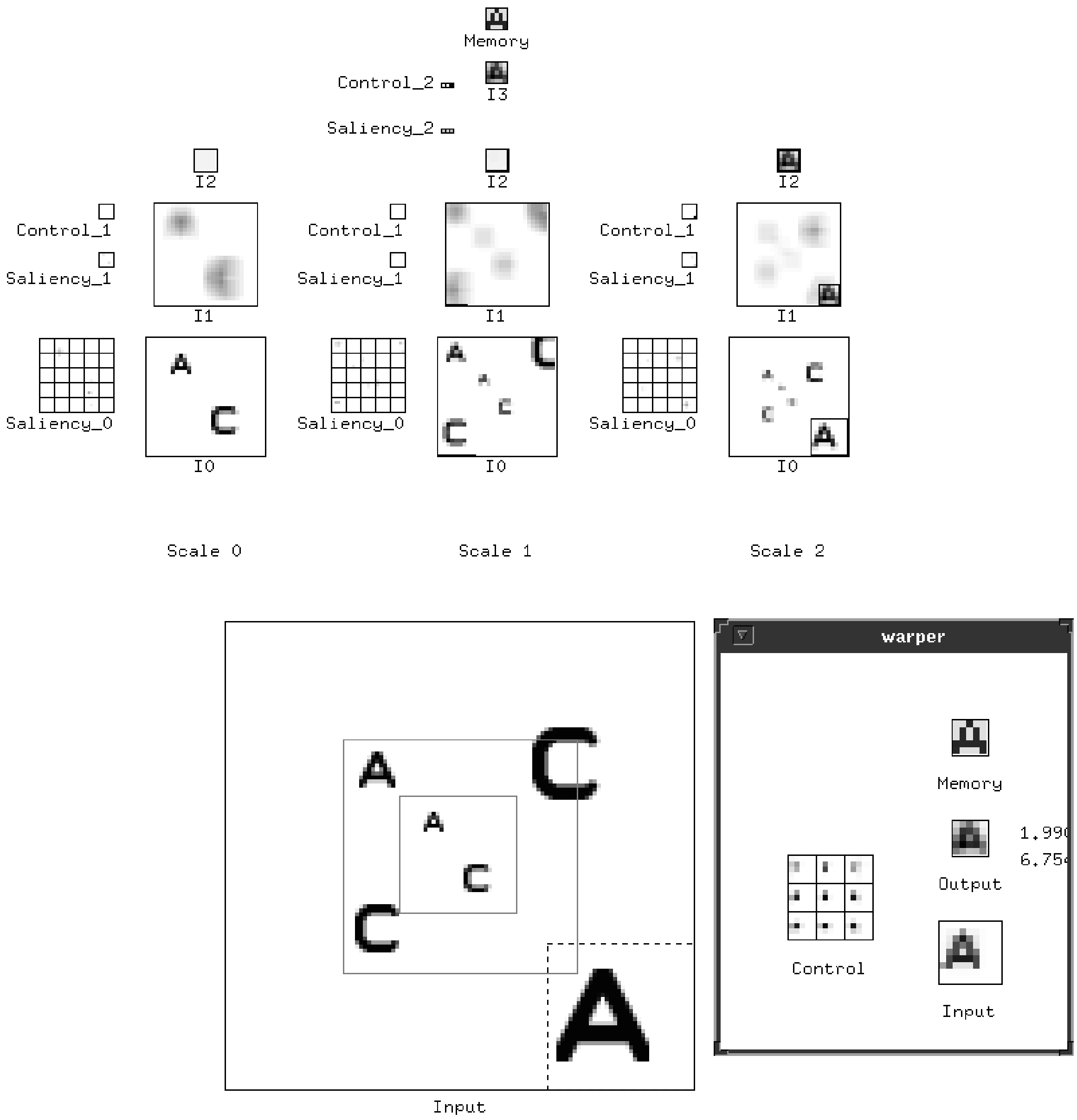

Figure 2.25: **Simulation of the stack circuit.**
The dotted square outlines in the input array denote the boundaries of the different levels of the stack. The dashed line indicates the window of attention. The circuit is shown attending to the 'A' at the lowest resolution level of the stack.