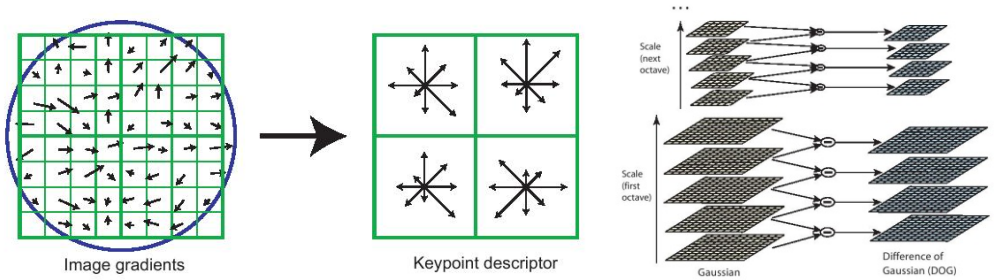
Neural Network and Deep Learning Approaches to Computer Vision

Sumeet Agarwal
Department of Electrical Engineering
IIT Delhi

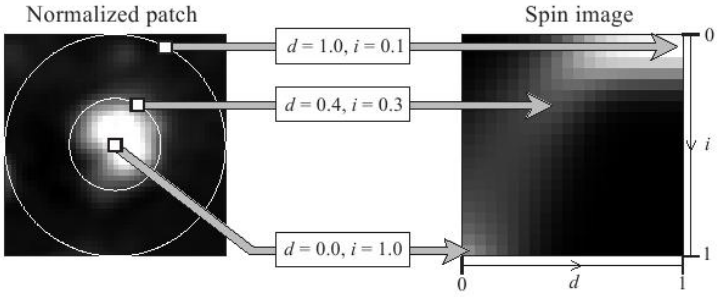
What is the key challenge in vision?

- Arguably, extracting meaningful *features* from images
- How do we construct increasingly complex/abstract representations, starting with raw pixels?
- These representations can be handcoded; but can they also be *learnt* automatically from data?
- Does the learning of such representations have to be guided/supervised, or can it also be achieved in an *unsupervised* fashion?

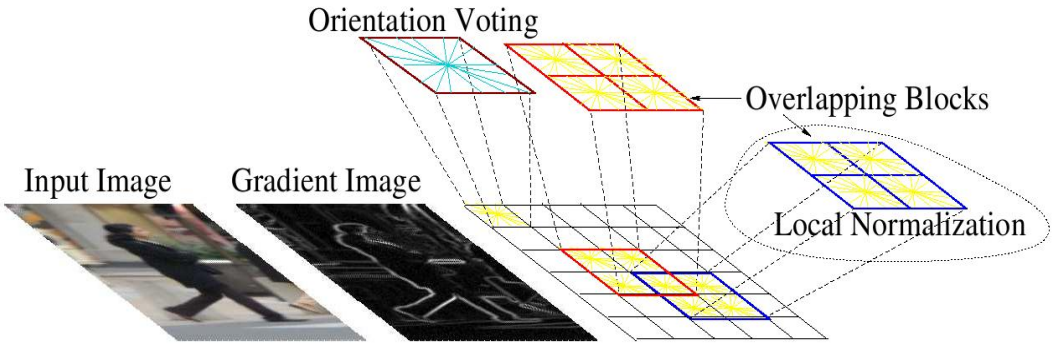
Computer vision features



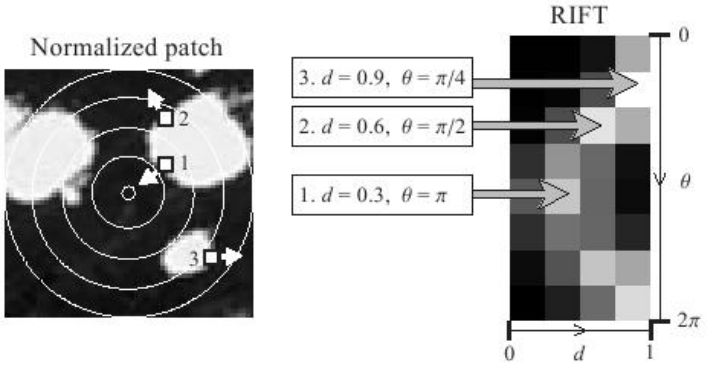
SIFT



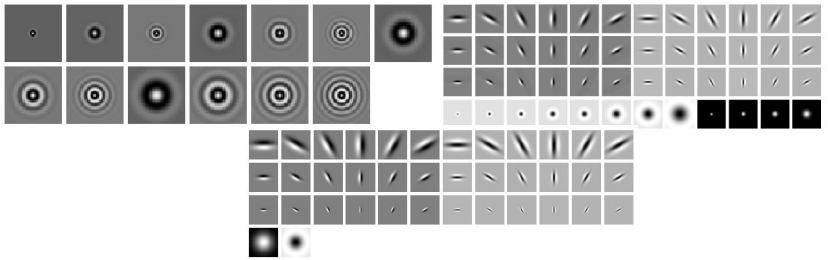
Spin image



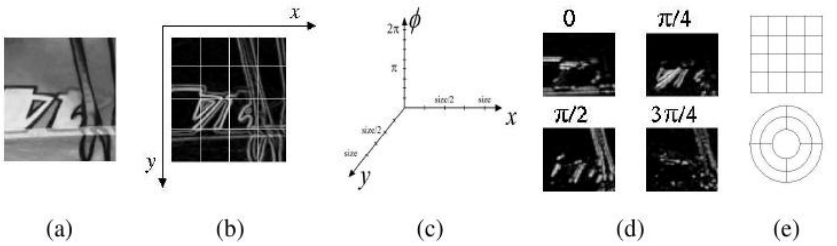
HoG



RIFT



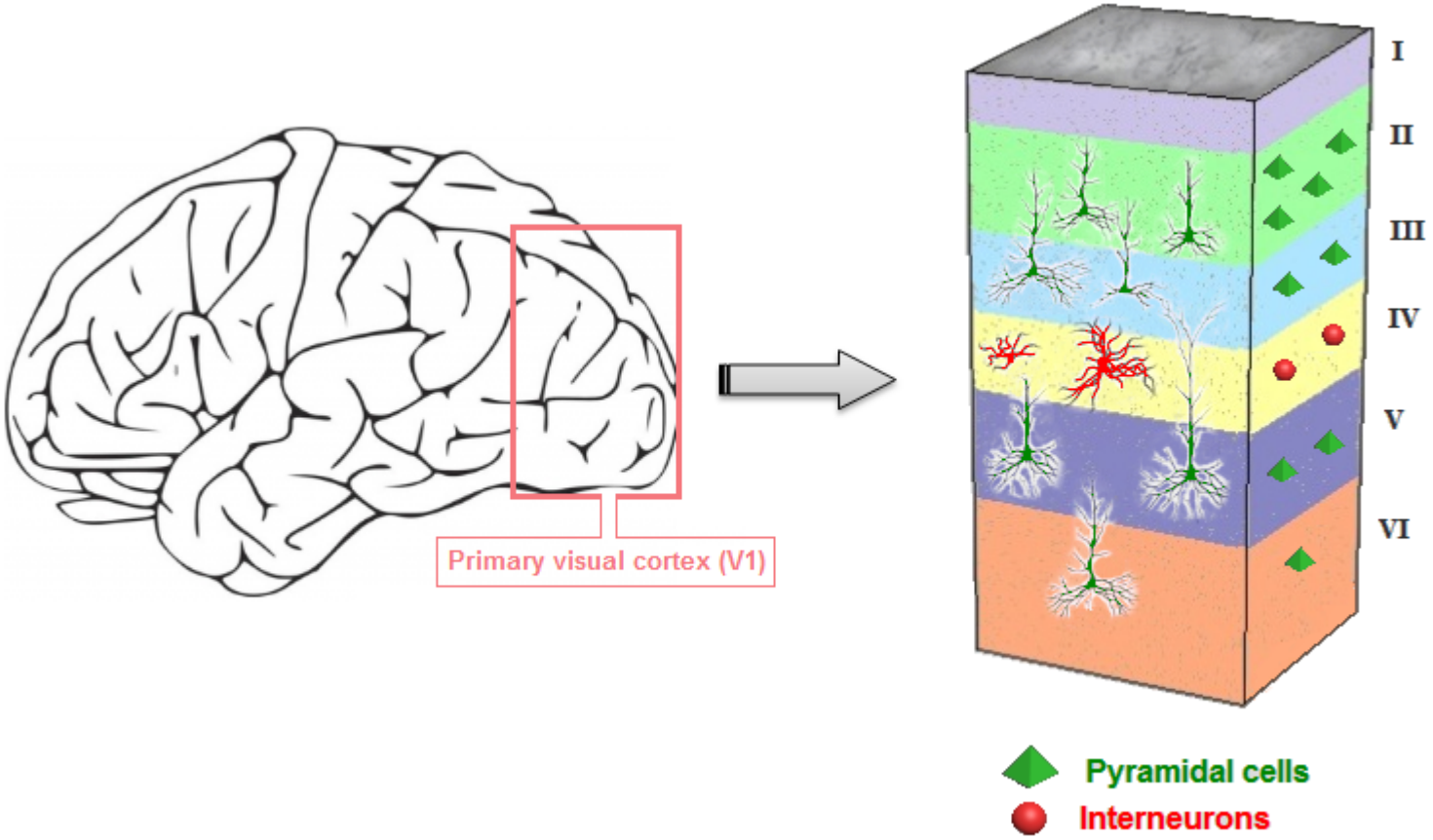
Textons



GLOH

[Andrew Ng]

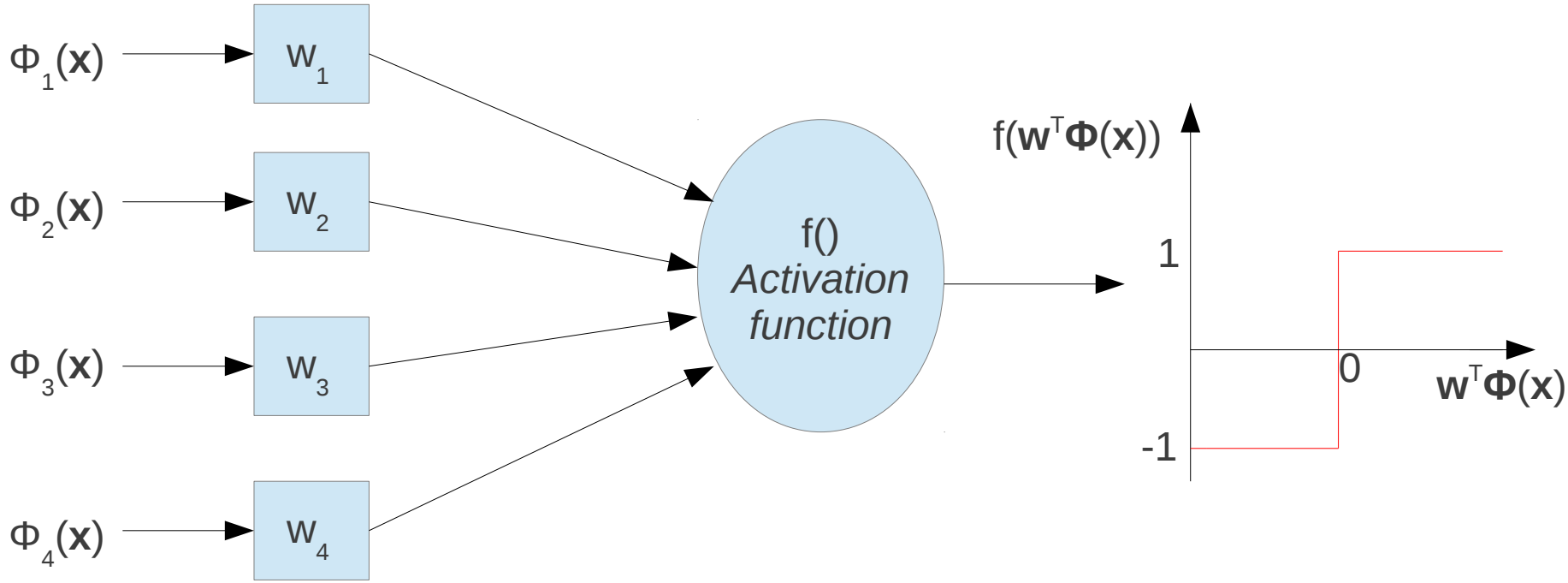
Human vision



[Bachatene *et al.*, 2012]

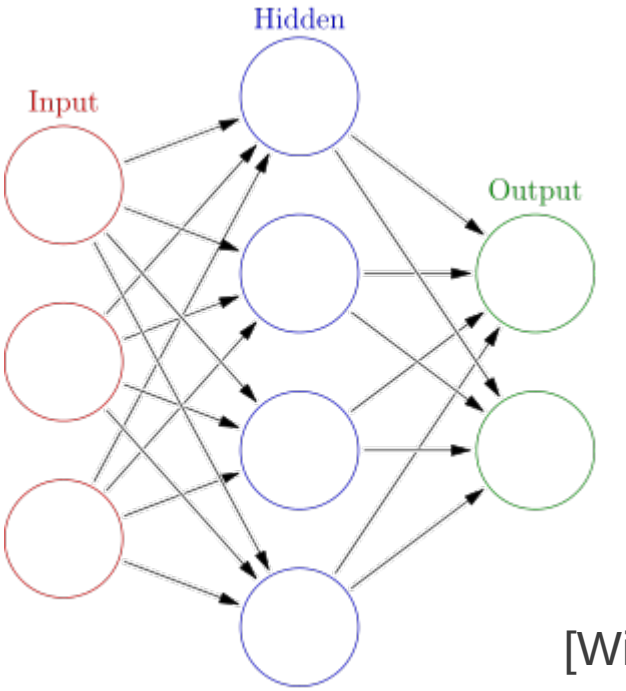
Neuronal networks build up a hierarchy of increasingly complex representations.

Learning models: The Perceptron



A non-linear transformation in the form of a step function is applied to the weighted sum of the input features. This is inspired by the way neurons appear to function, mimicking the *action potential*.

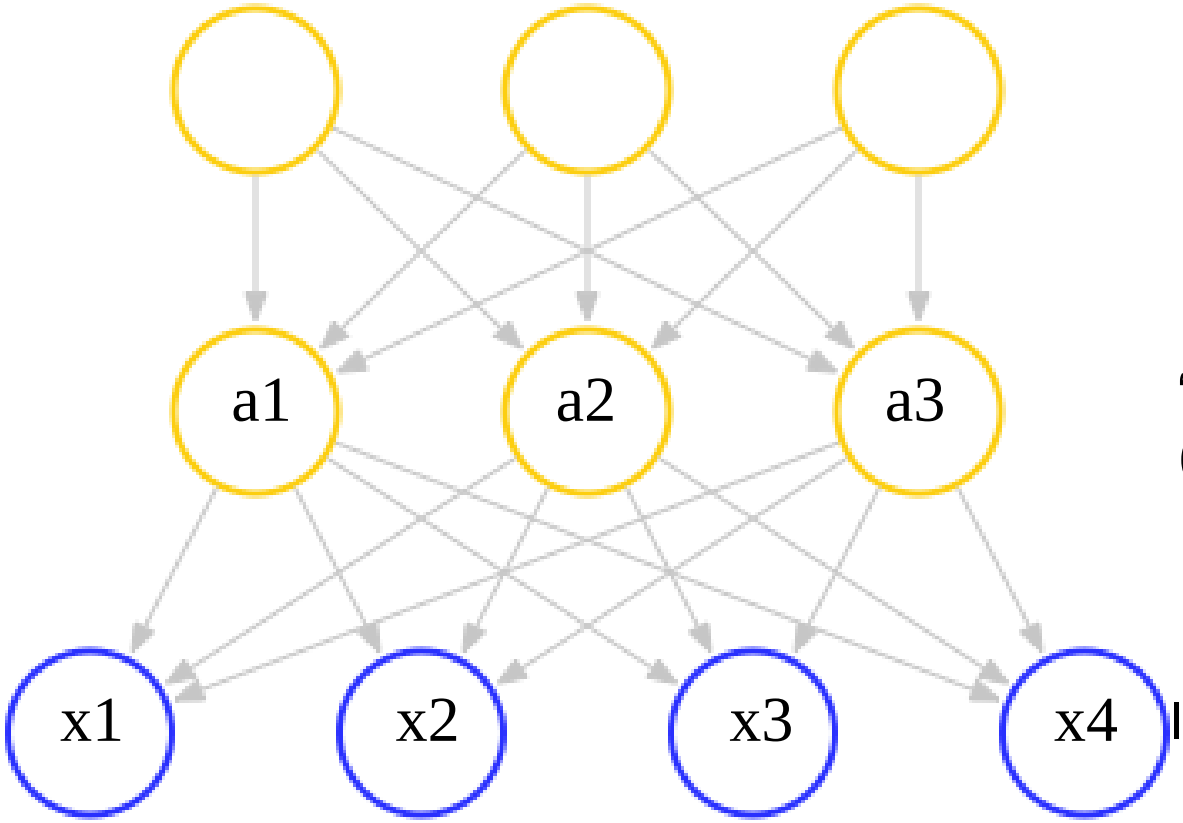
Neural Networks: *Multilayer Perceptrons*



[Wikipedia]

Usually, the non-linear activation function used is a logistic sigmoid: $y = f(\mathbf{w}^T\mathbf{\Phi}(\mathbf{x})) = \sigma(\mathbf{w}^T\mathbf{\Phi}(\mathbf{x}))$, where $\sigma(a) = 1/(1+e^{-a})$. This makes y a differentiable function of the input $\mathbf{x}$; each unit/neuron can now be thought of as simply a logistic regression classifier.

Learning feature hierarchies



Higher layer
(Combinations of edges;
cf. V2)

"Sparse coding"
(edges; cf. V1)

Input image (pixels)

[Technical details: Sparse autoencoder or sparse version of Hinton's DBN.]

[Lee, Ranganath & Ng, 2007]

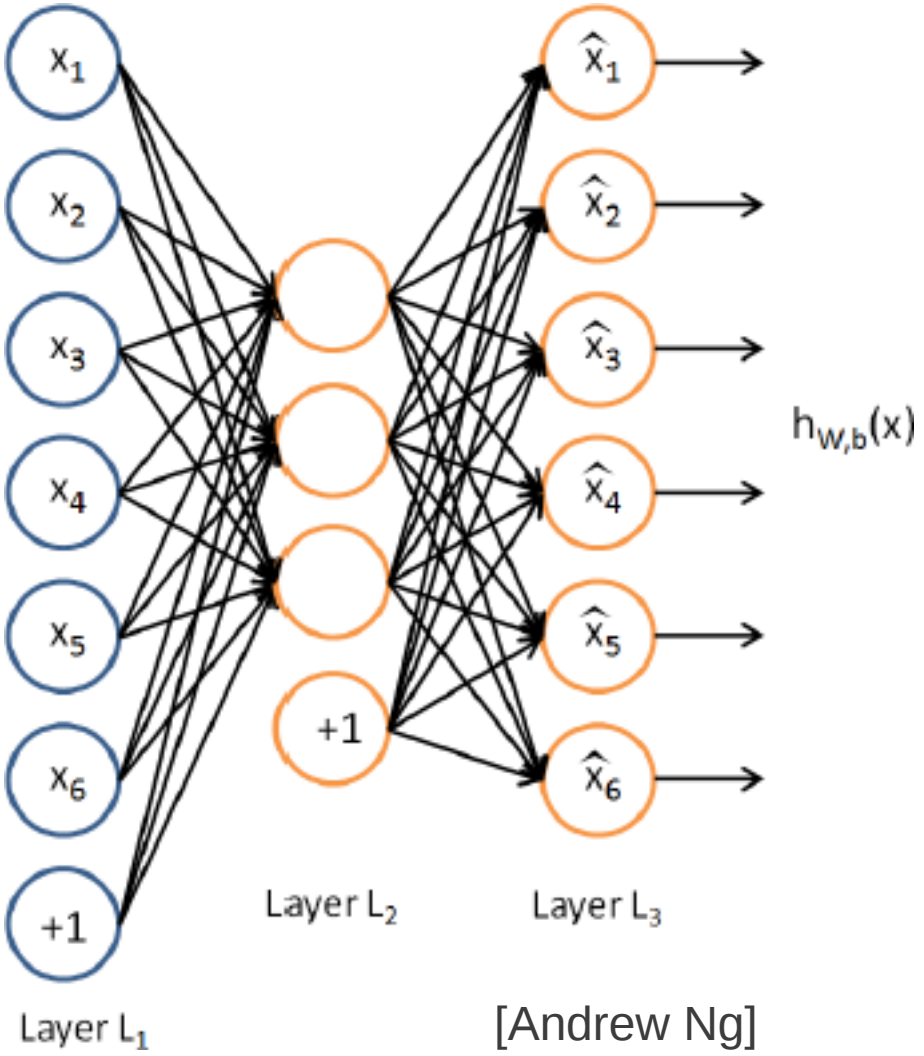
Supervised learning with neural nets

- Target values for the output(s) can be provided as categorical or continuous values, corresponding to classification and regression settings
- An appropriate error function is defined and minimised with respect to the network weights
- Typically done using gradient descent; the gradient of the error function can be computed via *backpropagation*

'Deep' learning

- Is just a fashionable term for the use of neural networks with many hidden layers
- The aim is for hidden neurons to be able to capture a hierarchy of representations, similar to the visual cortex
- Labelled training data may be limited; can useful representations also be learnt in an unsupervised fashion?

Sparse autoencoders



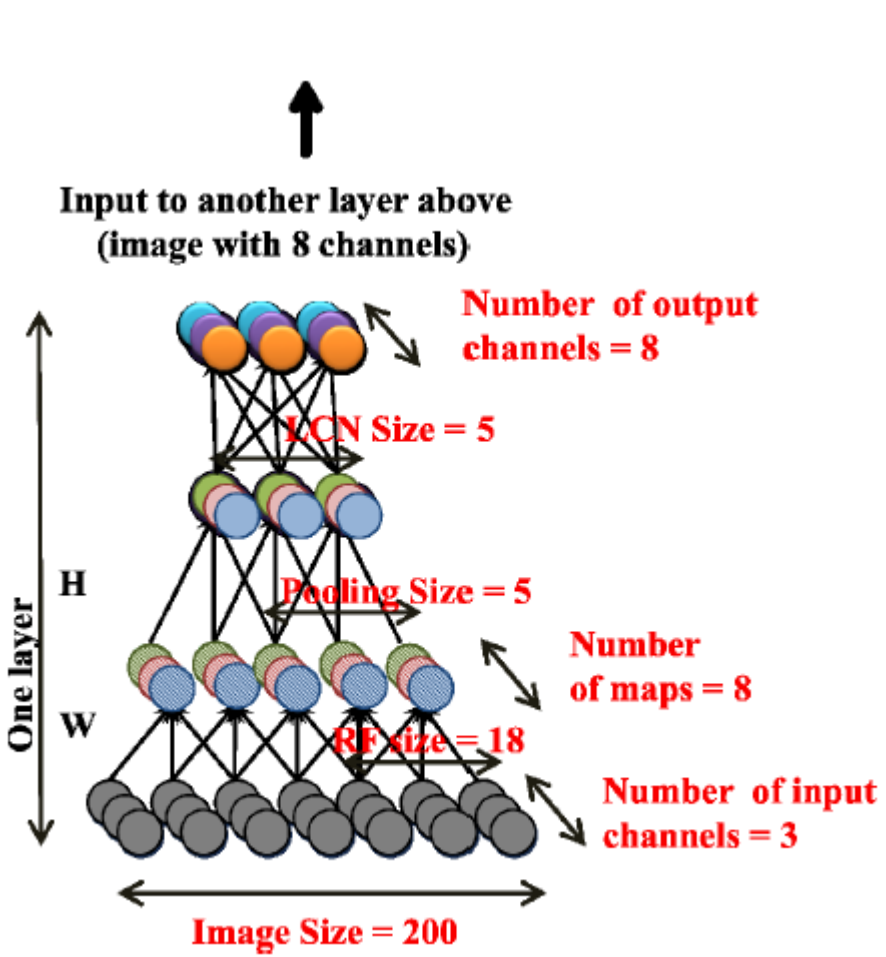
[Andrew Ng]

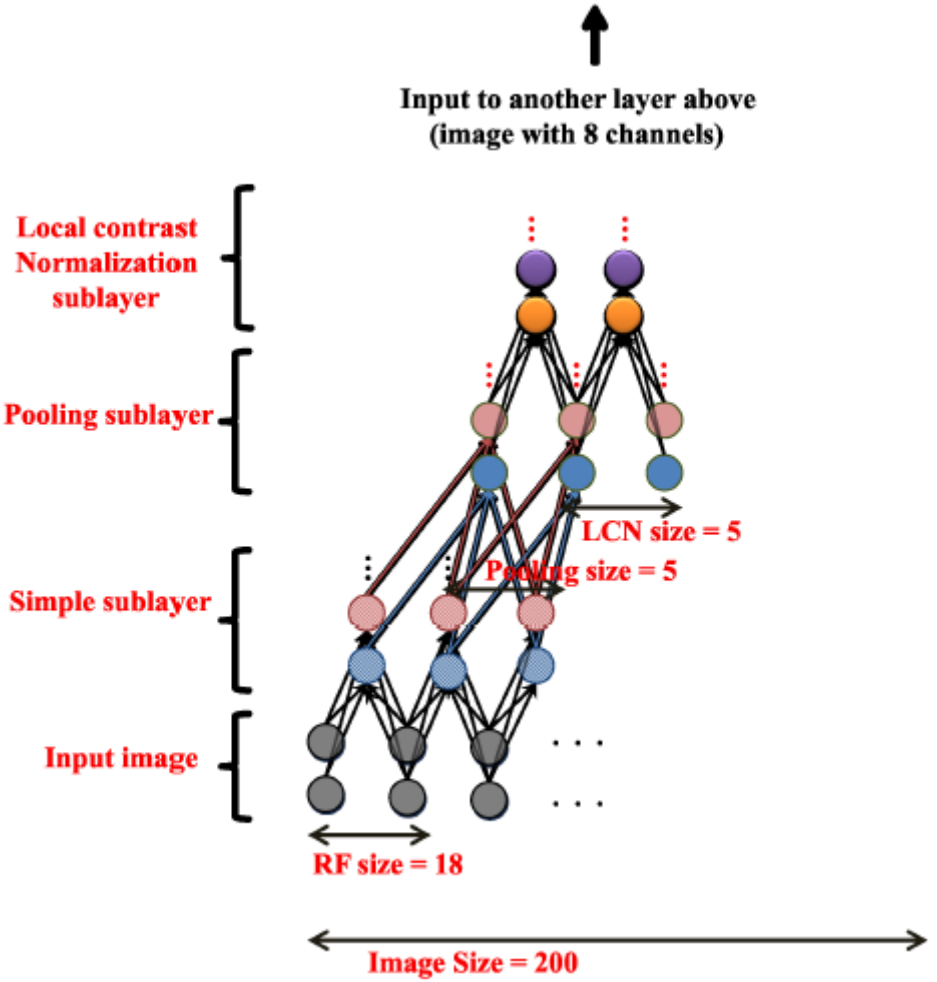
- A neural net with as many outputs as inputs
- The idea is to reproduce the input as closely as possible (minimise reconstruction error)
- How can this be done whilst retaining a relatively small number of features in the hidden layers, i.e., enforcing *sparsity*?
- Represents a form of *dimensionality reduction*

Large-scale deep learning

- Made possible by massive computing power and parallelisation
- *Google Brain*: A deep neural network with 9 layers and ~10^9 connections
- Still only one-millionth the size of a 3-year-old human brain!
- Important for demonstrating that complex concepts like faces can be discovered in an entirely *unsupervised* fashion

Visual input network architectures





[Le *et al*. 2012]

Conclusions

- Classical neural networks provide a biologically-inspired approach to the problem of learning appropriate visual representations
- Recent advances in technology have made it possible to train 'deep' networks, with millions or billions of connections
- Unsupervised learning by minimising reconstruction error whilst enforcing sparsity can be a powerful tool for feature/concept discovery

References

- Chris Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- Quoc V. Le *et al*. Building High-level Features Using Large Scale Unsupervised Learning. *ICML* 2012.
- Quoc V. Le *et al*. ICA with Reconstruction Cost for Efficient Overcomplete Feature Learning. *NIPS* 2011.
- Andrew Ng. Sparse autoencoder. *CS294A Lecture notes*, Stanford University, 2011.
- Lyes Bachatene *et al*. Adaptation and Neuronal Network in Visual Cortex. *Visual Cortex - Current Status and Perspectives*. InTech, 2012.