Communicated by Christoph von der Malsburg

Learning to Segment Images Using Dynamic Feature Binding

Michael C. Mozer
Department of Computer Science and Institute of Cognitive Science, University of Colorado, Boulder, CO 80309–0430 USA

Richard S. Zemel
Department of Computer Science, University of Toronto, Toronto, Ontario M5S 1A4

Marlene Behrmann
Department of Psychology and Faculty of Medicine and Rotman Research Institute of Baycrest Centre, University of Toronto, Toronto, Ontario M5S 1A1

Christopher K. I. Williams
Department of Computer Science, University of Toronto, Toronto, Ontario M5S 1A4

Despite the fact that complex visual scenes contain multiple, overlapping objects, people perform object recognition with ease and accuracy. One operation that facilitates recognition is an early segmentation process in which features of objects are grouped and labeled according to which object they belong. Current computational systems that perform this operation are based on predefined grouping heuristics. We describe a system called MAGIC that *learns* how to group features based on a set of presegmented examples. In many cases, MAGIC discovers grouping heuristics similar to those previously proposed, but it also has the capability of finding nonintuitive structural regularities in images. Grouping is performed by a relaxation network that attempts to dynamically bind related features. Features transmit a complex-valued signal (amplitude and phase) to one another; binding can thus be represented by phase locking related features. MAGIC's training procedure is a generalization of recurrent backpropagation to complex-valued units.

1 Introduction

Recognizing an isolated object in an image is a demanding computational task. The difficulty is greatly compounded when the image contains

Neural Computation 4, 650–665 (1992)

multiple objects because image features are not grouped according to which object they belong. Without the capability to form such groupings, it would be necessary to undergo a massive search through all subsets of image features. For this reason, most machine vision recognition systems include a component that performs feature grouping or *image segmentation* (e.g., Guzman 1968; Lowe 1985; Marr 1982). Psychophysical and neuropsychological evidence suggests that the human visual system performs a similar operation (Duncan 1984; Farah 1990; Kahneman and Henik 1981; Treisman 1982).

Image segmentation presents a circular problem: Objects cannot be identified until the image has been segmented, but unambiguous segmentation of the image requires knowledge of what objects are present. Fortunately, object recognition systems do not require precise segmentation: Simple heuristics can be used to group features, and although these heuristics are not infallible, they suffice for most recognition tasks. Further, the segmentation–recognition cycle can iterate, allowing the recognition system to propose refinements of the initial segmentation, which in turn refines the output of the recognition system (Hinton 1981; Hanson and Riseman 1978; Waltz 1975).

A multitude of heuristics have been proposed for segmenting images. Gestalt psychologists have explored how people group elements of a display and have suggested a range of grouping principles that govern human perception. For example, there is evidence for the grouping of elements that are close together in space or time, that appear similar, that move together, or that form a closed figure (Rock and Palmer, 1990). Computer vision researchers have studied the problem from a more computational perspective. They have investigated methods of grouping elements of an image based on *nonaccidental regularities*—feature combinations that are unlikely to occur by chance when several objects are juxtaposed, and are thus indicative of a single object. Kanade (1981) describes two such regularities, parallelism and skewed symmetry, and shows how finding instances of these regularities can constrain the possible interpretations of line drawings. Lowe and Binford (1982) find nonaccidental, significant groupings through a statistical analysis of images. They evaluate potential feature groupings with respect to a set of heuristics such as collinearity, proximity, and parallelism. The evaluation is based on a statistical measure of the likelihood that the grouping might have resulted from the random alignment of image features. Boldt *et al.* (1989) describe an algorithm for constructing lines from short line segments. The algorithm evaluates the goodness of fit of pairs of line segments in a small neighborhood based on relational measures (collinearity, proximity, and contrast similarity). Well matched pairs are replaced by longer segments, and the procedure is repeated.

In these earlier approaches, the researchers have hypothesized a set of grouping heuristics and then tested their psychological validity or computational utility. In our work, we have taken an *adaptive* approach to the

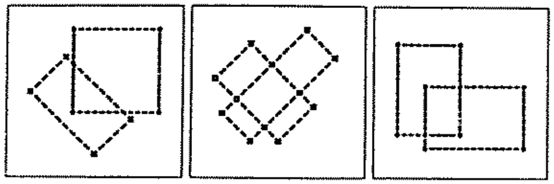

Figure 1: Examples of randomly generated two-dimensional geometric contours.

problem of image segmentation in which a system learns how to group features based on a set of examples. We call the system MAGIC, an acronym for multiple-object adaptive grouping of image components. In many cases MAGIC discovers grouping heuristics similar to those proposed in earlier work, but it also has the capability of finding nonintuitive structural regularities in images.

MAGIC is trained on a set of presegmented images containing multiple objects. By "presegmented" we mean that each image feature is labeled as to which object it belongs. MAGIC learns to detect configurations of the image features that have a consistent labeling in relation to one another across the training examples. Identifying these configurations then allows MAGIC to label features in novel, unsegmented images in a manner consistent with the training examples.

2 The Domain

Our initial work has been conducted in the domain of two-dimensional geometric contours, including rectangles, diamonds, crosses, triangles, hexagons, and octagons. The contours are constructed from four primitive feature types—oriented line segments at 0°, 45°, 90°, and 135°—and are laid out on a 25×25 grid. At each location on the grid are units, called *feature units*, that represent each of the four primitive feature types. In our present experiments, images contain two contours. We exclude images in which the two contours share a common edge. This permits a unique labeling of each feature. Examples of several randomly generated images containing rectangles and diamonds are shown in Figure 1.

3 Representing Feature Labelings

Before describing MAGIC, we must first discuss a representation that allows for the labeling of features. von der Malsburg (1981; von der

Malsburg and Schneider 1986), Gray *et al.* (1989), Eckhorn *et al.* (1988), and Strong and Whitehead (1989), among others, have suggested a biologically plausible mechanism of labeling through temporal correlations among neural signals, either the relative timing of neuronal spikes or the synchronization of oscillatory activities in the nervous system. The key idea here is that each processing unit conveys not just an activation value—average firing frequency in neural terms—but also a second, independent value that represents the relative phase of firing. The dynamic grouping or *binding* of a set of features is accomplished by aligning the phases of the features.

A flurry of recent work on populations of coupled oscillators (e.g., Baldi and Meir 1990; Grossberg and Somers 1991; Eckhorn *et al.* 1990; Kammen *et al.* 1990) has shown that this type of binding can be achieved using simple dynamic rules. However, most of this work assumes a relatively homogeneous pattern of connectivity among the oscillators and has not attempted to tackle problems in computer vision such as image segmentation, where each oscillator represents an image feature, and more selective connections between the oscillators are needed to simulate the selective binding of appropriate subsets of image features. A few exceptions exist (Goebel 1991a,b; Hummel and Biederman 1992; Lumer and Huberman 1991; Sporns *et al.* 1991); in these systems, the pattern of connectivity among oscillators is specified by simple predetermined grouping heuristics.[1]

In MAGIC, the activity of a feature unit is a complex value with *amplitude* and *phase* components. The phase represents a labeling of the feature, and the amplitude represents the confidence in that labeling. The amplitude ranges from 0 to 1, with 0 indicating a complete lack of confidence and 1 indicating absolute certainty. There is no explicit representation of whether a feature is present or absent in an image. Rather, absent features are clamped off—their amplitudes are forced to remain at 0—which eliminates their ability to influence other units, as will become clear when the activation dynamics are presented later.

4 The Architecture

When an image is presented to MAGIC, units representing features absent in the image are clamped off and units representing present features are assigned random initial phases and small amplitudes. MAGIC's task is to assign appropriate phase values to the units. Thus, the network performs a type of pattern completion.

[1] In the Sporns *et al.* model, the coupling strength between two connected units changes dynamically on a fast time scale, but this adaptation is related to achieving temporal correlations, not learning grouping principles.

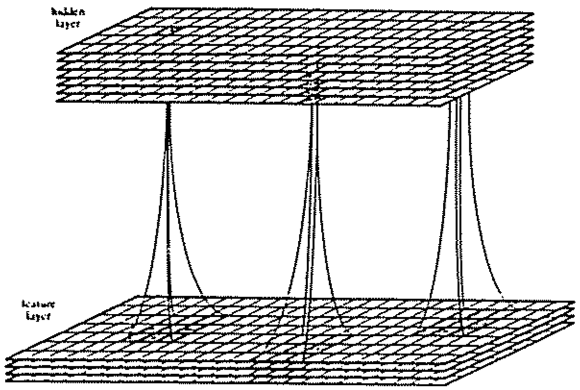

Figure 2: The architecture of MAGIC. The lower (input) layer contains the feature units; the upper layer contains the hidden units. Each layer is arranged in a spatiotopic array with a number of different feature types at each position in the array. Each plane in the feature layer corresponds to a different feature type. The grayed hidden units are reciprocally connected to all features in the corresponding grayed region of the feature layer. The lines between layers represent projections in both directions.

The network architecture consists of two layers of units, as shown in Figure 2. The lower (input) layer contains the feature units, arranged in spatiotopic arrays with one array per feature type. The upper layer contains hidden units that help to align the phases of the feature units; their response properties are determined by training. There are interlayer connections, but no intralayer connections. Each hidden unit is reciprocally connected to the units in a local spatial region of all feature arrays. We refer to this region as a *patch*; in our current simulations, the patch has dimensions 4×4. For each patch there is a corresponding fixed-size *pool* of hidden units. To achieve uniformity of response across the image, the pools are arranged in a spatiotopic array in which neighboring pools respond to neighboring patches and the patch-to-pool weights are constrained to be the same at all locations in the array.

The feature units activate the hidden units, which in turn feed back to the feature units. Through a relaxation process, the system settles on an assignment of phases to the features. One might consider an alternative architecture in which feature units were directly connected to one another (Hummel and Biederman 1992). However, this architecture is in principle not as powerful as the one we propose because it does not allow for higher order contingencies among features.

5 Network Dynamics

The dynamics of MAGIC are based on a mean-field approximation to a stochastic network of directional units, described in Zemel *et al.* (1992). A variant of this model was independently developed by Gislén *et al.* (1991). These papers provide a justification of the activation rule and error function in terms of an energy minimization formalism.

The response of each feature unit i, x_i, is a complex value in polar form, (a_i, p_i), where a_i is the amplitude and p_i is the phase. Similarly, the response of each hidden unit j, y_j, has components (b_j, q_j). The weight connecting unit i to unit j, w_{ji}, is also complex valued, having components (ρ_{ji}, θ_{ji}). The activation rule we propose is a generalization of the dot product to the complex domain. The net input to hidden unit j at time step $t+1$ is

$$\begin{aligned}
\text{net}_j(t+1) &= \mathbf{x}(t) \cdot \mathbf{w}_j \\
&= \textstyle\sum_i x_i(t) w_{ji}^* \\
&= \Bigg(\Big\{ \big(\textstyle\sum_i a_i(t)\rho_{ji} \cos[p_i(t) - \theta_{ji}]\big)^2 \\
&\qquad + \big(\textstyle\sum_i a_i(t)\rho_{ji} \sin[p_i(t) - \theta_{ji}]\big)^2 \Big\}^{1/2}, \\
&\qquad \tan^{-1}\left\{ \frac{\sum_i a_i(t)\rho_{ji} \sin[p_i(t) - \theta_{ji}]}{\sum_i a_i(t)\rho_{ji} \cos[p_i(t) - \theta_{ji}]} \right\} \Bigg)
\end{aligned}$$

where the asterisk denotes the complex conjugate. The net input is passed through a squashing nonlinearity that maps the amplitude of the response from the range $0 \to \infty$ to $0 \to 1$ but leaves the phase unaffected:

$$y_j(t) = \frac{\text{net}_j(t)}{m_j(t)} \frac{\mathrm{I}_1\left[m_j(t)\right]}{\mathrm{I}_0\left[m_j(t)\right]}$$

where $m_j(t)$ is the magnitude of the net input, $|\text{net}_j(t)|$, and I_k is the modified Bessel function of the first kind and order k. The squashing function $\mathrm{I}_1(m)/\mathrm{I}_0(m)$ is shown in Figure 3.

The intuition underlying the activation rule is as follows. The amplitude (confidence) of a hidden unit, b_j, should be monotonically related to how well the feature response pattern matches the hidden unit weight vector, just as in the standard real-valued activation rule. Indeed, one can readily see that if the feature and weight phases are equal ($p_i = \theta_{ji}$), the rule for b_j reduces to the real-valued case. Even if the feature and weight phases differ by a constant ($p_i = \theta_{ji} + c$), b_j is unaffected. This is a critical property of the activation rule: Because *absolute* phase values have no intrinsic meaning, the response of a unit should depend only on the *relative* phases. That is, its response should be rotation invariant. The activation rule achieves this by essentially ignoring the average difference

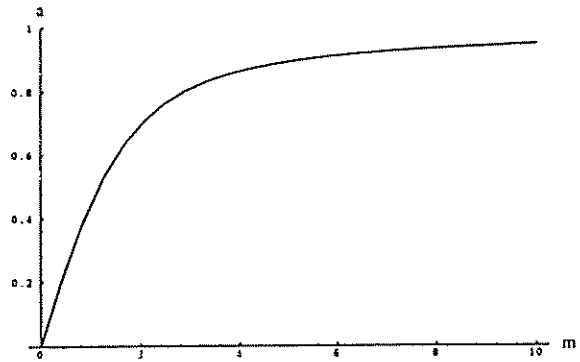

Figure 3: The squashing function $a = I_1(m)/I_0(m)$. The amplitude of the net input to a unit is passed through this function to obtain the output amplitude.

in phase between the feature units and the weights. The hidden phase, q_h, reflects this average difference.[2]

The flow of activation from the hidden layer to the feature layer follows the same dynamics as the flow from the feature layer to the hidden layer:

$$\text{net}_i(t+1) = \mathbf{y}(t+1) \cdot \mathbf{w}_i$$

and

$$x_i(t) = \frac{\text{net}_i(t)}{m_i(t)} \frac{I_1[m_i(t)]}{I_0[m_i(t)]}$$

if feature i is present in the image, or $x_i(t) = 0$ otherwise. Note that update is sequential by layer: the feature units activate the hidden units, which then activate the feature units.

In MAGIC, the weight matrix is constrained to be Hermitian, i.e., $w_{ji} = w_{ij}^*$. This is a generalization of weight symmetry to the complex domain. Weight symmetry ensures that MAGIC will converge to a fixed point. The proof of this is a generalization of Hopfield's (1984) result to complex units, discrete-time update, and a two-layer architecture with sequential layer updates and no intralayer connections.

[2] To elaborate, the activation rule produces a q_j that yields the minimum of the following expression:

$$d_i = \sum_i [a_i \cos p_i - \rho_{ji} \cos(\theta_{ji} + q_j)]^2 + [a_i \sin p_i - \rho_{ji} \sin(\theta_{ji} + q_j)]^2$$

This is a measure of the distance between the feature and weight vectors given a free parameter q_j that specifies a global phase shift of the weight vector.

6 Learning Algorithm

During training, we would like the hidden units to learn to detect configurations of features that reliably indicate phase relationships among the features. For instance, if the contours in the image contain extended horizontal lines, one hidden unit might learn to respond to a collinear arrangement of horizontal segments. Because the unit's response depends on the phase pattern as well as the activity pattern, it will be strongest if the segments all have the same phase value.

We have experimented with a variety of algorithms for training MAGIC, including an extension of soft competitive learning (Nowlan 1990) to complex-valued units, recurrent backpropagation (Almeida 1987; Pineda 1987), backpropagation through time (Rumelhart *et al.* 1986), a backpropagation autoencoder paradigm in which patches of the image are processed independently, and an autoencoder in which the patches are processed simultaneously and their results are combined. The algorithm with which we have had greatest success, however, is a relatively simple single-step error propagation algorithm. It involves running the network for a fixed number of iterations and, for each iteration, using backpropagation to adjust the weights so that the feature phase pattern better matches a target phase pattern. Each training trial proceeds as follows:

1. A training example is generated at random. This involves selecting two contours and instantiating them in an image. The features of one contour have target phase 0° and the features of the other contour have target phase 180°.

2. The training example is presented to MAGIC by setting the initial amplitude of a feature unit to 0.1 if its corresponding image feature is present, or clamping it at 0.0 otherwise. The phases of the feature units are set to random values in the range 0° to 360°.

3. Activity is allowed to flow from the feature units to the hidden units and back to the feature units.

4. The new phase pattern over the feature units is compared to the target phase pattern (see step 1), and an error measure is computed:

$$E = \sum_i \ln[I_0(m_i)] - \left\{ \left[\sum_i m_i \cos(\hat{p}_i - p_i)\right]^2 + \left[\sum_i m_i \sin(\hat{p}_i - p_i)\right]^2 \right\}^{1/2}$$

where m_i is the magnitude of the net input to feature unit i, p_i is the actual phase of unit i, and $\hat{p}_i$ is the target phase. This is a log likelihood error function derived from the formalism described in Zemel *et al.* (1992). In this formalism, the activities of units represent a probability distribution over phase values. The error function is the asymmetric divergence between the actual and target phase distributions. The aim is to minimize the difference between the target and actual phases and to maximize the amplitude, or confidence, of the response. The error measure factors out the absolute difference between the target and actual phases. That is, E is minimized when $\hat{p}_i - p_i$ is equal for all i, regardless of the value of $\hat{p}_i - p_i$.

5. Using a generalization of backpropagation to complex valued units, error gradients are computed for the feature-to-hidden and hidden-to-feature weights.

6. Steps 3–5 are repeated for a maximum of 30 iterations. The trial is terminated if the error increases on five consecutive iterations.

7. Weights are updated by an amount proportional to the average error gradient over iterations. The constraint that $w_{ji} = w^*_{ij}$ is enforced by modifying w_{ji} in proportion to $\nabla_{ji} + \nabla^*_{ij}$ and modifying w_{ij} in proportion to $\nabla^*_{ji} + \nabla_{ij}$, where ∇_{ij} denotes the gradient with respect to the weight to i from j. To achieve a translation-invariant response of the hidden units, hidden units of the same "type" responding to different regions of the image are constrained to have the same weights. This is achieved by having single set of underlying weight parameters that is replicated across the hidden layer. The appropriate gradient descent algorithm for these parameters is to adjust them in proportion to the sum of the gradients with respect to each of their instantiations.

The algorithm is far less successful when a target phase pattern is given just on the final iteration or final k iterations, rather than on each iteration. Surprisingly, the algorithm operates little better when error signals are propagated back through time.

The simulations reported below use a learning rate parameter of 0.005 for the amplitudes and 0.02 for the phases. On the order of 10,000 learning trials are required for stable performance, although MAGIC rapidly picks up on the most salient aspects of the domain.

7 Simulation Results

We trained a network with 20 hidden units per pool on examples like those shown in Figure 1. The resulting weights are shown in Figure 4. Each hidden unit attempts to detect and reinstantiate activity patterns that match its weights. One clear and prevalent pattern in the weights is the collinear arrangement of segments of a given orientation, all having the same phase value. When a hidden unit having weights of this form responds to a patch of the feature array, it tries to align the phases of the patch with the phases of its weight vector. By synchronizing the phases

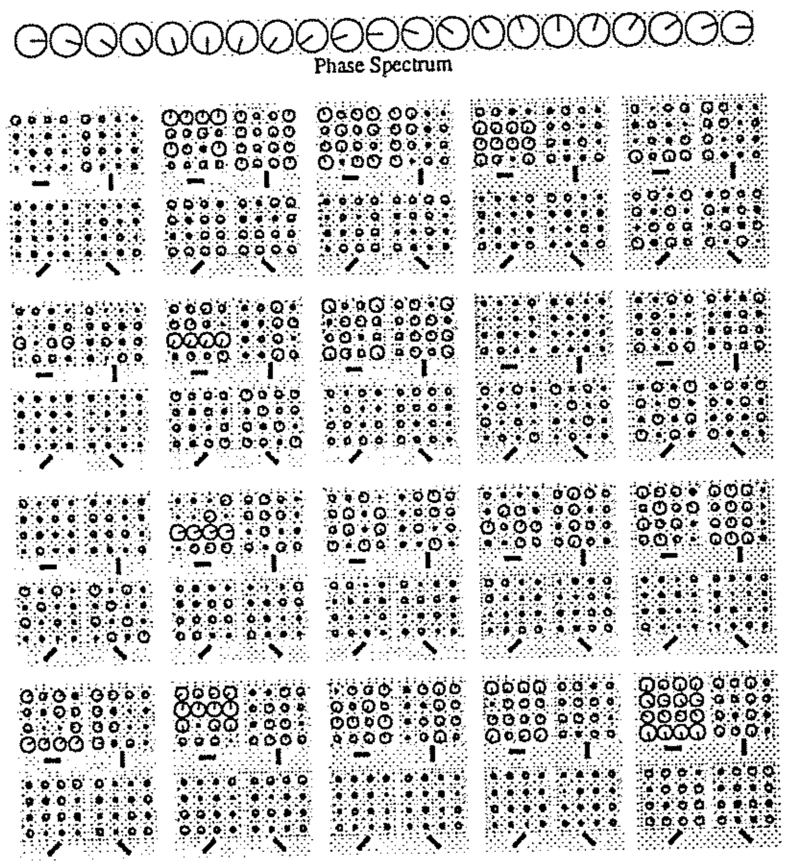

Figure 4: Complex feature-to-hidden connection weights learned by MAGIC. In this simulation, there are connections from a 4×4 patch of the image to a pool of 20 hidden units. (These connections are replicated for each patch in the image to achieve a uniformity of hidden unit response.) The connections feeding into each hidden unit are presented on a light gray background. Each hidden unit has a total of 64 incoming weights—4 × 4 locations in its receptive field and four feature types at each location. The weights are further grouped by feature type (dark gray background), and for each feature type they are arranged in a 4 × 4 pattern homologous to the image patch itself. The area of a circle is proportional to the amplitude of the corresponding weight, the orientation of the internal tick mark represents the phase angle. Due to the symmetry constraint, hidden-to-feature weights (not shown) mirror the feature-to-hidden weights.

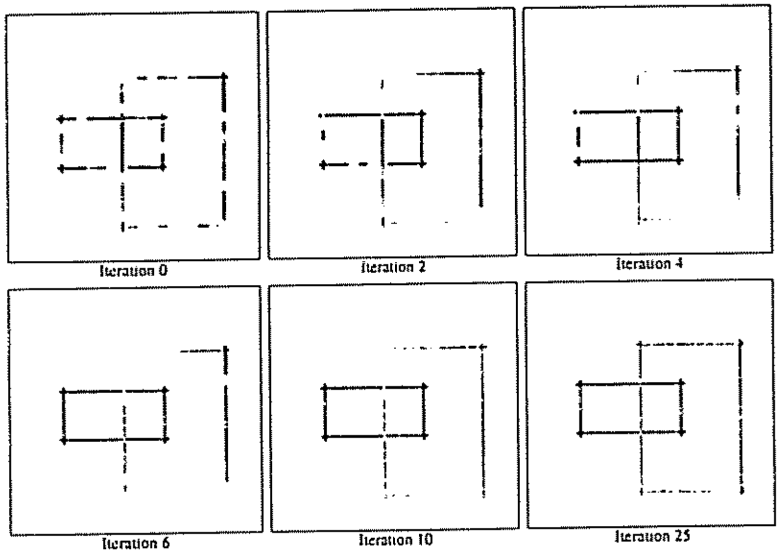

Figure 5: An example of MAGIC segmenting an image. The "iteration" refers to the number of times activity has flowed from the feature units to the hidden units and back. The phase value of a feature is represented by a gray level. The cyclic phase continuum can be approximated only by a linear gray level continuum, but the basic information is conveyed nonetheless.

of features, it acts to group the features. Thus, one can interpret the weight vectors as the rules by which features are grouped.

Whereas traditional grouping principles indicate the conditions under which features should be bound together as part of the same object, the grouping principles learned by MAGIC also indicate when features should be segregated into different objects. For example, the weights of the vertical and horizontal segments are generally 180° out of phase with the diagonal segments. This allows MAGIC to segregate the vertical and horizontal features of a rectangle from the diagonal features of a diamond (see Fig. 1, left panel). We had anticipated that the weights to each hidden unit would contain two phase values at most because each image patch contains at most two objects. However, some units make use of three or more phases, suggesting that the hidden unit is performing several distinct functions. As is the usual case with hidden unit weights, these patterns are difficult to interpret.

Figure 5 presents an example of the network segmenting an image. The image contains two rectangles. The top left panel shows the features of the rectangles and their initial random phases. The succeeding panels show the network's response during the relaxation process. The lower right panel shows the network response at equilibrium. Features of each object have been assigned a uniform phase, and the two objects are 180° out of phase. The task here may appear simple, but it is quite challenging due to the illusory rectangle generated by the overlapping rectangles.

8 Alternative Representation of Feature Labeling

To perform the image segmentation task, each feature unit needs to maintain two independent pieces of information: a label assigned to the feature and a measure of confidence associated with the label. In MAGIC, these two quantities are encoded by the phase and amplitude of a unit, respectively. This *polar* representation is just one of many possible encodings, and requires some justification due to the complexity of the resulting network dynamics. An alternative we have considered—which seems promising at first glance but has serious drawbacks—is the *rectangular* coordinate analog of the polar representation. In this scheme, a feature unit conveys values indicating belief in the hypotheses that the feature is part of object $\mathcal{A}$ or object $\mathcal{B}$, where $\mathcal{A}$ and $\mathcal{B}$ are arbitrary names. For example, the activities $(1,0)$ and $(0,1)$ indicate complete confidence that the feature belongs to object $\mathcal{A}$ or $\mathcal{B}$, respectively, $(0,0)$ indicates that nothing is known about which object the feature belongs to, and intermediate values indicate intermediate degrees of confidence in the two hypotheses. The rectangular and polar representations are equivalent in the sense that one can be transformed into the other.[3]

The rectangular scheme has two primary benefits. First, the activation dynamics are simpler. Second, it allows for the simultaneous and explicit consideration of multiple labeling hypotheses, whereas the polar scheme allows for the consideration of only one label at a time. However, these benefits are obtained at the expense of presuming a correspondence between absolute phase values and objects. (In the rectangular scheme we described, $\mathcal{A}$ and $\mathcal{B}$ always have phases 0° and 90°, respectively, obtained by transforming the rectangular coordinates to polar coordinates.) The key drawback of absolute phase values is that a local patch of the image cannot possibly determine which label is correct. A patch containing, say, several collinear horizontal segments can determine only that the segments should be assigned the same label. Preliminary simulations indicate that the resulting ambiguity causes severe difficulties in processing. In contrast, the polar scheme allows the network to express the *relative* labelings of two segments—e.g., that they should be assigned the same label—without needing to specify the particular label.

[3]Yann Le Cun (personal communication, 1991) has independently developed the notion of using the rectangular encoding scheme in the domain of adaptive image segmentation.

9 Current Directions

We are currently extending MAGIC in several directions, which we outline here.

- We have not addressed the question of how the continuous phase representation is transformed into a discrete object label. One may simply quantize the phase values such that all phases in a given range are assigned the same label. This quantization step has the extremely interesting property that it allows for a hierarchical decomposition of objects. If the quantization is coarse, only gross phase differences matter, allowing one object to be distinguished from another. As the quantization becomes finer, an object is divided into its components. Thus, the quantization level in effect specifies whether the image is parsed into objects, parts of objects, parts of parts of objects, etc.

  This hierarchical decomposition of objects can be achieved only if the phase values reflect the internal structure of an object. For example, in the domain of geometric contours, MAGIC would not only have to assign one contour a different phase value than another, but it would also have to assign each edge composing a contour a slightly different phase than each other edge (assuming that one considers the edges to be the "parts" of the contour). Somewhat surprisingly, MAGIC does exactly this because the linkage between segments of an edge is stronger than the linkage between two edges. This is due to the fact that collinear features occur in images with much higher frequency than do corners. Thus, the relative frequency of feature configurations leads to a natural principle for the hierarchical decomposition of objects.

- Although MAGIC is trained on pairs of objects, it has the potential of processing more than two objects at a time. For example, with three overlapping objects, MAGIC attempts to push each pair 180° out of phase but ends up with a best constraint satisfaction solution in which each object is 120° out of phase with each other. We are exploring the limits of how many objects MAGIC can process at a time.

- Spatially local grouping principles are unlikely to be sufficient for the image segmentation task. Indeed, we have encountered incorrect solutions produced by MAGIC that are locally consistent but globally inconsistent. To solve this problem, we are investigating an architecture in which the image is processed at several spatial scales simultaneously. Fine-scale detectors respond to the sort of detail shown in Figure 4, while coarser-scale detectors respond to more global structure but with less spatial resolution.

- Simulations are under way to examine MAGIC's performance on real-world images—overlapping handwritten letters and digits—where it is somewhat less clear to which types of patterns the hidden units should respond.

- Behrmann *et al.* (1992) are conducting psychological experiments to examine whether limitations of the model match human limitations.

Acknowledgments

This research was supported by NSF Presidential Young Investigator award IRI–9058450, Grant 90–21 from the James S. McDonnell Foundation, and DEC external research Grant 1250 to MM; by a National Sciences and Engineering Research Council Postgraduate Scholarship to RZ; and by an NSERC operating grant to MB. Our thanks to Paul Smolensky, Radford Neal, Geoffrey Hinton, and Jürgen Schmidhuber for helpful comments regarding this work.

References

Almeida, L. 1987. A learning rule for asynchronous perceptrons with feedback in a combinatorial environment. *Proceedings of the IEEE First Annual International Conference on Neural Networks*, Vol. 2, M. Caudill and C. Butler, eds., pp. 609–618. EEE Publishing Services, San Diego, CA.

Baldi, P., and Meir, R. 1990. Computing with arrays of coupled oscillators: An application to preattentive texture discrimination. *Neural Comp.* 2, 458–471.

Behrmann, M., Zemel, R. S., and Mozer, M. C. 1992. Perceptual organization and object-based attention. Manuscript in preparation.

Boldt, M., Weiss, R., and Riseman, E. 1989. Token-based extraction of straight lines. *IEEE Transact. Syst. Man Cybern.* 19, 1581–1594.

Duncan, J. 1984. Selective attention and the organization of visual information. *J. Exp. Psychol. General* 113, 501–517.

Eckhorn, R., Bauer, R., Jordan, W., Brosch, M., Kruse, W., Munk, M., and Reitboek, H. J. 1988. Coherent oscillations: A mechanism of feature linking in the visual cortex? *Biol. Cybern.* 60, 121–130.

Eckhorn, R., Reitboeck, H. J., Arndt, M., and Dicke, P. 1990. Feature linking via synchronization among distributed assemblies: Simulations of results from cat visual cortex. *Neural Comp.* 2, 293–307.

Farah, M. J. (1990). *Visual Agnosia*. The MIT Press/Bradford Books, Cambridge, MA.

Gislén, L., Peterson, C., and Söderberg, B. 1991. *Rotor Neurons—Basic Formalism and Dynamics* (LU TP 91–21). University of Lund, Department of Theoretical Physics, Lund, Sweden.

Goebel, R. 1991a. An oscillatory neural network model of visual attention, pattern recognition, and response generation. Manuscript in preparation.

Goebel, R. 1991b. The role of attention and short-term memory for symbol manipulation: A neural network model that learns to evaluate simple LISP expressions. In *Cognition and Computer Programming*, K. F. Wender, F. Schmalhofer, and H. D. Boecker, eds. Ablex Publishing Corporation, Norwood, NJ.

Gray, C. M., Koenig, P., Engel, A. K., and Singer, W. 1989. Oscillatory responses in cat visual cortex exhibit intercolumnar synchronization which reflects global stimulus properties. *Nature (London)* 338, 334–337.

Grossberg, S., and Somers, D. 1991. Synchronized oscillations during cooperative feature linking in a cortical model of visual perception. *Neural Networks* 4, 453–466.

Guzman, A. 1968. Decomposition of a visual scene into three-dimensional bodies. *AFIPS Fall Joint Comput. Conf.* 33, 291–304.

Hanson, A. R., and Riseman, E. M. 1978. *Computer Vision Systems*. Academic Press, New York.

Hinton, G. E. 1981. A parallel computation that assigns canonical object-based frames of reference. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pp. 683–685. Morgan Kaufmann, Los Altos, CA.

Hopfield, J. J. 1984. Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci. U.S.A.* 81, 3088–3092.

Hummel, J. E., and Biederman, I. 1992. Dynamic binding in a neural network for shape recognition. *Psychol. Rev.*, in press.

Kahneman, D., and Henik, A. 1981. Perceptual organization and attention. In *Perceptual Organization*, M. Kubovy and J. R. Pomerantz, eds., pp. 181–211. Erlbaum, Hillsdale, NJ.

Kammen, D., Koch, C., and Holmes, P. J. 1990. Collective oscillations in the visual cortex. In *Advances in Neural Information Processing Systems 2*, D. S. Touretzky, ed., pp. 76–83. Morgan Kaufmann, San Mateo, CA.

Kanade, T. 1981. Recovery of the three-dimensional shape of an object from a single view. *Artificial Intell.* 17, 409–460.

Lowe, D. G. 1985. *Perceptual Organization and Visual Recognition*. Kluwer Academic Publishers, Boston.

Lowe, D. G., and Binford, T. O. 1982. Segmentation and aggregation: An approach to figure-ground phenomena. *Proceedings of the DARPA IUS Workshop*, pp. 168–178. Palo Alto, CA.

Lumer, E., and Huberman, B. A. 1992. Binding hierarchies: A basis for dynamic perceptual grouping. *Neural Comp.* 4, 341–355.

Marr, D. 1982. *Vision*. Freeman, San Francisco.

Nowlan, S. J. 1990. *Max likelihood competition in RBF networks*. Tech. Rep. CRG-TR-90-2. Toronto, Canada: University of Toronto, Department of Computer Science, Connectionist Research Group.

Pineda, F. 1987. Generalization of back propagation to recurrent neural networks. *Phys. Rev. Lett.* 19, 2229–2232.

Rock, I., and Palmer, S. E. 1990. The legacy of Gestalt psychology. *Sci. Amer.* 263, 84–90.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. 1986. Learning internal representations by error propagation. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume I: Foundations*, D. E. Rumelhart and J. L. McClelland, eds., pp. 318–362. The MIT Press/Bradford Books, Cambridge, MA.

Sporns, O., Tononi, G., and Edelman, G. M. 1991. Modeling perceptual grouping and figure-ground segregation by means of active reentrant connections. *Proc. Natl. Acad. Sci.* 88, 129–133.

Strong, G. W., and Whitehead, B. A. 1989. A solution to the tag-assignment problem for neural networks. *Behav. Brain Sci.* 12, 381–433.

Treisman, A. 1982. Perceptual grouping and attention in visual search for features and objects. *J. Exp. Psychol. Human Percept. Perform.* 8, 194–214.

von der Malsburg, C. 1981. *The correlation theory of brain function* (Internal Report 81-2). Goettingen: Department of Neurobiology, Max Planck Institute for Biophysical Chemistry.

von der Malsburg, C., and Schneider, W. 1986. A neural cocktail-party processor. *Biol. Cybern.* 54, 29–40.

Waltz, D. A. 1975. Generating semantic descriptions from drawings of scenes with shadows. In *The Psychology of Computer Vision*. P. H. Winston, ed., pp. 19–92. McGraw-Hill, New York.

Zemel, R. S., Williams, C. K. I., and Mozer, M. C. 1992. Adaptive networks of directional units. Submitted for publication.

Received 29 August 1991; accepted 25 February 1992.