



ANNUAL
REVIEWS **Further**

Click here to view this article's
online features:

- Download figures as PPT slides
- Navigate linked references
- Download citations
- Explore related articles
- Search keywords

Deep Neural Networks: A New Framework for Modeling Biological Vision and Brain Information Processing

Nikolaus Kriegeskorte

Medical Research Council Cognition and Brain Sciences Unit, University of Cambridge,
Cambridge CB2 7EF, United Kingdom; email: nikolaus.kriegeskorte@mrc-cbu.cam.ac.uk

Annu. Rev. Vis. Sci. 2015. 1:417–46

The *Annual Review of Vision Science* is online at
vision.annualreviews.org

This article's doi:
10.1146/annurev-vision-082114-035447

Copyright © 2015 by Annual Reviews.
All rights reserved

Keywords

biological vision, computer vision, object recognition, neural network,
deep learning, artificial intelligence, computational neuroscience

Abstract

Recent advances in neural network modeling have enabled major strides in computer vision and other artificial intelligence applications. Human-level visual recognition abilities are coming within reach of artificial systems. Artificial neural networks are inspired by the brain, and their computations could be implemented in biological neurons. Convolutional feedforward networks, which now dominate computer vision, take further inspiration from the architecture of the primate visual hierarchy. However, the current models are designed with engineering goals, not to model brain computations. Nevertheless, initial studies comparing internal representations between these models and primate brains find surprisingly similar representational spaces. With human-level performance no longer out of reach, we are entering an exciting new era, in which we will be able to build biologically faithful feedforward and recurrent computational models of how biological brains perform high-level feats of intelligence, including vision.

INTRODUCTION

Unit: model abstraction of a neuron, typically computing a weighted sum of incoming signals, followed by a static nonlinear transformation

Backpropagation: a supervised neural network learning algorithm that efficiently computes error derivatives with respect to the weights by passing through the connectivity in reverse in order to iteratively minimize the error

The brain is a deep and complex recurrent neural network. The models of information processing that have dominated computational neuroscience, by contrast, are largely shallow architectures that perform simple computations. Unsurprisingly, complex tasks such as visual object recognition have remained beyond the reach of computational neuroscience (see the sidebar How Previous Attempts to Understand Complex Brain Information Processing Fell Short). In this article, I argue that recent advances in neural network models (LeCun et al. 2015) will usher in a new era of computational neuroscience, in which we will engage real-world tasks that require rich knowledge and complex computations.

Neural networks are an old idea, so what is new now? Indeed, the history of neural networks is roughly coextensive with that of modern computing machines (see the sidebar What Is Meant by the Term Neural Network?). John von Neumann and Alan Turing, whose ideas shaped modern computing technology, both explored network models inspired by the brain. An early mathematical model of a single neuron was suggested by McCulloch & Pitts (1943). Their binary threshold unit took a number of inputs, computed a weighted sum, and imposed a threshold, implementing a linear discriminant. Responding to a pattern of continuous inputs with a single binary output, the threshold unit provided an intuitive bridge between the biological hardware of a spiking neuron and categorization, a hallmark of cognition.

Discriminating categories that are not linearly separable in the input requires an intervening layer of nonlinear transformations between the input and the output units. The field took a while to find ways of automatically training such multilayer networks with input–output pairs. The most influential solution to this problem is the backpropagation algorithm, a gradient-descent method that makes iterative small adjustments to the weights in order to reduce the errors of the outputs (Werbos 1981, Rumelhart et al. 1986).

HOW PREVIOUS ATTEMPTS TO UNDERSTAND COMPLEX BRAIN INFORMATION PROCESSING FELL SHORT

The cognitive and brain sciences have gone through a sequence of transformations, with different fields dominating each period. Each field combined a different set of elements required for understanding how the brain works (**Table 1**). Cognitive psychology attempted to illuminate behaviorism’s black box with theories of information processing. However, it lacked fully explicit computational models. Cognitive science made information processing theory fully explicit. However, it lacked constraints from neurophysiological data, making it difficult to adjudicate between multiple alternative models consistent with the behavioral data. Connectionism within cognitive science offered a neurobiologically plausible computational framework. However, neural network technology was not sufficiently advanced to take on real-world tasks such as object recognition from photographs. As a result, neural networks did not initially live up to their promise as AI systems, and in cognitive science, modeling was restricted to toy problems. Cognitive neuroscience brought neurophysiological data into investigations of complex brain information processing. Our hands full with the new challenges of analyzing complex brain imaging data, however, our theoretical sophistication slipped back to the stage of cognitive psychology, and we began (perhaps reasonably) by mapping box-and-arrow models onto brain regions. Computational neuroscience uses fully explicit and biologically plausible computational models to predict neurophysiological and behavioral data. At this level of rigor, however, we have not been able to engage complex real-world computational challenges and higher-level brain representations. Now deep neural networks provide a framework for engaging complex cognitive tasks and predicting both brain and behavioral responses.

Table 1 Historical progress toward understanding how the brain works

Elements required for understanding how the brain works		Behaviorism	Cognitive psychology	Cognitive science	Cognitive neuroscience	Classical computational neuroscience	Future cognitive computational neuroscience
Data	Behavioral	✓	✓	✓	✓	✓	✓
	Neurophysiological				✓	✓	✓
Theory	Cognitive		✓	✓	✓		✓
	Fully computationally explicit			✓		✓	✓
	Neurally plausible			✓		✓	✓
Explanation of real-world tasks requiring rich knowledge and complex computations			✓		✓		✓
Explanation of how high-level neuronal populations represent and compute							✓

Backpropagation led to a second wave of interest in neural networks in cognitive science and artificial intelligence (AI) in the 1980s. In cognitive science, neural network models of toy problems fostered the theoretical notion of parallel distributed processing (Rumelhart & McClelland 1988). However, backpropagation models did not work well on complex, real-world problems such as vision. Models not as obviously inspired by the brain that used hand-engineered representations and machine learning techniques, such as support vector machines, appeared to provide better engineering solutions for computer vision and AI. As a consequence, neural networks fell out of favor in the 1990s.

WHAT IS MEANT BY THE TERM NEURAL NETWORK?

The term neural network originally refers to a network of biological neurons. More broadly, the term evokes a particular paradigm for understanding brain function, in which neurons are the essential computational units, and computation is explained in terms of network interactions. Note that this paradigm leaves aside many biological complexities, including functional contributions of neurochemical diffusion processes, glial cells, and hemodynamics (Moore & Cao 2008). Although neurons are biological entities, the term neural network has come to be used as a shorthand for *artificial* neural network, a class of models of parallel information processing that is inspired by biological neural networks but commits to several further major simplifications.

Although spiking models have an important place in the computational literature, the models discussed here are nonspiking and do not capture dendritic computation, other processes within each neuron (e.g., Gallistel & King 2011), and distinct contributions from different types of neurons. The spatial structure of a neuron is typically abstracted from and its spiking output is modeled as a real number analogous to the spike rate. The rate is modeled as a weighted sum of incoming activations passed through a static nonlinearity. Despite, and perhaps also because of, these simplifications, the neural network paradigm provides one of the most important paths toward understanding brain information processing. It appears likely that this approach will take a central role in any comprehensive future brain theory. Opinions diverge as to whether more biologically detailed models will ultimately be needed. However, neural networks as used in engineering are certainly neurobiologically plausible, and their success in AI suggests that their abstractions may be desirable, enabling us to explain at least some complex feats of brain information processing.

Generative model:

a model of the process that generated the data (e.g., the image) to be inverted in data analysis (e.g., visual recognition)

Feedforward

network: a network with connections that form a directed acyclic graph, precluding recurrent information flow

Despite a period of disenchantment among the wider brain and computer science communities, neural network research has an unbroken history (Schmidhuber 2015) in theoretical neuroscience and in computer science. Throughout the 1990s and 2000s neural nets were studied by a smaller community of scientists who realized that the difficulties encountered were not fundamental limitations of the approach, but merely high hurdles to be overcome through a combination of better learning algorithms, better regularization, and larger training sets. With computations boosted by better computer hardware, the efforts of this community have been fruitful. In the past few years, neural networks have finally come into their own. They are currently conquering several domains of AI, including the hard problem of computer vision.

Computer vision competitions such as ImageNet (Deng et al. 2009) use secret test sets of images, providing rigorous evaluations of state-of-the-art technology. In 2012, a neural network model built by Krizhevsky et al. (2012) won the ImageNet classification competition by a large margin. The deep convolutional architecture of this model had enabled a leap in performance. Human performance levels, although still superior, suddenly did not seem entirely unattainable for computer vision any longer—at least in restricted domains such as visual object classification. The model built by Krizhevsky et al. (2012) marked the beginning of the dominance of neural networks in computer vision. In the past three years, error rates have dropped further, roughly matching human performance in the domain of visual object classification. Neural networks have also been very successful recently in other domains, such as speech recognition (Sak et al. 2014) and machine translation (Sutskever et al. 2014).

Artificial intelligence has entered an era in which systems directly inspired by the brain dominate practical applications. The time has come to bring this brain-inspired technology back to the brain. We are now in a position to integrate neural network theory with empirical systems neuroscience and to build models that engage the complexities of real-world tasks, use biologically plausible computational mechanisms, and predict neurophysiological and behavioral data.

The theoretical and engineering developments are progressing at an unprecedented pace. Many of the insights gained in engineering will likely be relevant for brain theory. Recent methods for comparing internal representations in neural population codes between models and brains enable us to test neural-net models as theories of brain information processing (Dumoulin & Wandell 2008; Kay et al. 2008; Kriegeskorte 2011; Kriegeskorte & Kievit 2013; Kriegeskorte et al. 2008a,b; Mitchell et al. 2008; Nili et al. 2014).

This article introduces a broad audience of vision and brain scientists to neural networks, including some of the recent advances of this modeling framework in engineering, and reviews the first few studies that have used such models to explain brain data. What emerges is a new framework for bringing computational neuroscience to high-level cortical representations and complex real-world tasks.

The following section, titled “A Primer on Neural Networks,” introduces the basics of neural network models, including their learning algorithms and universal representational capacity. The section “Feedforward Neural Networks for Visual Object Recognition” describes the specific large-scale object recognition networks that currently dominate computer vision and discusses what these networks do and do not share with biological vision systems. The section titled “Early Studies Testing Deep Neural Nets as Models of Biological Brain Representations” reviews the first few studies to empirically compare internal representations between artificial neural networks and biological brains. The section titled “Recurrent Neural Networks for Vision” describes networks using recurrent computation. Recurrence is an essential component of biological brains, might implement inference on generative models of the formation of the input image, and represents a major frontier for computational neuroscience. Finally, the section titled “Conclusions” considers

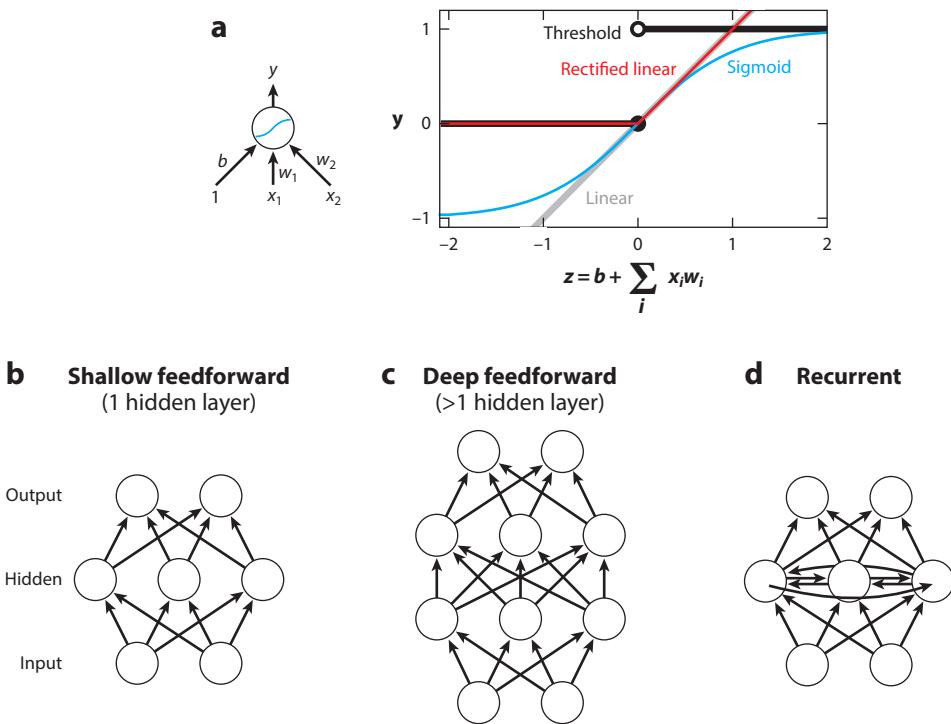


Figure 1

Artificial neural networks: basic units and architectures. (a) A typical model unit (*left*) computes a linear combination z of its inputs x_i using weights w_i and adding a bias b . The output y of the unit is a function of z , known as the activation function (*right*). Popular activation functions include linear (gray), threshold (black), sigmoid (hyperbolic tangent shown here, blue), and rectified linear (red) functions. A network is referred to as feedforward (*b,c*) when its directed connections do not form cycles and as recurrent (*d*) when they do form cycles. A shallow feedforward network (*b*) has zero or one hidden layers. Nonlinear activation functions in hidden units enable a shallow feedforward network to approximate any continuous function (with the precision depending on the number of hidden units). A deep feedforward network (*c*) has more than one hidden layer. Recurrent nets generate ongoing dynamics, lend themselves to the processing of temporal sequences of inputs, and can approximate any dynamical system (given a sufficient number of units).

critical arguments, upcoming challenges, and the way ahead toward empirically justified models of complex biological brain information processing.

A PRIMER ON NEURAL NETWORKS

A Unit Computes a Weighted Sum of Its Inputs and Activates According to a Nonlinear Function

We refer to model neurons as units to maintain a distinction between biological reality and highly abstracted models. The perhaps simplest model unit is a linear unit, which outputs a linear combination of its inputs (**Figure 1a**). Such units, combined to form networks, can never transcend linear combinations of the inputs. This insight is illustrated in **Figure 2b**, which shows how an output unit that linearly combines intermediate-layer linear-unit activations just adds up

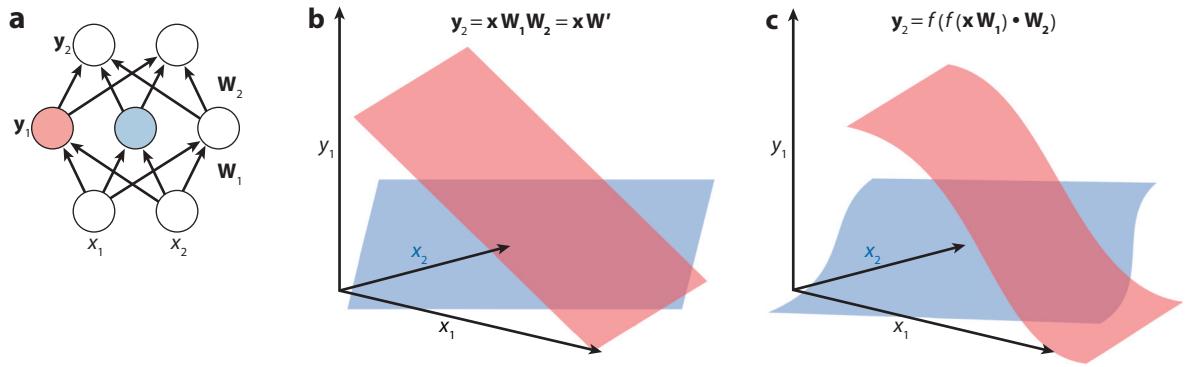


Figure 2

Networks with nonlinear hidden units can approximate arbitrary nonlinear functions. (a) A feedforward neural network with a single hidden layer. (b) Activation of the pink and blue hidden units as a function of the input pattern (x_1, x_2) when the hidden units have linear activation functions. Each output unit (y_2) will compute a weighted combination of the ramp-shaped (i.e., linear) activations of the hidden units. Thus, the output remains a linear combination of the input pattern. A linear hidden layer is not useful because the resulting network is equivalent to a linear network without a hidden layer intervening between input and output. (c) Activation of the pink and blue hidden units when these have sigmoid activation functions. Arbitrary continuous functions can be approximated in the output units (y_2) by weighted combinations of a sufficient number of nonlinear hidden-unit outputs (y_1).

ramp functions, and thus itself computes a ramp function. A multilayer network of linear units is equivalent to a single-layer network whose weights matrix \mathbf{W}' is the product of the weights matrices \mathbf{W}_i of the multilayer network. Nonlinear units are essential because their outputs provide building blocks (Figure 2c) whose linear combination one level up enables us to approximate any desired mapping from inputs to outputs, as described in the next section.

A unit in a neural network uses its input weights \mathbf{w} to compute a weighted sum z of its input activities \mathbf{x} and passes the result through a (typically monotonic) nonlinear function f to generate its activation y (Figure 1a). In early models, the nonlinearity was simply a step function (McCulloch & Pitts 1943, Rosenblatt 1958, Minsky & Papert 1972), making each unit a linear discriminant imposing a binary threshold. For a single threshold unit, the perceptron learning algorithm provides a method for iteratively adjusting the weights (starting with zeros or random weights) so as to get as many training input–output pairs as possible right. However, hard thresholding entails that, for a given pair of an input pattern and a desired output pattern, small changes to the weights will often make no difference to the output. This makes it difficult to learn the weights for a multilayer network by gradient descent, where small adjustments to the weights are made to iteratively reduce the errors. If the hard threshold is replaced by a soft threshold that continuously varies, such as a sigmoid function, gradient descent can be used for learning.

Universal function approximator:

model family that can approximate any function that maps input patterns to output patterns (with arbitrary precision when allowed enough parameters)

Networks with Nonlinear Hidden Units Are Universal Function Approximators

The particular shape of the nonlinear activation function does not matter to the class of input–output mappings that can be represented. Feedforward networks with at least one layer of hidden units intervening between input and output layers are universal function approximators: Given a sufficient number of hidden units, a network can approximate any function of the inputs in the output units. Continuous functions can be approximated with arbitrary precision by adding a sufficient number of hidden units and suitably setting the weights (Schäfer & Zimmermann 2007, Hornik 1991, Cybenko 1989). Figure 2c illustrates this process for two-dimensional inputs:

Adding up a sufficient number of sigmoid ramps, which can have any orientation, slope, and position, we can approximate any continuous function of the input.

To gain an intuition on why combining sigmoids (or any step-like functions) of the input enables a network to approximate any function, imagine we set the weights of a set of units so that they compute sigmoids whose plateaus (close to 1) overlap only in a particular region of the input space. If we now sum the outputs of these units in a unit with a high threshold, that unit can indicate (by an output close to 1) that we are in a certain region of the input space. If we build indicators in this fashion for all regions within the input space that require a different output, we can map any input to any required output approximately. The precision of this approximate mapping can always be improved by using more units to define more separate regions with indicators. Note that if the activation function is continuous (as it usually is), then the function represented by the network is also continuous. The network would use two hidden layers to represent what is essentially a lookup table of the training input–output pairs. (However, the network would have the nice feature of handling novel inputs by interpolation or extrapolation.) The cited theoretical results on the universality of feedforward nets go beyond this intuitive explanation and show that only a single hidden layer is needed to approximate any function and that the activation function need not resemble a step function.

A simple and powerful neural network architecture is the feedforward network (**Figure 1b,c**). A feedforward network is composed of a sequence of layers of units, with each unit sending its output only to units in higher layers. Thus, the units and connections of a feedforward network correspond to the nodes and edges, respectively, of a directed acyclic graph. In computer vision systems, units often receive inputs only from the immediately preceding layer. In addition, inputs in lower layers are usually restricted to local receptive fields, inspired by the visual hierarchy.

Modern models use a variety of nonlinear activation functions, including sigmoid (e.g., logistic or hyperbolic tangent) and rectified linear functions (**Figure 1a**). A rectified linear unit outputs the linear combination it computes, if it is positive, and 0 otherwise. Rectified linear units simplify the gradient-descent learning of the weights, enabling more rapid training, and have been demonstrated to work very well in computer vision and other domains.

Why Deep?

A feedforward network is said to be *deep* when it has more than one hidden layer. This technical definition notwithstanding, the term deep is also used in a graded sense. A deep net, thus, is a network with many layers, and one network can be deeper than another. *Deep learning* refers to the strategy of using architectures with many hidden layers to tackle difficult problems, including vision.

Why does depth help? We saw above that even shallow networks with a single layer of nonlinear hidden units are universal function approximators. Shallow networks are closely related to support vector machines, which can likewise learn arbitrary nonlinear functions, can be more efficiently trained than neural networks, and have been very successful tools of machine learning. The reason depth matters is that deep nets can represent many complex functions more concisely (i.e., with fewer units and weights) than shallow nets and support vector machines (Bengio 2009).

Consider a shallow network (i.e., a network with a single hidden layer) that computes some function. We can create a deeper network with the same number of units by distributing the units of the single hidden layer across multiple hidden layers in the new network. The deep network could have the same connectivity from the input to the hidden units and from the hidden units to the output. It can thus compute any function the shallow network can compute. The reverse is not true, however: The deep network is permitted additional nonzero weights from any given layer to higher layers, enabling *reuse* of the results of previous computations and extending the expressive

Deep learning: machine learning of complex representations in a deep neural network, typically using stochastic gradient descent by error backpropagation

Deep neural network: network with more than one hidden layer between the input and output layers; more loosely, a network with many hidden layers

power of the deep network. For many particular functions that a deep network might compute, one can show that a shallow network would need a much larger number of units (Bengio 2009).

Recurrent network:
a network with recurrent information flow, which produces dynamics and lends itself naturally to the perception and generation of spatiotemporal patterns

Universal approximator of dynamical systems:
a model family generating dynamics that can approximate any dynamical system (with arbitrary precision when allowed enough parameters)

Supervised learning:
a learning process requiring input patterns and additional information about the desired representation or the outputs (e.g., category labels)

It is instructive to consider the analogy to modern computing hardware. The von Neumann architecture is a fundamentally sequential model of computation that enables the reuse of results of previous computations. In special cases, in which many computations are to be performed independently (e.g., across image positions in graphics and vision), parallel hardware can speed up the process. Whereas independent computations can be performed either in parallel or sequentially, however, dependent computations can only be performed sequentially. The option to reuse previous results therefore extends the set of computable functions (if the total number of units is fixed).

In essence, a shallow network is a universal function approximator because it can piece together the target function like a lookup table. Many functions can be more concisely represented using a deeper network, however, taking advantage of redundancies and exploiting the inherent structure of the target function. Although every problem is different and the field is still learning when exactly depth helps, the practical success of deep learning in AI suggests that many real-world problems, including vision, may be more efficiently solved with deep architectures. Interestingly, the visual hierarchy of primate brains is also a deep architecture.

Recurrent Neural Networks Are Universal Approximators of Dynamical Systems

Feedforward networks compute static functions. An architecture with more interesting dynamics is a recurrent network, whose units can be connected in cycles. Such an architecture is more similar to biological neuronal networks, in which lateral and feedback connections are ubiquitous. The notion of separate hidden layers is meaningless in a recurrent network because every hidden unit can interact with every other hidden unit. Recurrent nets are therefore often depicted as a single interconnected set of hidden units, with separate sets of input and output units (**Figure 1d**). A layered architecture is a special case of a recurrent network in which certain connections are missing (i.e., their weights are fixed at 0).

In visual neuroscience, the theoretical concept of the visual hierarchy is based on a division of the connections into feedforward, lateral, and feedback connections, as identified by a connection's cortical layers of origin and termination, as well as on the fact that some neurons are separated from the input by many synapses and tend to represent more complex visual features. Although these criteria may not support a perfectly unambiguous assignment of ranks that would define a hierarchy for the primate visual system (Hilgetag et al. 2000), the hierarchical model continues to be a useful simplification.

Whereas a feedforward network computes a static function that maps inputs to outputs, a recurrent network produces dynamics: a temporal evolution of states that can be influenced by a temporal evolution of input patterns. The internal state of a recurrent network lends it a memory, enabling it to represent the recent stimulus history and detect temporal patterns. Whereas feedforward nets are universal function approximators, recurrent nets are universal approximators of dynamical systems (Schäfer & Zimmermann 2007). A variety of particular models have been explored by simulation and analytically.

In an echo-state network (Jaeger 2001, see also Maass et al. 2002, for a similar model with spiking dynamics), for example, the sequence of input patterns is fed into a set of hidden units that are sparsely and randomly connected. The wave of activity associated with each input pattern will reverberate among the hidden units for a while until it comes to be dominated by the effects of subsequent input patterns. Like concentric waves on the surface of a pond that enable us to infer an event at their center sometime in the past, the activity of the hidden units encodes information about the recent stimulus history. In echo-state networks, the weights among the hidden units are not trained (although their random setting requires some care to ensure that the memories do

not fade too quickly). Supervised learning is used to train a set of readout units to detect temporal patterns in the input.

Echo-state networks rely on random weights among the hidden units for their recurrent dynamics. Alternatively, the dynamics of a recurrent network can be explicitly learned through supervision, so as to optimize it to produce, classify, or predict temporal patterns (Graves & Schmidhuber 2009, Sutskever et al. 2014).

Representations Can Be Learned by Gradient Descent Using the Backpropagation Algorithm

The universality theorems assure us of the representational power of neural networks with sufficient numbers of units. However, these theorems do not tell us how to set the weights of the connections, so as to represent a particular function with a feedforward net or a particular dynamical system with a recurrent net. Learning poses a high-dimensional and difficult optimization problem. Models that can solve real-world problems will have large numbers of units and even larger numbers of weights. Global optimization techniques are not available for this nonconvex problem. The space of weight settings is so vast that simple search algorithms (for example, evolutionary algorithms) can cover only a vanishingly small subset of the possibilities and typically do not yield working solutions, except for small models restricted to toy problems.

The high dimensionality of the weight space makes global optimization intractable. However, the space contains many equivalent solutions (consider, for example, exchanging all incoming and outgoing weights between two neurons). Moreover, the total error (i.e., the sum of squared deviations between actual and desired outputs) is a locally smooth function of the weights. The current training method of choice is gradient descent, the iterative reduction of the errors through small adjustments to the weights.

The basic idea of gradient-descent learning is to start with a random initialization of the weights and to determine how much a slight change to each weight will reduce the error. The weight is then adjusted in proportion to the effect on the error. This method ensures that we move in the direction in weight space, along which the error descends most steeply.

The gradient, that is, how much the error changes with an adjustment of a weight, is the derivative of the error with respect to the weight. These derivatives can be computed easily for the weights connecting to the output layer of a feedforward network. For connections driving the preceding layers, an efficient way to compute the error derivatives is to propagate them backward through the network. This gives the method its name, backpropagation (Werbos 1981, Rumelhart et al. 1986).

Gradient descent sees only the local neighborhood in weight space and is not guaranteed to find globally optimal solutions. It can nevertheless find solutions that work very well in practice. The high dimensionality of the weight space is a curse in that it makes global optimization difficult. However, it is a blessing in that it helps local optimization find good solutions: With so many directions to move in, gradient descent is unlikely to get stuck in local minima, where the error increases in all directions and no further progress is possible.

Intriguingly, the same approach can be used to train recurrent networks. The error derivatives are then computed by backpropagation through time, with the process suffusing the loops in reverse through multiple cycles. To understand why this works, we can construe any recurrent network as the feedforward network obtained by replicating all units of the recurrent network along the dimension of time (for a sufficiently large number of time steps). Each time point of the recurrent computation corresponds to a layer of the feedforward net, each of which is connected to the next by the same weights matrix, the weights matrix of the recurrent network. By backpropagation through time, a recurrent network can learn weights that enable it to store short-term memories

Unsupervised learning: a learning process that requires only a set of input patterns and captures aspects of their probability distribution

in its dynamics, relating temporally separated events as needed to achieve desired classifications or predictions of temporal sequences. However, propagating error derivatives far enough backward through time for the network to learn how to exploit long-lag dependencies is hampered by the problem that the gradients tend to vanish or explode (Hochreiter 1991, Hochreiter et al. 2001). The problem occurs because a given weight's error derivative is the product of multiple terms, corresponding to weights and derivatives of the activation functions encountered along the path of backpropagation. One solution to this problem is offered by the long short-term memory (LSTM) architecture (Hochreiter & Schmidhuber 1997), in which special gated units can store short-term memories for extended periods. The error derivatives backpropagated through these units remain stable, enabling backpropagation to learn long-lag dependencies. Such networks, amazingly, can learn to remember information that will be relevant many time steps later in a sequential prediction, classification, or control task. Backpropagation adjusts the weights to ingrain a dynamics that selectively stores (in the activation state) the information needed later to perform the task.

Vanishing and exploding gradients can also pose a problem in training deep feedforward nets with backpropagation. The choice of nonlinear activation function can make a difference with regard to this problem. In addition, the details of the gradient-descent algorithm, regularization, and weight initialization all matter to making supervised learning by backpropagation work well.

Representations Can Also Be Learned with Unsupervised Techniques

In supervised learning, the training data comprise both input patterns and the associated desired outputs. An explicit supervision signal of this type is often unavailable in the real world. Biological organisms do not in general have access to supervision. In engineering, similarly, we often have a large number of unlabeled input patterns and only a smaller number of labeled input patterns (e.g., images from the web). Unsupervised learning does not require labels for a network to learn a representation that is optimized for natural input patterns and potentially useful for a variety of tasks. Natural images, for example, form a very small subset of all possible images, enabling unsupervised learning to find compressed representations.

An instructive example of unsupervised learning is provided by autoencoders (Hinton & Salakhutdinov 2006). An autoencoder is a feedforward neural network with a central code layer that has fewer units than the input. The network is trained with backpropagation to reconstruct its input in the output layer (which has the same number of units as the input layer). Although the learning algorithm is backpropagation and uses a supervision signal, the technique is unsupervised because it requires no separate supervision information (i.e., no labels), only the set of input patterns. If all layers, including the code layer, had the same dimensionality as the input, the network could just pass the input through its layers. Because the code layer has fewer units, however, it forms an informational bottleneck. To reconstruct the input, the network must learn to retain sufficient information about the input in its small code layer. An autoencoder therefore learns a compressed representation in its code layer, exploiting the statistical structure of the input domain. This representation will be specialized for the distribution of the input patterns used in training.

The layers from the input to the code layer are called the *encoder*, and the layers from the code layer to the output are called the *decoder*. If the encoder and decoder are linear, the network learns the linear subspace spanned by the first k principal components (for a code layer of k units). With nonlinear neural networks as encoders and decoders, nonlinearly compressed representations can be learned. Nonlinear codes can be substantially more efficient when the natural distribution of the input patterns is not well represented by a linear subspace. Natural images are a case in point.

Unsupervised learning can help pretrain a feedforward network when insufficient labeled training data are available for purely supervised learning. For example, a network for visual recogni-

can be pretrained layer by layer in the autoencoder framework using a large set of unlabeled images. Once the network has learned a reasonable representation of natural images, it can more easily be trained with backpropagation to predict the correct image labels.

FEEDFORWARD NEURAL NETWORKS FOR VISUAL OBJECT RECOGNITION

Computer vision has recently come to be dominated by a particular type of deep neural network: the deep feedforward convolutional network. These networks now robustly outperform the previous state of the art, which consisted in hand-engineered visual features (e.g., Lowe 1999) forming the input to shallow machine learning classifiers such as support vector machines. Interestingly, some of the earlier systems inserted an intermediate representation, often acquired by unsupervised learning, between the hand-engineered features and the supervised classifier. The insertion of this representation might have helped address the need for a deeper architecture.

The deep convolutional nets widely used computer vision today share several architectural features, some of which are loosely inspired by biological vision systems (Hubel & Wiesel 1968).

- **Deep hierarchy:** Like the primate ventral visual stream, these networks process information through a deep hierarchy of representations (typically 5 to 20 layers; see **Figure 3** for an example), gradually transforming a visual representation, whose spatial layout matches the image, to a semantic representation that enables the recognition of object categories.

Convolutional network:

network in which the preactivation of a layer (before the nonlinearity) implements convolutions of the previous layer with a number of weight-template patterns

Receptive field modeling:

predictive modeling of the response to arbitrary sensory inputs of neurons (or measured channels of brain activity)

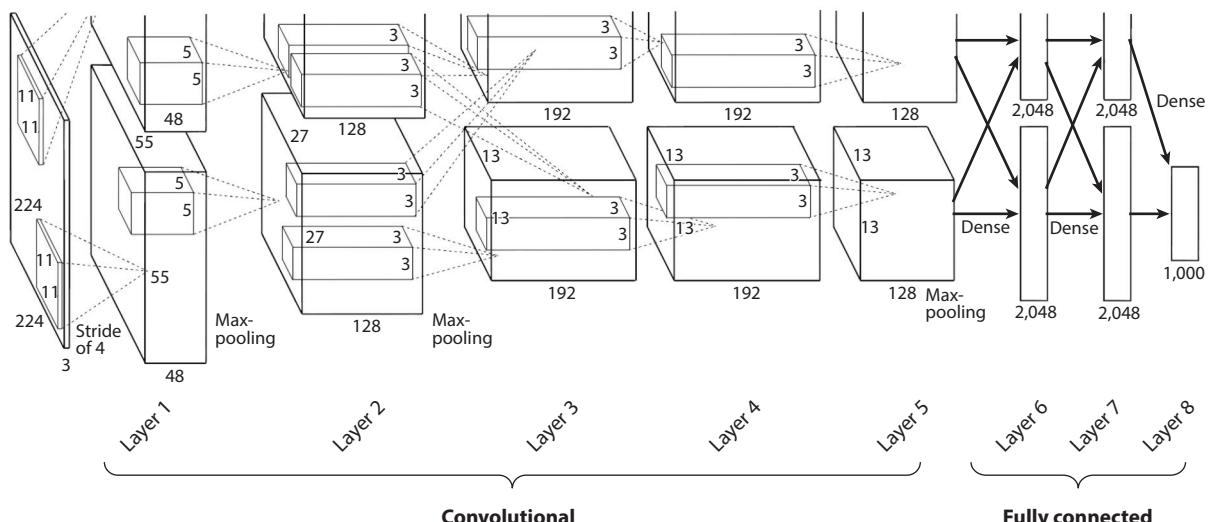


Figure 3

Deep convolutional feedforward architecture for object recognition. The figure shows the architecture used by Krizhevsky et al. (2012). The information flows from the input pixel image (*left*) (224 × 224 pixels, 3 color channels) through 7 hidden layers to the category output (*right*) (1,000 category detector units). The large boxes represent stacks of feature maps. For layer 2, for example, the lower large box represents 128 feature maps of size 27 (horizontal image positions) × 27 (vertical image positions). Note that the dimensions of the boxes are not drawn to scale. The small boxes represent the feature templates that are convolved with the representation in a given layer. Because convolution and max-pooling operate at strides greater than 1 pixel, the spatial extent of the feature maps decreases along the sequence of representations (224, 55, 27, 13, 13, 1, 1). The upper and lower large boxes represent the division of labor between two graphics processing units.

Max-pooling: summary operation implementing invariances by retaining only the maxima of sets of detectors differing in irrelevant properties (e.g., local position)

- **Convolution:** The lower layers contain local visual feature detectors with small receptive fields (RFs). Each detector is replicated all over the two-dimensional image, forming a feature map. This amounts to a convolution of the image with each feature pattern, followed by a static nonlinearity. The convolutional architecture is motivated by the insight that a feature useful in one position is likely to also be useful in another position. This architecture resembles early primate visual areas, which also detect qualitatively similar local visual features in many visual field positions (although the feature characteristics and the spatial distributions of the RFs are not completely uniform as in convolutional networks). The RFs of the units increase in size along the hierarchy. The restriction of the connections to a local region and the replication of the connection weights across spatial positions (same weight pattern at all locations for a given feature) greatly reduce the number of parameters that need to be learned (LeCun et al. 1989).
- **Local pooling and subsampling:** In between the convolutional stages, local pooling stages are inserted. Pooling combines the outputs of a local set of units by taking the maximum or the average. This confers a local tolerance to spatial shifts of the features, making the representation robust to small image shifts and small distortions of the configuration of image features (Fukushima 1980). Max-pooling is also used in neuroscientific vision models such as HMAX (Riesenhuber & Poggio 1999, Serre et al. 2007) to implement local tolerances. Pooling is often combined with subsampling of the spatial locations. The reduction in the number of distinct spatial locations represented frees up resources for an increase along the hierarchy in the number of distinct features computed at each location.

In the highest layers, units have global RFs, receiving inputs from all units of the previous layer. The final layer typically contains one unit per category and implements a softmax (normalized exponential) function, which strongly reduces all but the very highest responses and ensures that the outputs add up to 1. The output can be interpreted as a probability distribution over the categories when the training procedure is set up to minimize the crossentropy error.

The networks can be trained to recognize the category of the input image using backpropagation (LeCun et al. 1989, LeCun & Bengio 1995). When a network is trained to categorize natural images, the learning process discovers features that are qualitatively similar to those found in biological visual systems (**Figure 4**). The early layers develop Gabor-like features, similar to those that characterize V1 neurons. Similar features are discovered by unsupervised techniques such as sparse representational learning (Olshausen & Field 1997), suggesting that they provide a good starting point for vision, whether the goal is sparse representation or categorization. Subsequent stages contain units that are selective for slightly more complex features, including curve segments. Higher layers contain units that are selective for parts of objects and for entire objects, such as faces and bodies of humans and animals, and inanimate objects such as cars and buildings.

To understand what has been learned automatically, the field is beginning to devise methods for visualizing the RFs and selectivities of units within deep networks (Zeiler & Fergus 2014, Girshick et al. 2014, Simonyan et al. 2014, Tsai & Cox 2015, Zhou et al. 2015). **Figure 4** shows such visualizations, which support the idea that units learn selectivities to natural image features that increase in visual complexity along the hierarchy. However, two important caveats accompany such visualizations. First, because of the multiple nonlinear transforms across layers, a unit cannot be accurately characterized by an image template. If the high-level responses could be computed by template matching, a deep hierarchy would not be needed for vision. The visualizations merely show what drives the response in the context of a particular image. To get an idea of the selectivity of a unit, many images that drive it need to be considered (for multiple templates for each of a larger number of units, see Zeiler & Fergus 2014). Second, the units visualized in **Figure 4** have been selected because they confirm a theoretical bias for interpretable selectivities. Units similar

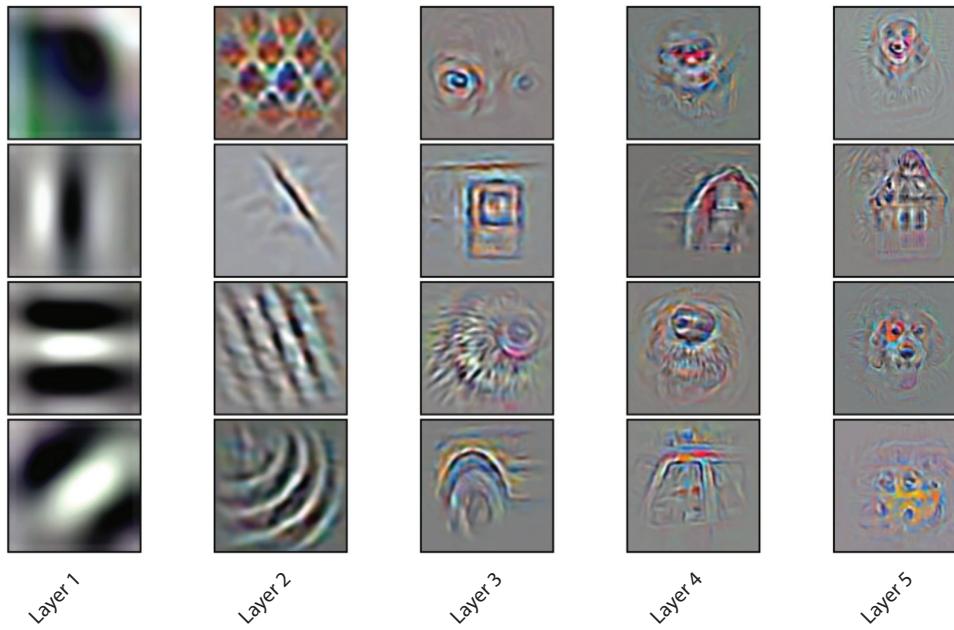


Figure 4

Deep supervised learning produces feature selectivities that are qualitatively consistent with neurophysiological findings. To understand representations in deep neural networks, we can visualize which image elements drive a given unit in a deep network. For 20 example units (4 from each of 5 layers), the images shown visualize what caused the response in the context of a particular image that strongly drove the unit. The visualization technique used here involves two steps: selection of an input image that strongly drives the unit, and inversion of the feedforward computation to generate the image element responsible. Convolutions along the feedforward pass are inverted by deconvolution (using the transposes of the convolution matrices). Max-pooling operations are inverted by storing the identity of the connection to the pooling unit that was maximally active in the feedforward pass. Note that a unit deep in a network does not perform a simple template-matching operation on the image and therefore cannot be fully characterized by any visual template. However, performing the above visualization for many images that drive a unit (not shown) can help us understand its selectivity and tolerances. The deconvolutional visualization technique shown was developed by Zeiler & Fergus (2014). The deep network is from Chatfield et al. (2014). The analysis was performed by Güçlü & van Gerven (2015). Figure adapted with permission from Güçlü & van Gerven (2015).

to those shown may be the exception rather than the rule, and it is unclear whether they are essential to the functionality of the network. For example, meaningful selectivities could reside in linear combinations of units rather than in single units, with weak distributed activities encoding essential information.

The representational hierarchy appears to gradually transform a space-based visual to a shape-based and semantic representation. The network acquires complex knowledge about the kinds of shapes associated with each category. In this context, shape refers to luminance- and color-defined features of various levels of complexity. High-level units appear to learn representations of shapes occurring in natural images, such as faces, human bodies, animals, natural scenes, buildings, and cars. The selectivities learned are not restricted to the categories detected by the output layer, but may include selectivities to parts of these objects or even to context elements. For example, the network by Krizhevsky et al. (2012) contains units that appear to be selective for text (Yosinski et al.

2015) and faces, although text and faces were not among the trained categories. Presumably, those responses help detect the categories represented in the output layer, because they are statistically related to the categories to be detected. For example, part-selective features may serve as stepping stones toward detection of entire objects (Jozwik et al. 2015). A verbal functional interpretation of a unit, e.g., as an eye or a face detector, may help our intuitive understanding and capture something important. However, such verbal interpretations may overstate the degree of categoricity and localization, and underestimate the statistical and distributed nature of these representations.

An influential example of a deep convolutional neural network for computer vision is the system built by Krizhevsky et al. (2012). The architecture (**Figure 3**) comprises five convolutional layers and three fully connected layers. The authors found that reducing the number of convolutional layers hurt performance, illustrating the need for a deep architecture. The system uses rectified linear units, max-pooling, and local normalization. The network was trained by backpropagation to recognize which of 1,000 object categories was shown in the input image. The training set comprised 1.2 million category-labeled images from the ImageNet set. This set was expanded by a factor of 2,048 by adding translated and horizontally reflected versions of the images. The training cycled through the resulting image set 90 times.

The training relied on dropout regularization (Hinton et al. 2012), a technique in which each unit is “dropped” (omitted from the computations) with a probability of 0.5 on each training trial. Thus, on a given trial, a random set of approximately half of the units is used in both the forward pass computing the output and the backpropagation pass adjusting the weights. This method prevents complex coadaptations of the units during learning, forcing each unit to make a useful contribution in the context of many different teams of other units. The network has a total of 650,000 units and 60 million parameters. The convolutional layers are defined by their small local weight templates, which constitute less than 5% of the parameters in total. Over 95% of the parameters define the upper three fully connected layers. Dropout was applied to the first two fully connected layers, each of which has many millions of incoming connections. Experiments showed that dropout was necessary to prevent overfitting.

The training was performed over the course of six days on a single workstation with two graphics processing units (GPUs), which parallelize and greatly accelerate the computations. The system was tested on a held-out set of images in the ImageNet Large-Scale Visual Recognition Challenge 2012, a computer-vision competition. It won the competition, beating the second-best system by a large margin and marking the beginning of the dominance of neural networks in computer vision. Since then, several convolutional neural networks using similar architectures have further improved performance (e.g., Zeiler & Fergus 2014, Chatfield et al. 2014).

The deep convolutional neural networks used in computer vision perform limited aspects of vision, such as category-level recognition. However, the range of visual tasks tackled is quickly expanding, and deep networks do represent a quantum leap compared with the earlier computer vision systems. Deep convolutional networks are not designed to closely parallel biological vision systems. However, their essential functional mechanisms are inspired by biological brains and could plausibly be implemented with biological neurons. This new technology provides an exciting framework for more biologically faithful brain-computational models that perform complex feats of intelligence beyond the current reach of computational neuroscience.

EARLY STUDIES TESTING DEEP NEURAL NETS AS MODELS OF BIOLOGICAL BRAIN REPRESENTATIONS

Several studies have begun to assess deep convolutional neural networks as models for biological vision, comparing both the internal representational spaces and performance levels between

Normalization:

an operation (e.g., division) applied to a set of activations so as to hold fixed a summary statistic (e.g., the sum)

Dropout: a regularization method for neural network training in which each unit is omitted from the architecture with probability 0.5 on each training trial

Graphics processing unit (GPU):

specialized computer hardware developed for graphics computations that greatly accelerates matrix–matrix multiplications and is essential for efficient deep learning

models and brains. One finding that is replicated and generalized across several studies (Yamins et al. 2013, 2014; Khaligh-Razavi & Kriegeskorte 2013, 2014) is that models that utilize representational spaces that are more similar to those of the inferior temporal (IT) cortex (Tanaka 1996) in human and nonhuman primates tend to perform better at object recognition. This observation affirms the intuition that computer vision can learn from biological vision. Conversely, biological vision science can look to engineering for candidate computational theories.

It is not true in general that engineering solutions closely follow biological solutions (consider planes, trains, and automobiles). In computer vision in particular, early failures to scale neural network models to real-world vision fostered a sense that seeking more brain-like solutions was fruitless. However, the recent successes of neural network models suggest that brain-inspired architectures for vision are extremely powerful. The empirical comparisons between representations in computer vision systems and brains discussed in this section suggest that the neural network models do not merely have architectural similarities. They also learn representations very similar to those of the primate ventral visual pathway.

It is impossible to prove that representations have to be similar to biological brains to support successful computer vision. However, an association between performance at object recognition and representational similarity to IT has been shown for a large set of automatically generated neural network architectures using random features (Yamins et al. 2013, 2014), for a wide range of popular hand-engineered computer vision features and neuroscientific vision models (Khaligh-Razavi & Kriegeskorte 2013, 2014), and for the layers of a deep neural network (Khaligh-Razavi & Kriegeskorte 2014). Within the architectures explored so far, at least, it appears that performance optimization leads to representational spaces similar to IT.

IT is known to emphasize categorical divisions in its representation (Kriegeskorte et al. 2008b). Models that perform well at categorization (which is implemented by linear readout) similarly tend to have stronger categorical divisions. This partially explains their greater representational similarity to IT. However, even the within-category representational geometries tend to be more similar to IT in the better-performing models (Khaligh-Razavi & Kriegeskorte 2014).

The best performing models are deep neural networks, and these networks are also best at explaining the IT representational geometry (Khaligh-Razavi & Kriegeskorte 2014, Cadieu et al. 2014). Khaligh-Razavi & Kriegeskorte (2014) tested a wide range of classical computer vision features; several neuroscientifically motivated vision models, including VisNet (Wallis & Rolls 1997, Tromans et al. 2011) and HMAX (Riesenhuber & Poggio 1999); and the deep neural network built by Krizhevsky et al. (2012) (**Figure 3**). The brain representations in their study were estimated from human functional magnetic resonance imaging (fMRI) and monkey cell recordings (monkey data from Kiani et al. 2007, Kriegeskorte et al. 2008b). Khaligh-Razavi & Kriegeskorte (2014) compared the internal representational spaces between models and brain regions using representational similarity analysis (Kriegeskorte et al. 2008a). For each pair of stimuli, this analysis measures the dissimilarity of the two stimuli in the representation. The vector of representational dissimilarities across all stimulus pairs is then compared between a model representation and a brain region.

Early layers of the deep neural network had representations resembling early visual cortex. Across the layers of the network, the representational geometry became monotonically less similar to early visual cortex and more similar to IT. These results are shown in **Figure 5** for human data. Similar results were obtained for monkey IT (not shown here).

At the highest layer, the representation did not yet fully explain the explainable variance in the IT data. However, a representation fitted to IT (by linear remixing and reweighting of the features of the deep neural network using independent image sets for training and validation) fully explained the IT data (Khaligh-Razavi & Kriegeskorte 2014). This IT-fitted deep neural

Representational similarity analysis:
method for testing computational models of brain information processing through statistical comparisons of representational distance matrices that characterize population-code representations

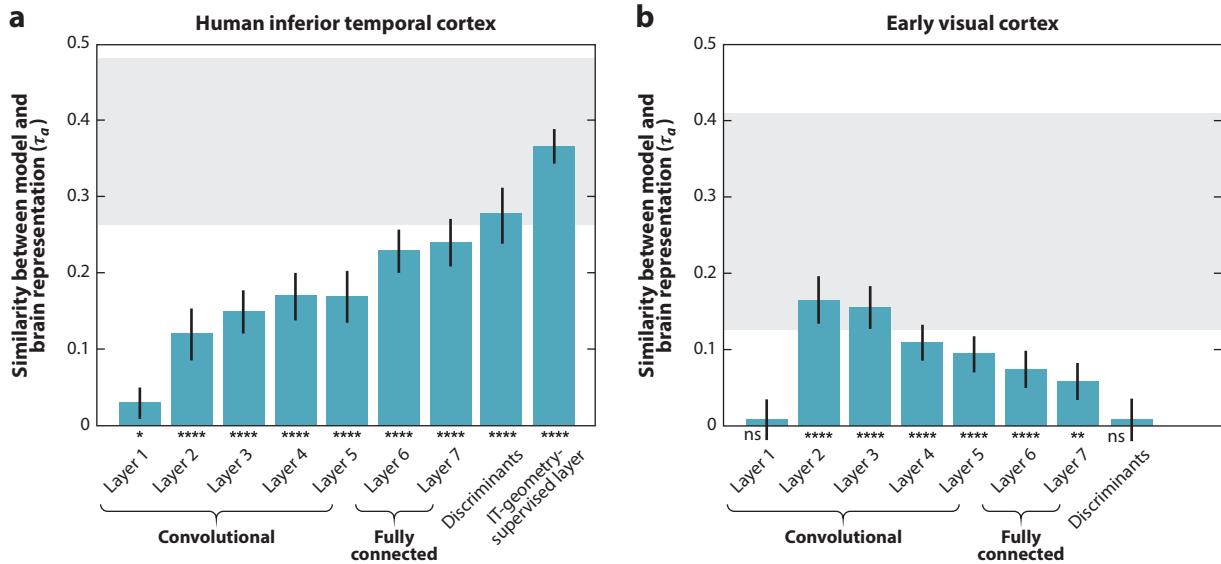


Figure 5

Deep neural network explains early visual and inferior temporal representations of object images. Each representation in model and brain was characterized by the dissimilarity matrix of the response patterns elicited by a set of real-world photos of objects.

(a) Representations become monotonically more similar to those of human inferior temporal (IT) cortex as we ascend the layers of the Krizhevsky et al. (2012) neural network. When the final representational stages are linearly remixed to emphasize the same semantic dimensions as IT using linear category discriminants (*second bar from the right*), and when each layer and each discriminant are assigned a weight to model the prevalence of different computational features in IT (cross-validated to avoid overfitting to the image set; *rightmost bar*), the noise ceiling (*gray shaded region*) is reached, indicating that the model fully explains the data. When the same method of linear combination with category discriminants and weighting was applied to traditional computer vision features (not shown here), the representation did not explain the IT data. Similar results were obtained for monkey IT (not shown here). (b) Lower layers of the deep neural network resemble the representations in the foveal confluence of early visual areas (V1–V3). Asterisks indicate accuracy above chance as follows: ns, not significant; *, $p < 0.05$; **, $p < 0.01$; ***, $p < 0.001$; ****, $p < 0.0001$. The similarity between each model representation and IT (vertical axes) was measured using Kendall's rank correlation coefficient τ_a to compare representational dissimilarity matrices (subject-group-average τ_a plotted). Results reproduced from Khaligh-Razavi & Kriegeskorte (2014).

network representation explained the IT representation substantially and significantly better than a similarly IT-fitted combination of the conventional computer vision features.

Cadieu et al. (2013, 2014) analyzed the internal representations of a population of IT cells alongside models of early vision, the HMAX model (Riesenhuber & Poggio 1999, Serre et al. 2007), a hierarchically optimized multilayer model from Yamins et al. (2013, 2014), and the deep neural networks from Krizhevsky et al. (2012) and Zeiler & Fergus (2014). The representations performing best at object categorization (Figure 6a) were the deep neural network built by Zeiler & Fergus (2014) and the biological IT representation (monkey neuronal recordings), followed closely by the deep network proposed by Krizhevsky et al. (2012). The other representations performed at much lower levels. The two deep networks explained the IT data equally well, as did neuronal recordings from an independent set of IT neurons (Figure 6b).

Several additional studies have yielded similar results and are beginning to characterize the extent to which representations at different depths can explain the representational stages of the ventral stream (Agrawal et al. 2014, Güçlü & van Gerven 2015). Overall, these early empirical comparisons between deep neural network models and the primate ventral stream suggest four

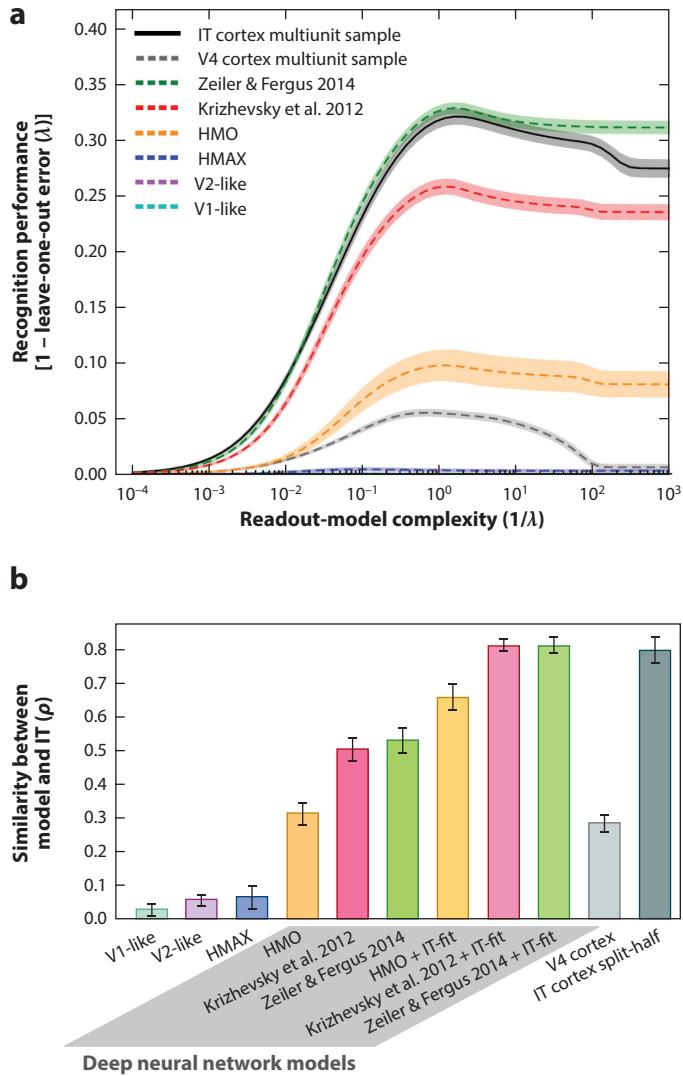


Figure 6

Deep neural networks beat simpler computational models at recognition and better explain the IT representation. (a) Object recognition performance of deep neural networks beats that of shallower models and rivals that of a population of IT neurons recorded in a monkey. Recognition performance (vertical axis) is plotted as a function of readout-model complexity (horizontal axis); high performance at low complexity indicates that the categories occupy easily separable regions in the representational space. (b) Deep neural network representations more closely resemble IT than do three simpler models (V1-like, V2-like, and HMAX). The similarity between each model and IT (vertical axis) was measured using the Spearman's rank correlation coefficient (ρ) to compare representational dissimilarity matrices. Results reproduced from Cadieu et al. (2014). Abbreviations: HMAX, hierarchical model and X (Riesenhuber & Poggio 1999, Serre et al. 2007, Tarr 1999); HMO, hierarchical modular optimization model (Yamins et al. 2014); IT, inferior temporal cortex.

Discriminative model: a model that extracts information of interest from the data (e.g., the image) without explicitly representing the process that generated the data

conclusions: (a) Only deep neural networks perform object recognition at levels comparable to human performance; (b) only deep neural networks explain the representational geometry of IT; (c) the representation appears to be gradually transformed, with lower layers resembling the earlier stages of the primate ventral stream; and (d) the high-level deep network representation resembles IT not merely in that it emphasizes categorical divisions, but also in its within-category representational geometry.

RECURRENT NEURAL NETWORKS FOR VISION

Feedforward networks are useful as models of the initial sweep of neuronal signaling through the visual hierarchy. They go some way toward explaining vision at a glance. However, feedforward networks are unlike the brain in terms of their connectivity and dynamics and are fundamentally limited to the computation of static functions. Rather than computing a static function on each of a series of image frames, vision takes a time-continuous input stream and interprets it through ongoing recurrent computations. The current network state likely represents the recent stimulus history, along with predictions of impending events and other behaviorally important information.

Recurrent computations probably contribute to the automatic rapid interpretation of even static visual images (Sugase et al. 1999, Brincat & Connor 2006, Freiwald & Tsao 2010, Carlson et al. 2013, Cichy et al. 2014, H. Tang et al. 2014, Clarke et al. 2015), leading to the emergence of representations that clearly distinguish particular objects and object categories. Individual faces, for example, become more clearly distinguishable in the monkey neuronal population codes at latencies that exceed the 100 ms or so that it takes the feedforward sweep to reach IT (Sugase et al. 1999, Freiwald & Tsao 2010). Similarly, at the level of object categories, evidence from human magnetoencephalography (MEG) suggests that strong categorical divisions arise only at latencies of over 200 ms after stimulus onset (Carlson et al. 2013, Cichy et al. 2014, Clarke & Tyler 2015, Clarke et al. 2015). Both category and exemplar representations, thus, may rely on recurrent processing to achieve invariance to irrelevant variation among images that carry the same essential meaning.

The brain might rely on a combination of feedforward and recurrent processing to arrive at a representation similar to that computed in feedforward convolutional networks trained for object recognition. We have seen that a recurrent network can be unfolded as a deep feedforward network. Conversely, when the numbers of units and connections are limited, a desirable function computed by a very large feedforward network might alternatively be approximated by recurrent computations in a smaller network. Recurrent dynamics can expand computational power by multiplying the limited physical resources for computation along time.

Recurrent neuronal dynamics likely also serve more sophisticated computations than those of feedforward convolutional networks. For example, assume the function of a visual neuron is to represent the presence of some piece of content in the image (a feature, an object part, or an object). The feedforward sweep alone might not provide the full evidence the neuron needs to confidently detect the piece of content that it represents. The neuron might therefore integrate later-arriving lateral and top-down signals to converge on its ultimate response. Multiple neurons might pass messages recurrently until the population converges on a stable interpretation of the image.

Recurrent computations might implement the iterative fitting to the image of a generative model of image formation, in which the fitted parameters specify the contents (and causes) of the image (see the sidebar The Deep Mystery of Vision: How to Integrate Generative and Discriminative Models). Assume, for simplicity, that the generative model is exactly invertible. This might be plausible if the model includes prior world knowledge sufficient to disambiguate visual images. Images and symbolic descriptions of their contents are then related by a one-to-one (bijective)

THE DEEP MYSTERY OF VISION: HOW TO INTEGRATE GENERATIVE AND DISCRIMINATIVE MODELS

The recent advances in computer vision were largely driven by feedforward neural networks. These models are discriminative: They discriminate categories among sets of images without an explicit model of the image-formation process. A more principled way to implement vision (or any data analysis) is to formulate a model of the process that generated the image (the data) and then to invert the process in order to infer the parameters of the model from the data (for a textbook on this approach in computer vision, see Prince 2012).

For vision, the generative model is an image-formation (or graphics) model that generates images from some high-level representation, such as a symbolic description of the visual scene. The first challenge is to define such a model. The second challenge is to perform inference on it—that is, to find the high-level representation that best explains the image, for example, the maximum a posteriori estimate or the full posterior probability distribution over all possible high-level representations, given the image. The idea of an active search for the interpretation that best explains the evidence in the context of our prior knowledge about the world is captured in von Helmholtz's (1866) description of vision as unconscious inference.

Research in computer vision and biological vision has always spanned the entire gamut from discriminative to generative approaches (Knill et al. 1996, Yuille & Kersten 2006, Prince 2012). However, the generative approach is practically challenging for computer vision and theoretically challenging for neuroscience. Most computer vision systems, whether using hand-engineered features or deep learning, therefore still rely primarily on discriminative models, learning mostly feedforward computations that process images to produce the desired outputs (but see Prince 2012). In neuroscience, similarly, feedforward models such as HMAX (Riesenhuber & Poggio 1999) have been influential.

Image formation involves nonlinear processes such as occlusion. Whereas the inversion of a linear generative model has a closed-form solution that can be implemented in a feedforward computation, inverting a graphics model is computationally much more challenging. Inferring a high-level scene description from an image requires consideration (at some level of abstraction) of a combinatorial explosion of possible configurations of objects and lights.

The deep mystery of vision is exactly how discriminative and generative models are integrated into a seamless and efficient process of inference. Vision might rely on a discriminative feedforward model for rapid recognition at a glance and on recurrent dynamics for iterative refinement of the inference, for correcting the errors of an initial feedforward estimate, or for probabilistic inference on hypotheses highlighted by the feedforward pass.

Recurrent neural networks can implement such dynamic inference processes and, given recent successes in the domain of language processing, seem poised for a fundamental advance in vision research.

mapping. In principle, the inverse of the generative model could be represented by a feedforward model (because of universality). Such a model might require too many neurons and connections, however, or its connectivity might be impossible to learn from limited training data. Instead of analyzing the image through feedforward computations, we can perform analysis by synthesis (Yuille & Kersten 2006), fitting a generative model of image formation to the particular image to be recognized.

The inversion of generative models has long been explored in both brain science and computer vision (Knill et al. 1996, Yuille & Kersten 2006, Prince 2012). The inference problem is difficult because a vast number of combinations of surfaces and lights can explain any image. To constrain the search space and disambiguate the solution, the brain must use massive prior knowledge about the world. Inference on a generative model might be tractable if performed on an intermediate-level representation of the image computed by discriminative mechanisms. How

the brain combines discriminative computations with inference on generative models to perceive the world is one of the fundamental unsolved problems in brain science.

The Helmholtz machine (Dayan et al. 1995) uses analysis by synthesis at the level of learning. A bottom-up recognition model and a top-down generative model are concurrently learned so as to best represent the distribution of the inputs in a maximum-likelihood sense. The learning can be performed using the wake-sleep algorithm (Hinton et al. 1995, Dayan 2003). In the wake phase, the recognition model “perceives” training images, and the generative model learns to better reconstruct these images from their internal representations. In the sleep phase, the generative model “dreams” of images, and the recognition model learns to better infer the internal representations from the images. By alternating wake and sleep phases, the two models coadapt and jointly discover a good representation for the distribution of images used in training.

A recurrent network could use the feedforward sweep to compute an initial rough estimate of the causes, and it could use subsequent recurrent computations to iteratively reduce the prediction error of the generative model and to explain nonlinear interactions of the parts, such as occlusion. The process could use predictive coding (Lee & Mumford 2003, Friston 2010) with recognized parts of the image explained away (and subtracted out of) lower-level representations. In such a process, the parts yet unexplained would be gradually uncluttered in the low-level representation and contextualized by the high-level representation as the easier, and then the more difficult, components of the image are successively explained.

Recurrent computations might converge on a point estimate of the parameters of a generative model of the image. Alternatively, they might implement probabilistic inference, converging on a representation of the posterior distribution over the parameters of the generative model. Recurrent message passing can implement belief propagation, an algorithm for probabilistic inference on a generative model. If the model captures the causal process giving rise to images, the recurrent dynamics can infer the specific causes (e.g., the objects, their properties, and the lighting) of a particular image. This process can be implemented in recurrent neural networks and might explain how the brain performs optimal cue combination, temporal integration, and explaining away (Lochmann & Deneve 2011). Belief propagation is a deterministic algorithm for probabilistic inference. Another deterministic proposal is based on probabilistic population codes (Ma et al. 2006).

Alternatively, a neural network might perform probabilistic inference by Markov chain Monte Carlo (MCMC) sampling, using neural stochasticity as a random generator (Hoyer & Hyvärinen 2003, Fiser et al. 2010, Buesing et al. 2011, McClelland 2013, Häfner et al. 2014). In this view, a snapshot of neural population activity represents a point estimate of the stimulus, and a temporal sequence of such snapshots represents the posterior distribution. For near-instantaneous readout of a probabilistic representation, several MCMC chains could operate in parallel (Savin & Deneve 2014). The sampling approach naturally handles the representation of joint probability distributions of multiple variables.

These proposals are exciting because they explain how the brain might perform formal probabilistic inference with neurons, linking the biological hardware to the high-level goal of rational information processing. The ultimate goal, of course, is not rational inference, but successful behavior, that is, survival and reproduction. We should expect the brain to perform probabilistic inference only to the extent to which it is expedient to do so in the larger context of successful behavior (Gershman et al. 2015).

How the probabilistic inference proposals of computational neuroscience scale up to the real-world challenges of vision remains to be seen. If they do, they might have a central future role in both brain theory and computer vision. The brain clearly handles uncertainty well in many contexts (Tenenbaum et al. 2006, Pouget et al. 2013), so it is helpful to view its inferences as approximations, however rough, to rational probabilistic inference.

At a larger timescale, vision involves top-down effects related to expectation and attentional scrutiny, as well as active exploration of a scene through a sequence of eye movements and through motor manipulations of the world. With the recurrent loop expanded to pass through the environment, these processes bring limited resources (the fovea, conscious attention) to different parts of a scene sequentially, selectively sampling the most relevant information while accumulating evidence toward an overall interpretation. Active perception is being explored in the computational literature. For example, Y. Tang et al. (2014b) built a model for face recognition that uses a convolutional feedforward pass for initialization and an attentional mechanism for selection of a region of interest, on which probabilistic inference is performed using a generative model, which itself is learned from data.

The challenge ahead is, first, to scale recurrent neural network models for vision to real-world tasks and human performance levels and, second, to fit and compare their representational dynamics to biological brains. Recurrent models are already successful in several domains of AI, including video-to-text description (Venugopalan et al. 2015), speech-to-text recognition (Sak et al. 2014), text-to-text language translation (Sutskever et al. 2014, Cho et al. 2014), and text-to-speech synthesis (Fan et al. 2014). In brain science, recurrent neural net models will ultimately be needed to explain every major function of information processing, including vision, other perceptual processes, cognition, and motor control.

CONCLUSIONS

Computational neuroscience has been very successful by asking what the brain *should* compute (Körding 2007). The normative goals proposed have often led to important insights before being replaced by larger goals. Should the brain efficiently encode sensory information [Barlow 1961 (2012)]? Or should it infer an accurate probabilistic representation of the world (Barlow 2001)? The ultimate goal is successful behavior.

Normative theory has driven advances at the cognitive and neuronal levels. Successes of this approach include theories of efficient coding [Barlow 1961 (2012), Olshausen & Field 1997, Simoncelli & Olshausen 2001], probabilistic neuronal coding and inference (Hoyer & Hyvärinen 2003, Fiser et al. 2010, Buesing et al. 2011, McClelland 2013, Pouget et al. 2013), Bayesian sensorimotor control (Körding & Wolpert 2006), and probabilistic cognition (Tenenbaum et al. 2006). For low-level sensory representations and for low-dimensional decision and control processes, normative theories prescribe beautiful and computationally simple inference procedures, which we know how to implement in computers and which might plausibly be implemented in biological brains. However, visual recognition and many other feats of brain information processing require inference using massive amounts of world knowledge. Not only are we missing a normative theory that would specify the optimal solution, but, until recently, we were not even able to implement any functioning solution.

Until recently, computers could not do visual object recognition, and image-computable models that could predict higher-level representations of novel natural images did not exist. Deep neural networks put both the task of object recognition and the prediction of high-level neural responses within our computational reach. This advance opens up a new computational framework for modeling high-level vision and other brain functions.

Deep neural net models are optimized for task performance. In this sense, the framework addresses the issue of what the brain *should* compute at the most comprehensive level: that of successful behavior. In its current instantiation, the deep net framework gives up an explicit probabilistic account of inference, in exchange for neurally plausible models that have sufficient capacity to

solve real-world tasks. We will see in the future whether explicitly probabilistic neural net models can solve the real-world tasks and explain biological brains even better.

Synthetic neurophysiology:

computational analysis of responses and dynamics of artificial neural networks aimed to gain a higher-level understanding of their computational mechanisms

Replacing One Black Box by Another?

One criticism of using complex neural networks to model brain information processing is that it replaces one impenetrably complex network with another. We might be able to capture the computations, but we are capturing them in a large net, the complexity of which defies conceptual understanding. There are two answers to the criticism of impenetrability.

First, it is true that our job is not done when we have a model that is predictive of neural responses and behavior. We must still strive to understand—at a higher level of description—how exactly the network transforms representations across the multiple stages of a deep hierarchy (and across time when the network is recurrent). However, once we have captured the complex biological computations in an artificial neural network, we can study its function efficiently in silico—with full knowledge of its internal dynamics. Synthetic neurophysiology, the analysis and visualization of artificial network responses to large natural and artificial stimulus sets, might help reveal the internal workings of these networks (Zeiler & Fergus 2014, Girshick et al. 2014, Simonyan et al. 2014, Tsai & Cox 2015, Zhou et al. 2015, Yosinski et al. 2015).

The second answer to the criticism of the impenetrability of neural network models is that we should be prepared to deal with mechanisms that elude a concise mathematical description and an intuitive understanding. After all, intelligence requires large amounts of domain-specific knowledge, and compressing this knowledge into a concise description or mathematical formula might not be possible. In other words, our models should be as simple as possible, but no simpler.

Similar to computational neuroscience, AI began with simple and general algorithms. These algorithms did not scale up to real-world applications, however. Real intelligence turned out to require incorporating large amounts of knowledge. This insight eventually led to the rise of machine learning. Computational neuroscience must follow in the footsteps of AI and acknowledge that most of what the brain does requires ample domain-specific knowledge learned through experience.

Are Deep Neural Net Models Similar to Biological Brains?

The answer to this question is in the eye of the beholder. We can focus on the many abstractions from biological reality and on design decisions driven by engineering considerations and conclude that they are very different. Alternatively, we can focus on the original biological inspiration and on the fact that biological neurons can perform the operations of model units, and conclude that they are similar.

Abstraction from biological detail is desirable and is in fact a feature of all models of computational neuroscience. A model is not meant to be identical to its object, but rather to explain it at an abstract level of description. Merely pointing out a difference to biological brains, therefore, does not constitute a legitimate challenge. For example, the fact that real neurons spike does not pose a challenge to a rate-coding model. It just means that biological brains can be described at a finer level of detail than the model does not address. If spiking were a computational requirement (e.g., Buesing et al. 2011) and a spiking model outperformed the best rate-coding model at its own game of predicting spike rates, or at predicting behavior, however, then this model would present a challenge to the rate-coding approach.

Many features of the particular type of deep convolutional feedforward network currently dominating computer vision deserve to be challenged in the context of modeling biological vision (see the sidebar Adversarial Examples Can Reveal Idiosyncrasies of Neural Networks). The features

ADVERSARIAL EXAMPLES CAN REVEAL IDIOSYNCRASIES OF NEURAL NETWORKS

Fooling vision can help us learn about its mechanisms. This is true for both biological and artificial vision. Researchers are exploring how artificial neural networks represent images by trying to fool them (Szegedy et al. 2014, Goodfellow et al. 2015, Nguyen et al. 2015). They use optimization techniques to design images that are incorrectly classified. An adversarial example is an image from category X (e.g., a bus or a noise image) that has been designed to be misclassified by a particular network as belonging to category Y (e.g., an ostrich). Such an image can be designed by taking a natural image from category X and adjusting the pixels to fool the net. The backpropagation algorithm, which usually serves to find small adjustments to the weights that reduce the error for an image, can be used to find small adjustments to the image that instead create an error. For the convolutional neural networks currently used in computer vision, adversarial examples can be created by very slight changes to the image that clearly do not render it a valid example of a different category. Such adversarial examples can look indistinguishable from the original image to humans. This has been taken as evidence of the limitations of current neural network architectures both as vision systems and as models of human vision.

An adversarial example created to fool an artificial neural network will not usually fool a human observer. However, it is not known whether adversarial examples can similarly be created for human visual systems. The technique described above for constructing adversarial examples requires full knowledge of the connections of the particular network to be fooled. Current psychophysical and neurophysiological techniques cannot match this process to fool biological vision. An intriguing possibility, thus, is that biological visual systems, too, are susceptible to adversarial examples. These could exploit idiosyncrasies of a particular brain, such that an adversarial example created to fool one person will not fool another. The purpose of vision systems is to work well under natural conditions, not to be immune to extremely savvy sabotage that requires omniscient access to the internal structure of a network and precise stabilization of the fooling image on the visual sensor array. Human vision is famously susceptible to visual illusions of various kinds. Moreover, from a machine learning perspective, it appears inevitable that adversarial examples can be constructed for any learning system—artificial or natural—that must rely on an imperfect inductive bias to generalize from a limited set of examples to a high-dimensional classification function.

What lessons do the adversarial examples hold about current neural network models? If adversarial examples fooled only the particular instance of a network for which they were constructed, exploiting the idiosyncrasies of that particular net, then they would be easy to dismiss. However, adversarial examples generalize across networks to some extent. If a new network is created by initializing the same architecture with new random weights and training it with the same set of labeled images, the resulting network will often still be fooled by an adversarial example created for the original network. Adversarial examples also generalize to slightly altered architectures if the same training set is used. If the training set is changed, adversarial examples created for the original network are not very effective anymore, but they may still be misclassified at a higher rate than natural images. This suggests that adversarial examples exploit network idiosyncrasies resulting largely from the training set, but also, to some extent, from the basic computational operations used. One possibility is that the linear combination computed by each unit in current systems makes these systems particularly easy to fool (Goodfellow et al. 2015). In essence, each unit divides its input space by a linear boundary (even if its activation rises smoothly as we cross the boundary for sigmoid or linearly on the preferred side for rectified linear activation functions). In contrast, networks using radial basis functions, in which each unit has a particular preferred pattern in its input space and the response falls off in all directions, might be harder to fool. However, these networks are also harder to train—and perhaps for the same reason. It will be intriguing to see this puzzle solved as we begin to compare the complex representational transformations between artificial and biological neural networks in greater detail.

that deserve to be challenged first are the higher-level computational mechanisms, such as the lack of bypass connections in the feedforward architecture, the lack of feedback and local recurrent connections, the linear–nonlinear nature of the units, the rectified linear activation function, the max-pooling operation, and the offline supervised gradient-descent learning. To challenge one of these features, we must demonstrate that measured neuronal responses or behavioral performance can be more accurately predicted using a different kind of model.

The neural network literature is complex and spans the gamut from theoretical neuroscience to computer science. This literature includes feedforward and recurrent, discriminative and generative, deterministic and stochastic, nonspiking and spiking models. It provides the building blocks for tomorrow’s more comprehensive theories of information processing in the brain. Now that these models are beginning to scale up to real-world tasks and human performance levels in engineering, we can begin to use this modeling framework in brain science to tackle the complex processes of perception, cognition, and motor control.

The Way Ahead

We will use modern neural network technology with the goal of approximating the internal dynamics and computational function of large portions of biological brains, such as their visual systems. An important goal is to build models with layers that correspond one-to-one to visual areas, and with receptive fields, nonlinear response properties, and representational geometries that match those of the corresponding primate visual areas. The requirement that the system perform a meaningful task such as object recognition provides a major functional constraint. Task training of neural networks with millions of labeled images currently provides much stronger constraints than neurophysiological data do on the space of candidate models. Indeed, the recent successes at predicting brain representations of novel natural images are largely driven by task training (Yamins et al. 2014, Khaligh-Razavi & Kriegeskorte 2014, Cadieu et al. 2014). However, advances in massively parallel brain-activity measurement promise to provide stronger brain-based constraints on the model space in the future. Rather than minimizing a purely task-based loss function, as commonly done in engineering, modeling biological brains will ultimately require novel learning algorithms that drive connectivity patterns, internal representations, and task performance into alignment with brain and behavioral measurements. In order to model not only the final processing mechanism but the learning process in a biologically plausible way, we will also need to employ unsupervised and reinforcement learning techniques (Sutton & Barto 1998, Mnih et al. 2015).

AI, machine learning, and the cognitive and brain sciences have deep common roots. At the cognitive level, these fields have recently converged through Bayesian models of inference and learning (Tenenbaum et al. 2006). Like deep networks, Bayesian nonparametric techniques (Ghahramani 2013) can incorporate large amounts of world knowledge. These models have the advantage of explicit probabilistic inference and learning. Explaining how such inference processes might be implemented in biological neural networks is one of the major challenges ahead. Neural networks have a long history in AI, in cognitive science, in machine learning, and in computational neuroscience. They provide a common modeling framework to link these fields. The current vindication in engineering of early intuitions about the power of brain-like deep parallel computation reinvigorates the convergence of these disciplines. If we can build models that perform complex feats of real-world intelligence (AI) and explain neuronal dynamics (computational neuroscience) and behavior (cognitive science), then—for the tasks tackled—we will understand how the brain works.

SUMMARY POINTS

1. Neural networks are brain-inspired computational models that now dominate computer vision and other AI applications.
2. Neural networks consist of interconnected units that compute nonlinear functions of their input. Units typically compute weighted combinations of their inputs followed by a static nonlinearity.
3. Feedforward neural networks are universal function approximators.
4. Recurrent neural networks are universal approximators of dynamical systems.
5. Deep neural networks stack multiple layers of nonlinear transformations and can concisely represent complex functions such as those needed for vision.
6. Convolutional neural networks constrain the input connections of units in early layers to local receptive fields with weight templates that are replicated across spatial positions. The restriction and sharing of weights greatly reduce the number of parameters that need to be learned.
7. Deep convolutional feedforward networks for object recognition are not biologically detailed and rely on nonlinearities and learning algorithms that may differ from those of biological brains. Nevertheless they learn internal representations that are highly similar to representations in human and nonhuman primate IT cortex.
8. Neural networks now scale to real-world AI tasks, providing an exciting technological framework for building more biologically faithful models of complex feats of brain information processing.

FUTURE ISSUES

1. We will build neural net models that engage complex real-world tasks and simultaneously explain biological brain-activity patterns and behavioral performance.
2. The models will have greater biological fidelity in terms of architectural parameters, nonlinear representational transformations, and learning algorithms.
3. Network layers should match the areas of the visual hierarchy in their response characteristics and representational geometries.
4. Models should predict a rich array of behavioral measurements, such as reaction times for particular stimuli in different tasks, similarity judgments, task errors, and detailed motor trajectories in continuous interactive tasks.
5. New supervised learning techniques will drive neural networks into alignment with measured functional and anatomical brain data and with behavioral data.
6. Recurrent neural network models will explain the representational dynamics of biological brains.
7. Recurrent neural network models will explain how feedforward, lateral, and feedback information flow interact to implement probabilistic inference on generative models of image formation.

8. We will tackle more complex visual functions beyond categorization, such as identification of unique entities, attentional shifts and eye movements that actively explore the scene, visual search, image segmentation, more complex semantic interpretations, and sensorimotor integration.

DISCLOSURE STATEMENT

The author is not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

The author thanks Seyed Khaligh-Razavi and Daniel Yamins for helpful discussions, and Patrick McClure and Katherine Storrs for comments on a draft of the manuscript. This research was funded by the UK Medical Research Council (Program MC-A060-5PR20), a European Research Council Starting Grant (ERC-2010-StG 261352), and a Wellcome Trust Project Grant (WT091540MA).

LITERATURE CITED

- Agrawal P, Stansbury D, Jitendra Malik J, Gallant JL. 2014. Pixels to voxels: modeling visual representation in the human brain. arXiv:1407.5104 [q-bio.NC]
- Barlow H. 2001. Redundancy reduction revisited. *Netw. Comput. Neural Syst.* 2(3):241–53
- Barlow HB. 1961 (2012). Possible principles underlying the transformations of sensory messages. In *Sensory Communication*, ed. WA Rosenblith, pp. 217–34. Cambridge, MA: MIT Press
- Bengio Y. 2009. *Learning Deep Architectures for AI*. Hanover, MA: Now
- Brincat SL, Connor CE. 2006. Dynamic shape synthesis in posterior inferotemporal cortex. *Neuron* 49:17–24
- Buesing L, Bill J, Nessler B, Maass W. 2011. Neural dynamics as sampling: a model for stochastic computation in recurrent networks of spiking neurons. *PLOS Comput. Biol.* 7(11):e1002211
- Cadieu CF, Hong H, Yamins DL, Pinto N, Ardisa D, et al. 2014. Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLOS Comput. Biol.* 10(12):e1003963
- Cadieu CF, Hong H, Yamins DL, Pinto N, Majaj NJ, DiCarlo JJ. 2013. *The neural representation benchmark and its evaluation on brain and machine*. Presented at Int. Conf. Learn. Represent., Scottsdale, AZ, May 2–4. arXiv:1301.3530 [cs.NE]
- Carlson T, Tovar DA, Alink A, Kriegeskorte N. 2013. Representational dynamics of object vision: the first 1000 ms. *J. Vis.* 13:1
- Chatfield K, Simonyan K, Vedaldi A, Zisserman A. 2014. *Return of the devil in the details: delving deep into convolutional nets*. Presented at Br. Mach. Vis. Conf., Nottingham, UK, Sept. 1–5. arXiv:1405.3531 [cs.CV]
- Cho K, van Merriënboer B, Gulcehre C, Bougares F, Schwenk H, Bengio Y. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. Presented at Conf. Empir. Methods Nat. Lang. Process., Doha, Qatar, Oct. 25–29. arXiv:1406.1078 [cs.CL]
- Cichy RM, Pantazis D, Oliva A. 2014. Resolving human object recognition in space and time. *Nat. Neurosci.* 17:455–62
- Clarke A, Devereux BJ, Randall B, Tyler LK. 2015. Predicting the time course of individual objects with MEG. *Cereb. Cortex* 25:3602–12
- Clarke A, Tyler LK. 2015. Understanding what we see: how we derive meaning from vision. *Trends Cogn. Sci.* 19:677–87
- Cybenko G. 1989. Approximation by superpositions of a sigmoid function. *Math. Control Signals Syst.* 2:303–14

- Dayan P. 2003. Helmholtz machines and wake-sleep learning. In *The Handbook of Brain Theory and Neural Networks*, ed. MA Arbib, pp. 520–25. Cambridge, MA: MIT Press. 2nd ed.
- Dayan P, Hinton GE, Neal RM, Zemel RS. 1995. The Helmholtz machine. *Neural Comput.* 7(5):889–904
- Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L. 2009. ImageNet: a large-scale hierarchical image database. *Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 20–25, Miami*, pp. 248–55. New York: IEEE
- Dumoulin SO, Wandell BA. 2008. Population receptive field estimates in human visual cortex. *NeuroImage* 39:647–60
- Fan Y, Qian Y, Xie F-L, Soong FK. 2014. TTS synthesis with bidirectional LSTM based recurrent neural networks. *Proc. 15th Annu. Conf. Int. Speech Commun. Assoc., Sept. 14–18, Singapore*, pp. 1964–68. Baixas, Fr.: ISCA
- Fiser J, Berkes P, Orbán G, Lengyel M. 2010. Statistically optimal perception and learning: from behavior to neural representations. *Trends Cogn. Sci.* 14:119–30
- Freiwald WA, Tsao DY. 2010. Functional compartmentalization and viewpoint generalization within the macaque face-processing system. *Science* 330(6005):845–51
- Friston K. 2010. The free-energy principle: a unified brain theory? *Nat. Rev. Neurosci.* 11(2):127–38
- Fukushima K. 1980. Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol. Cybern.* 36(4):193–202
- Gallistel CR, King AP. 2011. *Memory and the Computational Brain: Why Cognitive Science Will Transform Neuroscience*. Chichester, UK: Wiley-Blackwell
- Gershman SJ, Horvitz EJ, Tenenbaum JB. 2015. Computational rationality: a converging paradigm for intelligence in brains, minds, and machines. *Science* 349(6245):273–78
- Ghahramani Z. 2013. Bayesian non-parametrics and the probabilistic approach to modelling. *Philos. Trans. R. Soc. Lond. A* 371:20110553
- Girshick R, Donahue J, Darrell T, Malik J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv:1311.2524 [cs.CV]
- Goodfellow IJ, Shlens J, Szegedy C. 2015. Explaining and harnessing adversarial examples. arXiv:1412.6572v3 [stat.ML]
- Graves A, Schmidhuber J. 2009. Offline handwriting recognition with multidimensional recurrent neural networks. *Adv. Neural Inf. Process. Syst.* 21:545–52
- Güçlü U, van Gerven MA. 2015. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *J. Neurosci.* 35(27):10005–14
- Häfner RM, Berkes P, Fiser J. 2014. Perceptual decision-making as probabilistic inference by neural sampling. arXiv:1409.0257 [q-bio.NC]
- Hilgetag CC, O'Neill MA, Young MP. 2000. Hierarchical organization of macaque and cat cortical sensory systems explored with a novel network processor. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 355:71–89
- Hinton GE, Dayan P, Frey BJ, Neal RM. 1995. The “wake-sleep” algorithm for unsupervised neural networks. *Science* 268(5214):1158–61
- Hinton GE, Salakhutdinov RR. 2006. Reducing the dimensionality of data with neural networks. *Science* 313(5786):504–7
- Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. 2012. Improving neural networks by preventing co-adaptation of feature detectors. arXiv:1207.0580 [cs.NV]
- Hochreiter S. 1991. *Untersuchungen zu dynamischen neuronalen Netzen*. Master’s Thesis, Inst. Inform., Tech. Univ. München
- Hochreiter S, Bengio Y, Frasconi P, Schmidhuber J. 2001. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In *A Field Guide to Dynamical Recurrent Neural Networks*, ed. SC Kremer, JF Kolen, pp. 237–244. New York: IEEE
- Hochreiter S, Schmidhuber J. 1997. Long short-term memory. *Neural Comput.* 9(8):1735–80
- Hornik K. 1991. Approximation capabilities of multilayer feedforward networks. *Neural Netw.* 4(2):251–57
- Hoyer PO, Hyvärinen A. 2003. Interpreting neural response variability as Monte Carlo sampling of the posterior. *Adv. Neural Inform. Proc. Syst.* 15:293–300
- Hubel DH, Wiesel TN. 1968. Receptive fields and functional architecture of monkey striate cortex. *J. Physiol.* 195:215

- Jaeger H. 2001. *The “echo state” approach to analysing and training recurrent neural networks—with an erratum note*. GMD Tech. Rep. 148, Ger. Natl. Res. Cent. Inf. Technol., Bonn
- Jozwik KM, Kriegeskorte N, Mur M. 2015. Visual features as stepping stones toward semantics: explaining object similarity in IT and perception with non-negative least squares. *Neuropsychologia*. In press
- Kay KN, Naselaris T, Prenger RJ, Gallant JL. 2008. Identifying natural images from human brain activity. *Nature* 452:352–55
- Khaligh-Razavi S-M, Kriegeskorte N. 2013. *Object-vision models that better explain IT also categorize better, but all models fail at both*. Presented at COSYNE, Salt Lake City, UT
- Khaligh-Razavi S-M, Kriegeskorte N. 2014. Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLOS Comput. Biol.* 10(11):e1003915. doi:10.1371/journal.pcbi.1003915
- Kiani R, Esteky H, Mirpour K, Tanaka K. 2007. Object category structure in response patterns of neuronal population in monkey inferior temporal cortex. *J. Neurophysiol.* 97:4296–309
- Knill DC, Kersten D, Yuille A. 1996. Introduction: a Bayesian formulation of visual perception. In *Perception as Bayesian Inference*, ed. DC Knill, W Richards, pp. 1–21. Cambridge, UK: Cambridge Univ. Press
- Körding K. 2007. Decision theory: What “should” the nervous system do? *Science* 318(5850):606–10
- Körding KP, Wolpert DM. 2006. Bayesian decision theory in sensorimotor control. *Trends Cogn. Sci.* 10(7):319–26
- Kriegeskorte N. 2011. Pattern-information analysis: from stimulus decoding to computational-model testing. *NeuroImage* 56:411–21
- Kriegeskorte N, Kievit RA. 2013. Representational geometry: integrating cognition, computation, and the brain. *Trends Cogn. Sci.* 17:401–12
- Kriegeskorte N, Mur M, Bandettini P. 2008a. Representational similarity analysis—connecting the branches of systems neuroscience. *Front. Syst. Neurosci.* 2:4
- Kriegeskorte N, Mur M, Ruff DA, Kiani R, Bodurka J, et al. 2008b. Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron* 60:1126–41
- Krizhevsky A, Sutskever I, Hinton GE. 2012. ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* 25:1097–105
- LeCun Y, Bengio Y. 1995. Convolutional networks for images, speech, and time series. In *The Handbook of Brain Theory and Neural Networks*, ed. MA Arbib, pp. 255–58. Cambridge, MA: MIT Press
- LeCun Y, Bengio Y, Hinton G. 2015. Deep learning. *Nature* 521:436–44
- LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, et al. 1989. Backpropagation applied to handwritten zip code recognition. *Neural Comput.* 1(4):541–51
- Lee TS, Mumford D. 2003. Hierarchical Bayesian inference in the visual cortex. *J. Opt. Soc. Am. A* 20(7):1434–48
- Lochmann T, Deneve S. 2011. Neural processing as causal inference. *Curr. Opin. Neurobiol.* 21(5):774–81
- Lowe DG. 1999. Object recognition from local scale-invariant features. *Proc. 7th IEEE Int. Conf. Comput. Vis., Sept. 20–27, Kerkyra, Greece*, pp. 1150–57. New York: IEEE
- Ma WJ, Beck JM, Latham PE, Pouget A. 2006. Bayesian inference with probabilistic population codes. *Nat. Neurosci.* 9:1432–38
- Maass W, Natschläger T, Markram H. 2002. Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Comput.* 14(11):2531–60
- McClelland JL. 2013. Integrating probabilistic models of perception and interactive neural networks: a historical and tutorial review. *Front. Psychol.* 4:503
- McCulloch WS, Pitts W. 1943. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* 5(4):115–33
- Minsky M, Papert S. 1972. *Perceptrons: An Introduction to Computational Geometry*. Cambridge, MA: MIT Press
- Mitchell TM, Shinkareva SV, Carlson A, Chang K-M, Malave VL, et al. 2008. Predicting human brain activity associated with the meanings of nouns. *Science* 320:1191–95
- Moore CI, Cao R. 2008. The hemo-neural hypothesis: on the role of blood flow in information processing. *J. Neurophysiol.* 99(5):2035–47
- Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518:529–33

- Nguyen A, Yosinski J, Clune J. 2015. *Deep neural networks are easily fooled: high confidence predictions for unrecognizable images*. Presented at IEEE Conf. Comput. Vis. Pattern Recognit., June 7–12, Boston. arXiv:1412.1897v4 [cs.CV]
- Nili H, Wingfield C, Walther A, Su L, Marslen-Wilson W, Kriegeskorte N. 2014. A toolbox for representational similarity analysis. *PLOS Comput. Biol.* 10:e1003553
- Olshausen BA, Field DJ. 1997. Sparse coding with an overcomplete basis set: a strategy employed by V1? *Vis. Res.* 37(23):3311–25
- Pouget A, Beck JM, Ma WJ, Latham PE. 2013. Probabilistic brains: knowns and unknowns. *Nat. Neurosci.* 16(9):1170–78
- Prince SJ. 2012. *Computer Vision: Models, Learning, and Inference*. Cambridge, UK: Cambridge Univ. Press
- Riesenhuber M, Poggio T. 1999. Hierarchical models of object recognition in cortex. *Nat. Neurosci.* 2(11):1019–25
- Rosenblatt F. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.* 65(6):386
- Rumelhart DE, Hinton GE, Williams RJ. 1986. Learning representations by back-propagating errors. *Nature* 323(6088):533–36
- Rumelhart DE, McClelland JL, PDP Research Group. 1988. In *Parallel Distributed Processing* Vol. 1, pp. 354–62. New York: IEEE
- Sak H, Senior A, Beaufays F. 2014. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. arXiv:1402.1128 [cs.NE]
- Savin C, Deneve S. 2014. Spatio-temporal representations of uncertainty in spiking neural networks. *Adv. Neural Inf. Process. Syst.* 27:2024–32
- Schäfer AM, Zimmermann HG. 2007. Recurrent neural networks are universal approximators. *Int. J. Neural Syst.* 17(4):253–63
- Schmidhuber J. 2015. Deep learning in neural networks: an overview. *Neural Netw.* 61:85–117
- Serre T, Wolf L, Bileschi S, Riesenhuber M, Poggio T. 2007. Robust object recognition with cortex-like mechanisms. *IEEE Trans. Pattern Anal. Mach. Intell.* 29(3):411–26
- Simoncelli EP, Olshausen BA. 2001. Natural image statistics and neural representation. *Annu. Rev. Neurosci.* 24(1):1193–216
- Simonyan K, Vedaldi A, Zisserman A. 2014. Deep inside convolutional networks: visualising image classification models and saliency maps. arXiv:1312.6034 [cs.CV]
- Sugase Y, Yamane S, Ueno S, Kawano K. 1999. Global and fine information coded by single neurons in the temporal visual cortex. *Nature* 400(6747):869–73
- Sutton RS, Barto AG. 1998. *Reinforcement Learning: An Introduction*, Vol. 1. Cambridge, MA: MIT Press
- Sutskever I, Vinyals O, Le QV. 2014. Sequence to sequence learning with neural networks. *Adv. Neural Inf. Process. Syst.* 27:3104–12
- Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, et al. 2014. *Intriguing properties of neural networks*. Presented at Int. Conf. Learn. Represent., Apr. 14–16, Banff, Can. arXiv:1312.6199v4 [cs.CV]
- Tanaka K. 1996. Inferotemporal cortex and object vision. *Annu. Rev. Neurosci.* 19:109–39
- Tang H, Buia C, Madsen J, Anderson WS, Kreiman G. 2014. A role for recurrent processing in object completion: neurophysiological, psychophysical and computational evidence. arXiv:1409.2942 [q-bio.NC]
- Tang Y, Srivastava N, Salakhutdinov RR. 2014. Learning generative models with visual attention. *Adv. Neural Inf. Process. Syst.* 27:1808–16
- Tarr MJ. 1999. News on views: pandemonium revisited. *Nat. Neurosci.* 2:932–35
- Tenenbaum JB, Griffiths TL, Kemp C. 2006. Theory-based Bayesian models of inductive learning and reasoning. *Trends Cogn. Sci.* 10(7):309–18
- Tromans JM, Harris M, Stringer SM. 2011. A computational model of the development of separate representations of facial identity and expression in the primate visual system. *PLOS ONE* 6:e25616
- Tsai C-Y, Cox DD. 2015. Measuring and understanding sensory representations within deep networks using a numerical optimization framework. arXiv:1502.04972 [cs.NE]
- Venugopalan S, Xu H, Donahue J, Rohrbach M, Mooney R, Saenko K. 2015. Translating videos to natural language using deep recurrent neural networks. arXiv:1412.4729 [cs.CV]

- von Helmholtz H. 1866. *Handbuch der physiologischen Optik: Mit 213 in den Text eingedruckten Holzschnitten und 11 Tafeln*, Vol. 9. Voss
- Wallis G, Rolls ET. 1997. A model of invariant object recognition in the visual system. *Prog. Neurobiol.* 51:167–94
- Werbos PJ. 1981. Applications of advances in nonlinear sensitivity analysis. In *Proceedings of the 10th IFIP Conference*, pp. 762–70
- Yamins DL, Hong H, Cadieu CF, DiCarlo JJ. 2013. Hierarchical modular optimization of convolutional networks achieves representations similar to macaque IT and human ventral stream. *Adv. Neural Inf. Process. Syst.* 26:3093–101
- Yamins DLK, Hong H, Cadieu CF, Solomon EA, Seibert D, DiCarlo JJ. 2014. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *PNAS* 111:8619–24
- Yosinski J, Clune J, Nguyen A, Fuchs T, Lipson H. 2015. Understanding neural networks through deep visualization. arXiv:1506.06579 [cs.CV]
- Yuille A, Kersten D. 2006. Vision as Bayesian inference: analysis by synthesis? *Trends Cogn. Sci.* 10(7):301–8
- Zeiler MD, Fergus R. 2014. Visualizing and understanding convolutional networks. *Proc. 13th Eur. Conf. Comput. Vis., Sept. 6–12, Zurich*, pp. 818–833. New York: Springer
- Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A. 2015. *Object detectors emerge in deep scene CNNs*. Presented at Int. Conf. Learn. Represent., May 7–9, San Diego. arXiv:1412.6856 [cs.CV]

RELATED RESOURCES

- Bengio Y, Goodfellow I, Courville A. “Deep Learning” online book (in progress)
<http://www.iro.umontreal.ca/~bengioy/dlbook/>
- Hinton G. 2012. Coursera course “Neural Networks for Machine Learning”
<https://www.coursera.org/course/neuralnets>
- Ng A. Coursera course “Machine Learning”
<https://www.coursera.org/course/ml>
- Nielson M. “Neural Networks and Deep Learning” online book (in progress)
<http://neuralnetworksanddeeplearning.com/>

初心者のためのニューラルネットワーク

Deep Neural Networks: A New Framework for Modeling Biological Vision and Brain Information Processing

Nikolaus Kriegeskorte

2015年

- Annual Review Visison Science (2015) 1:417–46.
- URL:vision.annualreviews.org
- DOI: 10.1146/annurev-vision-082114-035447

キーワード: 生物学的視覚, コンピュータビジョン, 物体認識, ニューラルネットワーク, ディープラーニング, 人工知能, 計算論的神経科学

1 要約

近年, ニューラルネットワークのモデル化が進み, コンピュータビジョンをはじめとする人工知能の応用が大きく進展している。人間レベルの視覚認識能力が, 人工システムの手の届くところまで来ているのだ。人工ニューラルネットワークは, 脳にヒントを得ており, その計算は生物のニューロンに実装することができる。現在, コンピュータビジョンの主流となっている畳み込みフィードフォワードネットワークは, 犬長類の視覚階層のアーキテクチャからさらにインスピレーションを得ている。しかし, 現在のモデルは工学的な目的で設計されており, 脳の計算をモデル化したものではない。しかし, これらのモデルと犬長類の脳の内部表現を比較した初期の研究では, 表現空間が驚くほど似ていることがわかった。人間レベルの課題成績はもはや手の届かないものとなり, 私たちは, 生物学的に忠実なフィードフォワードおよびリカレント計算モデルを構築して, 生物の脳が視覚をはじめとする高レベルの知能をどのように発揮するかを知ることができる, エキサイティングな新時代を迎えようとしている。

2 用語集

ユニット: ニューロンの抽象化モデル, 典型的には入力信号の荷重和を計算し, 非線形変換を行う。

誤差逆伝播: 誤差を反復的に最小化するため, 接続性を逆に通過させることで, 重みに関する誤差導関数を効率的に計算する教師付きニューラルネットワーク学習アルゴリズム

生成モデル: データ処理(視覚認識など)の際に反転させたいデータ(画像など)を生成する過程のモデル

順伝播ネットワーク: 有向グラフを持つネットワークであり, 再帰結合を含まない

普遍関数近似器: 入力パターンと出力パターンを対応させるあらゆる関数を(十分なパラメータを与えれば任意の精度で)近似できるモデル群

深層学習: 複雑な表現を深層ニューラルネットワークで機械学習するもので、通常は確率的勾配降下法や誤差逆伝播法を用いる

深層ニューラルネットワーク: 入力層と出力層の間に2つ以上の隠れ層を持つネットワーク。

リカレントネットワーク: 情報の流れが再帰的で、時空間パターンの認識や生成に適したダイナミクスを持つネットワーク

力学系の普遍近似器 Universal approximator of dynamical systems: 任意の力学系を（十分なパラメータが与えられれば任意の精度で）近似できる力学を生成するモデル群。

教師あり学習: 入力パターンと、目的の表現や出力に関する追加情報（カテゴリーラベルなど）を必要とする学習プロセス

教師なし学習: 入力パターンのセットのみを必要とし、その確率分布の側面を把握する学習プロセス

畳み込みネットワーク: ある層の事前活性化（非線形性の前）により、いくつかの重みテンプレートパターンで前の層の畳み込みを実行するネットワーク

受容野モデリング: ニューロンの任意の感覚入力（または脳活動の測定チャネル）に対する反応を予測的にモデリングする手法

最大値プーリング: 無関係な特性（例えば、局所的な位置）が異なる検出器のセットの最大値のみを保持することで、不变性を実現する要約操作

正規化: 要約統計量を得るために活性値の集合に適用される操作（総和して除するなど）

ドロップアウト: ニューラルネットワークの学習における正則化手法の一つで、各学習試行において、各ユニットを確率0.5でアーキテクチャから除外する手法

GPU (Graphics Processing Unit): グラフィックス計算のために開発された特殊なコンピュータハードウェアで、行列と行列の乗算を大幅に高速化し、効率的な深層学習には欠かせない

表象類似性解析: 母集団コードの表現を特徴づける表象距離行列を統計的に比較することで、脳の情報処理の計算モデルを検証する手法

識別モデル: データを生成したプロセスを明示的に表現することなく、データ（画像など）から目的の情報を抽出するモデル

統合神経生理学: 人工的な神経ネットワークの反応やダイナミクスを計算機上で解析し、その計算メカニズムをより高度に理解することを目的とした学問

3 はじめに

脳は深くて複雑なリカレントニューラルネットワークである。一方、これまで計算論的神経科学の分野を席巻してきた情報処理モデルは、単純な計算を行う浅いアーキテクチャが中心であった。当然のことながら、視覚的物体認識のような複雑な課題は、計算神経科学の手の届かないところにあった（サイドバー「脳の複雑な情報処理を理解するためのこれまでの試みはいかにして失敗したか」参照）。本稿では、最近のニューラルネットワークモデルの進歩（LeCun et al. 2015）により、豊富な知識と複雑な計算を必要とする実世界の課題に取り組む計算神経科学の新時代が到来すると主張する。

ニューラルネットワークの歴史は古く、今何が新しいのだろうか。実際、ニューラルネットワークの歴史は、現代のコンピュータの歴史とほぼ同じである（サイドバー「ニューラルネットワークという言葉の意味」参照）。ジョン・フォン・ノイマンとアラン・チューリングは、現代のコンピュータ技術を形成するアイデアを持っていた。いずれも脳にヒントを得たネットワークモデルを探求していた。単一のニューロンの初期の数学的モデルは McCulloch & Pitts (1943) によって提案された。彼らの2値閾値ユニットは、いくつかの入力を受け取り、加重和を計算し、線形判別器を実装して閾値を設定した。閾値ユニットは、連続的な入力パターンに单一の2値出力で応答することで、スパイクニューロンの生物学的ハードウェアと、認知の特徴であるカテゴリ化との間の直感的な架け橋となつた。

入力では線形に分離できないカテゴリを判別するには、入力ユニットと出力ユニットの間に非線形変換の層を介在させる必要がある。このような入出力対を持つ多層ネットワークを自動的に学習する方法を見つけるのに、この分

野では時間がかかった。この問題に対する最も重要な解決策は、誤差逆伝播法（バックプロパゲーションアルゴリズム）である。バックプロパゲーションアルゴリズムは、出力の誤差を減らすために重みを反復的に小さく調整する勾配降下法である (Werbos 1981, Rumelhart et al. 1986)。

複雑な脳情報処理を如何に理解しようとしてきたのか？

認知科学や脳科学は、時代ごとに異なる分野が中心となって、一連の変革を遂げてきた。それぞれの分野では、脳の働きを理解するために必要な要素が異なっている（表1）。認知心理学は、行動主義のブラックボックスを情報処理の理論で解明しようとした。しかし、完全に明示的な計算モデルはなかった。認知科学は、情報処理理論を完全に明示した。しかし、神経生理学的なデータによる制約がないため、行動データと一致する複数の代替モデルを判断することが困難であった。認知科学のコネクショニズムは、神経生物学的に妥当な計算論的枠組みを提供した。しかし、ニューラルネットワークの技術は、写真からの物体認識のような実世界の課題に取り組むには十分ではなかった。その結果、ニューラルネットワークは当初、AIシステムとしての期待に応えることができず、認知科学ではモデル化はおもちゃの問題に限られていた。認知神経科学では、神経生理学的なデータを用いて、複雑な脳の情報処理を研究した。しかし、複雑な脳画像データの解析という新たな課題に手を焼いているうちに、理論的な洗練度は認知心理学の段階に戻ってしまい、（おそらく合理的に）以下のように始められた。箱と矢印のモデルを脳の領域にマッピングすることから始められた。計算論的神経科学では、完全に明示的で生物学的に妥当な計算モデルを用いて、神経生理学的数据や行動学的数据を予測する。しかし、このレベルの厳密さでは、現実世界の複雑な計算上の課題や、より高度な脳の表現に取り組むことはできなかった。ディープニューラルネットワークは、複雑な認知課題に取り組み、脳と行動の両方の反応を予測する枠組みを提供する。

Table 1 Historical progress toward understanding how the brain works

Elements required for understanding how the brain works		Behaviorism	Cognitive psychology	Cognitive science	Cognitive neuroscience	Classical computational neuroscience	Future cognitive computational neuroscience
Data	Behavioral	✓	✓	✓	✓	✓	✓
	Neurophysiological				✓	✓	✓
Theory	Cognitive		✓	✓	✓		✓
	Fully computationally explicit			✓		✓	✓
	Neurally plausible			✓		✓	✓
	Explanation of real-world tasks requiring rich knowledge and complex computations		✓		✓		✓
Explanation of how high-level neuronal populations represent and compute							✓

図1 表1

バックプロパゲーションをきっかけに、1980年代認知科学や人工知能（AI）の分野でニューラルネットワークが注目されるようになった。認知科学の分野では、おもちゃのような問題のニューラルネットワークモデルが、並列分散処理の理論的な概念を育てた (Rumelhart & McClelland, 1988)。しかし、バックプロパゲーションモデルは、視覚のような複雑な実世界の問題ではうまく機能しなかった。脳からの影響を受けていないモデルでは、サポートベクターマシンなどの機械学習技術を用いて、手作業で表現を行うモデルの方が、コンピュータビジョンやAIの工学的な解決策として優れていると考えられた。その結果、ニューラルネットワークは1990年代には人気がなくなってしまった。

ニューラルネットワーク研究は、脳科学やコンピュータサイエンスのコミュニティでは一時的に敬遠されていた。だが、理論的な神経科学やコンピュータサイエンスの分野では連綿とした歴史がある (Schmidhuber, 2015)。1990年代から2000年代にかけて、ニューラルネットは少数の科学者たちによって研究されていた。彼らが遭遇した困難はアプローチの根本的な限界ではなく、より優れた学習アルゴリズム、より優れた正則化、より大きなトレーニングセットの組み合わせによって克服すべき高いハードルに過ぎないことに気がついた。コンピュータのハードウェアの性

能向上に伴い、このコミュニティの努力は実りあるものとなった。ここ数年で、ニューラルネットワークはようやく本領を発揮するようになった。ニューラルネットワークは、コンピュータビジョンという難問を含む AI のいくつかの領域を制覇している。

ImageNet (Deng et al. 2009) のようなコンピュータビジョン競技会では、秘密のテストセットが用いられ、最先端の技術が厳密に評価される。2012 年の ImageNet では、Krizhevsky ら (2012) が開発したニューラルネットワークモデルが大差をつけて優勝した。このモデルは、深層畳み込みアーキテクチャにより、飛躍的な性能向上を実現した。人間の成績の方が依然として優れているが、少なくとも視覚物体分類のような限定された領域では、コンピュータビジョンにとって全く達成できないとは思えなくなった。Krizhevsky ら (2012) が構築したモデルは、コンピュータビジョンにおけるニューラルネットワークの優位性の始まりとなった。この 3 年間でエラー率はさらに低下し、視覚物体分類の分野では人間の性能とほぼ同じになった。また、音声認識 (Sak et al. 2014) や機械翻訳 (Sutskever et al.) でも同様の成績を収めている。

人工知能は、脳から直接インスピレーションを受けたシステムが実用化される時代に入った。今こそ、この脳にインスパイアされたテクノロジーを、脳に戻すべき時が来ている。我々は現在、ニューラルネットワーク理論と実証的なシステム神経科学を統合し、実世界の複雑な課題に取り組み、生物学的に妥当な計算メカニズムを使用し、神経生理学的および行動学的数据を予測するモデルを構築することができる立場にある。

理論と工学の発展は、かつてないほどのスピードで進んでいる。工学的に得られた知見の多くは、脳の理論にも関連すると思われる。モデルと脳の間で神経集団コードの内部表現を比較する最近の手法により、脳の情報処理理論としてのニューラルネットモデルを検証することができる (Dumoulin & Wandell 2008, Kay et al.; Kriegeskorte, 2011; Kriegeskorte & Kievit 2013; Kriegeskorte et al.; 2008a,b; Mitchell et al. 2008; Nili et al. 2014)。

本稿では、視覚や脳の科学者に向けて、ニューラルネットワークを紹介する。ニューラルネットワークは、このモデル化の枠組みを工学的に発展させたものであり、脳のデータを説明するためにこのようなモデルを使用した最初のいくつかの研究をレビューする。ここでは、計算論的神経科学を高レベルの皮質表現や複雑な実世界の課題に適用するための新しいフレームワークを紹介する。

次節では「A Primer on Neural Networks」と題して、学習アルゴリズムや普遍的な表現能力など、ニューラルネットワークモデルの基本を紹介している。「Feedforward Neural Networks for Visual Object Recognition」節では、現在コンピュータビジョンで主流となっている特定の大規模な物体認識ネットワークについて説明し、これらのネットワークが生物の視覚システムと何を共有し、何を共有していないかを議論している。「Early Studies Testing Deep Neural Nets as Models of Biological Brain Representations」と題した節では、人工的なニューラルネットワークと生物学的な脳の間の内部表現を経験的に比較した最初のいくつかの研究をレビューしている。「Recurrent Neural Networks for Vision」と題した節では、リカレント計算を用いたネットワークについて説明している。リカレント計算は、生物の脳の本質的な構成要素であり、入力画像の形成に関する生成モデルの推論を実現する可能性があり、計算論的神経科学の主要なフロンティアとなっている。最後に、「結論」節では、複雑な生物学的脳情報処理の経験的に正当化されたモデルに向けて、重要な議論、今後の課題、および今後の方向性について考察している。

4 ユニットは入力の重み付けされた和を計算し、活性化する

非線形関数に従うモデルニューロンをユニットと呼ぶのは、生物学的な現実と抽象度の高いモデルとを区別するためである。最も単純なモデルユニットは、入力の線形結合を出力する線形ユニットである (図1a)。このようなユニットを組み合わせてネットワークを構成しても、入力の線形結合を超えることはできない。図2b は、中間層の線形ユニットの活性化を線形に組み合わせた出力ユニットが、ランプ関数を加算し、ランプ関数を計算している様子を示している。線形ユニットの多層ネットワークは、重み行列 W が多層ネットワークの重み行列 W_i の積である単層ネットワークと等価である。非線形ユニットは、その出力がビルディングブロック (図2c) を提供し、その 1 レベル上の線

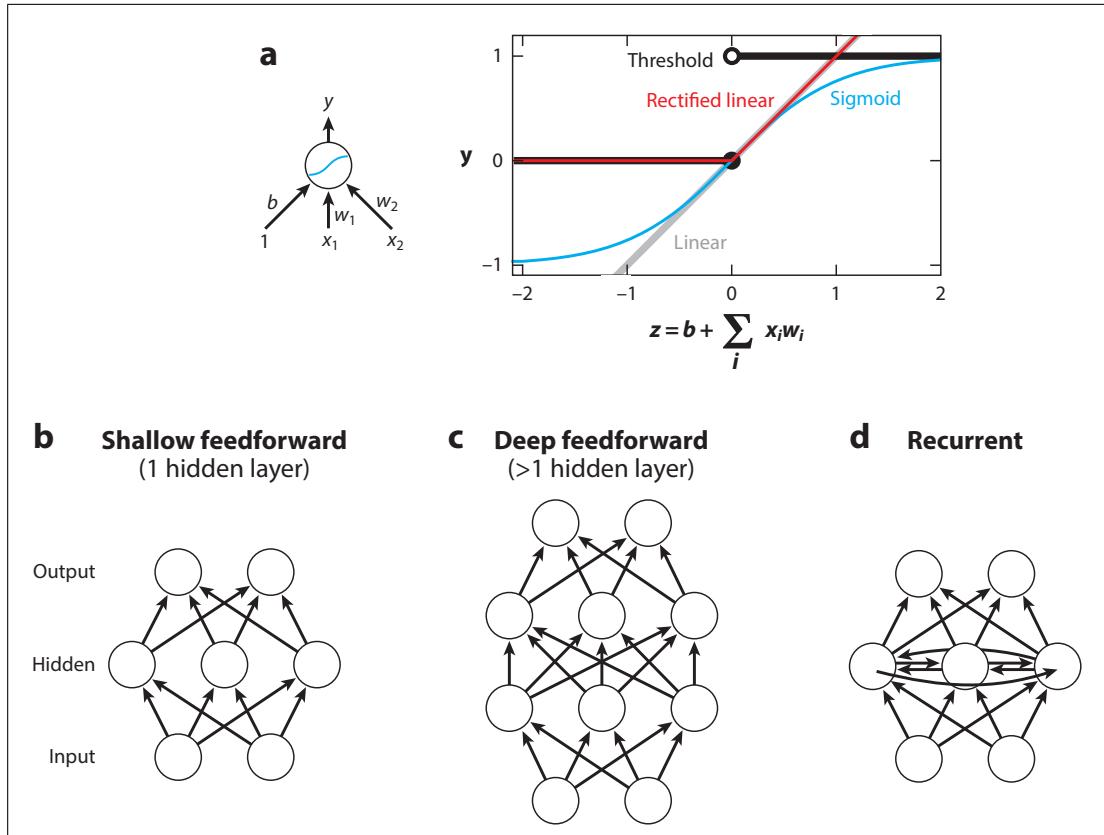


図2 人工ニューラルネットワーク: 基本ユニットとアーキテクチャ。

(a) 典型的なモデルユニット(図左)は、重み w_i とバイアス b を用いて、入力 x_i の線形結合 z を計算。ユニットの出力 y は z の関数であり、活性化関数と呼ばれる(図右)。一般的な活性化関数には線形(灰)、閾値(黒)、シグモイド(ここでは双曲接線、青)、整流線形(赤)などがある。ネットワークは、有向結合がサイクルを形成していない場合はフィードフォワード(b,c)、サイクルを形成している場合はリカレント(d)と呼ばれる。浅いフィードフォワードネットワーク(b)は、隠れ層が 0 または 1 である。隠れユニットの非線形活性化関数により浅いフィードフォワードネットワークはあらゆる連続関数を近似することができる(精度は隠れユニットの数に依存する)。深いフィードフォワードネットワーク(c)は複数の隠れ層を持つ。リカレントネットワークは、継続的なダイナミクスを生成し、入力の時間系列処理に適しており(十分な数のユニットがあれば)あらゆるダイナミカルシステムを近似することができる。

形結合によって、次のセクションで説明するように、入力から出力への任意のマッピングを近似するため、必要不可欠である。

ニューラルネットワークのユニットは、その入力重み w を使って、入力活性の加重和 z を計算し、その結果を(通常は単調な)非線形関数 f にして活性化 y を生成する(図1a)。初期モデルでは、非線形性は単純にステップ関数(McCulloch & Pitts 1943, Rosenblatt 1958, Minsky & Papert 1972)であり、各ユニットは二値の閾値を課す線形識別器となっていた。パーセプトロンの学習アルゴリズムでは、1つの閾値ユニットに対して、できるだけ多くの学習入出力ペアが正しくなるように、重みを反復的に調整する方法(ゼロまたはランダムな重みから始める)が用意されていた。しかし、ハードしきい値では、入力パターンと目的の出力パターンの組み合わせに対して、重みを少し変えただけでは出力に違いが出ないことがよくあります。これでは、多層ネットワークの重みを勾配降下法で学習することが難しくなる。勾配降下法では、重みを少しづつ調整して誤差を繰り返し減らしていく。ハードな閾値をシグモイド関数のような連続的に変化するソフトな閾値に置き換えれば、勾配降下法で学習することができる。

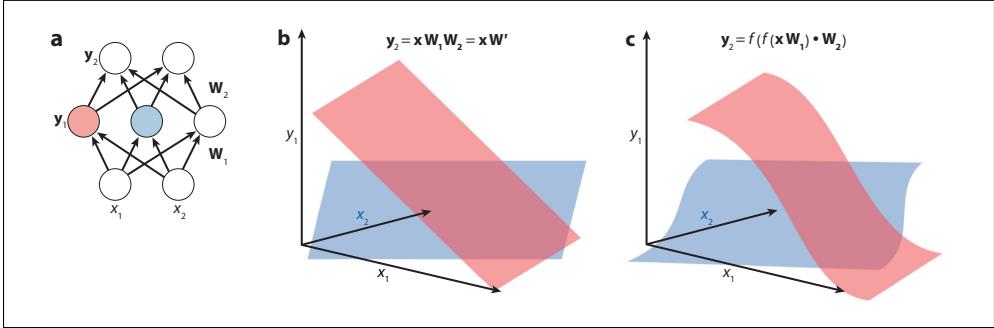


図3 非線形の隠れユニットを持つネットワークは、任意の非線形関数を近似することができる

- (a) 単一の隠れ層を持つフィードフォワードニューラル・ネットワーク
- (b) 隠れユニットが線形活性化関数を持つ場合、入力パターン (x_1, x_2) の関数としてのピンクと青の隠れユニットの活性化
各出力ユニット (y_2) は隠れユニットのランプ状（すなわち線形）の活性化の重み付けされた組み合わせを計算します。そのため、出力は入力パターンの線形結合のままとなる。
このようにして得られたネットワークは、入力と出力の間に隠れ層が介在しない線形ネットワークと同等であるため、線形の隠れ層は役に立たない。
- (c) ピンクとブルーの隠れユニットがシグモイド活性化関数を持つ場合の活性化。任意の連続関数は、十分な数の非線形隠れユニットの出力 (y_1) の重み付けされた組み合わせによって、出力ユニット (y_2) で近似することができる。

4.1 非線形隠れ単位を持つネットワークは普遍的な関数近似器である

非線形活性化関数の特定の形状は、表現可能な入力-出力写像のクラスには関係がない。入力層と出力層の間に少なくとも 1 層の隠れユニットを介在させたフィードフォワードネットワークは、普遍的な関数近似器である。十分な数の隠れユニットがあれば、ネットワークは入力のあらゆる関数を出力ユニットで近似することができる。十分な数の隠れユニットを追加し、重みを適切に設定することで、連続関数を任意の精度で近似することができる (Schäfer & Zimmermann, 2007; Hornik, 1991; Cybenko, 1989)。図2c は 2 次元の入力に対するこのプロセスを示している。十分な数のシグモイドランプ（任意の方向、傾き、位置を持つことができる）を追加することで、入力の任意の連続関数を近似することができる。

入力のシグモイド（または任意のステップ状の関数）を組み合わせることで、ネットワークが任意の関数を近似することができる理由を直感的に理解するために、入力空間の特定の領域でのみ（1 に近い）プラトーが重なるシグモイドを計算するように、一連のユニットの重みを設定したとする。これらのユニットの出力を高いしきい値を持つユニットにまとめると、そのユニットは、入力空間の特定の領域にいることを（1 に近い出力で）示すことができる。このようにして、異なる出力を必要とする入力空間内のすべての領域について指標を構築すれば、あらゆる入力をあらゆる必要な出力に近似的に対応させることができる。この近似マッピングの精度は、より多くのユニットを使用して、より多くの独立した領域を指標で定義することにより、常に向上させることができます。活性化関数が連続的であれば（通常はそうである）、ネットワークが表す関数も連続的であることに注意。ネットワークは 2 つの隠れ層を使って、基本的にトレーニング用の入力と出力のペアのルックアップテーブルを表現する。しかし、このネットワークは、補間や外挿によって新しい入力を扱うことができるという優れた特徴を持つ）。引用されているフィードフォワードネットの普遍性に関する理論的な結果は、この直感的な説明を超えて、どんな関数でも近似するために必要な隠れ層は 1 つだけであり、活性化関数はステップ関数に似ている必要はないことを示している。

シンプルで強力なニューラルネットワークのアーキテクチャとして、フィードフォワードネットワークがある（図1b, c）。フィードフォワードネットワークは、ユニットの層が連続しており、各ユニットはその出力を上位の層の

ユニットにのみ送信する。このため、フィードフォワードネットワークのユニットと接続は、それぞれ有向非環状グラフのノードとエッジに相当する。コンピュータビジョンシステムでは、ユニットは直前の層からの入力しか受け取らないことが多い。また、位の層の入力は、視覚的な階層構造からヒントを得て、局所的な受容野に制限されていることが多い。

最近のモデルでは、シグモイド（ロジスティックや双曲タンジェントなど）やReLU（整流された線形関数）など、さまざまな非線形活性化関数が使われている（図1a）。整形線形ユニットは、計算した線形結合が正であれば「0」を、そうでなければ「1」を出力する。整形線形ユニットは、重みの勾配降下学習を単純化して、より迅速な学習を可能にしており、コンピュータビジョンやその他の領域で非常に有効であることが実証されている。

5 なぜディープ（深層）

フィードフォワードネットワークは、複数の隠れ層を持つ場合に「深い」と言われる。この技術的な定義にもかかわらず、ディープという言葉は段階的な意味でも使われている。つまり、深いネットとは、多くの層を持つネットワークのことであり、あるネットワークが他のネットワークよりも深いこともある。深層学習とは、多くの隠れた層を持つアーキテクチャを用いて、視覚をはじめとする難しい問題に取り組む戦略のことである。

なぜ深さが役立つか？上述したように、非線形の隠れユニットが1層の浅いネットワークでも、普遍的な関数近似を行うことができる。浅いネットワークはサポートベクターマシンと密接な関係にあり、サポートベクターマシンも同様に任意の非線形関数を学習することができ、ニューラルネットワークよりも効率的に学習することができ、機械学習のツールとして大きな成功を収めている。深さが重要な理由は、深層ネットは、浅層ネットやサポートベクターマシンよりも、多くの複雑な関数をより簡潔に（すなわち、より少ないユニットと重みで）表現できるからである（Bengio 2009）。Why does depth help？We saw above that even shallow networks with a single layer of nonlinear hidden units are universal function approximators. Shallow networks are closely related to support vector machines, which can likewise learn arbitrary nonlinear functions, can be more efficiently trained than neural networks, and have been very successful tools of machine learning. The reason depth matters is that deep nets can represent many complex functions more concisely (i.e., with fewer units and weights) than shallow nets and support vector machines (Bengio 2009).

ある関数を計算する浅いネットワーク（隠れ層が1つのネットワーク）を考える。单一の隠れ層のユニットを、新しいネットワークの複数の隠れ層に分散させることで、同じ数のユニットを持つより深いネットワークを作ることができる。深いネットワークは、入力から隠れたユニットへの接続性と、隠れたユニットから出力への接続性が同じである。そのため、浅いネットワークが計算できるあらゆる関数を計算することができる。しかし、その逆はない。深層ネットワークでは、任意の層から上位の層に向けて、ゼロではない重みを追加することができるため、以前の計算結果を再利用することができ、深層ネットワークの表現力を拡張することができる。深層ネットワークが計算する可能性のある特定の機能については、浅層ネットワークの方がはるかに多くのユニット数を必要とすることが示されている（Bengio 2009）。

現代のコンピュータ・ハードウェアとの類似性を考えることは有益である。ファン・ノイマン・アーキテクチャは、基本的に逐次的な計算モデルであり、以前の計算結果を再利用することができる。多くの計算が独立して行われるような特殊なケース（例えば、グラフィックスやビジョンにおける画像の位置を超えた計算など）では、並列ハードウェアによって処理を高速化することができる。しかし、独立した計算は並列にも逐次にも実行できるのに対し、依存性のある計算は逐次にしか実行できない。そのため、以前の結果を再利用することで、（ユニットの総数が固定されている場合）計算可能な関数の数を増やすことができる。

本来、浅いネットワークは、ルックアップテーブルのように対象となる関数をつなぎ合わせるために、普遍的な関数近似器となる。しかし、多くの関数は、より深いネットワークを用いて、冗長性を利用したり、対象と

なる関数の固有の構造を利用したりすることで、より簡潔に表現することができる。しかし、AIにおける深層学習の実用的な成功は、視覚を含む多くの実世界の問題が、深層アーキテクチャを用いてより効率的に解決できる可能性を示唆している。興味深いことに、靈長類の脳の視覚階層も深層アーキテクチャである。

5.1 リカレントニューラルネットワークは力学系の普遍的な近似器である

フィードフォワードネットワークは静的な関数を計算する。より興味深いダイナミクスを持つアーキテクチャは、ユニットがサイクル状に接続されるリカレントネットワークである。このようなアーキテクチャは、横方向の接続やフィードバック接続が普遍的に存在する生物学的なニューロンネットワークに似ている。リカレントネットワークでは、すべての隠れユニットが他のすべての隠れたユニットと相互作用できるので、別々の隠れた層という概念は意味を持たない。そのため、リカレントネットワークは、入力ユニットと出力ユニットが別々に配置された、相互に接続された1つの隠れユニットのセットとして描かれることが多い(図1d)。レイヤードアーキテクチャは、リカレントネットワークの特殊なケースで、特定の接続が欠けている(すなわち重みが0に固定されている)。

視覚神経科学では、視覚階層の理論的概念は、接続をフィードフォワード接続、側方接続、フィードバック接続に分け、接続の起点と終点の皮質層によって識別することに基づいている。また、あるニューロンは入力から多くのシナプスで隔てられており、より複雑な視覚的特徴を表す傾向があるという事実に基づいている。これらの基準は、靈長類の視覚システムの階層を定義するような完全に明確なランクの割り当てをサポートしないかもしれないが(Hilgetag et al. 2000)，階層モデルは有用な単純化であり続けている。

フィードフォワードネットワークが入力を出力にマッピングする静的な関数を計算するのに対し、リカレントネットワークはダイナミクスを生み出す。リカレントネットワークは、入力パターンの時間的変化に影響されることができる状態の時間的変化を作り出す。リカレントネットワークの内部状態には記憶があり、最近の刺激の履歴を表現したり、時間的なパターンを検出したりすることができる。フィードフォワードネットワークが普遍的な関数近似であるのに対し、リカレントネットワークは動的システムの普遍的な近似である(Schäfer & Zimmermann 2007)。様々なモデルがシミュレーションや解析によって検討されている。

エコーステートネットワーク(Jaeger 2001, スパイキングダイナミクスを用いた同様のモデルについては Maass et al. 2002も参照)では、例えば一連の入力パターンは疎かつランダムに接続された隠れユニットのセットに供給される。各入力パターンに関連した活動の波は、後続の入力パターンの影響に支配されるようになるまで、しばらくの間、隠れユニットの間で反響する。池の表面の同心円状の波が、その中心にある過去の出来事を推測させるように、隠れユニットの活動は最近の刺激の履歴に関する情報を符号化する。反響状態ネットワークでは、隠れユニット間の重みは訓練されていない(ただし、ランダムに設定されているため、記憶がすぐに消えてしまわないように注意が必要である)。教師あり学習は、入力の時間的パターンを検出する読み出しユニットのセットを学習するために使用される。

反響状態ネットワークは、そのリカレントダイナミクスを隠れユニット間のランダムな重みに依存している。また、リカレントネットワークのダイナミクスを監視によって明示的に学習し、時間的パターンの生成、分類、予測に最適化することもできる(Graves & Schmidhuber 2009, Sutskever et al. 2014)。

5.2 バックプロパゲーションアルゴリズムを用いた勾配降下法による表現の学習方法

普遍性定理は、十分な数のユニットを持つニューラルネットワークの表現力を保証するものである。しかし、これらの定理は、特定の関数をフィードフォワードネットで表現したり、特定の力学系をリカレントネットで表現したりするために、接続の重みをどのように設定すればよいかを教えてくれない。学習は、高次元で難しい最適化問題である。現実の問題を解決できるモデルは、膨大な数のユニットと、さらに膨大な数の重みを持っている。この非凸問題には、大域的な最適化手法は使えない。重み設定の空間は非常に広大なので、単純な探索アルゴリズム(例えば進化的アルゴリズム)では、可能性のごく一部しかカバーできず、おもちゃの問題に限定された小さなモデルを除いて、

通常は実用的な解を得ることはできない。

重み空間は高次元であるため、大域的な最適化は困難である。しかし、この空間には多くの等価な解が含まれている（例えば、2つのニューロン間のすべての入出力を交換することを考えてみよ）。さらに、全誤差（実際の出力と希望の出力の間の二乗偏差の合計）は、重みの局所的に滑らかな関数である。現在選択されている学習方法は、勾配降下法であり、重みを少しづつ調整して誤差を反復的に減らしていく。

勾配降下法による学習の基本的な考え方は、重みをランダムに初期化した状態で開始し、各重みをわずかに変更することで誤差がどの程度減少するかを判断することである。そして、誤差への影響に比例して重みを調整する。この方法では、重み空間において、誤差が最も急峻に下降する方向に確実に移動する。

勾配とは、つまり、重みを調整したときに誤差がどの程度変化するかを示すもので、重みに対する誤差の微分である。これらの導関数は、フィードフォワードネットワークの出力層に接続する重みについて、簡単に計算することができる。前の層を駆動する接続では、誤差の微分を計算する効率的な方法は、ネットワークを介して後方に伝搬させることである。これにより、この方法はバックプロパゲーションと呼ばれている（Werbos 1981, Rumelhart et al. 1986）。

勾配降下法は、重み空間の局所的な近傍のみを見ており、大域的に最適な解を見つけることは保証されていない。しかし、それにもかかわらず、実際には非常によく機能する解を見つけることができる。重み空間の高次元化は、大域的な最適化を困難にするという呪いである。しかし、局所的な最適化で良い解が得られるという点では、幸いなことである。勾配降下法では、移動できる方向が非常に多いため、すべての方向で誤差が増加し、それ以上の進歩が見られないようなローカルミニマムに陥ることはない。

興味深いことに、リカレントネットワークの学習にも同じ手法を用いることができる。誤差導関数は、時間をかけてバックプロパゲーションにより計算され、このプロセスは複数のサイクルで逆にループを満たすことになる。この仕組みを理解するために、リカレントネットワークを、リカレントネットワークのすべてのユニットを時間の次元に沿って（十分に大きな数のタイムステップで）複製して得られるフィードフォワードネットワークと考えることができる。リカレント計算の各時点は、フィードフォワードネットワークの層に対応し、各層は同じ重み行列（リカレントネットワークの重み行列）によって次の層に接続されている。リカレントネットワークは、時間をバックプロパゲーションすることにより、そのダイナミクスに短期記憶を保存することができる重みを学習することができ、時間的に分離されたイベントを必要に応じて関連づけ、時間的シーケンスの望ましい分類または予測を達成することができる。しかし、ネットワークが長期的な依存性を利用する方法を学習するために、誤差導関数を時間的に十分に後方に伝播させることは、勾配が消滅または爆発する傾向があるという問題によって妨げられる（Hochreiter 1991, Hochreiter et al. 2001）。この問題は、与えられた重みの誤差導関数が、バックプロパゲーションの経路で遭遇する活性化関数の重みと導関数に対応する複数の項の積であるために起こる。この問題を解決する一つの方法として、長短期記憶（LSTM）アーキテクチャー（Hochreiter, 1991, Hochreiter et al. 2001）がある。このユニットを介してバックプロパゲーションされた誤差導関数は安定しており、バックプロパゲーションによる長期依存性の学習が可能である。このようなネットワークは、驚くべきことに、逐次的な予測、分類、制御などのタスクにおいて、何時間も後に関連する情報を記憶することを学ぶことができる。バックプロパゲーションは、重みを調整して、課題を実行するために必要な情報を選択的に（活性化状態で）記憶するダイナミクスを植え付ける。

バックプロパゲーションを用いたディープフィードフォワードネットのトレーニングでは、グラデーションの消失や爆発も問題となる。この問題については、非線形活性化関数の選択によって違いが生じる。さらに、バックプロパゲーションによる教師付き学習をうまく機能させるためには、勾配降下アルゴリズム、正則化、および重みの初期化の詳細が重要になる。

5.3 教師なしの手法でも表現を学習できる

教師付き学習では、学習データは、入力パターンとそれに関連する望ましい出力の両方で構成されている。しかし、現実の世界では、このような明示的な監視信号を利用できないことが多い。生物は、一般的に、教師信号を受けることができない。工学の分野でも、ラベル付けされていない多数の入力パターンと、ラベル付けされた少数の入力パターン（ウェブ上の画像など）が存在することがある。教師なし学習では、ネットワークがラベルを必要とせず、自然な入力パターンに最適化された表現を学習し、さまざまな課題に役立つ可能性がある。例えば、自然画像は、すべての可能な画像の非常に小さなサブセットを形成するため、教師なし学習によって圧縮された表現を見つけることができる。

教師なし学習の例として、自己符号化器が挙げられる (Hinton & Salakhutdinov, 2006)。自己符号化器は、入力よりも少ないユニットを持つ中央コード層を持つフィードフォワードニューラルネットワークである。このネットワークはバックプロパゲーションによって学習され、入力層と同じ数のユニットを持つ出力層で入力を再構成する。学習アルゴリズムはバックプロパゲーションで、教師信号を使用する。だが、この手法は、個別の教師信号（つまりラベル）を必要とせず、入力パターンのセットのみを必要とするため、教師なしである。もし、コード層を含むすべての層が入力と同じ次元を持っていたら、ネットワークは入力を各層に通すだけでよい。しかし、コード層はユニット数が少ないと、情報のボトルネックになってしまう。入力を再構成するためには、ネットワークはその小さなコード層に入力に関する十分な情報を保持するように学習しなければならない。そこで自己符号化器は、入力領域の統計的構造を利用して、コード層に圧縮表現を学習する。この表現は、学習に用いられる入力パターンの分布に合わせて特化される。

入力から符号層までの層を符号化器と呼び、符号層から出力までの層を復号化器と呼ぶ。符号化器と復号化器が線形であれば、ネットワークは最初の k 個の主成分で構成される線形部分空間を学習する（符号層が k 個の場合）。非線形ニューラルネットワークをエンコーダーおよびデコーダーとして使用すると、非線形に圧縮された表現を学習することができる。非線形符号は、入力パターンの自然な分布が線形部分空間でうまく表現されない場合に、より効率的になる。自然界の画像がその例である。

純粋な教師付き学習ではラベル付きの学習データが十分に得られない場合、教師なし学習はフィードフォワードネットワークの事前学習に役立つ。例えば、視覚認識のためのネットワークは、ラベルのない大量の画像を使って、オートエンコーダーの枠組みで層ごとに事前学習することができる。ネットワークが自然な画像の妥当な表現を学習した後、バックプロパゲーションを用いて、より簡単に正しい画像ラベルを予測するように学習することができる。

6 視覚的物体認識のためのフィードフォワードニューラルネットワーク

コンピュータビジョンでは、近年、ある種の深層ニューラルネットワーク（深層フィードフォワード畳み込みネットワーク）が主流になっている。これらのネットワークは、サポートベクターマシンなどの浅い機械学習分類器への入力として、手作業で視覚的特徴 (Lowe, 1999 など) を作成していた以前の技術を凌駕している。興味深いことに、初期のシステムの中には、手で作成した特微量と教師付き分類器の間に、教師なし学習によって得られた中間表現を挿入したものがあった。この表現を挿入することで、より深いアーキテクチャの必要性に対応できたのかもしれない。

今日、コンピュータビジョンで広く使用されている深層畳み込みネットワークには、いくつかのアーキテクチャ上の特徴があり、そのいくつかは生物の視覚システムからゆるやかに着想を得ている (Hubel & Wiesel 1968)。

深い階層: 灵長類の腹側視覚野と同様、これらのネットワークは、深い階層（通常5～20層）の表現を介して情報を処理し、空間的なレイアウトが画像と一致する視覚表現から、物体のカテゴリーを認識できる意味的な表現へ

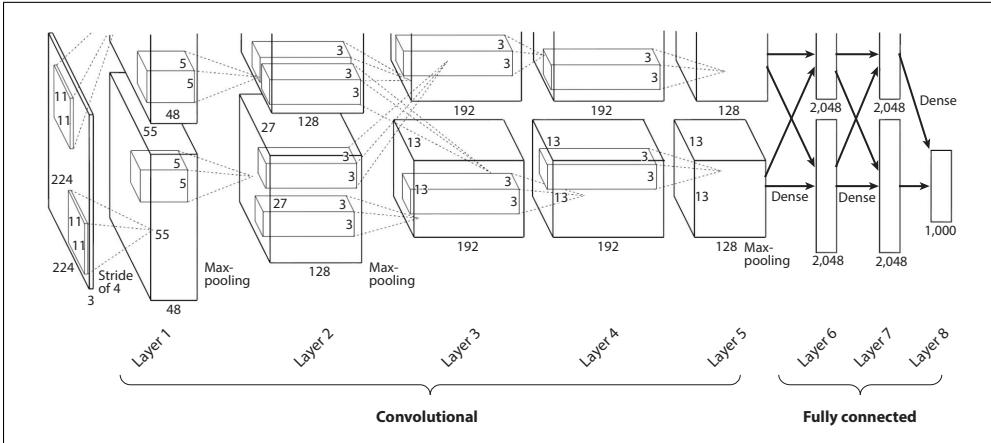


図4 物体認識のための深層畠み込みフィードフォワードアーキテクチャ。図は Krizhevskyら (2012) が使用したアーキテクチャを示している。情報は、入力画素画像（左）(224×224 画素, 3 色チャンネル) から 7 つの隠れ層を経て、カテゴリ出力（右）(1,000 個のカテゴリー検出ユニット) へと至る。大きな箱は特微量地図のスタックを表している。例えば、層 2 の場合、下の大きな箱は、27（横方向の画像位置）× 27（縦方向の画像位置）のサイズの 128 個の特微量地図を表している。なお、箱の大きさは縮尺がない。小さな箱は、ある層の表現と畠み込まれる特徴テンプレートを表している。畠み込みと最大プーリングは 1 ピクセル以上のストライドで行われるので、特微量地図の空間的な広がりは、表現系列 (224, 55, 27, 13, 13, 1, 1, 1) に沿って減少する。上と下の大きな箱、2 つの GPU の役割分担を表している。

と徐々に変化させていく。

畠み込み: 下位層には、小さな受容野 (RF) を持つ局所的な視覚特徴検出器がある。各検出器は、2 次元の画像全体に複製され、特徴マップを形成する。これは、各特徴パターンで画像を畠み込み、その後に静的な非線形性を加えたものである。畠み込みのアーキテクチャは、ある位置で有用な特徴は、別の位置でも有用である可能性が高いという洞察に基づく。このアーキテクチャは、靈長類の初期視覚野に似ており、靈長類視覚野もまた、多くの視野位置で質的に類似した局所的な視覚的特徴を検出する（ただし、特徴の特性と RF 空間分布は、畠み込みネットワークのように完全に均一ではない）。ユニットの RF は、階層に沿って大きさを増していく。局所領域への接続の制限と、空間位置間での接続重みの複製（与えられた特徴に対して全ての位置で同じ重みパターン）により、学習が必要なパラメータの数を大幅に減らすことができる (LeCun et al., 1989)。

局所プーリングとサブサンプリング: 畠み込段階の間に、局所プーリング段階が挿入される。プーリングでは、局所的なユニットのセットの出力を、最大値または平均値をとることで結合する。これにより、特徴の空間的なずれに対する局所的な耐性が付与され、小さな画像のズレや画像特徴の構成の小さな歪みに対してロバストな表現が可能になる(Fukushima 1980)。最大値プーリングは、HMAX ((Riesenhuber & Poggio) のような神経科学的な視覚モデルでも、局所的な耐性を実現するために用いられている。プーリングは、空間的な位置のサブサンプリングと組み合わされることが多い。表現される空間的な位置の数が減ることで、各位置で計算される特徴の数を階層的に増やすためのリソースが確保される。

最上層では、ユニットはグローバル RF を持ち、前の層のすべてのユニットからの入力を受け取る。最終層には、カテゴリーごとに 1 つのユニットが含まれ、ソフトマックス（正規化された指数関数）関数が実装されている。この関数は、非常に高い応答を除いてすべての応答を強く減少させ、出力の合計が 1 になるようにする。この出力は、クロスエントロピー誤差を最小化するように学習手順が設定されている場合、カテゴリーに対する確率分布として解釈することができる。

このネットワークは、バックプロパゲーションを用いて、入力画像のカテゴリーを認識するように学習することができる (LeCun et al. & Bengio, 1995)。自然の画像を分類するようにネットワークを学習させると、学習過程で、

生物の視覚システムに見られる特徴と質的に類似した特徴が発見される(図4)。初期の層では、V1ニューロンの特徴に似た、ガボールのような特徴が現れる。同様の特徴は、スパース表現学習などの教師なしの手法でも発見されており(Olshausen & Field, 1997),スパース表現やカテゴリー化を目的とする場合でも、視覚の出発点として適していることが示唆されている。続いて、カーブセグメントなどのやや複雑な特徴を選択するユニットがある。さらに上位層には、物体の一部や、人間や動物の顔や体、車や建物などの無生物など、物体全体に選択的なユニットがある。

何が自動的に学習されたかを理解するために、この分野では、ディープネットワーク内のユニットのRFや選択性を可視化する方法が考案され始めている(Zeiler & Fergus, 2014; Girshick et al. 2014, Simonyan et al. 2014, Tsai & Cox 2015, Zhou et al. 2015)。図4はそのような視覚化を示しており、ユニットは階層に沿って視覚的な複雑さを増していく自然な画像特徴に対する選択性を学習するという考えを支持している。しかし、このような視覚化には2つの重要な注意点がある。第一に、階層間で複数の非線形変換が行われるため、画像テンプレートではユニットを正確に特徴づけることができません。もし、テンプレートマッチングによって高レベルの反応が計算できるのであれば、視覚には深い階層は必要ないだろう。視覚化されたものは、特定の画像の文脈で何が反応を促進するかを示すだけである。あるユニットの選択性を知るために、そのユニットを駆動する多くの画像を検討する必要がある(より多くのユニットのそれぞれに対する複数のテンプレートについてはZeiler & Fergus 2014を参照)。第二に、図4で視覚化されたユニットは、解釈可能な選択性の理論的なバイアスを確認するために選択されている。これらのユニットは例外的なものであり、ネットワークの機能に不可欠なものであるかどうかは不明である。例えば、意味のある選択性は、単一のユニットではなく、ユニットの直線的な組み合わせに存在し、弱い分散活動が本質的な情報をコード化している可能性がある。

表現の階層は、空間ベースの視覚的な表現から、形状ベースの意味的な表現へと徐々に変化していくように見える。ネットワークは、各カテゴリーに関連する形状の種類に関する複雑な知識を獲得する。ここでいう「形」とは、輝度や色で定義された様々なレベルの複雑な特徴のことである。高レベルのユニットは、顔、人体、動物、自然の風景、建物、車など、自然の画像に含まれる形の表現を学習するようだ。学習される選択性は、出力層で検出されたカテゴリに限らず、これらの物体の一部や、文脈の要素に対する選択性も含まれている可能性がある。例えばKrizhevskyら(2012)のネットワークには、テキスト(Yosinski et al. 2015)や顔に選択性があると思われるユニットが含まれているが、テキストや顔は学習したカテゴリーには含まれていなかった。おそらく、それらの反応は、出力層で表現されるカテゴリーを検出するのに役立つと思われる。なぜなら、それらは検出されるべきカテゴリーと統計的に関連しているからである。例えば、部分選択性の特徴は、物体全体を検出するための足がかりとなる可能性がある(Jozwik et al. 2015)。ユニットを、例えば目や顔の検出器のように、口頭で機能的に解釈することは、私たちの直感的な理解を助け、重要な何かを捉えることができるかもしれない。しかし、このような言葉による解釈は、カテゴリー性や局在性の度合いを過大評価し、これらの表現の統計的・分散的性質を過小評価する可能性がある。

コンピュータビジョン用の深層畳み込みニューラルネットワークの有力な例としてKrizhevskyら(2012)が構築したシステムがある。このシステムのアーキテクチャ(図3)は、5つの畳み込み層と3つの完全連結層で構成されている。著者らは、畳み込み層の数を減らすと性能が低下することを発見し、深層アーキテクチャの必要性を示した。このシステムでは、整流線形ユニット(ReLU)、最大プーリング、局所的正規化が用いられている。ネットワークはバックプロパゲーションによって学習され、入力画像に表示されている1,000種類の物体カテゴリのうち、どれが表示されているかを認識できるようになっている。学習セットはImageNetセットから得た120万枚のカテゴリラベル付き画像で構成されている。このセットは、画像の翻訳版と水平反射版を追加することで、2,048倍に拡張された。学習は、結果として得られた画像セットを90回繰り返すことで行われた。

これは、各訓練試行において、各ユニットを0.5の確率で“ドロップ”(計算から除外)する手法である(Hinton et al. 201)。この手法では、各試行において、各ユニットを0.5の確率で”脱落”(計算から除外)させ、出力を計算するフォワードパスと重みを調整するバックプロパゲーションパスの両方で、約半数のユニットのランダムなセットを使用する。この方法により、学習中のユニットの複雑な共同適応を防ぎ、各ユニットが他のユニットの様々なチー

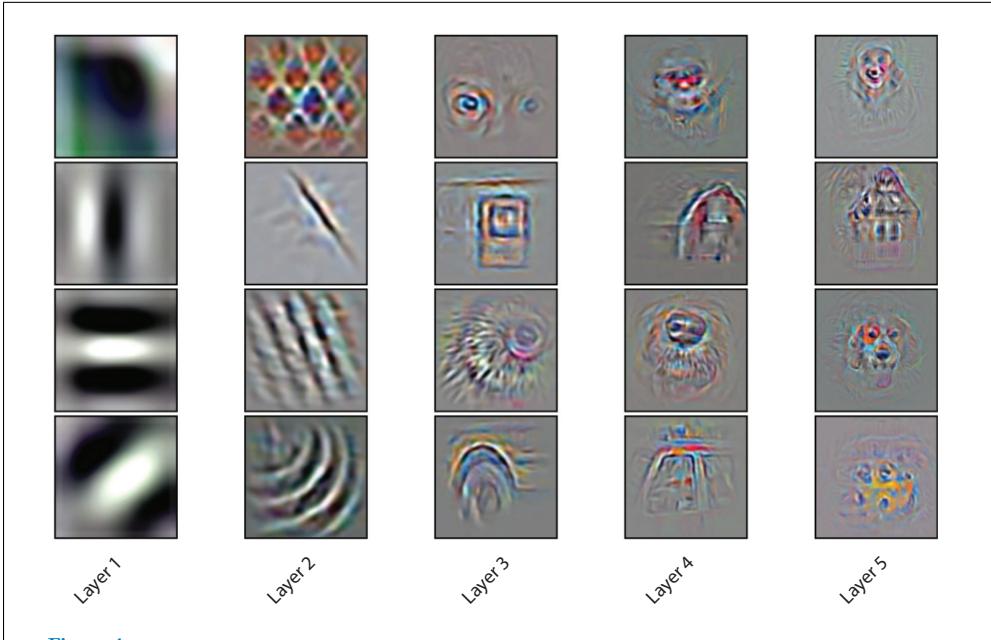


図5 教師付き深層学習は、神經生理学的知見と質的に一致する特徴選択性を生み出す。ディープニューラルネットワークにおける表現を理解するために、どの画像要素が深層ニューラルネットワークの特定のユニットを駆動するかを視覚化することができる。20個の例示ユニット(5層の各層から4個ずつ)について、そのユニットを強く駆動する特定の画像の文脈で何が反応を引き起こしたかを可視化した画像を示している。ここで使われている可視化技術には、ユニットを強く動かす入力画像の選択と、担当する画像要素を生成するためのフィードフォワード計算の反転という2つのステップがある。フィードフォワードパスに沿った畠み込みは、反転畠み込み(畠み込み行列の転置を使用)によって反転される。最大ブーリング演算は、フィードフォワードパスで最大活性であったブーリングユニットへの接続のアイデンティティ(単位行列)を格納することによって反転される。ネットワークの奥深くにあるユニットは、画像上で単純なテンプレートマッチング操作を行わないため、どのような視覚的テンプレートによっても完全には特徴づけられることに注意。しかし、あるユニットを駆動する多くの画像(図示せず)に対して上記の可視化を実行することで、その選択性や許容範囲を理解することができる。示されている反転畠み込み可視化技術は、Zeiler & Fergus (2014)によって開発された。ディープネットワークは Chatfield et al. (2014)による。The analysis was performed by Güclü & van Gerven (2015). Figure adapted with permission from Güclü & van Gerven (2015).

ムの中で有用な貢献をすることを可能にする。このネットワークには、合計65万個のユニットと6,000万個のパラメータがある。畠み込み層は、局所的な重みのテンプレートで定義されており、パラメータ全体の5%未満を占めている。95最初の2つの完全連結層には、何百万もの接続があるため、ドロップアウトを適用した。実験の結果、オーバーフィッティングを防ぐためには、ドロップアウトが必要であることがわかった。

この学習は、2つのGPU(Graphics Processing Unit)を搭載した1台のワークステーション上で6日間にわたって行われた。GPUは、計算を並列化して大幅に高速化する。このシステムは、コンピュータビジョンのコンテスト「ImageNet Large-Scale Visual Recognition Challenge 2012」画像セットでテストされた。その結果、2位のシステムに大差をつけて優勝し、コンピュータビジョンにおけるニューラルネットワークの優位性が確立された。その後、同様のアーキテクチャを用いたいくつかの畠み込みニューラルネットワークがさらに性能向上させている(例: Zeiler & Fergus 2014, Chatfield et al. 2014)。

コンピュータビジョンで使われている深層畠み込みニューラルネットワークは、カテゴリーレベルの認識など、限られた範囲の視覚処理を行うものである。しかし、取り組むべき視覚課題の範囲は急速に拡大しており、初期のコンピュータビジョンシステムと比較して、深層ネットワークは飛躍的な進歩を遂げている。深層畠み込みネットワークは、生物学的な視覚システムに近い形で設計されているわけではない。しかし、その本質的な機能メカニズムは、生

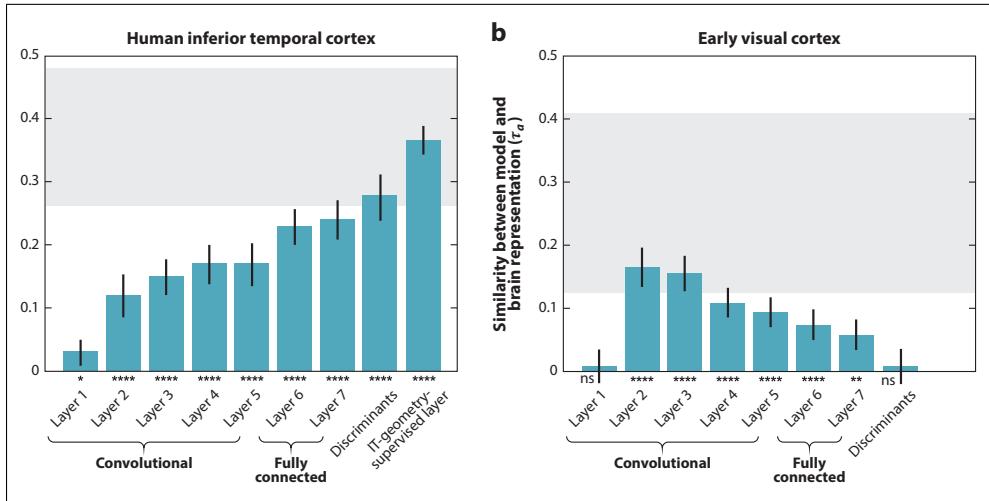


図6 ディープニューラルネットワークにより、物体画像の初期視覚表現と下側時間表現を説明。モデルと脳内の各表現は、実世界の物体の写真セットによって誘発される反応パターンの非類似性行列によって特徴づけられた。(a) Krizhevsky et al. (2012) の神経回路網の層を上るにつれて、表現はヒトの下側頭 (IT) 皮質の表現と単調に類似していく。最終的な表現段階で、線形カテゴリ判別を用いて、IT と同じ意味的次元を強調するように線形リミックスし(右から2番目のバー)，各層と各判別因子に重みを割り当てて、IT における異なる計算機能の抜卓率をモデル化すると(画像セットへのオーバーフィッティングを避けるためにクロスバリデーションを行った；右端のバー)，ノイズの上限(グレーの陰影領域)に達し、モデルがデータを完全に説明していることがわかる。従来のコンピュータビジョンの特徴量に、カテゴリー判別と重み付けを用いた同じ方法の線形結合を適用したところ(ここでは示していない)，この表現では IT データを説明できなかった。サルの IT についても同様の結果が得られた(ここでは示していない)。(b) ディープニューラルネットワークの下位層は、初期視覚野の中心窓合流部(V1-V3)の表現に似ている。アスタリスクはチャンス以上の精度であることを示す。ns, 有意差なし; , p < 0.05; , p < 0.01; , p < 0.001; , p < 0.0001。各モデルの表現と IT (縦軸) の類似性を Kendall の順位相関係数を用いて測定し、表現上の非類似性行列を比較した(被験者群平均をプロット)。結果は Khaligh-Razavi & Kriegeskorte (2014) より転載。

物の脳にヒントを得ており、生物のニューロンを使って実装することが可能である。この新しい技術は、現在の計算神経科学では実現できないような複雑な知能を発揮する、より生物学的に忠実な脳-計算モデルのためのエキサイティングなフレームワークとなる。

7 生物学的脳のモデルとしてのディープニューラルネットワークを検証する初期の研究

ディープ畠み込みニューラルネットワークを生物学的視覚のモデルとして評価し、モデルと脳の間で内部の表現空間と性能レベルの両方を比較する研究がいくつか始まっている。いくつかの研究で再現され、一般化された知見として、人間や靈長類の下側頭 (IT) 皮質 (Tanaka, 1996) の表現空間に近い表現空間を利用するモデルは、物体認識において優れた性能を発揮する傾向がある (Yamins et al. 2013, 2014; Khaligh-Razavi & Kriegeskorte 2013, 2014)。これは、コンピュータビジョンが生物の視覚から学ぶことができるという直感を裏付けるものである。逆に、生物学的視覚科学は、計算理論の候補を工学に求めることができる。

工学的な解決策が生物学的な解決策に近いということは、一般的にはない(飛行機、列車、自動車を考えてみよ)。特にコンピュータビジョンの分野では、ニューラルネットワークモデルを実世界の視覚に適用できなかった初期の段階では、より脳に近いソリューションを求めて無駄だという意識があった。しかし、最近のニューラルネットワークモデルの成功は、脳からヒントを得た視覚のためのアーキテクチャが非常に強力であることを示唆している。本節

では、コンピュータビジョンシステムの表現と脳の表現を経験的に比較し、ニューラルネットワークモデルが単にアーキテクチャ的に似ているだけではないことを示す。また、霊長類の腹側視覚経路の表現と非常によく似た表現を学習している。

コンピュータビジョンを成功させるために、表現が生物の脳と似ていなければならぬことを証明することは不可能である。しかし、ランダムな特徴を用いて自動的に生成されたニューラルネットワークアーキテクチャの大規模なセット (Yamins et al. 2013, 2014), 手作業で作成された一般的なコンピュータビジョンの特徴や神経科学的なビジョンモデル (Khaligh-Razavi & Kriegeskorte 2013, 2014), およびディープニューラルネットワークの層 (Khaligh-Razavi & Kriegeskorte 2014) において、物体認識の性能と IT との表現の類似性との間に関連性があることが示されている。少なくとも、これまでに検討されたアーキテクチャの中では、性能の最適化によって、IT に似た表現空間が実現されているように見える。

IT は、その表現においてカテゴリー的な区分を強調することが知られている (Kriegeskorte et al., 2008b)。線形読み出しによって実現されるカテゴリー化が得意なモデルは、同様にカテゴリー区分が強い傾向にある。このことは、IT との表現上の類似性が高いことを部分的に説明している。しかし、カテゴリー内の表現形式であっても、性能の良いモデルでは IT との類似性が高くなる傾向がある (Khaligh-Razavi & Kriegeskorte 2014)。

最も性能の高いモデルはディープニューラルネットワークであり、これらのネットワークは IT の表現幾何学を説明するのにも最適である (Khaligh-Razavi & Kriegeskorte 2014, Cadieu et al., 2014)。 Khaligh-Razavi & Kriegeskorte (2014) は、広範囲の古典的なコンピュータビジョンの特徴、VisNet ((Wallis & Rolls 1997, Tromans et al. 2011) や HMAX (Riesenhuber & Poggio 1999) などのいくつかの神経科学的に動機づけられたビジョンモデル、および Krizhevsky et al. (2012) が構築した深層ニューラルネットワークを検証した (図3)。彼らの研究における脳内表現は、ヒトの機能的磁気共鳴画像 (fMRI) とサルの細胞記録から推定されたものであった (サルのデータは Kiani et al. 2007, Kriegeskorte et al. 2008b より)。 Khaligh-Razavi & Kriegeskorte (2014) は、表象類似性分析を用いて、モデルと脳領域の間の内部表象空間を比較した (Kriegeskorte et al. 2008a)。この分析では、刺激の各対について、2つの刺激の表現上の非類似性を測定した。そして、全刺激対における表現上の非類似性ベクトルを、モデル表現と脳領域の間で比較した。

ディープニューラルネットワークの下位層では、初期視覚野に似た表現が見られた。ネットワークの層を越えて、表現形状は単調に初期視覚野には似ていなくなり、IT に似てきた。これらの結果を、ヒトのデータについて図 5 に示す。同様の結果は、サルの IT についても得られた (ここでは示されていない)。

最上位層では、IT データの説明可能な分散を完全には説明できなかった。しかし、IT に適合した表現 (訓練と検証のために独立した画像セットを用いて、ディープニューラルネットワークの特徴を線形に再混合・再重み付けしたもの) は、IT データを完全に説明した (Khaligh-Razavi & Kriegeskorte 2014)。この IT に適合したディープニューラルネットワーク表現は、従来のコンピュータビジョンの特徴量を同様に IT に適合させた組み合わせよりも、IT 表現を実質的かつ有意に説明した。

Cadieuら (2013, 2014) は、初期視覚のモデルである HMAX モデル (Riesenhuber & Poggio 1999, Serre et al. 2007), Yamins ら (2013, 2014) の階層的に最適化された多層モデル、Krizhevsky ら (2012) と Zeiler & Fergus (2014) のディープニューラルネットワークと並べて、IT 細胞集団の内部表現を分析した。物体分類で最も高い性能を示した表現は、Zeiler & Fergus (2014) が構築したディープニューラルネットワークと生物学的 IT 表現 (サルの神経細胞の記録) であり、Krizhevsky ら (2012) が提案したディープネットワークがそれに続いた。それ以外の表現は、かなり低いレベルであった。2つのディープネットワークは、独立した IT ニューロンのセットからのニューロン記録と同様に、IT データをよく説明した (図6b)。

さらにいくつかの研究でも同様の結果が得られており、異なる深さの表現が腹側ストリームの表現段階をどの程度説明できるのかが明らかになりつつある (Agrawal et al. 2014, Guc Gerven 2015)。全体として、ディープニューラルネットワークモデルと霊長類の腹側ストリームの間のこれらの初期の経験的な比較は、4つの結論を示唆してい

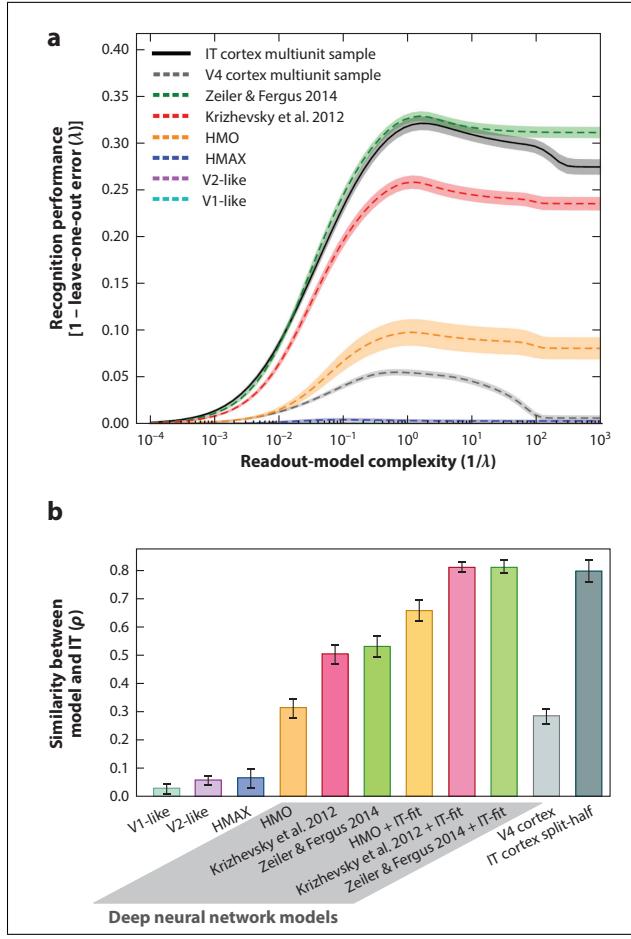


図7 深層ニューラルネットワークは、認識において、より単純な計算モデルを上回り、下側頭(IT)表現をよりよく説明する。 (a) 深層ニューラルネットワークの物体認識性能は、浅い計算モデルのそれを上回り、サルで記録されたITニューロンの集団のそれに匹敵する。認識性能(縦軸)は、読み出しモデルの複雑さ(横軸)の関数としてプロットされており、低い複雑さで高い性能を示しているのは、カテゴリーが表現空間において容易に分離可能な領域を占めていることを示している。 (b) ディープニューラルネットワークの表現は、より単純な3つのモデル(V1-like, V2-like, HMAX)よりも、ITに近いものとなっている。各モデルとIT(縦軸)の類似度は、スピアマンの順位相関係数(ρ)を用いて表現の非類似度行列を比較して測定した。結果は Cadieu et al. (2014) より。略語の説明 HMAX, hierarchical model and X (Riesenhuber & Poggio 1999, Serre et al. 2007, Tarr 1999); HMO, hierarchical modular optimization model (Yamins et al. 2014); IT, 下側頭皮質。

る。 (a) 深層ニューラルネットワークのみが、人間のパフォーマンスに匹敵するレベルの物体認識を行う。 (b) 深層ニューラルネットワークだけが、ITの表現上の幾何学性を説明する。 (c) 表現は徐々に変化しているようで、下位層は靈長類の腹側経路の初期段階に類似する。 (d) 高レベルの深層ネットワーク表現は、単にカテゴリー的な区分を強調するだけでなく、カテゴリー内の表現上の幾何学性においてもITに類似する。

8 視覚のためのリカレントニューラルネットワーク

フィードフォワードネットワークは、視覚階層におけるニューロンのシグナル伝達の最初の流れのモデルとして有用である。視覚を一瞥して説明するのに役立つ。しかし、フィードフォワードネットワークは、その接続性とダイナミクスの点で脳とは異なり、基本的には静的な機能の計算に限定されている。視覚は、一連の画像フレームのそれぞれに対して静的な関数を計算するのではなく、時間的に連続した入力ストリームを受け取り、進行中の再帰的な計算

によってそれを解釈する。現在のネットワークの状態は、最近の刺激の履歴や、差し迫ったイベントの予測、その他の行動上重要な情報を表していると考えられる。

再帰的な計算は、静止した視覚イメージを自動的に素早く解釈することに貢献していると考えられ (Sugase et al. 1999, Brincat & Connor 2006, Freiwald & Tsao 2010, Carlson et al. 2013, Cichy et al. 2014, H. Tang et al. 2014, Clarke et al. 2015), 特定の物体や物体カテゴリーを明確に区別する表現の出現につながっている。例えば、サルのニューロン集団のコードでは、フィードフォワードスイープが IT に到達するまでの 100 ms 程度の時間を超えると、個々の顔がより明確に区別できるようになる (Sugase et al. 1999, Freiwald & Tsao 2010)。同様に、物体カテゴリーのレベルでは、ヒトの脳磁図 (MEG) からの証拠によると、強いカテゴリー分割は、刺激開始後 200 ms 以上の潜時でのみ生じることが示唆されている (Carlson et al. 2013, Cichy et al. 2014, Clarke & Tyler 2015, Clarke et al. 2015)。このように、カテゴリー表現も模範表現も、本質的に同じ意味を持つ画像間の無関係な変化に対して不变性を得るために、リカレント処理に依存している可能性がある。

脳は、フィードフォワード処理とリカレント処理を組み合わせて、物体認識のために訓練されたフィードフォワード畳み込みネットワークで計算されるのと同じような表現を得ることができるかもしれない。リカレントネットワークは、ディープフィードフォワードネットワークとして展開できることを見てきた。逆に、ユニット数や接続数が限られている場合は、非常に大きなフィードフォワードネットワークで計算された望ましい関数を、より小さなネットワークでのリカレント計算で近似することもできる。リカレントダイナミクスは、計算のための限られた物理的資源を時間軸に沿って増やすことで、計算能力を拡大することができる。

リカレントニューロンダイナミクスは、フィードフォワード型の畳み込みネットワークに比べて、より高度な計算を行うことができます。例えば、視覚ニューロンの機能が、画像内のコンテンツの一部 (特徴、物体の一部、または物体) の存在を表すことだとする。フィードフォワードのスイープだけでは、ニューロンが表現するコンテンツの一部を自信を持って検出するために必要な完全な証拠を提供できない可能性がある。そのため、ニューロンは、後から届く横方向の信号とトップダウンの信号を統合して、最終的な反応に収束せざるを得ない。複数のニューロンがメッセージを再帰的に受け渡し、集団が画像の安定した解釈に収束するかもしれない。

リカレント計算では、画像形成の生成モデルを画像に繰り返しフィットさせ、フィットしたパラメータが画像の内容 (および原因) を特定することができるかもしれない (サイドバー「視覚のディープミステリー：生成モデルと識別モデルをいかに統合するか」参照)。) 簡単にするために、生成モデルが正確に反転可能であると仮定する。これは、モデルが、視覚的なイメージを曖昧にするのに十分な事前の世界知識を含んでいる場合には、説得力があるかもしれない。画像とその内容に関する記号的な記述は、一対一 (双射) の写像によって関連づけられる。原理的には、生成モデルの逆は、(普遍性のために) フィードフォワードモデルで表すことができる。しかし、そのようなモデルでは、ニューロンの数や接続数が多くなり、限られた学習データからその接続性を学ぶことができない可能性がある。フィードフォワード計算で画像を解析する代わりに、画像形成の生成モデルを認識すべき特定の画像に適合させて、合成による解析を行うことができる (Yuille & Kersten, 2006)。

視覚のディープミステリー：生成モデルと識別モデルの統合方法

近年のコンピュータビジョンの発展は、フィードフォワード型のニューラルネットワークが中心となっている。これらのモデルは識別モデルである。これらモデルは、画像形成過程を明示的にモデル化することなく、画像の集合からカテゴリーを識別する。視覚 (あるいは他のデータ解析) を実装するためのより原理的な方法は、画像を生成する過程 (データ) のモデルを定式化し、データからモデルのパラメータを推測するために、このプロセスを反転させることある (コンピュータビジョンにおけるこのアプローチの教科書は、Prince 2012 を参照)。

視覚の場合、生成モデルとは、視覚シーンの記号的記述などの高レベル表現から画像を生成する画像形成 (またはグラフィックス) モデルのことである。第一の課題は、このようなモデルを定義することである。つまり、画像を最もよく説明する高レベル表現を見つけることである。例えば、画像が与えられたとき、可能性なすべての高レベル表現に対する最大事後推定値や完全事後確率分布を見つけることである。von Helmholtz (1866) が「視覚は無意識の推

論である」と述べたように、世界についての事前の知識に基づいて、証拠を最もよく説明する解釈を積極的に探すという考え方である。

コンピュータビジョンや生物学的ビジョンの研究では、識別的アプローチから生成的アプローチまで、常に全領域に渡って行われている (Knill et al. 1996, Yuille & Kersten 2006, Prince 2012)。しかし、生成的アプローチは、コンピュータビジョンにとっては実用的に困難であり、神経科学にとっても理論的に困難である。そのため、ほとんどのコンピュータビジョンシステムは、手工業的に特徴量を使う場合でも、深層学習を使う場合でも、主に識別モデルに依存しており、画像を処理して目的の出力を生成するフィードフォワード計算を主に学習している (ただし、Prince 2012 参照)。神経科学の分野でも、同様に、HMAX (Riesenhuber & Poggio 1999) のようなフィードフォワードモデルが影響力を持っている。

画像形成には、隠蔽のような非線形プロセスが含まれる。線形生成モデルの反転は、フィードフォワード計算で実装可能な閉形式解であるが、グラフィックスモデルの反転は、計算上はるかに困難である。画像から高レベルのシーン記述を推定するには、オブジェクトと照明の可能な構成の組み合わせを (あるレベルの抽象度) 考慮する必要がある。

視覚のディープミステリー、識別モデルと生成モデルがどのように統合され、シームレスで効率的な推論過程になるのかという点にある。視覚は、一目で素早く認識するためには識別的なフィードフォワードモデルに依存し、推論の反復的な改良、最初のフィードフォワード推定のエラーの修正、またはフィードフォワードパスによって強調された仮説に対する確率的な推論にはリカレントダイナミクスに依存するかもしれない。

リカレントニューラルネットワークは、このような動的な推論プロセスを実装することができ、最近の言語処理の分野での成功を考えると、視覚研究においても基本的な進歩が期待できそうである。

9 結論

計算論的神経科学は、脳が何を計算すべきかを問うことで大きな成功を収めてきた (Kording, 2007)。提案された規範的な目標は、しばしば重要な洞察をもたらしたが、その後、より大きな目標に取って代わられた。脳は感覚情報を効率的にエンコードすべきなのか [Barlow 1961 (2012)]。あるいは、世界の正確な確率的表現を推論すべきか (Barlow 2001)。最終的な目標は成功した行動である。

規範理論は、認知やニューロンレベルでの進歩を促してきた。このアプローチの成功例としては、効率的コーディング理論 (Barlow, 1961 (2012), Olshausen & Field 1997, Simoncelli & Olshausen 2001)、確率的ニューロンコーディングと推論 (Hoyer & Hyvärinen 2003, Fiser et al. 2010, Buesing et al. 2011, McClelland 2013, Pouget et al 2013)、ベイズ感覚運動制御 (Kording & Wolpert 2006)、確率的認知 (Tenenbaum et al, 2006) などが挙げられる。低次元の感覚表現や低次元の意思決定・制御過程においては、規範理論は美しく、計算が簡単な推論手順を規定している。これらはコンピュータでの実装方法がわかつており、生物の脳でも実装される可能性がある。しかし、視覚認識をはじめとする脳の情報処理の多くは、膨大な量の世界知識を用いた推論を必要とする。最適解を示す規範理論がないだけでなく、最近まで、機能する解を実装することさえできなかった。

今まで、コンピュータは視覚的な物体認識を行うことができず、新規の自然画像の高次の表現を予測できる画像計算可能なモデルも存在しなかった。ディープニューラルネットワークは、物体認識と高次の神経反応の予測の両方を計算機上で可能にした。この進歩は、高次の視覚や他の脳機能をモデル化するための新しい計算フレームワークを開くものである。ディープニューラルネットのモデルは、課題成績に最適化されている。その意味で、このフレームワークは、脳が何を計算すべきかという問題を、最も包括的なレベル、つまり成功する行動のレベルで解決するものである。現在のところ、ディープネットワークフレームワークは、現実の課題を解決するのに十分な能力を持つ神経的に妥当なモデルと引き換えに、推論に関する明示的な確率論的説明を放棄している。将来的には、明示的な確率論的ニューラルネットモデルが実世界の課題を解決し、生物の脳をよりよく説明できるかどうかを見ることになる

だろう。

9.1 ブラックボックスを別のブラックボックスに置き換える？

複雑なニューラルネットワークを使って脳の情報処理をモデル化することに対する批判のひとつに、「複雑なネットワークを別のネットワークに置き換えてしまう」というものがある。計算を捉えることはできるかもしれないが、その計算を大きなネットに捉えているので、その複雑さは概念的に理解できない。この不可侵性批判に対する答えは2つある。

まず、神経の反応や行動を予測できるモデルができるても、私たちの仕事は終わらないのは事実である。私たちは、ネットワークが深い階層の複数のステージに渡って（また、ネットワークが再帰的である場合には時間に渡って）表現をどのように変換するのかを、より高いレベルの記述で理解する努力をしなければならない。しかし、いったん複雑な生物学的計算を人工ニューラルネットワークに取り込むことができれば、その内部ダイナミクスを完全に把握した上で、その機能を効率的にシリコンウェハー上で研究することができる。統合神経生理学は、大規模な自然および人工的な刺激セットに対する人工的なネットワークの反応を分析して視覚化するもので、これらのネットワークの内部動作を明らかにするのに役立つかもしれない（Zeiler & Fergus 2014, Girshick et al. 2014, Simonyan et al. 2014, Tsai & Cox 2015, Zhou et al. 2015, Yosinski et al. 2015）。

ニューラルネットワークモデルの不可侵性批判に対する2つ目の回答は、簡潔な数学的記述や直感的な理解を得られないメカニズムに対処する準備をしておくべきだということである。結局のところ、知能には大量の領域固有の知識が必要であり、その知識を簡潔な記述や数式に圧縮することは不可能かもしれない。言い換えれば、私たちのモデルは可能な限りシンプルでなければならぬ。

計算論的神経科学と同様、AIもシンプルで一般的なアルゴリズムから始った。しかし、これらのアルゴリズムは実世界での応用には適していなかった。本当の意味での知能を得るには、大量の知識を取り入れる必要があることがわかったのである。この洞察が、最終的に機械学習の発展につながったのである。計算論的神経科学は、AIに倣って、脳が行うことのほとんどは、経験を通じて学んだ十分な領域固有の知識を必要とすることを認めなければならない。

9.2 深層ニューラルネットモデルは生物の脳に似ているか？

この質問に対する答えは、見る人の目の中にある。生物学的な現実からの多くの抽象化や、工学的な配慮による設計上の決定に注目して、両者は大きく異なると結論づけることができる。一方、生物学的な発想の原点や、生物のニューロンがモデルユニットの操作を行うことができるという事実に着目して、両者は似ていると結論づけることもできる。

生物学的詳細からの抽象化は望ましいことであり、実際、計算論的神経科学のすべてのモデルの特徴となっている。モデルは対象と同一であることを意味するのではなく、抽象的な記述レベルで説明することを意味する。したがって、生物の脳との違いを指摘するだけでは、正当な挑戦とは言えない。例えば、本物の神経細胞が痙攣するという事実は、レートコーディングモデルにとっての挑戦ではない。それは、生物の脳が、モデルが対応していないより細かいレベルで記述できることを意味している。しかし、スパイクが計算上の必要条件であり（例えば、Buesing et al. 2011），スパイクモデルが、スパイクレートを予測するという独自のゲームにおいて、あるいは行動を予測するという点において、最高のレートコーディングモデルよりも優れている場合、このモデルはレートコーディングアプローチに対する挑戦となる。

現在、コンピュータビジョンで主流となっている深層畳み込みフィードフォワードネットワークの特徴の多くは、生物学的視覚をモデル化するという文脈では、疑問視されるべきものである（サイドバー「敵対的事例からニューラルネットワークの特性を明らかにする」を参照）。例えば、フィードフォワードアーキテクチャにおけるバイパス接続の欠如、フィードバックおよびローカルリカレント接続の欠如、ユニットの線形-非線形性、整流線形活性化関数、最

大プーリング操作、およびオンラインの教師付き勾配降下学習などである。これらの特徴に挑戦するためには、測定されたニューロン反応や行動のパフォーマンスが、別の種類のモデルを使った方がより正確に予測できることを示さなければならない。

ニューラルネットワークに関する文献は複雑で、理論的な神経科学からコンピュータサイエンスまで多岐にわたっている。この文献には、フィードフォワードモデルとリカレントモデル、識別モデルと生成モデル、決定論的モデルと確率論的モデル、非スパイクモデルとスパイクモデルが含まれている。これらの文献は、脳内の情報処理に関する将来のより包括的な理論を構築するための基礎となるものである。これらのモデルは、工学的に実世界の課題や人間の成績レベルにまでスケールアップし始めているので、脳科学におけるこのモデリングフレームワークを使って、知覚、認知、運動制御などの複雑な過程を取り組むことができるようになる。

敵対的事例からニューラルネットワークの特性を明らかにする

視覚を欺くにすることで、そのメカニズムを知ることができる。これは、生物学的な視覚でも人工的な視覚でも同様である。研究者たちは、人工ニューラルネットワークがどのように画像を表現するかを、誤魔化そうとして探っている (Szegedy et al. 彼らは最適化技術を用いて、誤って分類された画像を設計した。敵対的な例としては、カテゴリー X の画像 (例えば、バスやノイズ画像) が、特定のネットワークによってカテゴリー Y (例えば、ダチョウ) に属するものとして誤分類されるように設計されたものがある。このような画像は、カテゴリー X の自然な画像を取り出し、ネットを欺くために画素ルを調整することで設計できる。バックプロパゲーションアルゴリズムは、通常、画像の誤差を小さくするような重みの小さな調整を見つける役割を果たすが、代わりに誤差を生じさせるような画像の小さな調整を見つけるためにも使用できる。現在コンピュータビジョンで使用されている畳み込みニューラルネットワークでは、明らかに異なるカテゴリーの有効な例とはならないような画像のごくわずかな変更によって、敵対的な例が作られることがある。このような敵対的な例は、人間には元の画像と区別がつかないことがある。これは、現在のニューラルネットワークアーキテクチャが、ビジョンシステムとしても、人間の視覚のモデルとしても限界があることを示す証拠とされている。

人工ニューラルネットワークを欺くために作られた敵対的な例は、通常、人間の観察者を欺くことはできない。しかし、人間の視覚システムについても同様に敵対例を作ることができるかどうかは分かっていない。上述した敵対例を作る技術は、騙すべき特定のネットワークの接続を完全に知っている必要がある。現在の心理物理学や神経生理学の技術では、生物の視覚を騙すためのこのプロセスには対応できない。したがって、生物学的な視覚システムも敵対的な例に影響を受けやすいという興味深い可能性がある。これは、ある人を騙すために作られた敵対的事例が、別の人を騙すことができないような、特定の脳の特異性を利用している可能性がある。視覚システムの目的は、自然な条件下でうまく機能することであって、ネットワークの内部構造への全知全能のアクセスと、視覚センサーレイ上の騙し絵の正確な安定化を必要とする、極めて巧妙な妨害行為と無縁であることではない。人間の視覚は、様々な種類の視覚的錯覚を起こしやすいことで知られている。さらに、機械学習の観点からは、限られた例のセットから高次元の分類関数に一般化するために、不完全な帰納的バイアスに頼らなければならない人工的なものであれ、自然なものであれ、あらゆる学習システムに敵対的な例が構築されることは避けられないと思われる。

敵対的な例は、現在のニューラルネットワークモデルにどのような教訓を与えるのか？もし、敵対的な例が、それが構築されたネットワークの特定のインスタンスだけを騙し、その特定のネットの特異性を利用するものであれば、それを否定するのは簡単である。しかし、敵対的事例は、ある程度、ネットワークを越えて一般化する。同じアーキテクチャを新しいランダムな重みで初期化し、同じラベル付き画像のセットでトレーニングして新しいネットワークを作成しても、元のネットワーク用に作成された敵対的な例に騙されることがよくある。また、同じ学習セットを使用していれば、敵対的な例はアーキテクチャを少し変えただけで一般化する。学習セットを変更すると、元のネットワーク用に作成された敵対的な例はあまり効果的ではなくなるが、それでも自然な画像よりも高い割合で誤分類されることがある。このことから、敵対的な例は、主に学習セットに起因するネットワークの特異性を利用していることが示唆されるが、使用する基本的な計算処理にもある程度起因する。ひとつの可能性として、現在のシステム

では、各ユニットが計算する線形結合により、これらのシステムが特に騙されやすくなっていることが挙げられる (Goodfellow et al.2015)。要するに、各ユニットはその入力空間を線形の境界で分割している（シグモイドの場合は境界を越えると活性化が滑らかに上昇し、整流線形活性化関数の場合は好ましい側で線形に上昇するとしても）。一方、放射状基底関数を用いたネットワークは、各ユニットが入力空間に特定の好ましいパターンを持ち、応答がすべての方向に落ちていくので、騙すのが難しいかもしれない。しかし、このようなネットワークは、おそらく同じ理由で、訓練するのも難しい。人工的なニューラルネットワークと生物学的なニューラルネットワークの複雑な表現変換をより詳細に比較することで、このパズルが解かれることを期待している。

9.3 今後の展開

本研究では、最新のニューラルネットワーク技術を用いて、生物の脳の大部分である視覚系の内部ダイナミクスや計算機能を近似化することを目的としている。重要な目標は、視覚野に 1 対 1 で対応する層を持ち、受容野、非線形応答特性、および対応する霊長類の視覚野のものと一致する表現幾何学を持つモデルを構築することである。物体認識のような意味のある課題をシステムが実行するという要件は、機能上の大きな制約となる。現在、数百万枚のラベル付き画像を用いたニューラルネットワークのタスクトレーニングは、神経生理学的データよりもはるかに強い制約をモデル候補の空間に与えている。実際、新しい自然画像の脳内表現を予測する最近の成功例は、主にタスクトレーニングによってもたらされている (Yamins et al.2014; Khaligh-Razavi & Kriegeskorte 2014; Cadieu et al.2014)。しかし、超並列脳活動計測の進歩により、将来的にはモデル空間に対してより強い脳ベースの制約を与えることが期待されている。工学分野で一般的に行われているように、純粋に課題ベースの損失関数を最小化するのではなく、生物学的な脳をモデル化するには、接続パターン、内部表現、タスクパフォーマンスを脳や行動の測定値と一致させるような新しい学習アルゴリズムが必要になる。最終的な処理メカニズムだけでなく、学習プロセスを生物学的に妥当な方法でモデル化するためには、教師なし学習や強化学習の技術も必要になるだろう (Sutton & Barto 1998, Mnih et al.2015)。

AI、機械学習、そして認知科学や脳科学は、深い共通のルーツを持っている。認知レベルでは、これらの分野は、推論と学習のベイズモデルを通じて最近収束してきている (Tenenbaum et al. 2006)。ディープネットワークと同様に、ベイジアンノンパラメトリック手法 (Ghahramani 2013) は、大量の世界知識を取り込むことができる。これらのモデルには、明示的な確率的推論と学習という利点がある。このような推論プロセスを生物学的なニューラルネットワークにどのように実装するかを説明することは、今後の大きな課題の一つである。ニューラルネットワークは、AI、認知科学、機械学習、計算神経科学の分野で長い歴史を持ち、これらの分野をつなぐ共通のモデリングフレームワークとなっている。現在、脳のような深い並列計算の能力に関する初期の直感が工学的に実証されており、これらの分野の収束が再活性化されている。現実の世界で複雑な知能を発揮するモデルを構築し (AI)、神経細胞のダイナミクス (計算神経科学) や行動 (認知科学) を説明できれば、取り組むべき課題について、脳の仕組みを理解することができるだろう。

9.4 要約点

1. ニューラルネットワークは、脳からヒントを得て開発された計算モデルで、コンピュータビジョンをはじめとする人工知能の分野で主流となっている
2. ニューラルネットワークは、入力の非線形関数を計算する相互接続されたユニットで構成される。ユニットは通常、入力の重み付けされた組み合わせを計算し、続いて静的な非線形性を計算する。
3. フィードフォワードニューラルネットワークは、普遍関数近似器である。
4. リカレントニューラルネットワークは、動的システムの普遍的近似器である。
5. ディープニューラルネットワーク非線形変換を何層にも重ねることで、視覚などの複雑な機能を簡潔に表現で

きる。

6. 畳み込みニューラルネットワークは、初期の層にあるユニットの入力接続を、空間的な位置を超えて複製された重みテンプレートを用いて局所的な受容野に拘束する。重みの制限と共有により、学習が必要なパラメータの数が大幅に減少する。
7. 物体認識のための深い畳み込みフィードフォワードネットワークは、生物学的に詳細ではなく、生物の脳とは異なる非線形性や学習アルゴリズムに依存している。しかし、ヒトや霊長類の大脳皮質における表現と非常によく似た内部表現を学習する。
8. ニューラルネットワークは、現実の AI 課題にも対応しており、複雑な脳の情報処理をより生物学的に忠実にモデル化するためのエキサイティングな技術的枠組みを提供している。

9.5 将来の問題

1. 現実世界の複雑な課題に取り組むニューラルネットモデルを構築し、生物学的な脳活動のパターンと行動のパフォーマンスを同時に説明する。
2. アーキテクチャパラメータ、非線形表現変換、学習アルゴリズムの観点から、より生物学的に忠実なモデルを構築する。
3. ネットワーク層は、その応答特性と表現形状において、視覚階層の領域と一致する必要がある。
4. 異なる課題における特定の刺激に対する反応時間、類似性の判断、課題エラー、連続インタラクティブタスクにおける詳細な運動軌跡など、豊富な行動計測値を予測するモデル
5. New supervised learning techniques will drive neural networks to alignment with measuring functional and anatomical brain data and behavioral data.
6. リカレントニューラルネットワークモデルにより、生物の脳の表現力を説明する。
7. フィードフォワード、ラテラル結合、フィードバックの情報の流れを説明し、画像形成の生成モデルに確率的推論を導入する。
8. カテゴリー化を超えた複雑な視覚機能に取り組む。例えば、ユニークな実体の識別、シーンを積極的に探索する注意シフトや眼球運動、視覚探索、画像セグメンテーション、より複雑な意味解釈、感覚運動統合などが挙げられる。

Contents

Volume 1, 2015

An autobiographical article by Horace Barlow is available online at
www.annualreviews.org/r/horacebarlow.

Image Formation in the Living Human Eye <i>Pablo Artal</i>	1
Adaptive Optics Ophthalmoscopy <i>Austin Roorda and Jacque L. Duncan</i>	19
Imaging Glaucoma <i>Donald C. Hood</i>	51
What Does Genetics Tell Us About Age-Related Macular Degeneration? <i>Felix Grassmann, Thomas Ach, Caroline Brandl, Iris M. Heid, and Bernhard H.F. Weber</i>	73
Mitochondrial Genetics and Optic Neuropathy <i>Janey L. Wiggs</i>	97
Zebrafish Models of Retinal Disease <i>Brian A. Link and Ross F. Collery</i>	125
Angiogenesis and Eye Disease <i>Yoshihiko Usui, Peter D. Westenskow, Salome Murinello, Michael I. Dorrell, Leah Schepke, Felicitas Bucher, Susumu Sakimoto, Liliana P. Paris, Edith Aguilar, and Martin Friedlander</i>	155
Optogenetic Approaches to Restoring Vision <i>Zhuo-Hua Pan, Qi Lu, Anding Bi, Alexander M. Dizhoor, and Gary W. Abrams</i>	185
The Determination of Rod and Cone Photoreceptor Fate <i>Constance L. Cepko</i>	211
Ribbon Synapses and Visual Processing in the Retina <i>Leon Lagnado and Frank Schmitz</i>	235
Functional Circuitry of the Retina <i>Jonathan B. Demb and Joshua H. Singer</i>	263

Contributions of Retinal Ganglion Cells to Subcortical Visual Processing and Behaviors <i>Onkar S. Dhande, Benjamin K. Stafford, Jung-Hwan A. Lim, and Andrew D. Huberman</i>	291
Organization of the Central Visual Pathways Following Field Defects Arising from Congenital, Inherited, and Acquired Eye Disease <i>Antony B. Morland</i>	329
Visual Functions of the Thalamus <i>W. Martin Usrey and Henry J. Alitto</i>	351
Neuronal Mechanisms of Visual Attention <i>John Maunsell</i>	373
A Revised Neural Framework for Face Processing <i>Brad Duchaine and Galit Yovel</i>	393
Deep Neural Networks: A New Framework for Modeling Biological Vision and Brain Information Processing <i>Nikolaus Kriegeskorte</i>	417
Visual Guidance of Smooth Pursuit Eye Movements <i>Stephen G. Lisberger</i>	447
Visuomotor Functions in the Frontal Lobe <i>Jeffrey D. Schall</i>	469
Control and Functions of Fixational Eye Movements <i>Michele Rucci and Martina Poletti</i>	499
Color and the Cone Mosaic <i>David H. Brainard</i>	519
Visual Adaptation <i>Michael A. Webster</i>	547
Development of Three-Dimensional Perception in Human Infants <i>Anthony M. Norcia and Holly E. Gerhard</i>	569

Errata

An online log of corrections to *Annual Review of Vision Science* articles may be found at
<http://www.annualreviews.org/errata/vision>