Deep Learning: A Primer for Radiologists[1]

Gabriel Chartrand, PhD
Phillip M. Cheng, MD, MS
Eugene Vorontsov, BASc Eng Sci
Michal Drozdzal, PhD
Simon Turcotte, MD, MSc
Christopher J. Pal, PhD
Samuel Kadoury, PhD
An Tang, MD, MSc

Deep learning is a class of machine learning methods that are gaining success and attracting interest in many domains, including computer vision, speech recognition, natural language processing, and playing games. Deep learning methods produce a mapping from raw inputs to desired outputs (eg, image classes). Unlike traditional machine learning methods, which require hand-engineered feature extraction from inputs, deep learning methods learn these features directly from data. With the advent of large datasets and increased computing power, these methods can produce models with exceptional performance. These models are multilayer artificial neural networks, loosely inspired by biologic neural systems. Weighted connections between nodes (neurons) in the network are iteratively adjusted based on example pairs of inputs and target outputs by back-propagating a corrective error signal through the network. For computer vision tasks, convolutional neural networks (CNNs) have proven to be effective. Recently, several clinical applications of CNNs have been proposed and studied in radiology for classification, detection, and segmentation tasks. This article reviews the key concepts of deep learning for clinical radiologists, discusses technical requirements, describes emerging applications in clinical radiology, and outlines limitations and future directions in this field. Radiologists should become familiar with the principles and potential applications of deep learning in medical imaging.

Abbreviations: CNN = convolutional neural network, 3D = three-dimensional, 2D = two-dimensional

RadioGraphics 2017; 37:2113–2131

https://doi.org/10.1148/rg.2017170077

Content Code: IN

[1]From the Departments of Radiology (G.C., E.V., A.T.) and Hepatopancreatobiliary Surgery (S.T.), Centre Hospitalier de l'Université de Montréal, Hôpital Saint-Luc, 850 rue Saint-Denis, Montréal, QC, Canada H2X 0A9; Imagia Cybernetics, Montréal, Québec, Canada (G.C., M.D.); Department of Radiology, Keck School of Medicine, University of Southern California, Los Angeles, Calif (P.M.C.); Montreal Institute for Learning Algorithms, Montréal, Québec, Canada (E.V., M.D., C.J.P.); École Polytechnique, Montréal, Québec, Canada (E.V., C.J.P., S.K.); Department of Surgery, University of Montreal, Montréal, Québec, Canada (S.T.); and Centre de Recherche du Centre Hospitalier de l'Université de Montréal, Montréal, Québec, Canada (S.T., S.K., A.T.). Recipient of a Cum Laude award for an education exhibit at the 2016 RSNA Annual Meeting. Received April 3, 2017; revision requested August 11 and received August 23; accepted August 31. For this journal-based SA-CME activity, the authors G.C., E.V., C.J.P., and A.T. have provided disclosures (see end of article); all other authors, the editor, and the reviewers have disclosed no relevant relationships. **Address correspondence to** A.T. (e-mail: *an.tang@umontreal.ca*).

Supported by the Consortium for Research and Innovation in Medical Technologies in Quebec, MITACS–Cluster Accelerate (IT05356), the recruitment fund of the Centre de Recherche du Centre Hospitalier de l'Université de Montréal, Polytechnique Montréal, and Imagia Cybernetics. A.T. supported by the Fonds de Recherche du Québec en Santé and Fondation de l'Association des Radiologistes du Québec (Clinical Research Scholarship–Junior 1 Salary Award 26993).

SA-CME LEARNING OBJECTIVES

After completing this journal-based SA-CME activity, participants will be able to:

- Discuss the key concepts underlying deep learning with CNNs.
- Describe emerging applications of deep learning techniques to radiology for lesion classification, detection, and segmentation.
- List key technical requirements in terms of dataset, hardware, and software required to perform deep learning.

See www.rsna.org/education/search/RG.

Introduction

Medical image analysis and interpretation are fundamental cognitive tasks of a diagnostic radiologist. Effective computer automation of these tasks has historically been difficult despite technical advances in computer vision, a discipline dedicated to the problem of imparting visual understanding to a computer system. Recently, however, computer science researchers using a technique called deep learning have demonstrated breakthrough performance improvements in a variety of complex tasks, including image classification, object detection, speech recognition, language translation, natural language processing, and playing games (1,2).

TEACHING POINTS

- Deep learning is a type of representation learning in which the algorithm learns a composition of features that reflect a hierarchy of structures in the data. Complex representations are expressed in terms of simpler representations.
- Although neural networks have been used for decades, in recent years three key factors have enabled the training of large neural networks: *(a)* the availability of large quantities of labeled data, *(b)* inexpensive and powerful parallel computing hardware, and *(c)* improvements in training techniques and architectures.
- The basic unit of an artificial neural network, the artificial neuron or node, is a simplified model that mimics the basic mechanism found in the biologic neuron.
- Deep CNNs exploit the compositional structure of natural images so that shifts and deformations of objects in the images do not significantly affect the overall performance of the network.
- The creation of these large databases of labeled medical images and many associated challenges will be fundamental to foster future research in deep learning applied to medical images.

The success of deep learning with convolutional neural networks (CNNs) for images in nonmedical domains has fostered hopes for and research toward revolutionizing the automated analysis of medical images. At the same time, it has raised the necessity for clinical radiologists to become familiar with this rapidly developing technology, as some artificial intelligence experts have speculated that deep learning systems may soon surpass radiologists for certain image interpretation tasks (3,4).

Recently, these deep learning algorithms have been applied to medical imaging in several clinical settings, such as detection of breast cancer on mammograms (5,6), segmentation of liver metastases with computed tomography (CT) (7), brain tumor segmentation with magnetic resonance (MR) imaging (8), classification of interstitial lung disease with high-resolution chest CT (9), and generation of relevant labels pertaining to the content of medical images (10).

In this article, we review the premise and promise of deep learning by defining key terms in artificial intelligence and by reviewing the historical context that led to the emergence of deep learning systems. We describe the basic structure of neural networks and the CNN architecture. We briefly summarize technical and data prerequisites for deep learning. Finally, we explore types of emerging clinical applications and outline current limitations and future directions in the field.

What Is Deep Learning?

Deep learning is a class of machine learning algorithms characterized by the use of neural networks with many layers. This article focuses on CNNs (or "convnets"), since they are the most commonly used for image data. While other deep learning architectures exist for processing text in radiology reports (with natural language processing) or audio, these topics are beyond the scope of this article (11).

Definitions

The following terms from computer science are helpful for defining the context of deep learning.

Artificial Intelligence

Artificial intelligence is the branch of computer science devoted to creating systems to perform tasks that ordinarily require human intelligence. This is a broad umbrella term encompassing a wide variety of subfields and techniques; in this article, we focus on deep learning as a type of machine learning (Fig 1).

Machine Learning

Machine learning is the subfield of artificial intelligence in which algorithms are trained to perform tasks by learning patterns from data rather than by explicit programming (13). In classic machine learning, expert humans discern and encode features that appear distinctive in the data, and statistical techniques are used to organize or segregate the data on the basis of these features (Fig 2). For instance, for the purpose of analyzing an image, an expert in image processing might program an algorithm to decompose input images into basic elements of edges, gradients, and textures. Statistical analysis of the presence of these features in a given image can then be used to classify or interpret the image.

However, for many complex computer vision tasks, it is typically not clear even to an expert how to define the optimal image features for a machine learning algorithm to use. For example, it may not be obvious how to teach a computer to recognize an organ on the basis of pixel brightness (Fig 3). Therefore, it may be desirable for a computer system to not only learn the mappings of features to desired outputs, but to learn and optimize the features themselves.

Representation Learning

Representation learning is a type of machine learning in which no feature engineering is used. Instead, the algorithm learns on its own the best features to classify the provided data. With enough training examples, a system based on representation learning could potentially classify data better than with hand-engineered features. The challenge is in how a machine learning system could learn potentially complex features directly from raw data.

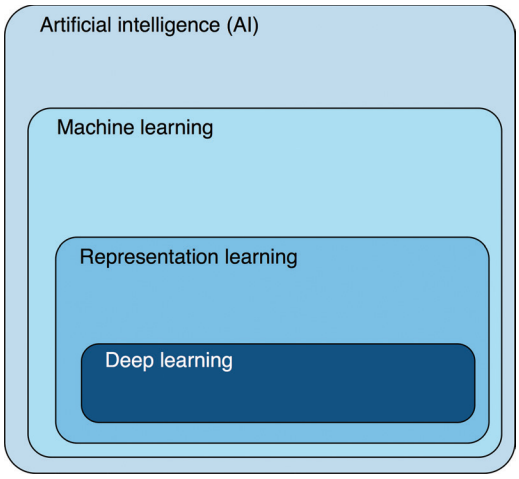


Figure 1. Venn diagram. Artificial intelligence is a subfield of computer science devoted to creating systems to perform tasks that ordinarily require human intelligence. Machine learning is a subfield of artificial intelligence where computers are trained to perform tasks without explicit programming. Classically, humans engineer features by which a computer can learn to distinguish patterns of data. Representation learning is a type of machine learning where no feature engineering is used; instead, the computer *learns* the features by which to classify the provided data. Deep learning is a type of representation learning where the learned features are compositional or hierarchical. (Adapted from reference 12.)

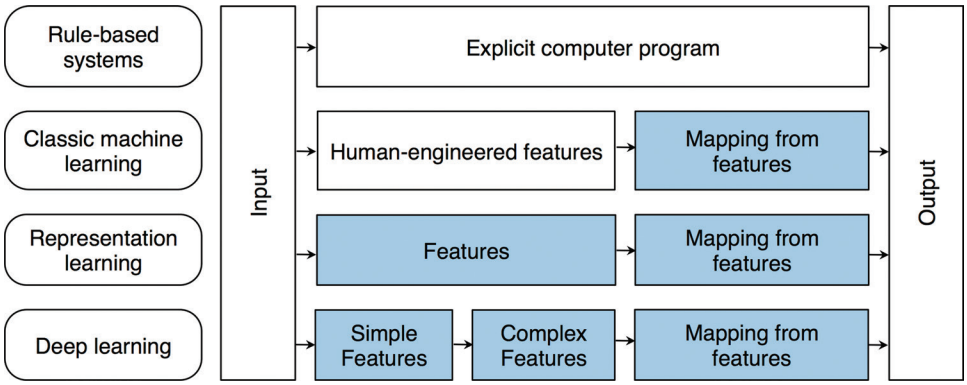


Figure 2. Classic machine learning depends on carefully designed features, requiring human expertise and complicated task-specific optimization. Deep learning bypasses feature engineering by taking advantage of large quantities of data and flexible hierarchical models. Deep learning has recently achieved striking performance improvements in diverse fields such as image classification, speech recognition, natural language processing, and playing games. Blue boxes represent components learned by fitting a model to example data; deep learning allows learning an end-to-end mapping from the input to the output. (Adapted from reference 12.)

Deep Learning

Deep learning is a type of representation learning in which the algorithm learns a composition of features that reflect a hierarchy of structures in the data. Complex representations are expressed in terms of simpler representations (12). These deep learning systems propose an end-to-end approach by learning simple features (such as signal intensity, edges, and textures) as components of more complex features such as shapes, lesions, or organs, therefore leveraging the compositional nature of images (Fig 4).

Historical Context

Deep learning systems encode features by using an architecture of artificial neural networks, an approach consisting of connected nodes inspired by biologic neural networks. Neural networks have a long history in artificial intelligence dating back to the 1950s (14). Systematic methods to train neural networks on the basis of a process called back-propagation were developed in the 1980s (15). However, success in training the deep multilayer neural networks needed for hierarchical representations was limited by the difficulty of the underlying optimization problem as well as the limits of the computing hardware of that early era. Consequently, research attention in machine learning for the next few decades drifted toward other techniques such as kernel methods and decision trees.

Although neural networks have been used for decades, in recent years three key factors have enabled the training of large neural networks: *(a)* the availability of large quantities of labeled data, *(b)* inexpensive and powerful parallel computing hardware, and *(c)* improvements in training techniques and architectures.

For processing images, a deep learning architecture known as the *convolutional neural network* has become dominant. CNNs of increasing depth and complexity have gained significant attention since 2012, when the winning entry in an annual international image classification competition

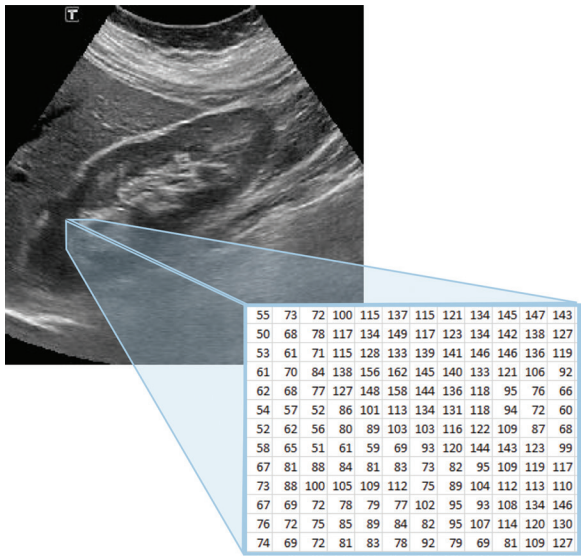


Figure 3. Human versus computer vision. A human expert easily classifies this image as an image of the right kidney. Why is this task difficult for a computer? Instead of shades of gray, a computer "sees" a matrix of numbers representing pixel brightness. Computer vision typically involves computing the presence of numerical patterns (features) in this matrix, then applying machine learning algorithms to distinguish images on the basis of these features.

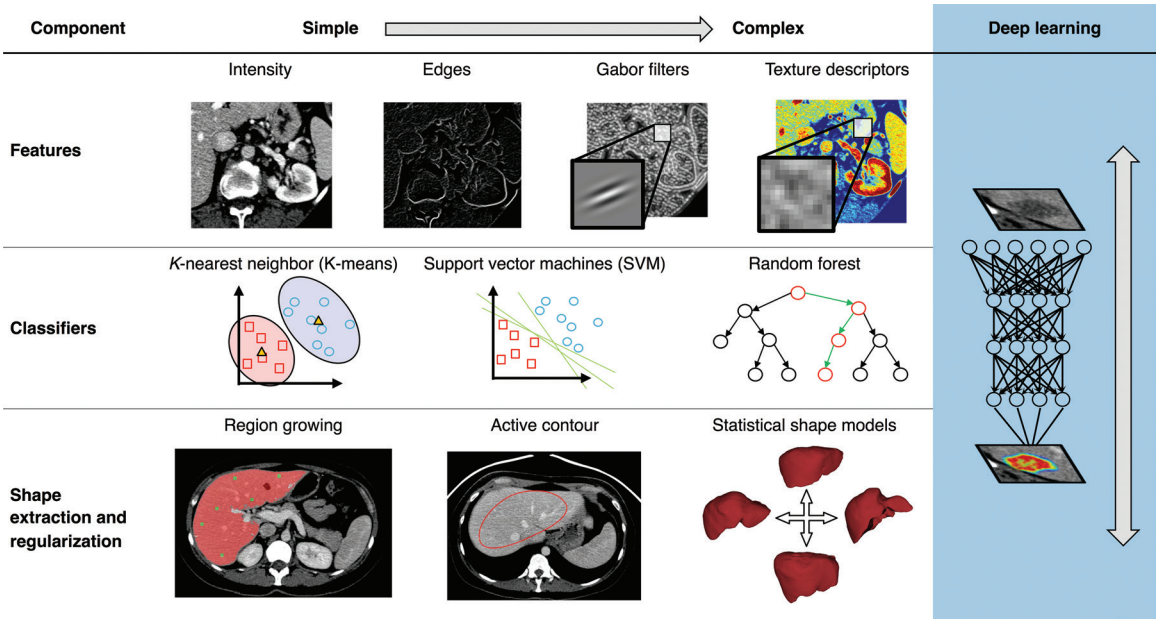


Figure 4. Computer vision tasks such as detection, segmentation, and classification are typically carried out with algorithms based on features, classifiers, and shape extraction methods. Recent approaches based on deep learning represent an important paradigm shift where features are not handcrafted, but learned in an end-to-end fashion. Features describe the appearance of organs and points of interest in medical images. Classifiers integrate features to output a decision. Shape extraction and regularization recover a consistent shape despite classification noise. Deep learning proposes an end-to-end approach where features are learned to maximize the classifier's performance. Shape regularization becomes implicit and often requires only mild postprocessing to recover the target shape.

(the ImageNet Large Scale Visual Recognition Challenge) used a deep CNN to produce a startling performance breakthrough compared with traditional computer vision techniques (16). Since 2012, all winning entries in this competition have used CNNs and have even exceeded human performance (17).

Neural Networks

In the brain, neurons exchange information via chemical and electrical synapses. Electrochemical signals are propagated from the synaptic area through the dendrites toward the soma, the body of the cell (Fig 5). When a certain excitation threshold is reached, the cell releases an activation signal through its axon toward synapses with neighboring neurons. Complex signals can be encoded by networks of neurons on the basis of this paradigm; for instance, a hierarchy of neurons in the visual cortex is able to detect edges by combining signals from independent visual receptors.

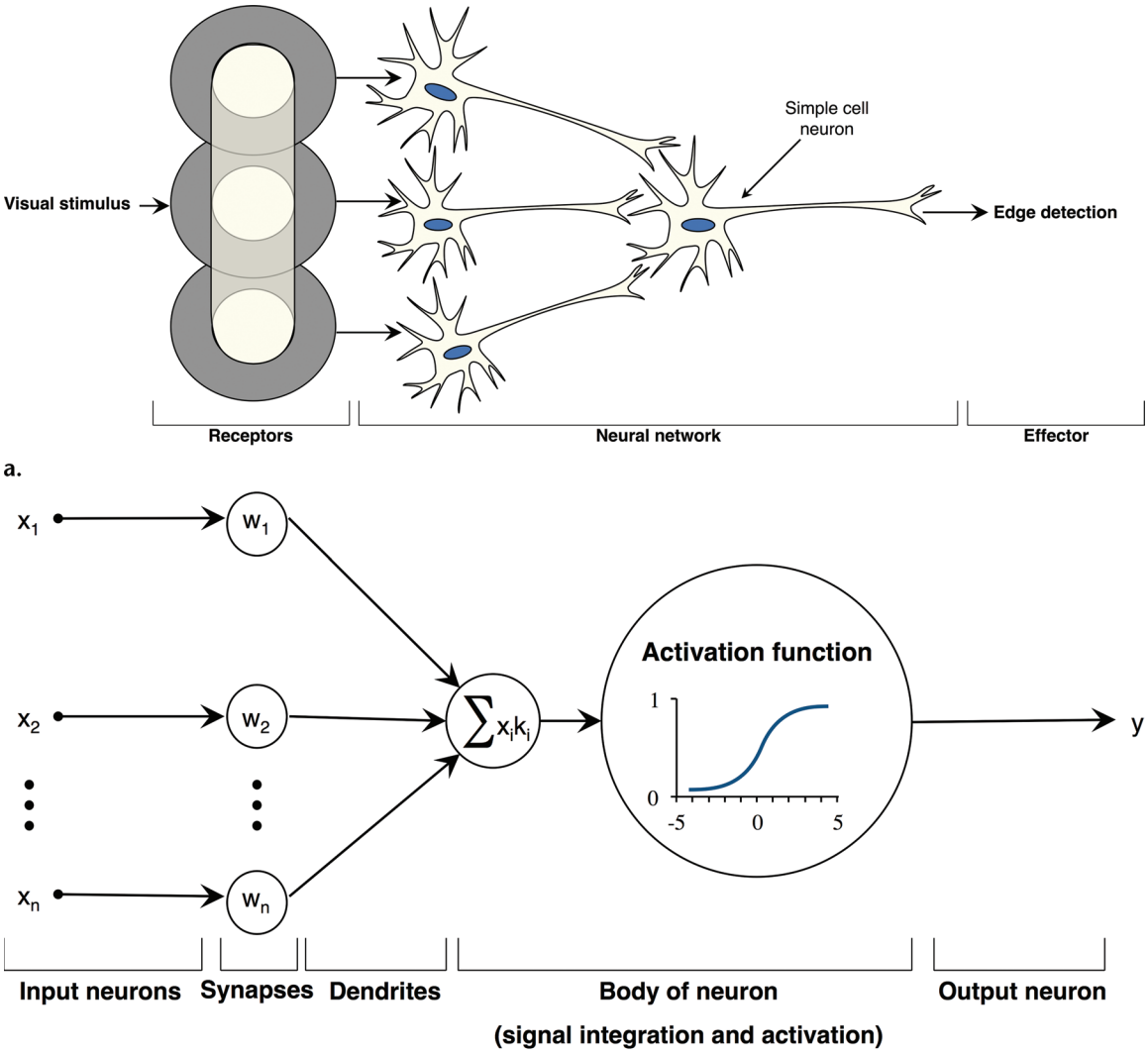


Figure 5. Conceptual analogy between components of biologic neurons **(a)** and artificial neurons **(b)**. The concept of neural networks stems from biologic inspiration. **(a)** In the visual cortex, there is a neural network able to detect edges from what is seen by the retina (gray circles = receptive areas of the retina). When the inner parts (smaller circles) of the three receptors are activated simultaneously, the simple cell neuron integrates the three signals and transmits an edge detection signal. **(b)** An artificial neural network is composed of interconnected artificial neurons. Each artificial neuron implements a simple classifier model, which outputs a decision signal based on a weighted sum of evidences, and an activation function, which integrates signals from previous neurons. Hundreds of these basic computing units are assembled together to build an artificial neural network computing device. The weights of the network are trained via a learning algorithm where pairs of input signals and desired output decisions are presented, much like the brain, which relies on external sensory stimuli to learn to achieve specific tasks.

Artificial neural networks are inspired by this biologic process. The basic unit of an artificial neural network, the artificial neuron or node, is a simplified model that mimics the basic mechanism found in the biologic neuron. The artificial neuron takes as an input a set of values representing features, each multiplied by a corresponding weight. The weighted features are summed and passed through a nonlinear activation function. In this way, an artificial neuron can be viewed as producing a decision by weighing a set of evidence.

Although an individual artificial neuron is simple, neural network architectures called multilayer perceptrons that consist of thousands of neurons can represent very complex nonlinear functions. These multilayer perceptrons are typically constructed by assembling multiple neurons to form a layer and by stacking these layers, connecting the output of one layer to the input of the following layer. The "deep" aspect of deep learning refers to the multilayer architecture of multilayer perceptrons (Fig 6). The first layer, called the input layer, represents input data such as individual pixel intensities, while the output layer produces target values such as a classification result. The intermediate layers of multilayer perceptrons are called hidden layers, since they do not directly produce visible desired outputs, but rather compute intermediate representations of the input features that are useful in the inference process.

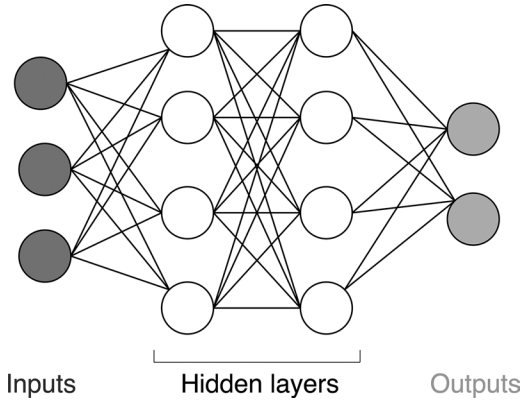


Figure 6. The basis for most deep learning research is the artificial neural network, a computational framework of interconnected nodes inspired by biologic neural networks. The "deep" aspect of deep learning refers to the multilayer architecture of these networks, which contain multiple hidden layers of nodes between the input and output nodes. This example has three input nodes, two hidden layers (each with four nodes), and two output nodes.

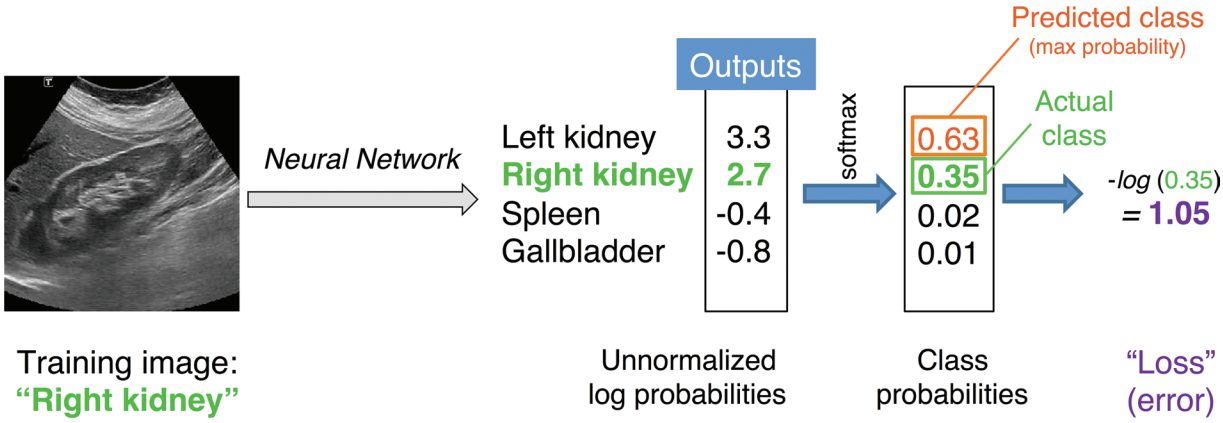


Figure 7. Softmax classifier. For classification, the output nodes of a neural network can be regarded as unnormalized log probabilities for each class. The softmax function converts these into class probabilities: During training, a "loss" value is computed to represent the error between the network's output-predicted class and the actual class of the input. This error is back-propagated from the final layer to adjust the weights throughout the network in a manner that minimizes the loss.

By stacking multiple layers, a network can represent a hierarchy of features that are an increasingly complex composition of low-level input features, thereby modeling higher levels of abstractions in the data. The compositional power of deep architectures allows neural networks to infer decisions on the basis of abstract concepts.

Obtaining a prediction from a sample observation (eg, an image) using a neural network involves computing sequentially the activation of each node of each layer, starting from the input layer up to the output layer, a process called forward propagation. In the setting of a classification task, the activation of the output layer is typically submitted to a softmax function, a normalized "squashing" function that maps a vector of real values to a probability distribution. Therefore, the softmax function converts raw activation signals from the output layer to target class probabilities (Fig 7).

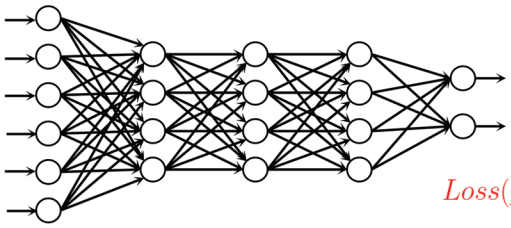
A neural network is trained by adjusting the parameters, which consist of the weights and biases of each node. Modern neural networks contain millions of such parameters. Starting from a random initial configuration, the parameters are adjusted via an optimization algorithm called gradient descent, which attempts to find a set of parameters that performs well on a training dataset (Fig 8). Each time predictions are computed from a given data sample (forward propagation), the performance of the network is assessed through a loss (error) function that quantitatively measures the inaccuracy of the prediction. Each parameter of the network is then adjusted by small increments in the direction that minimizes the loss, a process called back-propagation.

Owing to memory limitations and algorithmic advantages, the update of parameters is computed from a randomly selected subset of the training data at each iteration, a commonly used optimization method called stochastic gradient descent. After this training procedure is performed multiple times for each sample in the training dataset, the parameters approach values that maximize the model accuracy.

Deep CNNs

The composition of features in deep neural networks is enabled by a property common to all natural images: local characteristics and regularities dominate, and so complicated parts can be built

Figure 8. Learning process. Weights used by artificial neurons can nowadays amount to billions of parameters within a deep neural network. These parameters, randomly initialized, are progressively adjusted **(a)** via an optimization algorithm called gradient descent **(b)**. When presenting a series of training samples to the network, a loss function measures quantitatively how far the prediction is to the target class or regression value. All parameters are then slightly updated in the direction that will favor minimization of the loss function.



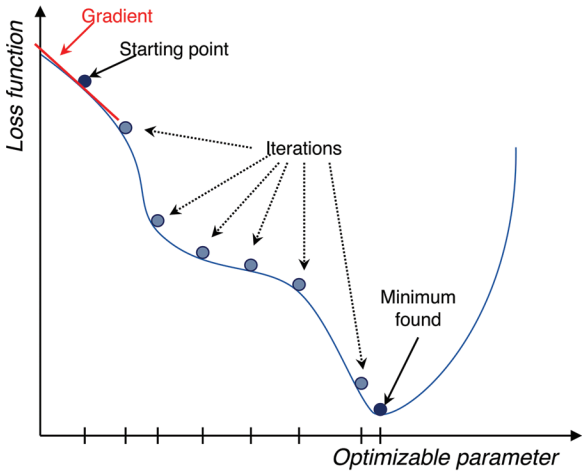
Prediction q	Target p
0.89	1.0
0.11	0.0

$$Loss(p,q) = -\sum_{x} p(x) log(q(x))$$

Learning algorithm

1. Present a batch of training samples to the network to evaluate a prediction based on its current configuration.
2. Evaluate the loss function by comparing the output prediction with the target values or classes.
3. Compute the gradient of the loss function with respect to every parameter of the model.
4. Update the weights of the model.
5. Repeat steps 1 to 4 until the loss function reaches a minimum.

a.



b.

from small local features. CNNs exploit the same property to efficiently process larger and more variable inputs than is reasonable with multilayer perceptrons. Unlike CNNs, multilayer perceptrons perform poorly on images in which the object of interest tends to vary in shape, orientation, and position because they must encode redundant representations for the many feature arrangements that this results in.

On the other hand, a CNN introduces some robustness to these variations by passing each feature detector over every part of the image in a convolution operation. Each feature detector is then limited to detecting local features in its immediate input, which is acceptable for natural images. Thus, deep CNNs exploit the compositional structure of natural images so that shifts and deformations of objects in the images do not significantly affect the overall performance of the network. They address complex tasks such as image classification with an efficient model architecture based on the following components: convolutions, activation functions, pooling, and softmax function.

Convolutions

This type of network relies on convolution operations: the linear application of a filter or kernel to local neighborhoods of points in an input (Fig 9a). Common image filters in PACS (picture archiving and communication system) workstations, such as those for image sharpening and smoothing, work using such operations. Filters representing features are usually defined by a small grid of weights (eg, 3×3). If an input has n channels (eg, different color channels), then the size of the filters would be $n \times 3 \times 3$.

Since a feature may occur anywhere in the image, the filters' weights are shared across all the image positions. Thus, image features can be modeled with fewer parameters, increasing model efficiency. Typically, multiple different convolutional filters are learned for each layer, yielding many different feature maps, each highlighting where different characteristics of the input image or of the previous hidden layer have been detected (Fig 9b).

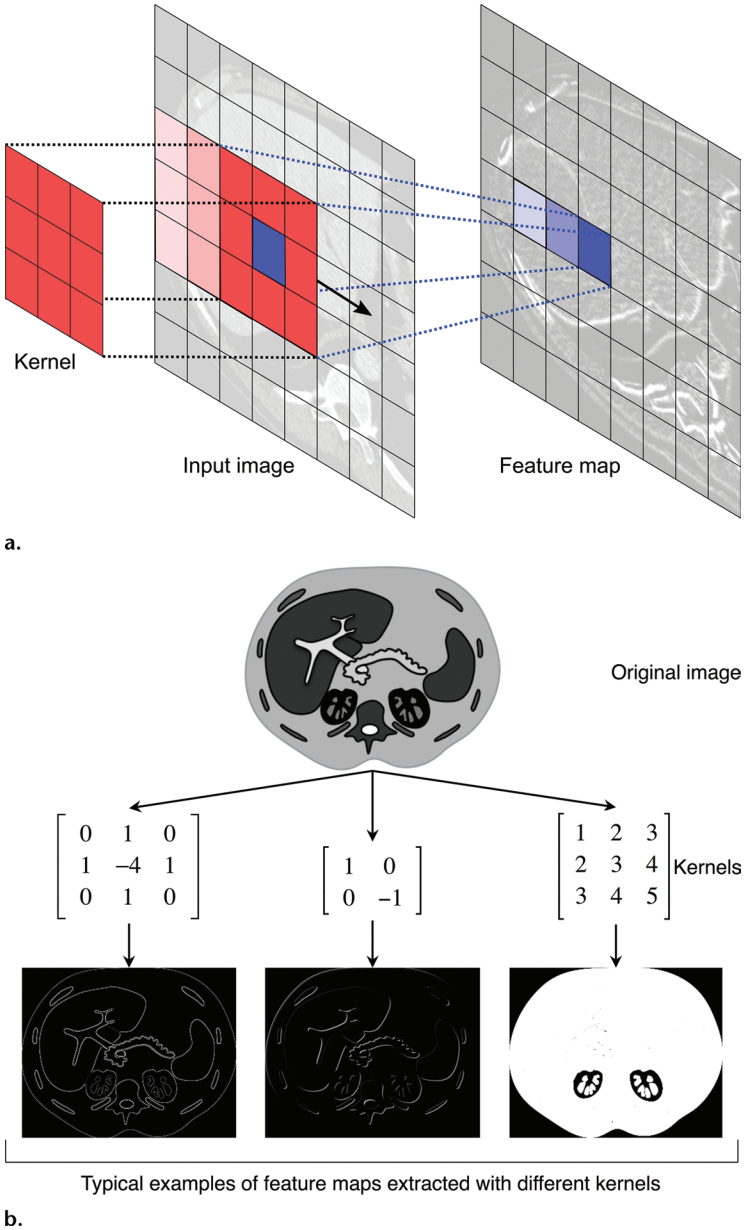


Figure 9. Convolutions. For each neuron to consider each pixel of a 512 × 512 image as input values to a neural network, an enormous amount of computer memory would be required. Instead, only the parameters of specialized filter operators, called convolutions, are learned. This mathematical operation describes the multiplication of local neighbors of a given pixel by a small array of learned parameters called a kernel. **(a)** Diagram shows the convolution of an image by a typical 3 × 3 kernel. The image pixels are multiplied by the nine values of a 3 × 3 kernel (red) and summed to produce the value of the blue pixel. This operation is repeated to cover the whole image. **(b)** By learning meaningful kernels, this operation mimics the extraction of visual features such as edges and corners, just like the visual cortex does. Convolutions are a key component of CNNs and their immense success in image processing tasks such as segmentation and classification.

Activation Function

A key component of deep neural networks is the activation function, a nonlinear function that is applied to the outputs of linear operations such as convolutions. Stacking these allows the input to be mapped to a representation that is linearly separable by a linear classifier. The activation function is inspired by observations regarding the basic behavior of biologic neurons. The role of an activation function in a neural network layer is typically that of a selection function, which allows some features to pass through to the output.

Historically, sigmoidal and hyperbolic tangent functions were used, as they were considered to be biologically plausible (18). Today, most CNNs now use a rectified linear unit (ReLU) in their hidden layers. This activation function is perfectly linear for positive inputs, passing them through unchanged, and blocks negative inputs (ie, evaluates to zero) (17,19,20).

Downsampling

An additional component of CNNs is the downsampling (or pooling) operation. This operation groups feature map activations into a lower-resolution feature map (Fig 10a). Downsampling increases the effective scope, or receptive field, of subsequent filters. Moreover, combined with convolutions, this operation also reduces the model's sensitivity to small shifts of the objects, since deeper layers rely increasingly on spatially low-resolution but contextually rich information. An added benefit of downsampling is the reduction of a model's memory footprint; for instance, the size of each feature map will decrease by four each time a 2 × 2 pooling operator is applied.

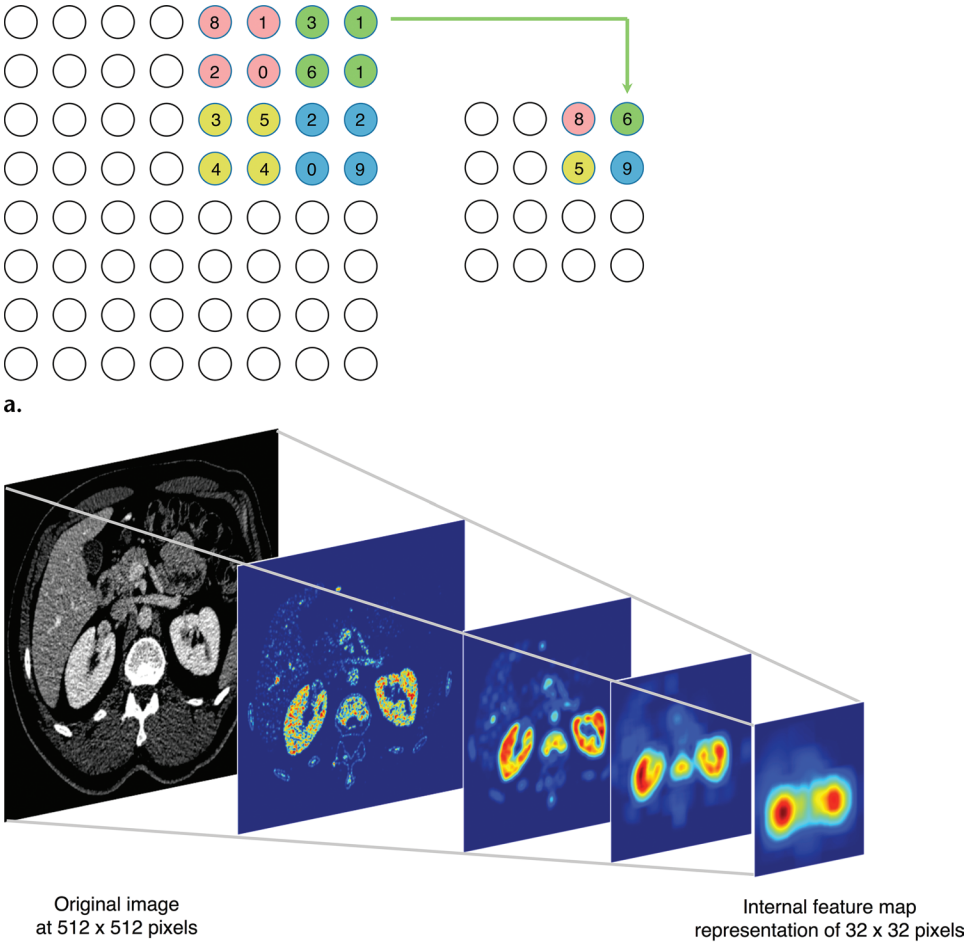


Figure 10. A CNN creates an internal representation of a hierarchy of visual features by stacking convolutional layers. To capture an increasingly larger field of view, features maps are progressively spatially reduced by downsampling images. **(a)** The max pooling layer, typically used to achieve downsampling, propagates only the maximum activation to the next layer. Subsequent convolutional layers become less sensible to small shifts or distortion of the target object in the extracted feature maps. **(b)** Downsampled representations of the kidneys from contrast-enhanced CT. This operation not only substantially reduces the memory requirements but also allows the network to be robust to the shape and position of the detected kidneys (ie, features of interest) in the images.

A common form of downsampling is max pooling, which propagates the maximum activation within a pooling region into a lower-resolution feature map. Successive pooling operations result in maps that have progressively lower resolution but represent increasingly richer information on the structure of interest (Fig 10b).

Convolutional Neural Networks

The first CNNs to employ back-propagation were used for handwritten digit recognition (21). These early CNNs were inspired by the Neocognitron, an early neural network architecture capable of visual pattern recognition by combining layers of simple cells and complex cells (22). The design of the Neocognitron drew its biologic inspiration from the work of Hubel and Wiesel (23), who described these two types of cells in the visual primary cortex, a discovery for which they were awarded the Nobel Prize in Physiology and Medicine in 1981.

CNNs can compose features consisting of incrementally larger spatial extent. Near the input, we need to care about only local features, captured by convolutional kernels; distant interactions between pixels appear weak. The same pattern occurs at every layer of representation in the model. In a CNN, the deeper the layer of representation, the coarser the characterization of the feature's spatial position (due to downsampling/pooling); thus, kernels in these deeper layers consider features over increasingly larger spatial scales.

Convolutional layers and activation functions transform the feature maps, while downsampling/pooling layers reduce the spatial resolution (Fig 11). In a typical network trained for classification, the coarse feature representation near the output of the network is typically

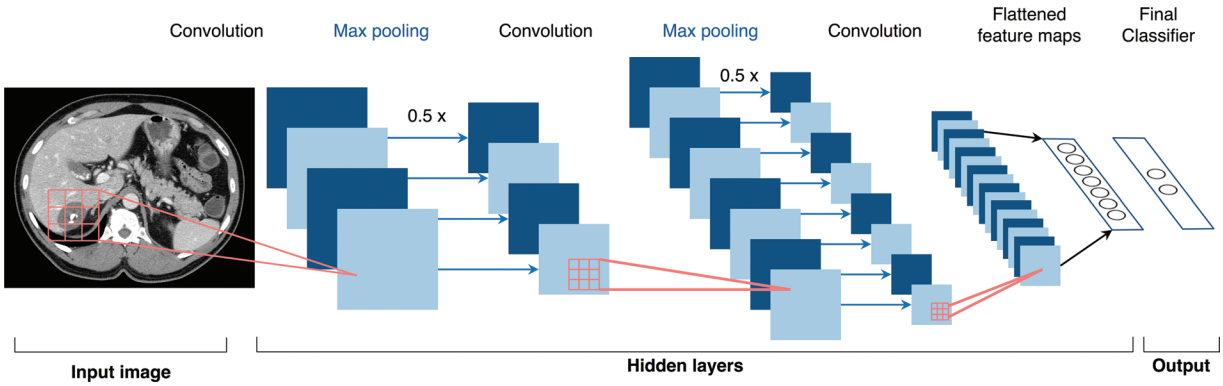


Figure 11. Integration of several concepts outlined in previous figures into a general diagram. Starting on the left, the input image is submitted to a series of convolutions with learned kernels, producing a stack of features maps containing low-level features such as edges and corners. These feature maps are then downsampled by a max pooling layer and further submitted to another set of learned convolutions, producing higher-level features such as parts of organs. Convolutions and max pooling layers can be stacked alternately until the network is deep enough to properly capture the structure of the image that is salient for the task at hand. Higher-level features are typically flattened into a single vector to perform the final classification or regression for the target task.

transformed into a vector form through fully connected layers. In a fully connected layer, each neuron is connected to all neurons in the previous layer. Fully connected layers allow reasoning about the entire content of the image. For classification, the output nodes of a neural network can be regarded as a vector of unnormalized log probabilities for each class. A softmax function at the final layer of a CNN can be used to normalize the output of a neural network so that it parameterizes a categorical distribution for class probabilities.

Neural networks have a reputation for being inscrutable "black boxes" due to their complexity and feature learning capability. However, certain strategies can be used to gain a better understanding of the underlying decision process of a trained CNN. For instance, one may gain insight into the role played by a given feature map by inspecting the associated receptive field in the image that caused the highest activation (Fig 12). For example, one may observe that low-level feature maps are active when their receptive field is positioned over various types of edges and corners, while midlevel feature maps are active on parts of organs and high-level feature maps encapsulate information about whole organs and large structures (24).

Alternatively, the last layer of the network before the final classification layer (pre-softmax layer) also provides insight on the CNN. The pre-softmax layer represents the whole image as a high-dimensional feature vector (eg, 4096-element feature vector). Since it is impossible to directly visualize such a high-dimensional vector, we apply dimensionality-reduction techniques to project the vectors into a two-dimensional (2D) space that we can easily visualize. A common dimensionality-reduction technique for this setting is *t*-stochastic neighbor embedding (*t*-SNE), which tends to preserve euclidean distances; that is, nearby vectors in the high-dimensional space are close to each other in the low-dimensional projection (Fig 13).

Training the Model

Data

To train a model, we need data. There are two general types of machine learning approaches that differ in the type of data that is needed to train them: supervised learning and unsupervised learning. With supervised learning, each example in the dataset is labeled. For instance, in a machine learning system for classifying renal tumors, a particular tumor image may be labeled "oncocytoma."

In unsupervised learning, the data examples are not labeled (ie, images are not annotated); instead, the model aims to cluster images into groups based on their inherent variability. The machine learning algorithm then tries to discover some structure in the data that might later be used to solve some task (eg, classification or segmentation of tumors). However, completely unsupervised learning is an open-ended research problem for which achieving good results remains difficult in practice. A hybrid approach called semisupervised learning makes use of a large quantity of unlabeled data combined with a usually small number of labeled data examples (26).

The proper dataset size for adequately training deep learning models is variable and depends on the nature and complexity of the task. Deep learning methods scale well with the quantity of data and can often leverage extremely large datasets for good performance. Although a large quantity of data is desirable, obtaining high-quality labeled data can be costly and time-consum-

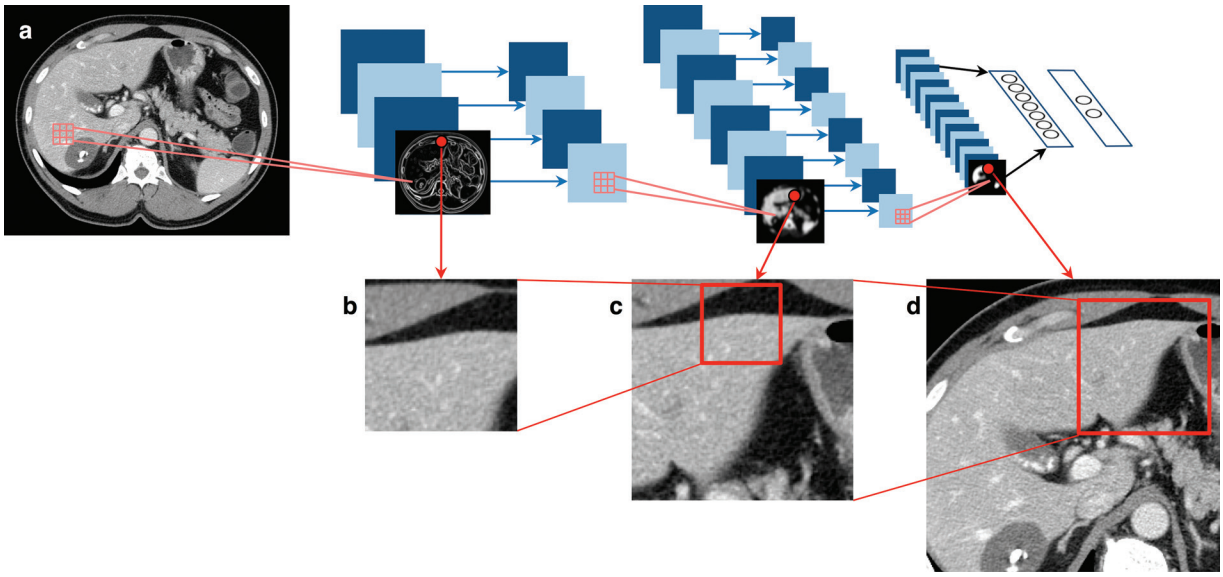


Figure 12. Stacking multiple convolutional and max pooling layers allows the model to learn a hierarchy of feature representations. While neurons close to the input image *(a)* will be activated by the presence of edges and corners formed by a few pixels, neurons located deeper in the network will be activated by combinations of edges and corners that represent characteristic parts of organs and eventually whole organs. At each successive level of representation, neurons gain a larger receptive field in the input image, as seen in *b*, *c*, and *d*. The final classification task thus relies on a rich set of hidden features that represent a large receptive field and integrate multiscale information in a meaningful way.

ing. For instance, completing a 20-minute image segmentation task on 1000 cases may require two experts to work full-time for 1 month.

The Table shows some examples of datasets used to train deep learning models in both the computer vision community and the medical imaging community; computer vision datasets are orders of magnitude larger than the medical imaging datasets. In some cases, the dataset acquisition costs can be reduced by crowd-sourcing, but relying entirely on outsourced labels may be problematic. Crowd-sourcing was investigated in the setting of mitotic activity detection on histologic slides of breast cancer cells (33).

Having small amounts of good-quality data is certainly better than having no data at all. Data augmentation can be used to artificially enlarge the size of a small dataset. The idea is to apply random transformations to the data that do not change the appropriateness of the label assignments. Possible random transformations that can be applied to images include flipping, rotation, translation, zooming, skewing, and elastic deformation. Hence, with data augmentation, image variants from an original dataset are created to enlarge the size of a training dataset of images presented to the deep learning models (34).

It is standard practice in machine learning to divide available data into three subsets: a training set, a validation set, and a test set. The training set is used to train and optimize the parameters of the neural network. This training involves repeatedly running training images through it and using the errors to adjust the weights of the network connections. The validation set is used to monitor the performance of the model during the training process; this dataset should also be used to perform model selection. The validation data are the best proxy we can have about the model performance on the test set. Once all the parameters of the model are fixed, we can measure its performance on the test set. This set is used only at the very end of a study to report the final model performance.

Learning

Designing neural network architectures requires consideration of numerous parameters that are not learned by the model (hyperparameters), such as the network topology, the number of filters at each layer, and the optimization parameters. Hyperparameters are typically selected through random search, a lengthy process where each configuration is instantiated and trained to establish which architecture performs best (35).

Once we have a proper dataset and a neural network architecture, we can proceed to learning the model parameters. An important machine learning pitfall is overfitting, where a model learns idiosyncratic statistical variations of the training set rather than generalizable patterns for a particular problem. Overfitting is usually detected by analysis of model accuracy on the training and validation sets (Fig 14). If the model performs well on the training set and poorly on the validation set, we say that the model has overfit the training data.

If we evaluate the model extensively on the validation data, we can overfit both the training and validation data (ie, the model performs well on the training and validation sets but poorly on the test set). A model overfits because it has too many parameters and can memorize the training data at the expense of being able to generalize to new data. If the model is overfitting, we should consider reducing its capacity or flexibility (eg, reducing the number of parameters) or adding more data (eg, applying more aggressive data augmentation).

Technical Requirements

Hardware

End-to-end training of a modern deep learning model typically requires a great deal of computation. The success of deep CNNs was made possible by the development of inexpensive parallel computing hardware in the form of graphics processing units (GPUs). While GPUs were initially created for computer gaming, they have demonstrated their utility as a flexible hardware for general-purpose parallel computation and are typically considered essential for training large modern deep neural networks in a reasonable amount of time. The speedup in performance over using conventional central processing units is typically 10 times to 40 times, allowing complex models consisting of tens of millions of parameters to be trained in a few days as opposed to weeks or months.

Software

Many software frameworks are now available for constructing and training multilayer neural networks (including convolutional networks). Frameworks such as Theano, Torch, TensorFlow, CNTK, Caffe, and Keras implement efficient low-level functions from which developers can describe neural network architectures with very few lines of code, allowing them to focus on higher-level architectural issues (36–40).

These libraries also allow researchers to efficiently tap into available computing resources such as GPUs. Most of these software tools are free and open-source, meaning that anyone can inspect and contribute to their codebase. By freely sharing code, models, data, and publications, the academic and industrial research communities are collaborating on machine learning problems at an accelerating pace.

Clinical Applications of Deep Learning

Deep learning has demonstrated impressive performance on tasks related to natural images (ie, photographs). This technique was quickly adopted by the medical image processing community. A recently published survey revealed more than 300 applications of deep learning to medical images—most of which were published over the past year—from different imaging modalities (radiography, CT, MR imaging) (41).

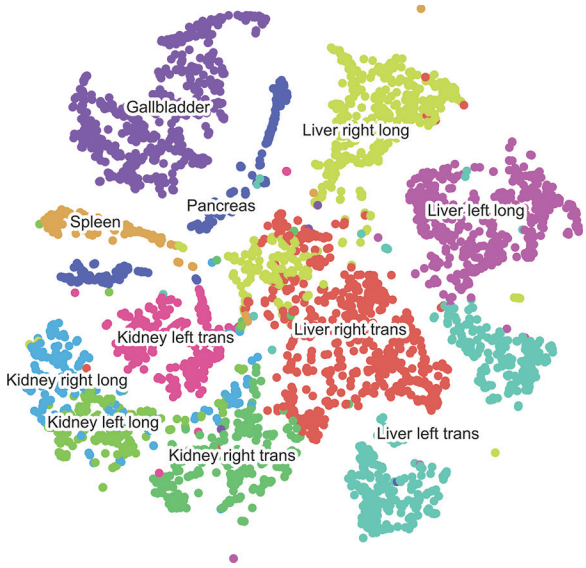


Figure 13. *t*-SNE visualization. Map shows the distribution of the 4096-element vectors to which the training cases of ultrasonographic (US) images with organ labels were mapped. Areas of overlap correspond to potential areas of classification confusion. For instance, there is significant overlap between longitudinal *(long)* views of the left and right kidney and between longitudinal and transverse *(trans)* views of the right hepatic lobe. Maps like these provide insight into the performance of the neural network classification (25).

This section focuses on recent applications of deep learning for classification, segmentation, and detection tasks. For each task, we provide a brief description, describe the training pipeline, summarize evaluation metrics, and provide examples of clinical applications.

Classification

Description.—Classification tasks in radiology typically consist of predicting some target class (eg, lesion category or condition) at the patient level from an image or region of interest. This task encompasses a wide range of applications, from determining the presence or absence of a disease to identifying the type of malignancy.

Training Pipeline.—Using deep learning, these tasks are commonly solved using CNNs. After forward propagation of input images, the softmax layer will produce a vector of class probabilities from which the highest value represents the predicted class.

Compared with traditional computer vision and machine learning algorithms, deep learning algorithms are data hungry. One of the

Example Datasets Used for Natural Image Processing and Medical Image Processing Tasks

Image Database*	Sample Size	Type of Data	Task
ImageNet (27)	1.4 million images	Natural images (eg, plants, flowers, animals, vehicles)	Classification of objects in natural images, 1000 classes
MNIST (28)	70 000 thumbnail images (60 000 training, 10 000 testing)	Handwritten digits	Classification between 0 and 9
Pascal VOC 2012 (29)	~10 000 images	Natural images	Segmentation of objects in natural images, 20 classes
YouTube-8M (30)	8 million videos	YouTube video URLs	Classification of video topics, 4800 classes
BRATS 2016 (31)	300 cases, four raters	Brain MR images	Segmentation of brain tumors on MR images (T1W, T1W CE, T2W, T2W FLAIR)
EM ISBI 2012 (32)	30 slices for training, 30 slices for testing	Electron micrographs	Segmentation of neurons
Shin et al (10)	Radiology reports from 780 000 imaging examinations, 216 000 annotated images	Radiology images and reports (private)	Correlate high-level topics and presence of common diseases with medical images

Note.—The size of datasets required to train a model is specific to every task but can amount to a large quantity of labeled data. CE = contrast-enhanced, FLAIR = fluid-attenuated inversion-recovery, T1W = T1-weighted, T2W = T2-weighted, URL = uniform resource locator.
*Numbers in parentheses are references.

main challenges faced by the community is the scarcity of labeled medical imaging datasets. While millions of natural images can be tagged using crowd-sourcing (27), acquiring accurately labeled medical images is complex and expensive. Further, assembling balanced and representative training datasets can be daunting given the wide spectrum of pathologic conditions encountered in clinical practice.

To manage the scarcity of labeled images, a common strategy is to pretrain a CNN first on a task for which there is a sufficient amount of data available, a technique called transfer learning (Fig 15). Since models pretrained on the popular ImageNet challenge dataset are now widely available, many authors have achieved good performances by reusing pretrained generic architectures and fine-tuning the final layers of the network to fit a relatively small and specialized dataset (42).

Evaluation Metrics.—The performance of these models is typically assessed using accuracy: the ratio of correctly predicted samples over all predictions. For tasks comprising multiple target classes, it is common in image classification competitions to report the top-five accuracy, a similar metric that assesses whether the correct label belongs to the five highest predicted class probabilities (16). One way to visualize the performance of the neural network is to generate a confusion matrix reporting predicted and true labels.

Applications.—Large-scale mining of a picture archiving and communication system (PACS) and a radiology information system (RIS) has been performed at the National Institutes of Health using a deep learning system to determine the semantic associations between images and reports. The researchers used a combination of unsupervised and supervised learning to train their system on 216 000 2D key images selected by radiologists for diagnostic reference. Transfer learning with natural images from ImageNet to medical imaging modalities (mostly CT and MR imaging) increased image classification performance.

Given new images from patient data acquisitions, the system was able to predict semantic labels (topics and key words) pertaining to the content of the images (10), with top-one and top-five accuracy values of 61%–66% and 93%–95%, respectively. Predictions at this level of semantic precision are likely not yet ready for integration into clinical practice; however, these directions show great promise for the future.

Segmentation

Description.—Segmentation can be defined as the identification of pixels or voxels composing an organ or structure of interest. From a machine learning perspective, it can be considered as a pixel-level classification task, where we aim to define whether a given pixel belongs to the

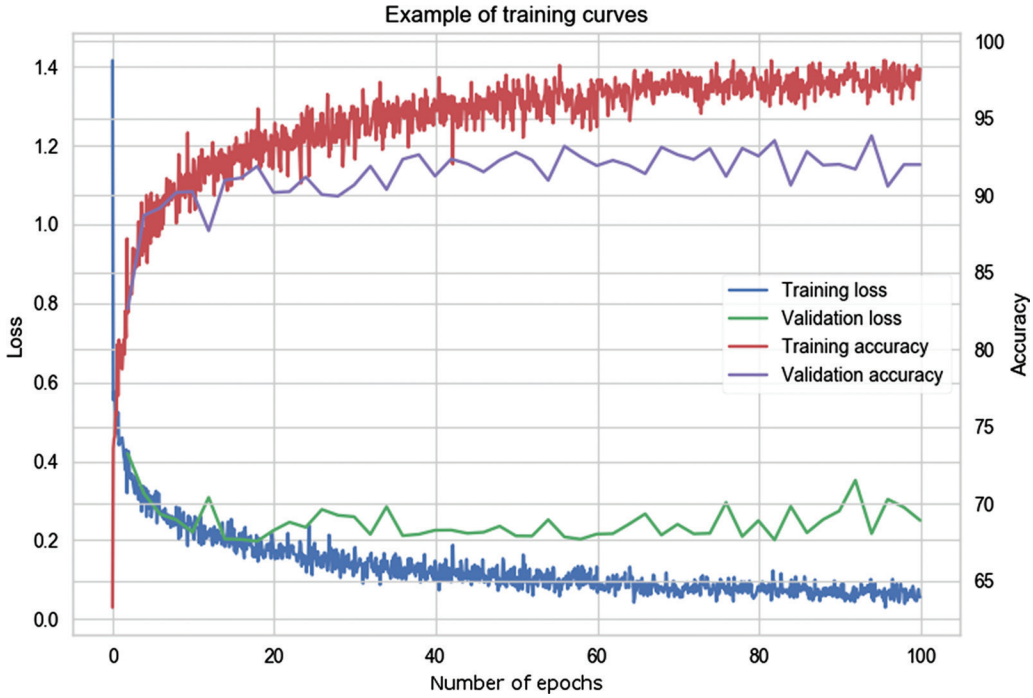


Figure 14. Training curves. Training a neural network involves repeatedly computing the forward propagation of batches of training images and back-propagating the loss to adjust the weights of the network connections. We can monitor the progress of training by plotting the training loss for each batch, which decreases toward zero, and the training accuracy, which increases toward 100%. It is also customary to evaluate the loss and the accuracy on the validation set every time the network runs through the entire training dataset (every epoch). The training of a neural network will typically be halted once the validation accuracy has not improved for a given number of epochs (eg, five epochs). The state of the model that yielded the best performance on the validation set is then used to compute the final results on a separate test set.

background or to a target class (eg, prostate, liver, lesions). Image masks resulting from this classification can subsequently be used to perform various quantitative analyses such as virtual surgery planning, radiation therapy planning, or quantitative lesion follow-up.

Training Pipeline.—There are two deep learning approaches to image segmentation. The first is a patch-based approach where the center pixel of a patch is classified; the whole segmentation map can be obtained by applying the model in a sliding-window fashion over the whole image and progressively building the output segmentation mask by segmenting the central pixel of each patch. This simple approach requires many model evaluations to obtain a segmentation map for a single image and thus is computationally inefficient.

A second approach is based on a CNN that directly produces a full-resolution segmentation output (Fig 16). While CNNs typically consist of a contracting path composed of convolutional, downsampling, and fully connected layers, in this segmentation model the fully connected layers are replaced by an expanding path, which also recovers the spatial information lost during the downsampling operations. In most cases, the expanding path is built with *(a)* upsampling operations, responsible for increasing the spatial resolution of feature maps, and *(b)* skip connections, used to pass the information from the contracting path of the network (bypassing the deeper layers). This architecture is known as the U-net (because of its U-shaped architecture) and is currently broadly used in medical image segmentation approaches (43,44).

Evaluation Metrics.—Pixel-wise classification accuracy is a poor metric for segmentation, since most pixels in an image do not contain the target class (eg, liver). Moreover, it does not account for the variability in size of different lesions and does not faithfully reflect the segmentation quality. Therefore, most authors will report metrics based on the intersection over union of the segmentation mask against a ground-truth segmentation mask created by an expert. In the medical field, the Dice score is commonly used as an evaluation metric and takes a value of 0 when both masks do not overlap at all and 1 for a perfect overlap.

Applications.—Automated liver and tumor segmentation from CT and MR imaging has been reported using cascaded fully convolutional CNNs. The first network segmented the liver, and the second network segmented lesions within

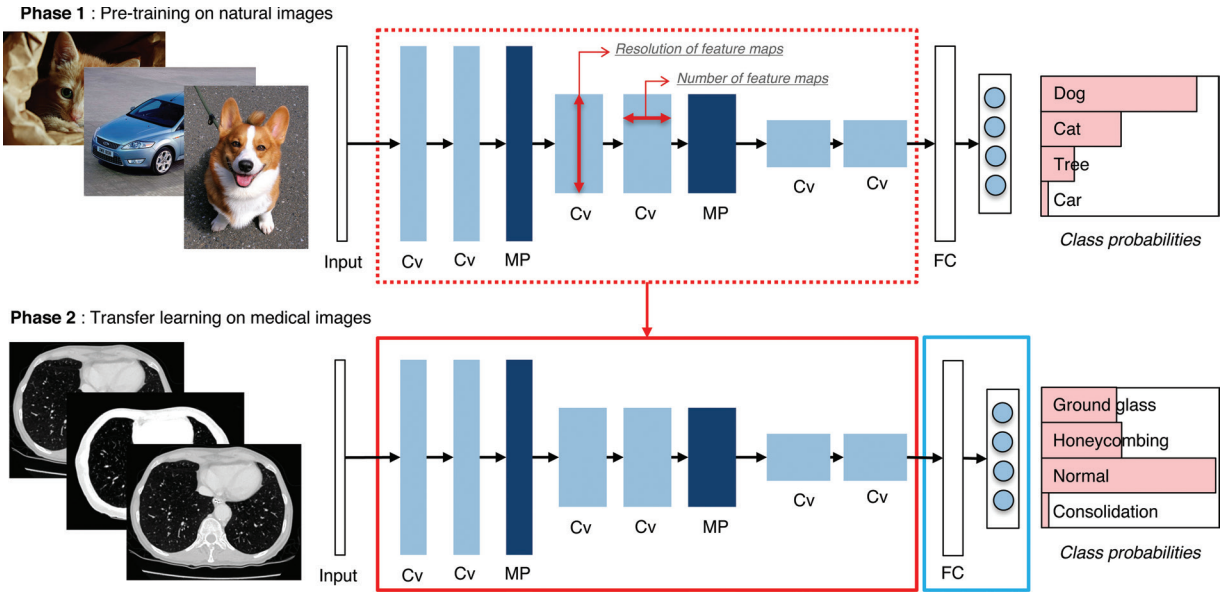


Figure 15. Transfer learning. Training CNNs for medical images can be challenging owing to the relative lack of large labeled medical image datasets for training and testing. One approach to solve this problem is transfer learning, where a network is initialized using weights derived from training on a large dataset; only a portion of the network (typically the final fully connected *[FC]* layers, outlined in light blue) needs to be retrained for a new smaller dataset. The underlying assumption is that basic image features may be shared among seemingly disparate datasets. The height and width of blue boxes respectively represent the resolution and number of feature maps resulting from the current layer operation. *Cv* = convolution, *MP* = max pooling.

the liver. Training was performed on 100 CT examinations. Dice scores over 94% were reported for the liver segmentations. Remarkably, the model was sufficiently robust to work on 38 MR imaging examinations (45).

Automated prostate segmentation from MR images using three-dimensional (3D) CNNs has been reported. Training was based on 50 MR imaging examinations from the Prostate MR Image Segmentation challenge dataset (46). An architecture using long and short residual connections improved the segmentation performance, achieving Dice scores of 81%–87%.

Detection

Description.—Detection of focal lesions such as lung nodules, hepatic lesions, or colon polyps is a prerequisite before characterization by a radiologist. While classification tasks aim to predict labels, detection tasks aim to predict the location of potential lesions, often in the form of points, regions, or bounding boxes of interest. From a machine learning perspective, these three types of outputs give rise to three different approaches to detection.

It can be viewed as a classification task, where a preidentified set of candidate patches sampled around points of interest are further classified as positive (eg, malignant lesion) or negative (benign lesion, normal parenchyma) samples. As seen earlier, it can be directly considered as a segmentation task, in which detection becomes implicit as individual connected areas of the resulting mask are considered detected samples. Finally, it can be considered as a regression task, where the coordinates of bounding boxes outlining target objects are directly inferred from the input image, a technique broadly applied for natural images (47). In this section, we focus on the first approach.

Training Pipeline.—When classifying voxels in a volume for detection or segmentation, a common challenge is that the target class tends to have relatively few examples, whereas the background class tends to be more numerous and more variable. A common strategy to train a CNN for detection in this setting is to generate a surrogate dataset based on small patches extracted from the original images. Patches will typically be sampled in equal number from the target class and the background class, providing a simple mechanism to mitigate the class imbalance naturally occurring in detection tasks.

A CNN is then trained on this patch dataset as if it were a classification task. It can subsequently be applied *(a)* in a sliding-window fashion across an input image or *(b)* on a subset of preselected image patches previously obtained with a sensitive candidate selection method (Fig 17). By casting the detection task as a classification one, pretrained architectures can again be leveraged to achieve good performances with small datasets.

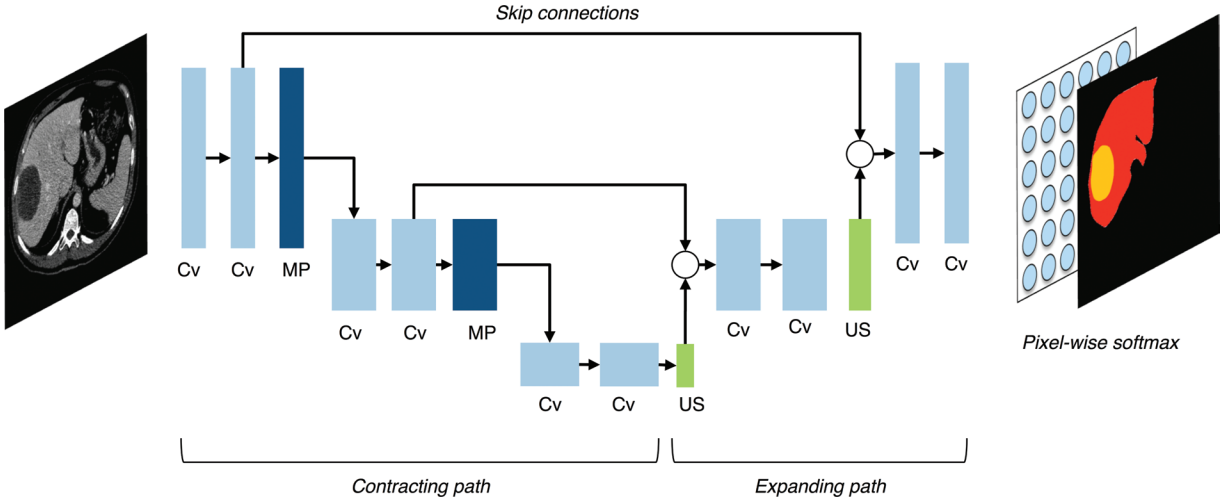


Figure 16. Lesion segmentation. A particular architecture of CNNs called the U-net is designed to output complete segmentation masks from input images. By expanding the max pooling *(MP)* layers with corresponding upsampling *(US)* layers, the output dimension of the final classification layer matches the dimension of the original input image. Skip connections connect the contracting path (left) with the expanding path (right). *Cv* = convolution.

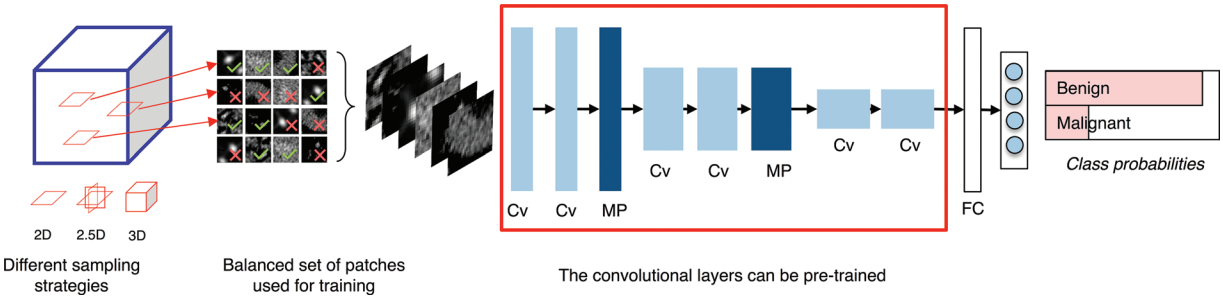


Figure 17. Training with patches. Detection tasks in the medical field are commonly solved by training convolutional networks on a surrogate dataset composed of small patches extracted from the original images. For volumetric modalities, different sampling strategies can be used to integrate 3D contextual information, such as using 3D patches or cross-like 2.5D patches. Patches are typically sampled in equal number from both classes to mitigate the class imbalance naturally occurring in detection tasks. Just as for classification, the CNN can be pretrained on an existing database and fine-tuned for the target application. *Cv* = convolution, *FC* = fully connected, *MP* = max pooling.

Evaluation Metrics.—In this setting, reporting performance on the basis of accuracy is not very informative, since most of the image contains normal tissue (true negative), which is likely to overshadow missed lesions (false negative). It is therefore common to report a combination of evaluation metrics that do not account for true negatives, such as sensitivity (also known as true-positive rate), positive predictive value (also known as precision in computer science), F score, and average false-positive per patient. To explore various operating point trade-offs of the trained model, it is also common to report—instead of the traditional receiver operating characteristic (ROC) curve—the free-response ROC curve (FROC) (48), which plots the lesion localization ratio against the nonlesion detection ratio for various cutoff points.

Applications.—Automated detection of malignant lesions on screening mammograms using deep CNNs has been reported. The training dataset included 44 090 mammographic images obtained as part of a screening program (6). The area under the ROC curve (AUC) was 0.93 for the CNN, 0.91 for the reference CAD (computer-aided diagnosis) system, and 0.84–0.88 for three human readers. Another team reported similar findings, with AUCs of 0.82 for a deep CNN and 0.77–0.87 for radiologists, although the radiologists' results were consistently less sensitive and more specific than those of the neural network (5).

Automated detection of cerebral microbleeds on susceptibility-weighted MR images using a cascade of two CNNs has been reported. The first one, a fully convolutional network, was aimed at providing a probability map of candidate locations with very high sensitivity. This high sensitivity was achieved by continually training on samples classified as false negative. The second one, a 3D CNN trained solely on a balanced subset of extracted 3D patches and false-positive samples (eg, flow voids, calcifications, cavernous

malformations), was able to achieve high specificity. Using this two-stage strategy to manage class imbalance yielded a system able to achieve a sensitivity of 93% while maintaining an average of 2.74 false positives per subject (49).

Various detection tasks have been performed with CNNs, such as coronary calcification detection with gated CT angiography (50), lung nodule detection with CT (51), and lymph node detection (42). To integrate 3D contextual information when working with volumetric modalities, patches sampled from different anatomic orientation planes can be aggregated and used as multichannel inputs (42).

Limitations of Deep Learning

Despite the variety of recent successes of deep learning, there are limitations in the application of the technique. First, deep learning is not the optimal machine learning technique for all data analysis problems. For problems in which data are well structured or optimal features are well-defined, other simpler machine learning methods such as logistic regression, support vector machines, and random forests are typically easier to apply and more effective (52).

Even in computer vision, where CNNs have become a dominant method, there are important limitations for deep learning. The most prominent limitation is that deep learning is an intensely data-hungry technology; learning weights for a large network from scratch requires a very large number of labeled examples to achieve accurate classification. However, unlike traditional approaches to computer vision and machine learning, which do not scale well with dataset size, deep learning does scale well with large datasets.

As noted earlier, transfer learning has recently received research attention as a potentially effective way of mitigating the data requirements. However, current projects applying transfer learning typically reuse weights from networks trained on ImageNet, a large labeled collection of relatively low-resolution 2D color photographs. Applications in radiology would be expected to process higher-resolution volumetric images with higher bit depths, for which pretrained networks are not yet readily available.

As a result, building large labeled public medical image datasets is an important step for further progress in applying deep learning to radiology. Barriers to this effort include privacy concerns for clinical images, as well as the costs and difficulties of obtaining accurate ground-truth labels from multiple experts or pathology diagnoses. Nevertheless, several efforts are under way to create large datasets of labeled medical images, such as the Cancer Imaging Archive (53).

The creation of these large databases of labeled medical images and many associated challenges (54) will be fundamental to foster future research in deep learning applied to medical images. It is also possible that further breakthroughs in deep learning methods can significantly reduce the data requirements for training deep learning systems; after all, humans do not require nearly as many labeled examples as current deep learning systems to learn to perform accurate image classification and interpretation.

Another limitation of deep learning systems is that they are relatively opaque compared with other machine learning methods. Despite the useful application of visualization techniques involving deconvolution and dimensionality reduction, it remains difficult to clearly define what different parts of a large network do, making it challenging to delineate limitations in the network or debug errors in image interpretation without a large comprehensive set of test examples.

One could argue that an accurate opaque system is preferable to an inaccurate transparent one, and that a human expert's image analysis can similarly be relatively opaque to a nonexpert. Nevertheless, it is currently much easier to interrogate a human expert's thought process than to decipher the inner workings of a deep neural network with millions of weights. Furthermore, an automated system's ability to clearly justify its analysis would be highly desirable for it to become widely acceptable for making critical judgments regarding patients' health.

Deep learning systems currently excel in emulating the kind of human judgment that is based purely on pattern recognition, where the most informative patterns can be discerned from previous training. However, no finite training set can fully represent the variety of cases that might be seen in clinical practice. More complex radiology interpretation problems typically require deductive reasoning using knowledge of pathologic processes and selective integration of information from prior examinations or the patient's health record. It is presently not clear how to train a deep learning system to emulate these more complex thought processes.

Future Directions

The role of deep learning and its application to the practice of radiology must still be defined. Deep learning systems may be conceived as a new form of diagnostic test with various clinical usage scenarios (55). A triage approach would run these automated image analysis systems in the background to detect life-threatening conditions or search through large amounts of clinical, genomic, or imaging data (56). A replacement approach would

use these systems for generating figure captions (57) or even fully automated interpretation of imaging examinations (56). An add-on approach would support the radiologist by performing time-consuming tasks such as lesion segmentation to assess total tumor burden (45).

Conclusion

Artificial neural networks have been used in artificial intelligence since the 1950s. Advances in training techniques and network architectures, combined with the recent availability of large amounts of labeled data and powerful parallel computing hardware, have enabled rapid development of deep learning algorithms. Deep learning is a powerful and generic artificial intelligence technique that can solve image detection, recognition, and classification tasks that previously required human intelligence. The introduction of deep learning techniques in radiology will likely assist radiologists in a variety of diagnostic tasks. Familiarity with the concepts, strengths, and limitations of computer-assisted techniques based on deep learning is critical to ensure optimal patient care.

***Disclosures of Conflicts of Interest.*—G.C.** *Activities related to the present article:* grant from Imagia Cybernetics. *Activities not related to the present article:* disclosed no relevant relationships. *Other activities:* disclosed no relevant relationships. **E.V.** *Activities related to the present article:* grant from Imagia Cybernetics. *Activities not related to the present article:* disclosed no relevant relationships. *Other activities:* disclosed no relevant relationships. **C.J.P.** *Activities related to the present article:* grant from Imagia Cybernetics. *Activities not related to the present article:* disclosed no relevant relationships. *Other activities:* disclosed no relevant relationships. **A.T.** *Activities related to the present article:* grant from Imagia Cybernetics. *Activities not related to the present article:* disclosed no relevant relationships. *Other activities:* disclosed no relevant relationships.

References

1. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521(7553):436–444.
2. Silver D, Huang A, Maddison CJ, et al. Mastering the game of Go with deep neural networks and tree search. Nature 2016;529(7587):484–489.
3. Hinton G. Geoff Hinton: on radiology. https://www.youtube.com/watch?v=2HMPRXstSvQ. Published 2016. Accessed March 19, 2017.
4. Erickson BJ. Deep learning: how it will change everything. https://siim.org/page/web16_deep_learning. Accessed August 17, 2016.
5. Becker AS, Marcon M, Ghafoor S, Wurnig MC, Frauenfelder T, Boss A. Deep learning in mammography: diagnostic accuracy of a multipurpose image analysis software in the detection of breast cancer. Invest Radiol 2017;52(7):434–440.
6. Kooi T, Litjens G, van Ginneken B, et al. Large scale deep learning for computer aided detection of mammographic lesions. Med Image Anal 2017;35:303–312.
7. Drozdzal M, Chartrand G, Vorontsov E, et al. Learning normalized inputs for iterative estimation in medical image segmentation. arXiv:170205174. https://arxiv.org/abs/1702.05174. Published February 16, 2017. Accessed October 2, 2017.
8. Havaei M, Guizard N, Larochelle H, Jodoin PM. Deep learning trends for focal brain pathology segmentation in MRI. In: Holzinger A, ed. Machine learning for health informatics. Lecture Notes in Computer Science, vol 9605. Cham, Switzerland: Springer International, 2016; 125–148.
9. Anthimopoulos M, Christodoulidis S, Ebner L, Christe A, Mougiakakou S. Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. IEEE Trans Med Imaging 2016;35(5):1207–1216.
10. Shin HC, Kim L, Seff A, Yao J, Summers RM. Interleaved text/image deep mining on a large-scale radiology database for automated image interpretation. J Mach Learn Res 2016;17(107):1–31.
11. Pons E, Braun LM, Hunink MG, Kors JA. Natural language processing in radiology: a systematic review. Radiology 2016;279(2):329–343.
12. Goodfellow I, Bengio Y, Courville A. Deep learning. Cambridge, Mass: MIT Press, 2016; 1–26.
13. Erickson BJ, Korfiatis P, Akkus Z, Kline TL. Machine learning for medical imaging. RadioGraphics 2017;37(2):505–515.
14. Rosenblatt F. The perceptron: a probabilistic model for information storage and organization in the brain. Psychol Rev 1958;65(6):386–408.
15. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. Cognitive Modeling 1988;5(3):1.
16. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, December 3–6, 2012. Red Hook, NY: Curran Associates, 2012; 1097–1105.
17. He K, Zhang X, Ren S, Sun J. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. ArXiv e-prints [serial online]; vol 1502. http://adsabs.harvard.edu/abs/2015arXiv150201852H. Published February 2015. Accessed October 2, 2017.
18. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9(8):1735–1780.
19. Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. Proceedings of the 14th International Conference on Artificial Intelligence and Statistics. Journal of Machine Learning Research 2011;15:315–323. http://proceedings.mlr.press/v15/glorot11a.html. Accessed October 2, 2017.
20. Maas AL, Hannun AY, Ng AY. Rectifier nonlinearities improve neural network acoustic models. In: Proceedings of the 30th International Conference on Machine Learning 2013;30(1). https://web.stanford.edu/~awni/papers/relu_hybrid_icml2013_final.pdf. Accessed October 2, 2017.
21. LeCun Y, Boser B, Denker J, et al. Handwritten digit recognition with a back-propagation network. In: Neural information processing systems 2. San Francisco, Calif: Morgan Kaufmann, 1990; 396–404.
22. Fukushima K. Neocognitron: a hierarchical neural network capable of visual pattern recognition. Neural Netw 1988;1(2):119–130.
23. Hubel DH, Wiesel TN. Receptive fields of single neurones in the cat's striate cortex. J Physiol (Paris) 1959;148:574–591.
24. Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. In: European Conference on Computer Vision. Cham, Switzerland: Springer International, 2014; 818–833.
25. Cheng PM, Malhi HS. Transfer learning with convolutional neural networks for classification of abdominal ultrasound images. J Digit Imaging 2017;30(2):234–243.
26. Kingma D, Rezende D, Mohamed S, Welling M. Semi-supervised learning with deep generative models. Adv Neural Inf Proc Syst 2014;27:3581–3589.
27. Jia D, Wei D, Socher R, Li-Jia L, Kai L, Li FF. ImageNet: a large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009. New York, NY: Institute of Electrical and Electronics Engineers, 2009; 248–255.
28. LeCun Y, Cortes C, Burges CJ. The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/. Published 1998. Accessed October 2, 2017.
29. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A. The Pascal visual object classes (voc) challenge. Int J Comput Vis 2010;88(2):303–338.

30. Abu-El-Haija S, Kothari N, Lee J, et al. YouTube-8m: a large-scale video classification benchmark. arXiv preprint arXiv:160908675. https://arxiv.org/abs/1609.08675. Published September 27, 2016. Accessed October 2, 2017.
31. Menze BH, Jakab A, Bauer S, et al. The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE Trans Med Imaging 2015;34(10):1993–2024.
32. Cardona A, Saalfeld S, Preibisch S, et al. An integrated micro- and macroarchitectural analysis of the Drosophila brain by computer-assisted serial section electron microscopy. PLoS Biol 2010;8(10):e1000502.
33. Albarqouni S, Baur C, Achilles F, Belagiannis V, Demirci S, Navab N. AggNet: deep learning from crowds for mitosis detection in breast cancer histology images. IEEE Trans Med Imaging 2016;35(5):1313–1321.
34. Roth HR, Lu L, Liu J, et al. Improving computer-aided detection using convolutional neural networks and random view aggregation. IEEE Trans Med Imaging 2016;35(5):1170–1181.
35. Bergstra J, Bengio Y. Random search for hyper-parameter optimization. J Mach Learn Res 2012;13(1):281–305.
36. Jia Y, Shelhamer E, Donahue J, et al. Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM International Conference on Multimedia. New York, NY: Association for Computing Machinery, 2014; 675–678.
37. Abadi M, Agarwal A, Barham P, et al. TensorFlow: large-scale machine learning on heterogeneous distributed systems. Preliminary white paper, November 9, 2015. https://www.tensorflow.org/about/bib. Accessed October 2, 2017.
38. Collobert R, Kavukcuoglu K, Farabet C. Torch7: a Matlab-like environment for machine learning. In: BigLearn NIPS Workshop 2011 (No. EPFL-CONF-192376). https://infoscience.epfl.ch/record/192376. Accessed October 2, 2017.
39. Bergstra J, Breuleux O, Bastien F, et al. Theano: a CPU and GPU math compiler in Python. In: Proceedings of the Python for Scientific Computing Conference, 2010; 1–7. https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.691.4524. Accessed October 2, 2017.
40. Theano Development Team. Theano: a Python framework for fast computation of mathematical expressions. arXiv e-prints. abs/1605.02688. https://arxiv.org/abs/1605.02688. Published May 9, 2016. Accessed October 2, 2017.
41. Litjens G, Kooi T, Bejnordi BE, et al. A survey on deep learning in medical image analysis. arXiv preprint arXiv:170205747. https://arxiv.org/abs/1702.05747. Published February 19, 2017. Accessed October 2, 2017.
42. Shin HC, Roth HR, Gao M, et al. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE Trans Med Imaging 2016;35(5):1285–1298.
43. Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015. New York, NY: Institute of Electrical and Electronics Engineers, 2015; 3431–3440.
44. Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation. arXiv:1505.04597v1. https://arxiv.org/abs/1505.04597. Published May 18, 2015. Accessed October 2, 2017.
45. Christ PF, Ettlinger F, Grün F, et al. Automatic liver and tumor segmentation of CT and MRI volumes using cascaded fully convolutional neural networks. arXiv preprint arXiv:1702.05970. https://arxiv.org/abs/1702.05970. Published February 20, 2017. Accessed October 2, 2017.
46. Yu L, Yang X, Chen H, Qin J, Heng PA. Volumetric convnets with mixed residual connections for automated prostate segmentation from 3D MR images. https://appsrv.cse.cuhk.edu.hk/~lqyu/papers/AAAI17_Prostate.pdf. Published 2017. Accessed October 2, 2017.
47. Redmon J, Farhadi A. YOLO9000: better, faster, stronger. arXiv preprint arXiv:161208242. https://arxiv.org/abs/1612.08242. Published December 25, 2016. Accessed October 2, 2017.
48. Chakraborty DP. A brief history of free-response receiver operating characteristic paradigm data analysis. Acad Radiol 2013;20(7):915–919.
49. Qi Dou, Hao Chen, Lequan Yu, et al. Automatic detection of cerebral microbleeds from MR images via 3D convolutional neural networks. IEEE Trans Med Imaging 2016;35(5):1182–1195.
50. Wolterink JM, Leiner T, de Vos BD, van Hamersvelt RW, Viergever MA, Išgum I. Automatic coronary artery calcium scoring in cardiac CT angiography using paired convolutional neural networks. Med Image Anal 2016;34:123–136.
51. Setio AA, Ciompi F, Litjens G, et al. Pulmonary nodule detection in CT images: false positive reduction using multi-view convolutional networks. IEEE Trans Med Imaging 2016;35(5):1160–1169.
52. Domingos P. Plenary panel: is deep learning the new 42? https://youtu.be/furfdqtdAvc?t=21m35s. Published September 1, 2016. Accessed March 27, 2017.
53. Cancer Imaging Archive. https://www.cancerimagingarchive.net/. Accessed March 26, 2017.
54. Grand challenges in biomedical image analysis. https://www.grand-challenge.org. Accessed March 26, 2017.
55. Bossuyt PM, Irwig L, Craig J, Glasziou P. Comparative accuracy: assessing new tests against existing diagnostic pathways. BMJ 2006;332(7549):1089–1092.
56. Summers RM. Progress in fully automated abdominal CT interpretation. AJR Am J Roentgenol 2016;207(1):67–79.
57. Vinyals O, Toshev A, Bengio S, Erhan D. Show and tell: a neural image caption generator. arXiv:14114555v1. https://arxiv.org/abs/1411.4555. Published November 17, 2014. Accessed October 2, 2017.