MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY
and
CENTER FOR BIOLOGICAL INFORMATION PROCESSING
WHITAKER COLLEGE

A.I. Memo No. 1253
C.B.I.P. Memo No. 50

January 1990

A Theory of How the Brain Might Work

Tomaso Poggio

ABSTRACT

I wish to propose a quite speculative new version of the grandmother cell theory to explain how the brain, or parts of it, may work. In particular, I discuss how the visual system may learn to recognize 3D objects. The model would apply directly to the cortical cells involved in visual face recognition. I will also outline the relation of our theory to existing models of the cerebellum and of motor control. Specific biophysical mechanisms can be readily suggested as part of a basic type of neural circuitry that can learn to approximate multidimensional input-output mappings from sets of examples and that is expected to be replicated in different regions of the brain and across modalities. The main points of the theory are:

- the brain uses modules for multivariate function approximation as basic components of several of its information processing subsystems.
- these modules are realized as HyperBF networks (Poggio and Girosi, 1990a,b).
- HyperBF networks can be implemented in terms of biologically plausible mechanisms and circuitry.

The theory predicts a specific type of population coding that represents an extension of schemes such as look-up tables. I will conclude with some speculations about the trade-off between memory and computation and the evolution of intelligence.

This paper describes research done within the Center for Biological Information Processing, in the Department of Brain and Cognitive Sciences, and at the Artificial Intelligence Laboratory. This research is sponsored by a grant from the Office of Naval Research (ONR), Cognitive and Neural Sciences Division; by the Artificial Intelligence Center of Hughes Aircraft Corporation; by the Alfred P. Sloan Foundation; by the National Science Foundation; by the Artificial Intelligence Center of Hughes Aircraft Corporation (S1-801534-2); by E.I. DuPont DeNemours and Company; and by the NATO Scientific Affairs Division (0403/87). Support for the A. I. Laboratory's artificial intelligence research is provided by the Advanced Research Projects Agency of the Department of Defense under Army contract DACA76-85-C-0010, and in part by ONR contract N00014-85-K-0124. Tomaso Poggio is supported by the Uncas and Helen Whitaker Chair at the Massachusetts Institute of Technology, Whitaker College.

Contents

1 The Grandmother Neuron Theory

A classical theme in the neurophysiological literature at least since the work of Hubel and Wiesel (1962) is the idea of information processing in the brain as leading to "grandmother" neurons responding selectively to the precise combination of visual features that are associated with one's grandmother. Even when not explicitily stated, this notion seems to capture how many neuroscientists believe that the brain works. The "grandmother" neuron theory is of course not restricted to vision and applies as well to other sensory modalities and even to motor control under the form of cells corresponding to elemental movements. Why is this idea so attractive? The idea is attractive because of its simplicity: it replaces complex information processing with the superficially simpler task of accessing a memory. The problem of recognition and motor control would be solved by simply accessing look-up tables containing appropriate descriptions of objects and of motor actions. The human brain can probably exploit a vast amount of memory with its 10^{14} or so synapses, making attractive any scheme that replaces computation with memory. In the case of vision the apparent simplicity of this solution hides the difficult problems of an appropriate representation of an object and of how to extract it from complex images. But even assuming that these problems of representation, feature extraction and segmentation could be solved by other mechanisms, a fundamental difficulty seems to be intrinsic to the "grandmother" cell idea. The difficulty consists of the combinatorial explosion in the number of cells that any scheme of the look-up table type would reasonably require for either vision or motor control. In the case of 3D object recognition, for instance, there should be for each object as many entries in the look-up table as there are 2-D views of the object, in principle an infinite number.

The difficulty of a combinatorial explosion lies at the heart of theories of intelligence that attempt to replace information processing with look-up tables of precomputed results. In this paper we suggest a scheme that avoids the combinatorial problem, while retaining the attractive features of the look-up table. The basic idea is to use only a few entries and interpolate or approximate among them. A mathematical theory based on this idea leads to a powerful scheme of learning from examples that is equivalent to a parallel network of simple processing elements. The scheme has an intriguingly simple implementation in terms of plausible biophysical mechanisms. We will discuss in particular the case of 3D object recognition but will propose that the scheme is possibly used by the brain for several different information processing tasks. Many information processing problems can be represented as the composition of one or more multivariate functions that map an input signal into an output signal in a smooth way. These modules could be synthesized from a sufficient set of input-output pairs – the examples – by the scheme described here. Because of the power and general applicability of this mechanism, we speculate that a part of the machinery of the brain – including perhaps some of the cortical circuitry which is somewhat similar across the different modalities – may be dedicated to the task of function approximation.

2 How to synthesize through Learning the basic Approximation Module: Regularization Networks

This section describes a technique for synthesizing the approximation modules discussed above through learning from examples. I first explain how to rephrase the problem of learning from examples as a problem of approximating a multivariate function. The material in this section is from Poggio and Girosi (1989, 1990a, 1990b), where more details can be found.

To illustrate the connection, let us draw an analogy between learning an input-output mapping and a standard approximation problem, 2-D surface reconstruction from sparse data points. *Learning* simply means collecting the *examples*, i.e., the input coordinates x_i, y_i and the corresponding output values at those locations, the heights of the surface d_i. *Generalization* means estimating d at locations x, y where there are no examples, i.e. no data. This requires interpolating or, more generally, approximating the surface (i.e. the function) between the data points (interpolation is the limit of approximation when there is no noise in the data). In this sense, learning is a problem of *hypersurface reconstruction* (Poggio et al., 1988, 1989; Omohundro, 1987).

From this point of view, learning a smooth mapping from examples is clearly ill-posed, in the sense that the information in the data is not sufficient to reconstruct uniquely the mapping at places where data are not available. In addition, the data are usually noisy. *A priori* assumptions about the mapping are needed to make the problem well-posed. One of the simplest assumptions is that the mapping is *smooth*: small changes in the inputs cause a small change in the output. Techniques that exploit smoothness constraints in order to transform an ill-posed problem into a well-posed one are well known under the term of *regularization theory*, and have interesting Bayesian applications (Tikhonov and Arsenin, 1977; Poggio, Torre and Koch, 1985; Bertero, Poggio and Torre, 1988). We have recently shown that that the solution to the approximation problem given by regularization theory can be expressed in terms of a class of multilayer networks that we call regularization networks or Hyper Basis Functions (see Fig. 1). Our main result (Poggio and Girosi, 1989) is that the regularization approach is equivalent to an expansion of the solution in terms of a certain class of functions:

$$f(\mathbf{x}) = \sum_{i=1}^{N} c_i G(\mathbf{x}; \boldsymbol{\xi}_i) + p(\mathbf{x}) \tag{1}$$

where $G(\mathbf{x})$ is one such function and the coefficients c_i satisfy a linear system of equations that depend on the N "examples", i.e. the data to be approximated. The term $p(\mathbf{x})$ is a polynomial that depends on the smoothness assumptions. In many cases it is convenient to include up to the constant and linear terms. Under relatively broad assumptions, the Green's function G is radial and therefore the approximating function becomes:

$$f(\mathbf{x}) = \sum_{i=1}^{N} c_i G(\|\mathbf{x} - \boldsymbol{\xi}_i\|^2) + p(\mathbf{x}), \tag{2}$$

which is a sum of radial functions, each with its *center* $\boldsymbol{\xi}_i$ on a distinct data point and of constant and linear terms (from the polynomial, when restricted to be of degree one). The number of radial functions, and corresponding centers, is the same as the number of examples.

Our derivation shows that the type of basis functions depends on the specific *a priori* assumption of smoothness. Depending on it we obtain the Gaussian $G(r) = e^{-(\frac{r}{c})^2}$, the well known "thin plate spline" $G(r) = r^2 \ln r$, and other specific functions, radial and not. As observed by Broomhead and Lowe (1989) in the radial case, a superposition of functions like Eq. 1 is equivalent to a network of the type shown in Fig. 1b. The interpretation of Eq. 2 is simple: in the 2D case, for instance, the surface is approximated by the superposition of, say, several two dimensional Gaussian distributions, each centered on one of the data points.

The network associated with Eq. 2 can be made more general in terms of the following extension

$$f^*(\mathbf{x}) = \sum_{\alpha=1}^{n} c_\alpha G(\|\mathbf{x} - \mathbf{t}_\alpha)\|_W^2) + p(\mathbf{x}) \tag{3}$$

where the parameters $\mathbf{t}_\alpha$, that we call "centers", and the coefficients c_α are unknown, and are in general much fewer than the data points ($n \leq N$). The norm is a *weighted norm*

$$\|\mathbf{x} - \mathbf{t}_\alpha\|_W^2 = (\mathbf{x} - \mathbf{t}_\alpha)^T W^T W (\mathbf{x} - \mathbf{t}_\alpha) \tag{4}$$

where W is an unknown square matrix and the superscript T indicates the transpose. In the simple case of diagonal W the diagonal elements w_i assign a specific weight to each input coordinate, determining in fact the units of measure and the importance of each feature (the matrix W is especially important in cases in which the input features are of a different type and their relative importance is unknown). Equation 3 can be implemented by the network of Fig. 1. Notice that a sigmoid function at the output may sometimes be useful without increasing the complexity of the system (see Poggio and Girosi, 1989). Notice also that there could be more than one set of Green's functions, for instance a set of multiquadrics and a set of Gaussians, each with its own $\mathbf{W}$. Notice that two or more sets of Gaussians, each with a diagonal $\mathbf{W}$, are equivalent to sets of Gaussians with their own σs.

2.1 Learning

Iterative methods can be used to find the optimal values of the various sets of parameters, the c_α, the w_i and the $\mathbf{t}_\alpha$, that minimize an error functional on the set of examples. Steepest descent is *the* standard approach that requires calculations of derivatives. An even simpler method that does not require calculation of derivatives (suggested and found surprisingly efficient in preliminary work by Caprile and Girosi, personal communication) is to look for random changes (controlled in appropriate ways) in the parameter values that reduce the error. We define the error functional – also called energy – as

$$H[f^*] = H_{\mathbf{c},\mathbf{t},\mathbf{W}} = \sum_{i=1}^{N} (\Delta_i)^2,$$

with

$$\Delta_i \equiv y_i - f^*(\mathbf{x}) = y_i - \sum_{\alpha=1}^{n} c_\alpha G(\|\mathbf{x}_i - \mathbf{t}_\alpha\|_{\mathbf{W}}^2).$$

In the first method the values of c_α, $\mathbf{t}_\alpha$ and $\mathbf{W}$ that minimize $H[f^*]$ are regarded as the coordinates of the stable fixed point of the following dynamical system:

$$\dot{c}_\alpha = -\omega \frac{\partial H[f^*]}{\partial c_\alpha}, \quad \alpha = 1, \ldots, n$$
$$\dot{\mathbf{t}}_\alpha = -\omega \frac{\partial H[f^*]}{\partial \mathbf{t}_\alpha}, \quad \alpha = 1, \ldots, n$$
$$\dot{\mathbf{W}} = -\omega \frac{\partial H[f^*]}{\partial \mathbf{W}},$$

where ω is a parameter. The derivatives are rather complex (see Poggio and Girosi, 1990a and Notes section).

The second method is simpler: random changes in the parameters are made and accepted if $H[f^*]$ decreases. Occasionally, changes that increase $H[f^*]$ may also be accepted (similarly to the Metropolis algorithm).

2.2 Interpretation of the network

The interpretation of the network of Fig. 1 is the following. *After learning*, the centers of the basis functions are similar to prototypes, since they are points in the multidimensional input space. Each unit computes a (weighted) distance of the inputs from its center, that is a measure of their similarity, and applies to it the radial function. In the case of the Gaussian, a unit will have maximum activity when the new input exactly matches its center. The output of the network is the linear superposition of the activities of all the basis functions in the network, plus direct, weighted connections from the inputs (the linear terms of $p(\mathbf{x})$) and from a constant input (the constant term). Notice that in the limit case of the basis functions approximating delta functions, the system becomes equivalent to a look-up table. *During learning* the weights $\mathbf{c}$ are found by minimizing a measure of the error between the network's prediction and each of the examples. At the same time, the centers of the radial functions and the weights in the norm are also updated during learning. Moving the centers is equivalent to modifying the corresponding prototypes and corresponds to task-dependent clustering. Finding the optimal weights $\mathbf{W}$ for the norm is equivalent to transforming appropriately, for instance scaling, the input coordinates and corresponds to task-dependent dimensionality reduction.

Regularization networks- of which HyperBFs are the most general and powerful version – represent a general framework for learning smooth mappings that rigorously connects approximation theory, generalized splines and regularization with feedforward multilayer networks. They also contain as special cases the Radial Basis Functions technique (Micchelli, 1986;

Powell, 1987; Broomhead and Lowe, 1988) and several well-known algorithms, especially in the pattern recognition literature.

3 A Proposal for a Biological Implementation

In this section we point out some remarkable properties of Gaussian HyperBF, that may have implications for neurobiology.

3.1 Factorizable Radial Basis Functions

The synthesis of (weighted) radial basis functions in high dimensions may be easier if they are factorizable. It is easily seen that *the only radial basis function which is factorizable is the Gaussian (with diagonal* **W***).* A multidimensional Gaussian function can be represented as the product of lower dimensional Gaussians. For instance a 2D Gaussian radial function centered in t can be written as:

$$G(\|\mathbf{x}-\mathbf{t}\|_W^2) \equiv e^{-\|\mathbf{x}-\mathbf{t}\|_W^2} = e^{-\frac{(x-t_x)^2}{2\sigma_x^2}} e^{-\frac{(y-t_y)^2}{2\sigma_y^2}} , \tag{5}$$

with $\sigma_x = 1/w_1$ and $\sigma_y = 1/w_2$, where w_1 and w_2 are the elements of the matrix **W** assumed, in this section, to be diagonal.

This dimensionality factorization is especially attractive from the physiological point of view, since it is difficult to imagine how neurons could compute $G(\|\mathbf{x}-\mathbf{t}_\alpha\|^2)$. The scheme of figure 2, on the other hand, is physiologically plausible. Gaussian radial functions in one, two and possibly three dimensions can be implemented as *receptive fields* by weighted connections from the sensor arrays (or some retinotopic array of units representing with their activity the position of features). Gaussians in higher dimensions can then be synthesized as products of one and two dimensional receptive fields.

This scheme has three additional interesting features:

1. the multidimensional radial functions are synthesized directly by appropriately weighted connections from the sensor arrays, without any need of an explicit computation of the norm and the exponential.

2. 2D Gaussians operating on the sensor array or on a retinotopic array of features extracted by some preprocessing transduce the implicit position of features in the array into a number (the activity of the unit).

3. 2D Gaussians acting on a retinotopic map can be regarded each as representing one 2D "feature",i.e. a component of the input vector, while each center represents the "template", resulting from the conjunction of those lower-dimensional features. Notice that in this analogy the radial basis function is the AND of several features and could also include the negation of certain features, that is the AND NOT of them. **W** weights the importance of the different features.

3.2 Biophysical Mechanisms

3.2.1 The network

The multiplication operation required by the previous interpretation of Gaussian GRBFs to perform the "conjunction" of Gaussian receptive fields is not too implausible from a biophysical point of view. It could be performed by several biophysical mechanisms (see Koch and Poggio, 1987). Here we mention three mechanisms:

1. inhibition of the silent type and related circuitry (see Torre and Poggio, 1978; Poggio and Torre, 1978)
2. the AND-like mechanism of NMDA receptors
3. a logarithmic transformation, followed by summation, followed by exponentiation. The logarithmic and exponential characteristic could be implemented in appropriate ranges by the sigmoid-like pre-to-postsynaptic voltage transduction of many synapses.

If the first or the second mechanism is used, the product of figure 3 can be performed directly on the dendritic tree of the neuron representing the corresponding radial function (alternatively, each dendritic tree may perform pairwise products only, in which case a logarithmic number of cells would be required). The scheme also requires a certain amount of memory per basis unit, in order to store the center vector. In the case of Gaussian receptive fields used to synthesize Gaussian radial basis functions, the center vector is effectively stored in the position of the 2D (or 1D) receptive fields and in their connections to the product unit(s). This is plausible physiologically.

The linear terms (the direct connections from the inputs to the output in figure 1) can be realized directly as inputs to the output neuron that summates linearly its synaptic inputs (an output nonlinearity is allowed and will not change the basic form of the model, see Poggio and Girosi, 1989). They may also be realized through intermediate linear units.

3.2.2 Mechanisms for learning

Do the update schemes have a physiologically plausible implementation? Consider first the steepest descent methods, which require derivatives. Equation (6) or a somewhat similar, quasi-hebbian scheme is not too unlikely and may require only a small amount of neural circuitry. Equation (7) seems more difficult to implement for a network of real neurons.

Methods such as the random descent method, which do not require calculation of derivatives are biologically much more plausible and seem to perform very well in preliminary experiments. In the Gaussian case, with basis functions synthesized through the product of Gaussian receptive fields, moving the centers means establishing or erasing connections to the product unit. A similar argument can be made also about the learning of the matrix **W**. Notice that in the diagonal Gaussian case the parameters to be changed are exactly the σ of the Gaussians, i.e. the spread of the associated receptive fields. Notice also that the σ for all centers on one particular dimension is the same, suggesting that the learning of w_i may involve the modification of the scale factor in the input arrays rather than a change in the dendritic spread of the postsynaptic neurons.

In all these schemes the real problem consists in how to provide the "teacher" input (but see figure 5).

4 Visual Recognition of 3D Objects and Face sensitive Neurones

We have recently suggested and demonstrated how to use a HyperBF network to learn to recognize a 3D object. This section reviews very briefly this work (Poggio and Edelman, 1990) and then suggests that the brain may use a similar strategy. Face sensitive neurons are discussed as a specific instance.

4.1 HyperBF networks for recognizing 3D objects

A 3D object gives rise to an infinite variety of 2D images or views, because of the infinite number of possible poses relative to the viewer, and because of arbitrarily different illumination conditions. Is it possible to synthesize a module that can recognize an object from any viewpoint, after it learns its 3D structure from a small set of perspective views? We have have recently shown (Poggio and Edelman, 1990) that the HyperBF scheme may provide a solution to the problem provided that relatively stable and uniquely identifiable features (that we will call "labeled" features) can be extracted from the image.

In our scheme a view is represented as a $2N$ vector $x_1, y_1, x_2, y_2, \ldots, x_N, y_N$ of the coordinates on the image plane of N labeled and visible feature points on the object. We assume that a view of an object is a vector of this type (instead of position in the image of feature points we have also used angles between corners and length of segments or both), in general augmented by components that represent other properties of the object not necessarily related to its geometric shape, such as color or texture. We also assume that the function that maps the views into $0, 1$ (0 if the view is of another object, 1 if the view is of the correct object) can be approximated by a smooth function (if this were false, one could approximate the mapping from the view to a "standard" view and then apply a radial function to the result, see Poggio and Edelman, 1990).

The network used for this task is shown in Figure 3 (see also Figure 4). In the simplest version (fixed centers) the centers correspond to some of the examples, i.e. some views of the object. Updating the centers is equivalent to modifying the corresponding "prototypical views". Updating the weights of the matrix $\mathbf{W}$ corresponds to changing the relative importance of the various features that define the views of an object. This is important in the case in which these features are of a completely different type: a large w indicates a larger weight in the feature in the measure of similarity and is equivalent to a small σ in the Gaussian function. Features with a small role have a very large σ: their exact position or value does not matter much.

An interesting conclusion of this work consists of the small number of views that is required to recognize an object from the infinite number of possible views. The results clearly show that the scheme avoids the main problem of look-up table schemes, the explosion in the number of entries. Furthermore, the performance of the HyperBF recognition scheme resembles human performance in a related task. As discussed in Poggio and Edelman (1990), the number of training views necessary to achieve an acceptable recognition rate on novel views, 80-100 for the full viewing sphere, is broadly compatible with the finding that people have trouble recognizing a novel wire-frame object previously seen from one viewpoint if it is rotated away from that viewpoint by about 30^o (it takes 72 $30^o \times 30^o$ patches to cover the viewing sphere).

Recently, Buelthoff and Edelman (1990) have obtained interesting psychophysical results that support this model for human recognition of a certain class of 3D objects against other possible models. In general, the experimental results fit closely the prediction of theories of the 2D interpolation variety and appear to contradict theories that involve 3D models.

4.2 Face sensitive neurones

The HyperBF recognition scheme we have outlined has suggestive similarities with some of the data about visual neurons responding to faces obtained by Perrett and coworkers recording from the temporal association cortex (see Perrett et al., 1987 and references therein, Poggio and Edelman, 1990). Let us consider the network of figure 3 as the skeleton for a model of the circuitry involved in the recognition of faces. One expects different modules one for each different object of the type of the network of Figure 3. One also expects hyerarchical organizations: for instance a network of the HyperBF type may be used to recognize certain types of eyes and then may serve as input to another network involved in recognizing a certain class of faces, which may be itself one of the inputs to a network for a specific face. Different types of cells may then be expected. The overall output of a network for a specific face may be identified with the behavioral responses associated with recognition and may or may not coincide with an individual neuron. There should be cells or parts of cells corresponding to the centers, i.e. to the prototypes used by the networks. The response of these neurons should be a Gaussian function of the distance of the input to the template. These units would be somewhat similar to "grandmother" filters with a graded response, rather than binary detectors, each representing a prototype. They would be synthesized as the conjunction of, for instance, two-dimensional Gaussian receptive fields looking at a retinotopic map of features. During learning, the weights of the various prototypes in the network output are modified to find the optimal values that minimize the overall error. The prototypes themselves are slowly changed to find optimal prototypes for the task. The weights of the different input features is also modified to perform task-dependent dimensionality reduction.

Some of these expectations are consistent with the experimental findings of Perret et al. (1987). Some of the neurons described have several of the properties expected from the units of a HyperBF network with a center, i.e. a prototype that corresponds to a view of a specific face.

Some of the main data (from Perret et al., 1987 and references therein)

- The majority of cells responsive to faces are sensitive to the general characteristics of the face and they are somewhat invariant to its exact position and attitude.

- Presenting parts of the face in isolation revealed that some of the cells responded to different subsets of features: some cells are more sensitive to parts of the face such as eyes or mouth.

- There are cells selective for a particular view of the head. Some cells were maximally sensitive to the front view of a face, and their response fell off as the head was rotated into the profile view, and others were sensitive to the profile view with no response to the front view of the face.

- There are cells that are specific to the views of one individual. It seems that for each known person there would be a set of 'face recognition units'. Our model applies most directly to these neurons.

5 Theories of the Cerebellum and of Motor Control

5.1 Marr's and Albus models of the Cerebellum

The cerebellum is a part of the brain that is important in the coordination of complex muscle movements. The neural organization of the cerebellum is highly regular and well known (see Figure 5). Marr (1969) and Albus (1972) modeled the cerebellum as a look-up table. The critical part of their theories is the assumption that the synapses between the parallel fibers and the Purkinje cells are modified as a function of the Purkinje cell activity *and* the climbing fibres input. I suggest (see figure 5) that the cerebellum is a HyperBF network or set of networks (one for each Purkinje cell). Instead of a simple look-up table, the cerebellum would be a *function approximation module* (in a sense, "an approximating look-up table"). In our conjecture, basket and Golgi cells would have different roles from the roles assumed in the Marr-Albus theory. In particular, the Golgi cells, which receive inputs from the parallel fibers and whose axons synapse on the granule cells-mossy fibers clusters, may be used to change the norm weights **W**.

Key assumptions

- granule cells correspond to basis units (there may be as many as 200,000 granule cells per Purkinje cell) representing as many "examples"