BEWARE: These are preliminary notes. In the future, they will become part of a textbook on Visual Object Recognition.

Chapter 10: First steps towards *in silico* vision

We have been traveling through the wonderful territory of visual cortex, examining the properties of different brain areas and neuronal circuits, how neurons respond to visual stimuli and their transformations and how neurons communicate with each other. It is now time to try to put all this biological knowledge into a theory of visual recognition and instantiate this theory through a computational model that aims to perform visual recognition. En route towards this goal, here we take initial steps to describe how scientists describe neuronal circuits in computational models[1].

10.1. Why bother with computational models?

I have to start by admitting that I am quite biased here. I think that Theoretical Neuroscience and Computational Neuroscience are *essential* to rigorously understand how neuronal circuits work. The language of Science is Mathematics. Any description that is not rigorously substantiated by mathematical thinking is weak and prone to failure.

In the course of doing science, designing experiments and interpreting the results, we implicitly assume the validity of several hypotheses and make multiple assumptions. Quantitative models force us to think about and formalize these hypotheses and assumptions. This process of explicitly stating the assumptions can help design better experiments, discover logical flaws in our thinking and further understand the results.

It is often the case that the same questions or related questions are posed and attacked from different angles, using different experimental systems or the same systems in different laboratories. It is not always trivial to compare the results across different reports[2]. Quantitative models can integrate and summarize observations across experiments, resolutions and laboratories. Sometimes seemingly unrelated observations can be linked together and interpreted using a common powerful framework. Sometimes a quantitative

[1] This is by no means intended to be an exhaustive sample or presentation of work in Theoretical and Computational Neuroscience. We would like to illustrate some of the ideas, questions, tools, methodologies and results in Computational Neuroscience. There are several books that I would recommend for those who are interested in learning more Dayan, P., and Abbott, L. (2001). Theoretical Neuroscience (Cambridge: MIT Press), Gabbiani, F., and Cox, S. (2010). Mathematics for Neuroscientists (London: Academic Press), Hertz, J., Krogh, A., and Palmer, R. (1991). Introduction to the theory of neural computation (Santa Fe: Santa Fe Institute Studies in the Sciences of Complexity), Koch, C. (1999). Biophysics of Computation (New York: Oxford University Press)..

[2] Consider the following simple question. What is multi-unit activity? Wildly different definitions permeate through the literature.

theoretical framework can help explain intriguing differences across apparently similar experiments. A model can point to important missing data, critical information and decisive experiments.

A good model can lead to (non-intuitive) experimental predictions. It is often the case that experimentalists rightly or wrongly believe that they can come up with predictions for the next set of experiments based on their intuitions. And often enough, this is certainly the case. Yet, intuition sometimes fails (unfortunately). The power of abstraction is sometimes critical to be able to extrapolate and push the frontiers of knowledge.

A quantitative model, implemented through simulations, can be useful from an engineering viewpoint. Consider for example, the problem of face recognition. A theoretical model that describes how the primate visual cortex identifies faces can lead to a computational algorithm of diverse and wide applicability.

10.2. Do I have to be a "professional theoretician" to build a model?

No, no and no! I have often encountered brilliant scientists that seem to be afraid of adventuring into the wonderful lands of computational models and theoretical neuroscience. One of the reasons may be the perennial and lamentable fear towards mathematics. Yet in other cases, scientists believe that they have to be "professional theoreticians" to build quantitative models. I would like to strongly argue against this notion. Some of the most provocative, insightful and pioneering computational models have come from people who probably do not consider themselves theoreticians and who spend most of their lives perfecting insightful experiments. One could provide a very long list of neat theoretical and computational insights provided by experimentalists. As a brief sample, see (Blake, 1989; Brincat and Connor, 2006; Carandini et al., 1997; Hubel and Wiesel, 1962; Laurent, 2002; Prinz AA, 2004).

10.3. Example: a model for orientation tuning in simple cells in primary visual cortex

I would like to spend a few minutes to dig deeper into the misconception that models are built exclusively by theoreticians through an example of illuminating theoretical ideas coming from brilliant experimentalists. Consider the pioneering work of Hubel and Wiesel discussed in **Chapter 3** (Hubel and Wiesel, 1962). They recorded the activity of neurons in the macaque and cat primary visual cortex and discovered that neurons are particularly tuned to a bar of a certain orientation within their receptive fields. In addition to the neat and careful description of the empirical findings, they went on to propose a simple model of how orientation tuning could arise. They considered a feed-forward model that pooled the activity of multiple units in the lateral geniculate nucleus (LGN) with circular center-surround receptive fields. They proposed that orientation tuning in

V1 arises by combining the activity of LGN units with receptive fields that are aligned along the preferred orientation of the V1 unit. Hubel and Wiesel further went on to propose a model that could explain the differences between so-called simple and complex cells in V1. There has been a large body of computational work in the field trying to describe the activity of V1 units. Yet, the insights of Hubel and Wiesel have played a key role in inspiring generations of experimentalists and theoreticians alike. Many computational models of vision today can trace their roots to the models proposed by Hubel and Wiesel (e.g. (Carandini et al., 2005; Fukushima, 1980; Serre et al., 2007a)).

10.4. A nested family of single neuron models

Multiple models have been proposed and used to describe the activity of individual neurons. These models range from the use of filter operations to describe the firing rate of a neuron all the way to simulations that include dendritic spines and even individual ionic channels. Roughly speaking, we can distinguish the following categories:

- Filter models
- Integrate-and-fire models
- Hodgkin-Huxley models
- Multi-compartmental models
- Models including dendritic subcompartments, spines, ion channels and perhaps even realistic geometries

As we move from filter operations towards realistic geometries and models including individual channels, there is a significant increase in the biological accuracy of the model. Analytical solutions[3] become more complicated or even nonexistent as we increase the complexity of the model. There is also an increase in the computational cost of the simulations as we move towards more complex models. It should be noted that biologically more accurate and more complex models are not necessarily better[4]. As the famous Argentinean writer Borges once said: “To think is to forget a difference, to generalize, to abstract”.

10.5. Geometrically accurate models vs. spherical cows with point masses

If we want to model the activity of a neuron, there are several questions that we need to think about. The answers to some of these questions may depend on which specific aspects of the neuronal responses we are interested in capturing. Let us consider a simple analogy. If you want to understand how an object of mass *m*, say a cow, will accelerate as you apply some force *F*, you can

[3] An equation (e.g. a differential equation) is said to have an *analytical solution* if we can explicitly write down an expression that represents the solution. Many equations are solvable even if there is no analytical solution.

[4] See the delightful short story on the value and accuracy of maps: http://www.sccs.swarthmore.edu/users/08/bblonder/phys120/docs/borges.pdf

consider a very simple model that assumes that the object is a point mass, that is, that the entire mass is concentrated on a point where you apply the force and write a one parameter model *F* = *m* . *a*. Now, you and I are well aware that cows are not point masses. Yet, this very simple model can capture essential ingredients of the problem and it can even help us understand that some of the same principles behind the cow's movement also explain the movement of the planets[5].

In a similar vein, theoreticians often think of neurons as spherical or ignore their shape, their dendrites and axons. To follow up on the Hubel-Wiesel example mentioned earlier, one can model the activity of individual V1 neurons as a filter operation on the visual input and describe several aspects of the V1 responses without getting into the details of dendritic computation, biophysics of action potential generation and other interesting neuronal properties (e.g. (Carandini and Heeger, 1994; Hubel and Wiesel, 1968)).

Yet sometimes it may be critical to consider multiple compartments, such as a soma, an axon and one or a few dendrites. Depending on the question, other modelers rightly argue that we need to pay attention to the exact 3D shape of every single branch and the spatial distribution of spines and synapses on each branch. Einstein famously stated: "Make things as simple as possible, but not simpler…" Easier said than done.

10.6. The leaky integrate-and-fire model

The leaky integrate-and-fire model is arguably one of the most often used models for single units in computational neuroscience. It dates back to 1907 (Lapicque[6]). The simplest instantiation of a leaky I&F model is given by an RC circuit. A current *I(t)* is integrated through a capacitance (*C*) and is leaked through a resistance (*R*). The dynamics of the intracellular voltage *V(t)* can be described by:

$$C\frac{dV(t)}{dt} = -\frac{V(t)}{R} + \mathrm{I(t)} \ .$$

Whenever the voltage crosses a threshold, a spike is emitted, the voltage is reset and an absolute refractory period is imposed. This highly oversimplified version of a real neuron captures some of our most basic intuition about neuronal integration. Synaptic inputs are conveyed from dendrites onto the soma, where information is integrated and an output action potential is generated when the somatic voltage exceeds a threshold. This oversimplified model does not capture several phenomena including spike rate adaptation, multiple compartments and

[5] Although a trivial point, it should be noted that this one-parameter model does not do a very good job in describing the movement of the cow, in the presence of friction and other realistic variables. Plus, nowadays, animal activists could get upset upon considering that cows are point masses.

[6] See translated version: Lapicque L. Quantitative investigations of electrical nerve excitation treated as polarization. Biological Cybernetics (2007) 97:341-349.

spike generation outside the soma, sub-millisecond biophysics, neuronal geometry and many other important nuances of neurons.

It is quite straightforward to write code to simulate the dynamic behavior of integrate-and-fire units. A lot of people in Computational Neuroscience write code in a language called MATLAB, which is pretty easy[7]. Here is an ultra-simple (and not entirely correct for the aficionados) implementation of the integrate-and-fire unit.

```
V(1)=V_res;                 % initial voltage
for t=2:n                   % for each time in the simulation from 1 to n
        V(t)=V(t-1)+(dt/tau_m) * (E_L - V(t-1) + R_m * I_e(t));
        % Change in voltage at time t
        if (V(t)>V_th)      % If V(t) is above threshold V_th
                spk(t)=1;   % Emit a spike
                V(t)=V_res; % And reset the voltage to a value V_res
        end
end
```

In just a couple of lines, one can simulate a very simple differential equation and create spikes (`spk`) in response to arbitrary input currents (given by `I_e(t)`). As an example, you can set `E_L=-65 mV`, `V_res=E_L`, `V_th=-50 mV`, `tau_m=10 ms`, `R_m=10 Mohm`, `n=1000 time steps`, `dt=0.1 ms`. You can play with different input patterns (e.g. `I_e=2+3*randn(n,1)`)[8]. The shape of the action potentials is not modeled in the I&F unit. In some of the slide figures presented in the lecture, the action potential shape was artificially imposed every time the voltage crosses threshold. The I&F model can describe some of the basic instantaneous firing properties of cortical neurons. For example, when current is injected into a pyramidal neuron in cat primary visual cortex the initial firing rate computed from the first two spikes can be well approximated by an I&F model. Real neurons are fancier devices. Among other properties, neurons show adaptation and the firing rate for subsequent spikes (beyond the first two spikes) is not well described by the simple integrate-and-fire model (but adjustments can be made to describe adaptation (Gabbiani and Cox, 2010; Koch, 1999)).

10.7. The Hodgkin-Huxley model

In another remarkable example of powerful intuition and computational principles described by the experimenters collecting the data, Hodgkin and Huxley provided some of the fundamental insights into the generation of action potentials, even well before much of the subsequent biological characterization of different ionic channels (Hodgkin and Huxley, 1952).

[7] If you are interested in learning to program in MATLAB, there are lots of easy-to-follow tutorials. For example, see http://www.mathworks.com/academia/student_center/tutorials/launchpad.html

[8] You can download annotated MATLAB code and related materials from: http://tinyurl.com/3mza84y

The model incorporates the key sodium and potassium currents that are responsible for membrane depolarization and repolarization and characterize the shape of the action potential:

$$I(t) \ = C\frac{dV}{dt} + \overline{g}_L(V - E_L) + \overline{g}_K n^4(V - E_K) + \overline{g}_{Na} m^3 h(V - E_{Na})$$

where E_L, E_K and E_{Na} reflect the leak, potassium and sodium reversal potentials. g_L is the leak conductance. $\overline{g}_K n^4$ describes the time and voltage-dependent potassium conductance and $\overline{g}_{Na} m^3 h$ describes the time and voltage-dependent sodium conductance.

Again, it is relatively straightforward to write the necessary MATLAB code to simulate the dynamics in a Hodgkin-Huxley model unit[9]. The Hodgkin-Huxley model provides a significantly richer and more sophisticated view of intracellular voltage dynamics compared to the simpler integrate-and-fire models. Hodgkin-Huxley neuron models are also widely used when exploring the properties of neurons and neuronal networks.

10.8. Network models

Even highly oversimplified neurons can perform interesting computations when connected in sophisticated ways. Collective computation refers to the emergent functional properties of a group of interconnected neurons. Ultimately, to understand the output of a complex system like the brain, we need to think about circuits of neurons and their interactions. Intuition often breaks down quickly when considering the activity of the circuit as a whole and network models can provide help to understand the emergence of properties of the circuit as a whole.

To study fluid mechanics, one can abstract from the details of the collisions and trajectories of individual molecules in the fluid and instead characterize properties of the fluid such as temperature and viscosity. Similarly, most network models idealize and simplify the component units.

Artificial networks can be built from simple electronic devices (operational amplifiers replace neurons; cables, resistors and capacitors replace axons, dendrites and synapses). The dynamics of such systems can also be simulated in computers (or clusters of computers).

Some basic definitions and nomenclature can be helpful here in understanding the discussions in the literature. Theoreticians often describe

[9] I recommend the neat book by Gabbiani and Cox, which includes a lot of MATLAB code for multiple different types of neuronal models: Gabbiani F, Cox S (2010) Mathematics for Neuroscientists. London: Academic Press.

circuits of interconnected neurons, sometimes connected in an all-to-all fashion, sometimes organized into layers. The layers may be organized hierarchically (e.g. in lose analogy to the hierarchical organization of the visual system). In such a hierarchy, one can distinguish between bottom-up connections (from a unit in a lower layer to a unit in a higher layer), top-down connections (from a unit in a higher layer to a unit in a lower layer) and horizontal or recurrent synapses (connecting units in the same layer). The models may incorporate excitatory units or both excitatory and inhibitory units.

10.9. Firing rate network models

Firing rate network models constitute a simple yet important class of circuits. In the simplest instantiation, consider a feed-forward circuit with N units projecting to a given output unit. The input activity is given by the vector $\boldsymbol{u}$. We can think of the components of $\boldsymbol{u}$ as the firing rate of each input unit. The output firing rate is given by v (v is a scalar). A synaptic kernel K_s describes how the input firing rate is (linearly) converted into an input current for the output unit. Theoreticians often represent the strength of a given synapse b ($b=1,...,N$) by a scalar value w_b. This value could represent a combination of the probability of synaptic release from the pre-synaptic neuron and the amplitude of the post-synaptic potential (positive or negative) evoked by the incoming neurotransmitters. The total input to the output unit I_s is given by:

$$I_s = \sum_{b=1}^{N} w_b \int_{-\infty}^{t} d\tau K_s(t-\tau) u_b(\tau)$$

where w_b represents the weight or strength of each synapse. Using an exponential kernel, the dynamics of this circuit can be described by:

$$\tau_s \frac{dI_s}{dt} = -I_s + \sum_{b=1}^{N} w_b u_b$$

The firing rate of the output unit is usually a non-linear function of the total input current: $v = F(I_s)$. F could be a sigmoid function or a rectifying threshold function.

10.10. The perceptron and gradient descent

So far, I have not quite told you how the weights $\boldsymbol{w}$ are set in the previous model. Now I would like to give you an example of a way of learning those weights that illustrates an interesting computation that this type of simple circuit can perform. The set of weights can be learnt to perform some interesting computation. For example, in one of the earliest instantiations of a biologically-inspired computational algorithm the *perceptron* can be trained to perform interesting classification tasks. Imagine that we have some data that you want to classify into two possible groups. For example, there may be a collection of face images and you want to separate them into male faces and female faces. Each image, indexed by *m*, is a matrix of grayscale values that can be vectorized and represented by $\boldsymbol{u}_m$. With each image, we have an associated label v_m=+1 (female) or -1 (male). We have a training set[10] consisting of multiple such examples. In the two-layer perceptron network, we will consider the input to the output unit given by $\mathbf{w}.\mathbf{u}$. The output v will take the value +1 if $\mathbf{w}.\mathbf{u}-\gamma \geq 0$ and -1 otherwise. The perceptron learning rule tells us how to choose the weights $\boldsymbol{w}$ to minimize the error in this classification task (Bishop, 1995).

Instead of a binary classification task, we may be interested in approximating a given output function *h(s)* (for example, *h(s)* could represent the firing rate of a neuron in cortex in response to a stimulus s). Gradient descent refers to changing $\boldsymbol{w}$ so as to minimize the error in this task by moving $\boldsymbol{w}$ along the direction of greatest change in the error $\mathbf{w} \rightarrow \mathbf{w}+\epsilon \nabla_{w} E$.

10.11. Example: digit recognition in a feed-forward network trained by gradient descent

As an example of the application of some of these ideas, consider the task of learning to recognize hand-written digits from 0 to 9. LeCun and colleagues developed a simple feed-forward network, trained by gradient descent, that can perform this task quite well (LeCun et al., 1998).

10.12. Extreme biological realism: the "blue brain" project

Many biologists strongly feel that oversimplified networks like the ones just described fail to capture the complexity and richness of neurobiological circuitry. At the other end of the spectrum in network models, one encounters efforts like the "blue brain" project (Markram, 2006). This project aims to introduce a significant amount of biological realism using complex and intensive network simulations. These networks often have large number of free parameters given that we still do not have sufficient data to constrain realistic models. The brief discussion above regarding the appropriate level of abstraction and realism in modeling single neurons is equally applicable here in the context of network models.

10.13. Algorithms and methods for data analysis

Many computational neuroscientists are also interested in the development of tools and resources to quantitatively and rigorously analyze neural data. I am not going to spend much time in this lecture describing these efforts but they do represent an important and rich repertoire of work. As a non-exhaustive list of such efforts, some people are interested in the time-frequency analysis of neural signals (e.g. (Pesaran et al., 2002)), in the development of algorithms to perform spike sorting (e.g. (Lewicki, 1998)), in decoding the activity

[10] In this type of exercise, it is always very important to separate the data into a training set (used to fit parameters) and a test set (used to evaluate performance). This is referred to as cross-validation. Without cross-validation, training may lead to overfitting the data without power to generalize.

of neural ensembles (e.g. (Hung et al., 2005; Wilson and McNaughton, 1993)), in using information theory or other approaches to characterize neural coding (e.g. (Bialek et al., 1991; Gabbiani et al., 1996)).

10.14. Example: computational models of visual recognition

I will focus now on how some of the definitions and ideas above are applied in the context of a specific problem, namely, how to understand the computations underlying visual object recognition. I will summarize some of the initial steps towards a theoretical understanding of the computational principles behind transformation-invariance visual recognition in the primate cortex.

10.14.1 Defining the problem

We start by defining what needs to be explained and the necessary constraints to solve the problem. A theory of visual object recognition, implemented by a computational model, should be able to explain the following phenomena and have the following characteristics:

1. *Selectivity*. The primate visual system shows a remarkable degree of selectivity and can differentiate among shapes that appear to be very similar at the pixel level (e.g. arbitrary 3D shapes created from paperclips, different faces, etc.). Critical to object recognition, a model should be able to discriminate among physically similar but distinct shapes.
2. *Transformation tolerance*. A trivial solution to achieve high selectivity would be to memorize all the pixels in the object. The problem with this type of algorithm is that it would not tolerate any changes in the image. An object can cast an infinite number of projections onto the retina. These changes arise due to changes in object position with respect to fixation, object scale, plane or depth rotation, changes in contrast or illumination, color, occlusion and others. The importance of combining selectivity and tolerance has been emphasized by many investigators (e.g. (Deco and Rolls, 2004b; Logothetis and Sheinberg, 1996; Olshausen et al., 1993; Riesenhuber and Poggio, 1999; Rolls, 1991; Serre et al., 2007a) among others).
3. *Speed*. Visual recognition is very fast, as emphasized by many psychophysical investigations (Kirchner and Thorpe, 2006; Potter and Levy, 1969; Serre et al., 2007b), scalp EEG measurements (Thorpe et al., 1996) and neurophysiological recordings in humans (Liu et al., 2009) and monkeys (e.g. (Hung et al., 2005; Keysers et al., 2001; Richmond et al., 1983) among others). This speed imposes an important constraint to the number of computational steps that the visual system can use for pattern recognition (Rolls, 1991; Serre et al., 2007a).
4. *Generic*. We can recognize a large variety of objects and shapes. Estimates about the exact number of objects or object categories that primates can discriminate vary widely depending on several assumptions and extrapolations (e.g. (Abbott et al., 1996; Biederman, 1987; Brady et

al., 2008; Standing, 1973)). Certain types of shapes may be particularly interesting, they may have more cortical real estate associated with them, they could be processed faster and could be independently impaired. For example, there has been extensive discussion in the literature about faces, their representation and how they can be different from other visual stimuli. Yet, independently of precise figures about the number of shapes that primates can discriminate and independently also of whether natural objects and faces are special or not, it is clear that there exists a generic system capable of discriminating among multiple arbitrary shapes. For simplicity and generality, we focus first on this generic shape recognition problem. Face recognition, or specialization for natural objects versus other shapes constitute interesting and important specific instantiations and sub problems of the general one that we try to address here.

5. *Implementable in a computational algorithm*. A successful theory of visual object recognition needs to be described in sufficient detail to be implemented through computational algorithms. This requirement is important because the computational implementation allows us to run simulations and hence to quantitatively compare the performance of the model against behavioral metrics. The simulations also lend themselves to a direct comparison of the model's computational steps and neurophysiological responses at different stages of the visual processing circuitry. The algorithmic implementation forces us to rigorously state the assumptions and formalize the computational steps; in this way, computational models can be more readily compared than "armchair" theories and models. The implementation can also help us debug the theory by discovering hidden assumptions, bottlenecks and challenges that the algorithms cannot solve or where performance is poor. There are multiple fascinating ideas and theories about visual object recognition that have not been implemented through computational algorithms. These ideas can be extremely useful and helpful for the field and can inspire the development of computational models. Yet, we emphasize that we cannot easily compare theories that can be and have been implemented against other ones that have not.

6. *Restricted to primates*. Here we restrict the discussion to object recognition in primates. There are strong similarities in visual object recognition at the behavioral and neurophysiological levels between macaque monkeys (one of the prime species for neurophysiological studies) and humans (e.g. (Kriegeskorte et al., 2008; Liu et al., 2009; Logothetis and Sheinberg, 1996; Myerson et al., 1981; Nielsen et al., 2006; Orban, 2004).

7. Biophysically plausible. There are multiple computational approaches to visual object recognition. Here we restrict the discussion to models that are biophysically plausible. In doing so, we ignore a vast literature in Computer Vision where investigators are trying to solve similar problems without direct reference to the cortical circuitry. These engineering approaches are extremely interesting and useful from a

practical viewpoint. Ultimately, in the same way that computers can become quite successful at playing chess without any direct connection to the way humans play chess, computer vision approaches can achieve high performance without mimicking neuronal circuits. Here we restrict the discussion to biophysically plausible algorithms.

8. *Restricted to the visual system*. The visual system is not isolated from the rest of the brain and there are plenty of connections between visual cortex and other sensory cortices, between visual cortex and memory systems in the medial temporal lobe and between the visual cortex and frontal cortex. It is likely that these connections also play an important role in the process of visual recognition, particularly through feedback signals that incorporate expectations (e.g. the probability that there is a lion in an office setting is very small), prior knowledge and experience (e.g. the object appears similar to another object that we are familiar with), cross-modal information (e.g. the object is likely to be a musical instrument because of the sound). To begin with and to simplify the problem, we restrict the discussion to the visual system.

10.14.2 Visual recognition goes beyond identifying objects in single images

We emphasize that visual recognition is far more complex than the identification of specific objects. Under natural viewing conditions, objects are embedded in complex scenes and need to be separated from their background. How this segmentation occurs constitutes an important challenge in itself. Segmentation depends on a variety of cues including sharp edges, texture changes and object motion among others. Some object recognition models assume that segmentation must occur prior to recognition. There is no clear biological evidence for segmentation prior to recognition and therefore this constitutes a weakness in such approaches. We do not discuss segmentation here (see (Borenstein et al., 2004; Sharon et al., 2006) for recent examples of segmentation algorithms).

Most object recognition models are based on studying static images. Under natural viewing conditions, there are important cues that depend on the temporal integration of information. These dynamic cues can significantly enhance recognition. Yet, it is clear that we can recognize objects in static images and therefore many models focus on the reduced version the pattern recognition problem using static objects. Here we also focus on static images.

We can perform a variety of complex tasks that rely on visual information that are different from identification. For example, we can put together images of snakes, lions and dolphins and categorize them as animals. Categorization is a very important problem in vision research and it also constitutes a formidable challenge for computer-based approaches. Here we focus on the question of object identification.

10.14.3 Modeling the ventral visual stream – Common themes

Several investigators have proposed computational models that aim to capture some of the essential principles behind the transformations along the primate ventral visual stream. Before discussing some of those models in more detail, we start by providing some common themes that are shared by many of these models.

The input to the models is typically an image, defined by a matrix that contains the grayscale value of each pixel. Object shapes can be discriminated quite well in grayscale images and, therefore, most models ignore the added complexities of color processing (but eventually it will also be informative and important to add color to these models). Because the focus is often on the computational properties of ventral visual cortex, several investigators often ignore the complexities of modeling the computations in the retina and LGN; the pixels are meant to coarsely represent the output of retinal ganglion cells or LGN cells. This is of course one of the many oversimplifications in several computation models given that we know that images go through a number of transformations before retinal ganglion cells convey information to the LGN and on to cortex (Meister, 1996).

Most models have a hierarchical and deep structure that aims to mimic the approximately hierarchical architecture of ventral visual cortex (Felleman and Van Essen, 1991; Maunsell, 1995). The properties of deep networks has received considerable attention in the computational world, even if the mathematics of learning in deep networks that include non-linear responses is far less understood than shallow counterparts (Poggio and Smale, 2003). It seems that neocortex and computer modelers have adopted a *Divide and Conquer* strategy whereby a complex problem is divided into many simpler tasks.

Most computational models assume, explicitly or implicitly, that cortex is cortex, and hence that there exist canonical microcircuits and computations that are repeated over and over throughout the hierarchy (Douglas and Martin, 2004; Riesenhuber and Poggio, 1999; Serre et al., 2007a).

As we ascend through the hierarchical structure of the model, units in higher levels typically have larger receptive fields, respond to more complex visual features and show an increased degree of tolerance to transformations of their preferred features.

10.14.4 A panoply of models

We summarize here a few important ideas that have been developed to describe visual object recognition. The presentation here is neither an exhaustive list nor a thorough discussion of each of these approaches. For a more detailed discussion of several of these approaches, see (Deco and Rolls, 2004a; LeCun et al., 1998; Riesenhuber and Poggio, 2002; Serre et al., 2005a; Ullman, 1996).

Straightforward template matching does not work for pattern recognition. Even shifting a pattern by one pixel would pose significant challenges for an algorithm that merely compares the input with a stored pattern on a pixel-by-pixel fashion. As noted at the beginning of this chapter, a key challenge to recognition is that an object can lead to infinite number of retinal images depending on its size, position, illumination, etc. If all objects were always presented in a standardized position, scale, rotation and illumination, recognition would be considerably easier (DiCarlo and Cox, 2007; Serre et al., 2007a). Based on this notion, several approaches are based on trying to transform an incoming object into a canonical prototypical format by shifting, scaling and rotating objects (e.g. (Ullman, 1996)). The type of transformations required is usually rather complex, particularly for non-affine transformations. While some of these problems can be overcome by ingenious computational strategies, it is not entirely clear (yet) how the brain would implement such complex calculations nor is there currently any clear link to the type of neurophysiological responses observed in ventral visual cortex.

A number of approaches are based on decomposing an object into its component parts and their interactions. The idea behind this notion is that there could be a small dictionary of object parts and a small set of possible interactions that act as building blocks of all objects. Several of these ideas can be traced back to the prominent work of David Marr (Marr, 1982; Marr and Nishihara, 1978) where those constituent parts were based on generalized cone shapes. The artificial intelligence community also embraced the notion of structural descriptions (Winston, 1975). In the same way that a mathematical function can be decomposed into a sum over a certain basis set (e.g. polynomials or sine and cosine functions), the idea of thinking about objects as a sum over parts is attractive because it may be possible and easier to detect these parts in a transformation-invariant manner (Biederman, 1987; Mel, 1997). In the simplest instantiations, these models are based on merely detecting a conjunction of object parts, an approach that suffers from the fact that part rearrangements would not impair recognition but they should (e.g. a house with a garage on the roof and the chimney on the floor). More elaborate versions include part interactions and relative positions. Yet, this approach seems to convert the problem of object recognition to the problem of object part recognition plus the problem of object parts interaction recognition. It is not entirely obvious that object part recognition would be a trivial problem in itself nor is it obvious that *any* object can be uniquely and succinctly described by a universal and small dictionary of simpler parts. It is not entirely trivial how recognition of complex shapes (e.g. consider discriminating between two faces) can be easily described in terms of a structural description of parts and their interactions. Computational implementations of these structural descriptions have been sparse (see however (Hummel and Biederman, 1992)). More importantly, it is not entirely apparent how these structural descriptions relate to the neurophysiology of the ventral visual cortex (see however (Vogels et al., 2001)).

A series of computational algorithms, typically rooted in the neural network literature (Hinton, 1992), attempt to build deep structures where inputs can be reconstructed (for a recent version of this, see e.g. (Hinton and Salakhutdinov, 2006). In an extreme version of this approach, there is no information loss along the deep hierarchy and backward signals carry information capable of re-creating arbitrary inputs in lower visual areas. There are a number of interesting applications for such "auto-encoder" deep networks such as the possibility of performing dimensionality reduction. From a neurophysiological viewpoint, however, it seems that the purpose of cortex is precisely the opposite, namely, to lose information in biologically interesting ways. It is not clear why one build an entire network to copy the input (possibly with fewer units). In other words, as emphasized at the beginning of this chapter, it seems that a key goal of ventral visual cortex is to be able to extract biologically relevant information (e.g. object identity) in spite of changes in the input at the pixel level.

Particularly within the neurophysiology community, there exist several "metric" approaches where investigators attempt to parametrically define a space of shapes and then record the activity of neurons along the ventral visual stream in response to these shapes (Brincat and Connor, 2004; Connor et al., 2007; Tanaka, 1996). This dictionary of shapes can be more or less quantitatively defined. For example, in some cases, investigators start by presenting different shapes in search of a stimulus that elicits strong responses. Subsequently, they manipulate the "preferred" stimulus by removing different parts and evaluating how the neuronal responses are modified by these transformations. While interesting, these approaches suffer from the difficulties inherent in considering arbitrary shapes that may or may not constitute truly "preferred" stimuli. Additionally, in some cases, the transformations examined only reveal anthropomorphic biases about what features could be relevant. Another approach is to define shapes parametrically. For example, Brincat and colleagues considered a family of curvatures and modeled responses in a six-dimensional space defined by a sum of Gaussians with parameters given by the curvature, orientation, relative position and absolute position of the contour elements in the display. This approach is intriguing because it has the attractive property of allowing investigators to plot "tuning curves" similar to the ones used to represent the activity of units in earlier visual areas. Yet, it also makes strong assumptions about the type of shapes preferred by the units. Expanding on these ideas, investigators have tried to start from generic shapes and use genetic algorithms whose trajectories are guided by the neuronal preferences (Yamane et al., 2008). What is particularly interesting about this approach is that it seems to be less biased than the former two. The key limitation here is the recording time and this type of algorithm, particularly with small data sets, may converge onto local minima or even not converge at all. Genetic algorithms can be more thoroughly examined in the computational domain. For example, investigators can examine a huge variety of computational models and let them "compete" with each other through evolutionary mechanisms (Pinto et al., 2009). To guide the evolutionary paths, it is necessary to define a cost function; for example, evolution can be constrained by rewarding models that achieve better performance in certain recognition tasks. This type of approach can lead to models with high performance (although it also suffers from difficulties related to local minima). Unfortunately, it is not obvious that better performance necessarily implies any better approximation to the way in which cortex solves the visual recognition problem.

10.14.5 Bottom-up hierarchical models of the ventral visual stream

A hierarchical network model can be described by a series of layers $i = 0,1,...,N$. Each layer contains $n(i) \times n(i)$ units arranged in a matrix. The activity of each unit in each layer can be represented by the matrix $\mathbf{x}_i$ ($\mathbf{x}_i \in \mathbb{R}^{n(i)\times n(i)}$). In several models, $x_i(j,k)$ (i.e., the activity of unit at position *j,k* in layer *i*) is a scalar value interpreted as the firing rate of the unit. The initial layer is defined as the input image; $\mathbf{x}_0$ represents the (grayscale) values of the pixels a given image.

Equations 1 and 2 above constitute the initial steps for many object recognition models and capitalize on the more studied parts of the visual system, the pathway from the retina to primary visual cortex. The output of Equation 2, after convolving the output of center-surround receptive fields with a Gabor function, can be thought of as a first order approximation to the edges in the image. As noted above, our understanding of ventral visual cortex beyond V1 is far more primitive and it is therefore not surprising that this is where most models diverge. In a first order simplification, we can generically describe the transformations along the ventral visual stream as:

$$\mathbf{x}_{i+1} = f_i(\mathbf{x}_i) \qquad \textbf{Equation 14.1}$$

This assumes that the activity in a given layer only depends on the activity pattern in the previous layer. This assumption implies that at least three types of connections are ignored: (i) connections that "skip" a layer in the hierarchy (e.g. synapses from the LGN to V4 skipping V1); (ii) top-down connections (e.g. synapses from V2 to V1 (Virga, 1989)) and (iii) connections within a layer (e.g. horizontal connections between neurons with similar preferences in V1 (Callaway, 1998)) are not included in **Equation 14.1**.

The sub index *i* in the function *f* indicates that the transformation from one layer to another is not necessarily the same. A simple form that *f* could take is a linear function:

$$\mathbf{x}_{i+1} = \mathbf{W}_i\mathbf{x}_i \qquad \textbf{Equation 14.2}$$

where the matrix **W***i* represents the linear weights that transform activity in layer *i* into activity in layer *i+1*. Not all neurons in layer *i* need to be connected to all neurons in layer *i+1*; in other words, many entries in **W** can be 0. This simple formulation fins some empirical evidence; for example, Hubel and Wiesel proposed that the oriented filters in primary visual cortex could arise from a linear summation of the activity of neurons in the lateral geniculate nucleus with appropriately aligned center-surround receptive fields (Hubel and Wiesel, 1962). Unfortunately, neurons are far more intricate devices and non-linearities abound in their response properties. For example, Hubel and Wiesel also described the activity of so-called complex cells that are also orientation tuned but show a non-linear response as a function of spatial frequency or bar length.

It is tacitly assumed by most modelers that there exist general rules, often summarized in the epithet "cortex is cortex", such that only a few such transformations are allowed. One of the early models that aimed to describe object recognition, inspired by the neurophysiological findings of Hubel and Wiesel, was the neocognitron (Fukushima, 1980) (see also earlier theoretical ideas in (Sutherland, 1968)). This model had two possible operations, a linear tuning function (performed by "simple" cells) and a non-linear OR operation (performed by "complex" cells). These two operations were alternated and repeated through the multiple layers in the deep hierarchy. This model demonstrated the feasibility of such linear/non-linear cascades in achieving scale and position tolerance in a letter recognition task. Several subsequent efforts (Amit and Mascaro, 2003; Deco and Rolls, 2004b; LeCun et al., 1998; Olshausen et al., 1993; Riesenhuber and Poggio, 1999; Wallis and Rolls, 1997) were inspired by the Neocognitron architecture.

One such effort to expand on the computational abilities of the Neocognitron in the computational model developed in the Poggio group (Riesenhuber and Poggio, 1999; Serre et al., 2005a; Serre et al., 2007a). This model is characterized by a purely feed-forward and hierarchical architecture. An image, represented by grayscale values, is convolved with Gabor filters at multiple scales and positions to mimic the responses of simple cells in primary visual cortex. Like other efforts, the model consists of a cascade of linear and non-linear operations. These operations come in only two flavors in the model: a tuning operation and soft-max operation. Both operations can be expressed in the following form:

$$x_{i+1}[k] = g\left(\frac{\sum_{j=1}^{n} w[j,k]\ x_i^p[j]}{\alpha + \left(\sum_{j=1}^{n} x_i^q[j] \right)^r} \right) \qquad \textbf{Equation 14.3}$$

where $x_{i+1}[k]$ represents the activity of unit *k* in layer *i+1*, $w[j,k]$ represents the connection weight between unit *j* in layer *i* and unit *k* in layer *b+1*, *p, q, r* are integer constants, *a* is a normalization constant and *g* is a nonlinear function (e.g. sigmoid). Depending on the values of *p*, *q* and *r* different interesting behaviors can be obtained. In particular, taking *r=1/2*, *p=1*, *q*=2, leads to a *tuning operation*:

$$x_{i+1}[k] = g\left(\frac{\sum_{j=1}^{n} w[j,k]\ x_i[j]}{\alpha + \sqrt{\sum_{j=1}^{n} x_i^2[j]}} \right) \qquad \textbf{Equation 14.3'}$$

The responses of the unit are controlled by the weights **w**. As emphasized above, tuning is a central aspect of any computational model of visual recognition, allowing units along the hierarchy to respond to increasingly more elaborate features. Taking **w**=1, *p=q+1*, *r=1*, leads to a softmax operation, particularly for large values of *q*:

$$x_{i+1}[k] = g\left(\frac{\sum_{j=1}^{n} x_i^{q+1}[j]}{\alpha + \sum_{j=1}^{n} x_i^q[j]} \right) \qquad \textbf{Equation 14.3''}$$

Of note, the unit with response $x_{i+1}[k]$ shows similar response tuning to the units with response $x_i[j]$ for $j=1,...,n$. Yet, the higher-level unit shows a stronger degree of tolerance to those aspects of the response that differentiate the responses of different units with similar tuning in layer *i*. For example, different units in layer *i* may show identical feature preferences but have slightly different receptive fields. A winner-take-all unit in layer i+1 that takes as input the responses of those units would inherit the same feature preferences but would reveal a larger receptive and tolerate changes in the position of the feature within the larger receptive field. Both operations can be implemented through relatively simple biophysical circuits (Kouh and Poggio, 2004).

This and similar architectures have been subjected to several tests including comparison with psychophysical measurements (e.g. (Serre et al., 2007b)), comparison with neurophysiological responses (e.g. (Cadieu et al., 2007; Deco and Rolls, 2004b; Hung et al., 2005; Lampl et al., 2004; Serre et al., 2005a) and quantitative evaluation of performance in computer vision recognition tasks (e.g. (LeCun et al., 1998; Mutch and Lowe, 2006; Serre et al., 2005b)).

10.14.6 Top-down signals in visual recognition

In spite of the multiple simplifications, the success of bottom-up architecture in describing a large number of visual recognition phenomena suggest that they may not be a bad first cut. As emphasized above, bottom-up architectures constitute only an approximation to the complexities and wonders of neocortical computation.One of the several simplifications in bottom-up models is the lack of top-down signals. We know that there are abundant back-projections in neocortex (e.g. (Callaway, 2004; Douglas and Martin, 2004; Felleman and Van Essen, 1991)). The functions of top-down connections have been less studied at the neurophysiological level but there is no shortage of computational models illustrating the rich array of computations that emerge with such connectivity. Several models have used top-down connections to guide attention to specific locations or specific features within the image (e.g. (Itti and Koch, 2001; Olshausen et al., 1993))(Chikkerur et al., 2009; Compte and Wang, 2006; Deco and Rolls, 2005; Rao, 2005; Tsotsos, 1990). The allocation of attention to specific parts of an image can significantly enhance recognition performance by alleviating the problems associated with image segmentation and with clutter.

Top-down signals can also play an important role in recognition of occluded objects. When only partial object information is available, the system must be able to perform object completion and interpret the image based on prior knowledge. Attractor networks have been shown to be able to retrieve the identity of stored memories from partial information (e.g. (Hopfield, 1982)). Some computational models have combined bottom-up architectures with attractor networks at the top of the hierarchy (e.g. (Deco and Rolls, 2004b)).

During object completion, top-down signals could play an important role by providing prior stored information that influences the bottom-up sensory responses. Several proposals have argued that visual recognition can be formulated as a Bayesian inference problem (Chikkerur et al., 2009; Lee and

Mumford, 2003; Mumford, 1992; Rao, 2004; Rao et al., 2002; Yuille and Kersten, 2006). Considering three layers of the visual cascade (e.g. LGN, V1 and higher areas), and denoting activity in those layers as $\mathbf{x}_0$, $\mathbf{x}_1$ and $\mathbf{x}_h$ respectively, then the probability of obtaining a given response pattern in V1 depends both on the sensory input as well as feedback from higher areas:

$$P(\mathbf{x}_1|\mathbf{x}_0) = \frac{P(\mathbf{x}_0|\mathbf{x}_1)P(\mathbf{x}_1|\mathbf{x}_h)}{P(\mathbf{x}_0|\mathbf{x}_h)}$$ **Equation 14.8**

where $P(\mathbf{x}_1|\mathbf{x}_h)$ represents the feedback biases conveying prior information. An intriguing idea proposed by Rao and Ballard argues that top-down connections provide predictive signals whereas bottom-up signals convey the difference between the sensory input and the top-down predictions (Rao and Ballard, 1999).

10.15. Hopfield networks

Hopfield developed a nice formalism to understand the properties of certain classes of networks (Hopfield, 1982; Tank and Hopfield, 1987). What is particularly attractive about these networks (no pun intended) is that there are emergent properties of the circuit that are not easy to identify or describe upon considering only individual units without paying attention to the interactions. The circuits can solve rather challenging computational problems and they have interesting properties such as robustness to perturbations and the possibility of performing pattern completion.

References

Abbott, L.F., Rolls, E.T., and Tovee, M.J. (1996). Representational capacity of face coding in monkeys. Cerebral Cortex 6, 498-505.

Amit, Y., and Mascaro, M. (2003). An integrated network for invariant visual detection and recognition. Vision research 43, 2073-2088.

Bialek, W., Steveninck, R., and Warland, D. (1991). Reading a neural code. Science 252, 1854-1857.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. Psychological Review 24, 115-147.

Bishop, C.M. (1995). Neural Networks for Pattern Recognition (Oxford: Clarendon Press).

Blake, R. (1989). A neural theory of binocular rivalry. Psychological Review 96, 145-167.

Borenstein, E., Sharon, E., and Ullman, S. (2004). Combining Top-Down and Bottom-Up Segmentation. In IEEE Conference on Computer Vision and Pattern Recognition.

Brady, T.F., Konkle, T., Alvarez, G.A., and Oliva, A. (2008). Visual long-term memory has a massive storage capacity for object details. Proc Natl Acad Sci U S A 105, 14325-14329.

Brincat, S., and Connor, C. (2006). Dynamic Shape Synthesis in Posterior Inferior Temporal Cortex. Neuron 49, 17-24.

Brincat, S.L., and Connor, C.E. (2004). Underlying principles of visual shape selectivity in posterior inferotemporal cortex. Nature neuroscience 7, 880-886.

Cadieu, C., Kouh, M., Pasupathy, A., Connor, C., Riesenhuber, M., and Poggio, T. (2007). A model of V4 shape selectivity and invariance. Journal of Neurophysiology 98, 1733-1750.

Callaway, E.M. (1998). Local circuits in primary visual cortex of the macaque monkey. Annu Rev Neurosci 21, 47-74.

Callaway, E.M. (2004). Feedforward, feedback and inhibitory connections in primate visual cortex. Neural Netw 17, 625-632.

Carandini, M., Demb, J.B., Mante, V., Tolhurst, D.J., Dan, Y., Olshausen, B.A., Gallant, J.L., and Rust, N.C. (2005). Do we know what the early visual system does? The Journal of neuroscience : the official journal of the Society for Neuroscience 25, 10577-10597.

Carandini, M., and Heeger, D.J. (1994). Summation and division by neurons in primate visual cortex. Science 264, 1333-1336.

Carandini, M., Heeger, D.J., and Movshon, J.A. (1997). Linearity and normalization in simple cells of the macaque primary visual cortex. The Journal of neuroscience : the official journal of the Society for Neuroscience 17, 8621-8644.

Chikkerur, S., Serre, T., and Poggio, T. (2009). A Bayesian inference theory of attention: neuroscience and algorithms. MIT-CSAIL-TR, ed. (Cambridge, MIT).

Compte, A., and Wang, X.J. (2006). Tuning curve shift by attention modulation in cortical neurons: a computational study of its mechanisms. Cereb Cortex 16, 761-778.

Connor, C.E., Brincat, S.L., and Pasupathy, A. (2007). Transformation of shape information in the ventral pathway. Current opinion in neurobiology 17, 140-147.

Dayan, P., and Abbott, L. (2001). Theoretical Neuroscience (Cambridge: MIT Press).

Deco, G., and Rolls, E.T. (2004a). Computational Neuroscience of Vision (Oxford Oxford University Press).

Deco, G., and Rolls, E.T. (2004b). A neurodynamical cortical model of visual attention and invariant object recognition. Vision research 44, 621-642.

Deco, G., and Rolls, E.T. (2005). Attention, short-term memory, and action selection: a unifying theory. Prog Neurobiol 76, 236-256.

DiCarlo, J.J., and Cox, D.D. (2007). Untangling invariant object recognition. Trends Cogn Sci 11, 333-341.

Douglas, R.J., and Martin, K.A. (2004). Neuronal circuits of the neocortex. Annu Rev Neurosci 27, 419-451.

Felleman, D.J., and Van Essen, D.C. (1991). Distributed hierarchical processing in the primate cerebral cortex. Cerebral Cortex 1, 1-47.

Fukushima, K. (1980). Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 36, 193-202.

Gabbiani, F., and Cox, S. (2010). Mathematics for Neuroscientists (London: Academic Press).

Gabbiani, F., Metzner, W., Wessel, R., and Koch, C. (1996). From stimulus encoding to feature extraction in weakly electric fish. Nature 384, 564-567.

Hertz, J., Krogh, A., and Palmer, R. (1991). Introduction to the theory of neural computation (Santa Fe: Santa Fe Institute Studies in the Sciences of Complexity).

Hinton, G. (1992). How neural networks learn from experience. Scientific American 267, 145-151.

Hinton, G.E., and Salakhutdinov, R.R. (2006). Reducing the dimensionality of data with neural networks. Science 313, 504-507.

Hodgkin, A.L., and Huxley, A.F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. Journal of Physiology 117, 500-544.

Hopfield, J.J. (1982). Neural networks and physical systems with emergent collective computational abilities. PNAS 79, 2554-2558.

Hubel, D.H., and Wiesel, T.N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of physiology 160, 106-154.

Hubel, D.H., and Wiesel, T.N. (1968). Receptive fields and functional architecture of monkey striate cortex. The Journal of physiology 195, 215-243.

Hummel, J.E., and Biederman, I. (1992). Dynamic binding in a neural network for shape recognition. Psychol Rev 99, 480-517.

Hung, C., Kreiman, G., Poggio, T., and DiCarlo, J. (2005). Fast Read-out of Object Identity from Macaque Inferior Temporal Cortex. Science 310, 863-866.

Itti, L., and Koch, C. (2001). Computational modelling of visual attention. Nat Rev Neurosci 2, 194-203.

Keysers, C., Xiao, D.K., Foldiak, P., and Perret, D.I. (2001). The speed of sight. Journal of Cognitive Neuroscience 13, 90-101.

Kirchner, H., and Thorpe, S.J. (2006). Ultra-rapid object detection with saccadic eye movements: visual processing speed revisited. Vision research 46, 1762-1776.

Koch, C. (1999). Biophysics of Computation (New York: Oxford University Press).

Kouh, M., and Poggio, T. (2004). A general mechanism for tuning: gain control circuits and synapses underlie tuning of cortical neurons. M.A. Memo, ed. (Cambridge, MIT).

Kriegeskorte, N., Mur, M., Ruff, D.A., Kiani, R., Bodurka, J., Esteky, H., Tanaka, K., and Bandettini, P.A. (2008). Matching categorical object representations in inferior temporal cortex of man and monkey. Neuron 60, 1126-1141.

Lampl, I., Ferster, D., Poggio, T., and Riesenhuber, M. (2004). Intracellular measurements of spatial integration and the MAX operation in complex cells of the cat primary visual cortex. J Neurophysiol 92, 2704-2713.

Laurent, G. (2002). Olfactory Network Dynamics and the coding of multidimensional signals. Nature Reviews Neuroscience 3, 884-895.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. Proc of the IEEE 86, 2278-2324.

Lee, T.S., and Mumford, D. (2003). Hierarchical Bayesian inference in the visual cortex. J Opt Soc Am A Opt Image Sci Vis 20, 1434-1448.

Lewicki, M.S. (1998). A review of methods of spike sorting: the detection and classification of neural action potentials. Network: Computation and Neural Systems 9, R53-R78.

Liu, H., Agam, Y., Madsen, J.R., and Kreiman, G. (2009). Timing, timing, timing: Fast decoding of object information from intracranial field potentials in human visual cortex. Neuron 62, 281-290.

Logothetis, N.K., and Sheinberg, D.L. (1996). Visual object recognition. Annual Review of Neuroscience 19, 577-621.

Markram, H. (2006). The blue brain project. Nat Rev Neurosci 7, 153-160.

Marr, D. (1982). Vision (Freeman publishers).

Marr, D., and Nishihara, H.K. (1978). Representation and recognition of the spatial organization of three-dimensional shapes. Proc R Soc Lond B Biol Sci 200, 269-294.

Maunsell, J.H.R. (1995). The brain's visual world: representatioin of visual targets in cerebral cortex. Science 270, 764-769.

Meister, M. (1996). Multineuronal Codes in Retinal Signaling. PNAS 93, 609-614.

Mel, B. (1997). SEEMORE: Combining color, shape and texture histogramming in a neurally inspired approach to visual object recognition. Neural Computation 9, 777.

Mumford, D. (1992). On the computational architecture of the neocortex. II. The role of cortico-cortical loops. Biol Cybern 66, 241-251.

Mutch, J., and Lowe, D. (2006). Multiclass Object Recognition with Sparse, Localized Features. In CVPR (New York), pp. 11-18.

Myerson, J., Miezin, F., and Allman, J. (1981). Binocular rivalry in macaque monkeys and humans: a comparative study in perception. Behavioral Analysis Letters 1, 149-159.

Nielsen, K.J., Logothetis, N.K., and Rainer, G. (2006). Discrimination strategies of humans and rhesus monkeys for complex visual displays. Current biology : CB 16, 814-820.

Olshausen, B.A., Anderson, C.H., and Van Essen, D.C. (1993). A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information. The Journal of neuroscience : the official journal of the Society for Neuroscience 13, 4700-4719.

Orban, G.A., Van Essen, D., Vanduffel, W. (2004). Comparative mapping of higher visual areas in monkeys and humans. Trends in Cognitive Sciences 8, 315-324.

Pesaran, B., Pezaris, J., Sahani, M., Mitra, P., and Andersen, R. (2002). Temporal structure in neuronal activity during working memory in macaque parietal cortex. Nature neuroscience 5, 805-811.

Pinto, N., Doukhan, D., DiCarlo, J.J., and Cox, D.D. (2009). A high-throughput screening approach to discovering good forms of biologically inspired visual representation. PLoS Comput Biol 5, e1000579.

Poggio, T., and Smale, S. (2003). The mathematics of learning: dealing with data. Notices of the AMS 50, 537-544.

Potter, M., and Levy, E. (1969). Recognition memory for a rapid sequence of pictures. Journal of experimental psychology 81, 10-15.

Prinz AA, B.D., Marder E (2004). Similar network activity from disparate circuit parameters. Nat Neurosci 7, 1345-1352.

Rao, R.P. (2004). Bayesian computation in recurrent neural circuits. Neural Comput 16, 1-38.

Rao, R.P. (2005). Bayesian inference and attentional modulation in the visual cortex. Neuroreport 16, 1843-1848.

Rao, R.P., and Ballard, D.H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. Nature neuroscience 2, 79-87.

Rao, R.P.N., Olshausen, B.A., and Lewicki, M.S., eds. (2002). Probabilistic Models of the Brain: Perception and Neural Function (Cambridge: MIT Press).

Richmond, B., Wurtz, R., and Sato, T. (1983). Visual responses in inferior temporal neurons in awake Rhesus monkey. Journal of Neurophysiology 50, 1415-1432.

Riesenhuber, M., and Poggio, T. (1999). Hierarchical models of object recognition in cortex. Nature neuroscience 2, 1019-1025.

Riesenhuber, M., and Poggio, T. (2002). Neural mechanisms of object recognition. Current opinion in neurobiology 12, 162-168.

Rolls, E. (1991). Neural organization of higher visual functions. Current opinion in neurobiology 1, 274-278.

Serre, T., Kouh, M., Cadieu, C., Knoblich, U., Kreiman, G., and Poggio, T. (2005a). A theory of object recognition: computations and circuits in the feedforward path of the ventral stream in primate visual cortex. (Boston, MIT), pp. CBCL Paper #259/AI Memo #2005-2036.

Serre, T., Kreiman, G., Kouh, M., Cadieu, C., Knoblich, U., and Poggio, T. (2007a). A quantitative theory of immediate visual recognition. Progress In Brain Research 165C, 33-56.

Serre, T., Oliva, A., and Poggio, T. (2007b). Feedforward theories of visual cortex account for human performance in rapid categorization. PNAS 104, 6424-6429.

Serre, T., Wolf, L., and Poggio, T. (2005b). Object Recognition with Features Inspired by Visual Cortex. In CVPR.

Sharon, E., Galun, M., Sharon, D., Basri, R., and Brandt, A. (2006). Hierarchy and adaptivity in segmenting visual scenes. Nature 442, 810-813.

Standing, L. (1973). Learning 10,000 pictures. Q J Exp Psychol 25, 207-222.

Sutherland, N.S. (1968). Outlines of a theory of visual pattern recognition in animals and man. Proc R Soc Lond B Biol Sci 171, 297-317.

Tanaka, K. (1996). Inferotemporal cortex and object vision. Annual Review of Neuroscience 19, 109-139.

Tank, D., and Hopfield, J. (1987). Collective computation in neuron-like circuits. Scientific American 257, 104-&.

Thorpe, S., Fize, D., and Marlot, C. (1996). Speed of processing in the human visual system. Nature 381, 520-522.

Tsotsos, J. (1990). Analyzing Vision at the Complexity Level. Behavioral and Brain Sciences 13-3, 423-445.

Ullman, S. (1996). High-Level Vision (Cambridge, MA: The MIT Press).

Virga, A., Rockland, KS (1989). Terminal Arbors of Individual "Feedback" Axons Projecting from Area V2 to V1 in the Macaque Monkey: A Study Using Immunohistochemistry of Anterogradely Transported Phaseolus vulgaris-leucoagglutinin. The Journal of Comparative Neurology 285, 54-72.

Vogels, R., Biederman, I., Bar, M., and Lorincz, A. (2001). Inferior temporal neurons show greater sensitivity to nonaccidental than to metric shape differences. J Cogn Neurosci 13, 444-453.

Wallis, G., and Rolls, E.T. (1997). Invariant face and object recognition in the visual system. Progress in Neurobiology 51, 167-194.

Wilson, M.A., and McNaughton, B.L. (1993). Dynamics of the Hippocampal Ensemble Code for Space. Science 261, 1055-1058.

Winston, P. (1975). Learning structural descriptions from examples. In The psychology of computer vision, P. Winston, ed. (McGraw-Hill), pp. 157-209.

Yamane, Y., Carlson, E.T., Bowman, K.C., Wang, Z., and Connor, C.E. (2008). A neural code for three-dimensional object shape in macaque inferotemporal cortex. Nature neuroscience 11, 1352-1360.

Yuille, A., and Kersten, D. (2006). Vision as Bayesian inference: analysis by synthesis? Trends Cogn Sci 10, 301-308.