Sparse deep belief net model for visual area V2

Authors: Honglak Lee, Chaitanya Ekanadham, Andrew Y. Ng
Computer Science Department, Stanford University, Stanford, CA 94305

PRESENTER: HOSSEIN MEHRZADFAR
BILKENT UNIVERSITY, NEUROSCIENCE DEPARTMENT
Introduction

Representation of the Visual Field in V1
Introduction

Representation of the Visual Field in V1
Introduction

The first stage of cortical processing of visual signals takes place in area V1.

Area V2 is a narrow strip of cortex located anterior and adjacent to area V1.
Introduction

Torsten Wiesel (1924-2013)

David Hubel (1926-2013)

Simple Cell Orientation Sensitivity

Best (Preferred) orientation

Figure 11.9 Neurons in the primary visual cortex respond selectively to oriented edges. (A) An anesthetized animal is fitted with contact lenses to focus the eyes on a screen, where images can be projected; an extracellular electrode records the neuronal responses. (B) Neurons in the primary visual cortex typically respond vigorously to a bar of light oriented at a particular angle and weakly—or not at all—to other orientations.
Introduction
Introduction
Introduction

Area V2 is a narrow strip of cortex located anterior and adjacent to area V1.

Responses of V2 neurons appear to signal somewhat more advanced features of the visual scene including contours, surfaces, and corners.
Introduction
Introduction

**Thick Stripes:** gets input from V1 layer 4B; many cells are orientation, direction, and disparity selective.

**Thin Stripes:** gets input from V1 blobs; many cells are color selective and not tuned for orientation or direction of motion.

**Pale Stripes:** gets input from V1 interblobs; mixture of the above properties.
Introduction
Introduction

[DIAGRAM OF NEURAL PATHWAYS AND FLOW CHARTS]

PLANAR

CIRCULAR

RADIAL

[DIFFERENT ARROW PATTERNS AND DIRECTIONS]

100 deg

LEFT

RIGHT

DOWN

UP

COUNTERCLOCKWISE

CLOCKWISE

IN

CUT
Introduction

Inferotemporal (IT) cortex

Object Recognition
Introduction

Face recognition
Introduction

Jennifer Aniston Cell (recorded in human IT cortex)
Introduction

Halle Berry
Introduction

The last few years have seen significant interest in “deep” learning algorithms that learn layered, hierarchical representations and much of this work appears to have been motivated by the hierarchical organization of the cortex.

Authors frequently compare their algorithms’ output to the oriented simple cell receptive fields found in visual area V1.

What about V2, V4, ...?
Introduction

Deep architectures attempt to learn hierarchical structure, and hold the promise of being able to first learn simple concepts, and then successfully build up more complex concepts by composing together the simpler ones.

For example:

- **Hinton et al.** proposed an algorithm based on learning individual layers of a hierarchical probabilistic graphical model from the bottom up.

- **Bengio et al.** proposed a similarly greedy algorithm, based on autoencoders.

- **Ranzato et al.** developed an energy-based hierarchical algorithm, based on a sequence of sparsified autoencoders/decoders.
Biological comparison

**Features in early visual cortex: area V1**

selectivity of neurons for oriented bar stimuli

gabor-like properties of V1 simple cells:
The receptive field of simple cells in V1 are localized, oriented, bandpass filters that resemble gabor filters.
Biological comparison

Features in visual cortex area V2

They respond to more complex stimuli such as: Corner, angel, color

it is uncertain what type of stimuli cause V2 neurons to respond optimally.

One V2 study reported that the receptive fields in this area were similar to those in the neighboring areas V1 and V4.
Biological comparison

Ito and Komatsu investigated how V2 neurons responded to angular stimuli.

By making several axial measurements within the profile, the authors were able to compute various statistics about each neuron’s selectivity for angle **width**, angle **orientation**, and for each separate line component of the angle.
Biological comparison

They found neurons that were selective for only one line component of its peak angle as well as neurons selective for both line components.

several neurons exhibited a high amount of selectivity for its peak angle producing angle profiles
Biological comparison

V2 were not selective for angle width or orientation.

No neurons were found that had more elongation in a diagonal axis than in the horizontal or vertical axes.

Therefore, an important conclusion was that a V2 neuron’s response to an angle stimulus is highly dependent on its responses to each individual line component of the angle.
Algorithm

Hinton et al. proposed an algorithm for learning deep belief networks, by treating each layer as a restricted Boltzmann machine (RBM) and greedily training the network one layer at a time from the bottom up.

Based on results from other methods e.g., sparse coding, ICA, heavy-tailed models and energy based models, sparseness seems to play a key role in learning gabor-like filters.

Therefore, we modify Hinton et al.’s learning algorithm to enable deep belief nets to learn sparse representations.
Algorithm

Sparse restricted Boltzmann machines

Restricted Boltzmann Machine (RBM) has a set of hidden units $h$, a set of visible units $v$, and symmetric connections weights between these two layers represented by a weight matrix $W$.

Suppose that we want to model $k$ dimensional real-valued data using an undirected graphical model with $n$ binary hidden units. The negative log probability of any state in the RBM is given by the following energy function:

$$-\log P(v, h) = E(v, h) = \frac{1}{2\sigma^2} \sum_i v_i^2 - \frac{1}{\sigma^2} \left( \sum_i c_i v_i + \sum_j b_j h_j + \sum_{i,j} v_i w_{ij} h_j \right).$$
Algorithm

Here, σ is a parameter, hj are hidden unit variables, vi are visible unit variables. Informally, the maximum likelihood parameter estimation problem corresponds to learning wij, ci and bj so as to **minimize the energy of states drawn from the data distribution**, and **raise the energy of states that are improbable given the data**.

\[- \log P(v, h) = E(v, h) = \frac{1}{2\sigma^2} \sum_i v_i^2 - \frac{1}{\sigma^2} \left( \sum_i c_i v_i + \sum_j b_j h_j + \sum_{i,j} v_i w_{ij} h_j \right) \]
Algorithm

Under this model, we can easily compute the conditional probability distributions. Holding either h or v fixed, we can sample from the other as follows:

\[
P(v_i | h) = N \left( c_i + \sum_j w_{ij} h_j, \sigma^2 \right),
\]
\[
P(h_j | v) = \text{logistic} \left( \frac{1}{\sigma^2} (b_j + \sum_i w_{ij} v_i) \right)
\]
Algorithm 1 Sparse RBM learning algorithm

1. Update the parameters using contrastive divergence learning rule. More specifically,
   \[ w_{ij} := w_{ij} + \alpha \left( \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{recon}} \right) \]
   \[ c_i := c_i + \alpha \langle v_i \rangle_{\text{data}} - \langle v_i \rangle_{\text{recon}} \]
   \[ b_j := b_j + \alpha \langle b_j \rangle_{\text{data}} - \langle b_j \rangle_{\text{recon}} , \]

   where \( \alpha \) is a learning rate, and \( \langle \cdot \rangle_{\text{recon}} \) is an expectation over the reconstruction data, estimated using one iteration of Gibbs sampling (as in Equations 2,3).

2. Update the parameters using the gradient of the regularization term.

3. Repeat Steps 1 and 2 until convergence.
Algorithm

Learning deep networks using sparse RBM

Once a layer of the network is trained, the parameters $w_{ij}$, $b_j$, $c_i$’s are frozen and the hidden unit values given the data are inferred. These inferred values serve as the “data” used to train the next higher layer in the network.

In our experiments using natural images, we learn a network with two hidden layers, with each layer learned using the sparse RBM algorithm.
Learning “strokes” from handwritten digits:

We applied the sparse RBM algorithm to the MNIST handwritten digit dataset. We learned a sparse RBM with 69 visible units and 200 hidden units.

Many bases found by the algorithm roughly represent different “strokes” of which handwritten digits are comprised.
Visualization

Learning from natural images:

We also applied the algorithm to a training set a set of 14-by-14 natural image patches, taken from a dataset compiled by van Hateren. We learned a sparse RBM model with 196 visible units and 400 hidden units.

The learned bases are oriented, gabor-like bases and resemble the receptive fields of V1 simple cells.

Figure 3: 400 first layer bases learned from the van Hateren natural image dataset, using our algorithm.
Visualization

Learning a **two-layer** model of natural images using sparse RBMs:

We further learned a two-layer network by **stacking one sparse RBM on top of another**

After learning, the second layer weights were **quite sparse**—most of the weights were very small, and only a few were either **highly positive** or **highly negative**.

**Positive** weights represent **excitatory** connections between model V1 and model V2 units, whereas **negative** elements represent **inhibitory** connections.
Visualization

By visualizing the second layer bases as shown in Figure below, we observed bases that encoded co-linear first layer bases as well as edge junctions.

This shows that by extending the sparse RBM to two layers and using greedy learning, the model is able to learn bases that encode contours, angles, and junctions of edges.
Evaluation experiments

We generated a stimulus set consisting of the same set of angles (pairs of edges).

To identify the “center” of each model neuron’s receptive field, we translate all stimuli densely over the 14x14 input image patch, and identify the position at which the maximum response is elicited.

Using these stimuli, we compute the hidden unit probabilities from our model V1 and V2 neurons. In other words, for each stimulus we compute the first hidden layer activation probabilities, then feed this probability as data to the second hidden layer and compute the activation probabilities again in the same manner.
Evaluation experiments

Visualization of a number of model V2 neurons that maximally respond to various complex stimuli.

the leftmost image shows a linear combination of the top three weighted V1 components that comprise the V2 basis; the next three images show the top three optimal stimuli; and the last three images show the top three weighted V1 bases
Evaluation experiments
Conclusion

When trained on natural images, this model learns local, oriented, edge filters in the first layer. More interestingly, the second layer captures a variety of both colinear (“contour”) features as well as corners and junctions, that in a quantitative comparison to measurements of V2 taken by Ito & Komatsu, appeared to give responses that were similar along several dimensions.

we believe that these results also suggest that deep learning algorithms, such as our sparse variant of deep belief nets, hold promise for modeling higher-order features such as might be computed in the ventral visual pathway in the cortex.
Thank you for your attention.