Neural Networks and Deep Learning

Dr. Jerome J. Braun

This Lecture:
Convolutional Neural Networks IV

Course: Neural Networks and Deep Learning
IE 7615

This Lecture

- CNNs in practice
- Gabor functions
- CNN generating inputs

CNN Training

- Local minima of error-function landscape
  - Likely similar
  - Local minima not as much issue as traditionally thought
- Long plateaus of error-function landscape
  - Runtime impact
- Challenges
  - Training large number of parameters
  - Breaking symmetries between parameters
  - High-dimensional spaces — number of parameters and exemplars
  - Scalability

CNNs in Practice (1)

- Architecture choice — task dependent
- More data —> more layers, more kernels
  - Computational cost

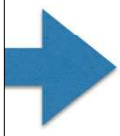
CNNs in Practice (2)

- Gradient descent (SGD) with momentum
- ReLU
- Learning rate
  - Choose initially on subset of data
  - Start with large learning rate, decrease until loss does not diverge
  - Higher rate of decrease towards end of training
- Parameter initialization — features should have similar variance
- Visualize features — should be uncorrelated and high variance
- Visualize trained kernels
  - Should be uncorrelated
  - Should have structure
- Do numeric checking of gradients — by finite differences

CNNs in Practice (3)

- Training fails?
  - Perhaps learning rate too large
  - BackProp bugs — do numerical check (finite differences)
- Performance too low?
  - If loss minimized — perhapse loss function not appropriate for given task
  - Perhaps network too small
  - Perhaps issues with units or parameters — use visualization
  - BackProp bugs
- Runtime too slow?
  - Perhaps network excessively large?
  - Consider distributed frameworks, GPU

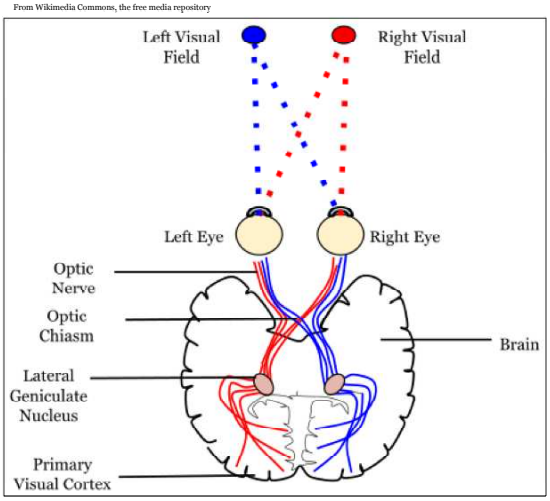
This Lecture



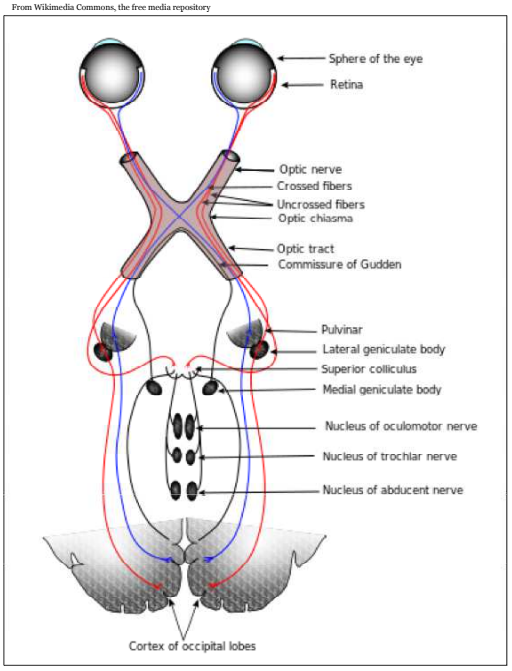
- CNNs in practice
- Gabor functions
- CNN generating inputs

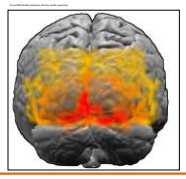
Recap: Human Vision

From Wikimedia Commons, the free media repository



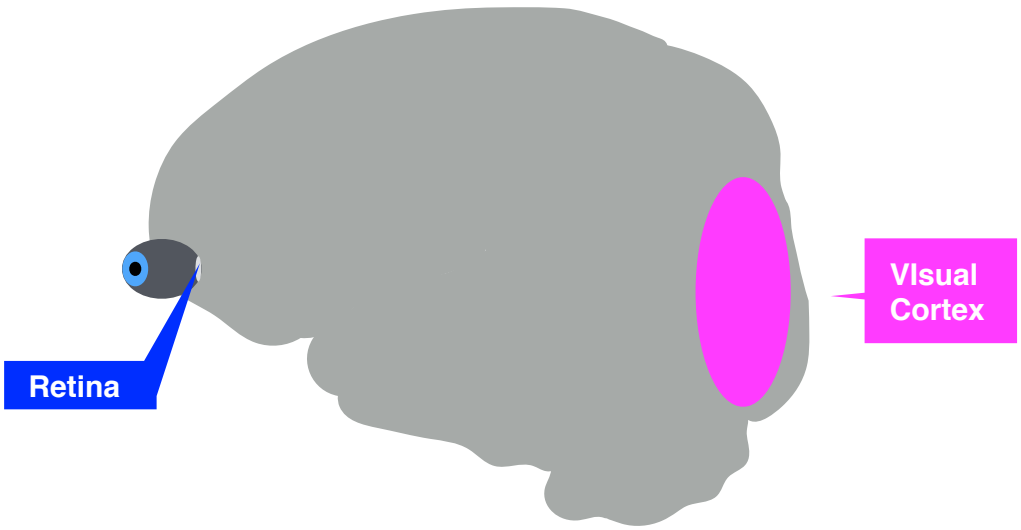
From Wikimedia Commons, the free media repository



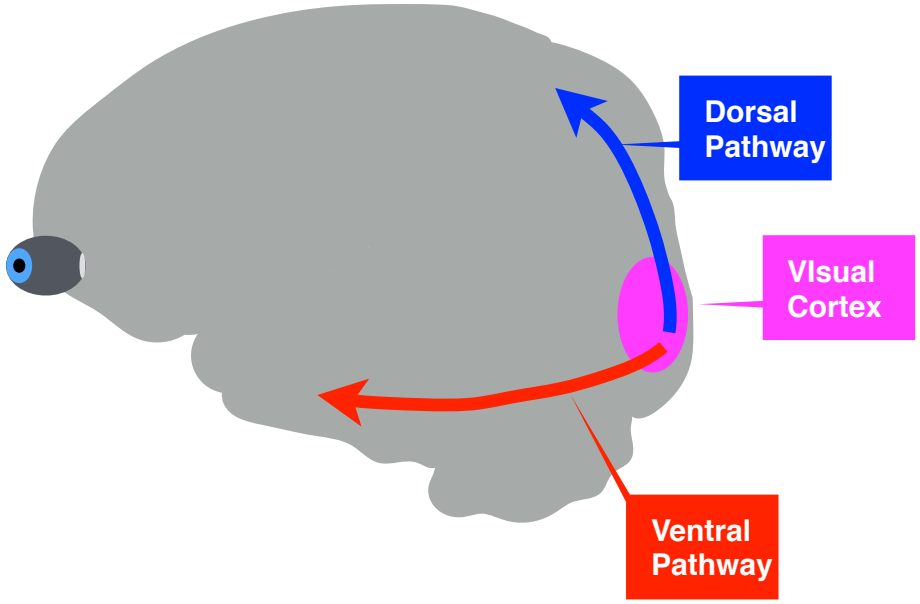


- Visual Cortex
- Caudal (posterior) view
- Brodmann areas 17, 18, 19

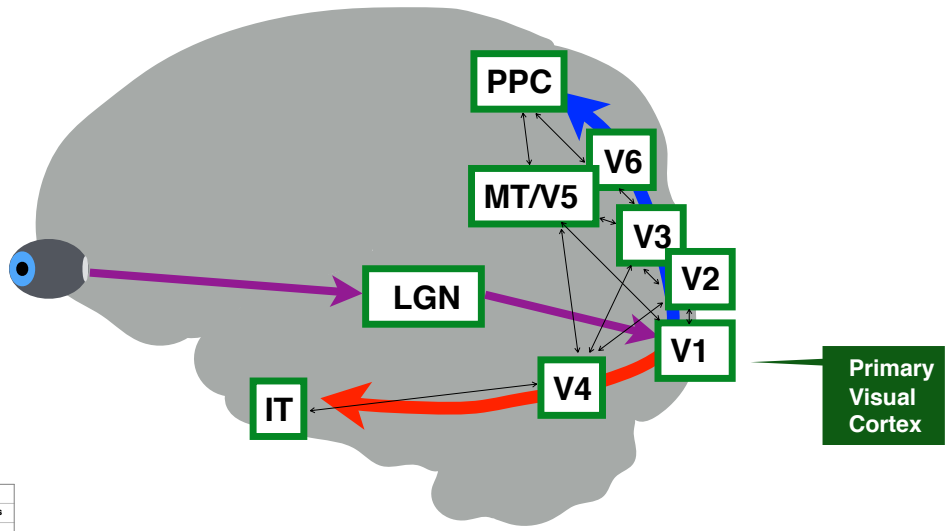
Recap: Human Vision



Recap: Pathways From Primary Visual Cortex



Recap: Visual Information Flow (Simplified)



IT	Inferotemporal cortex
LGN	Lateral geniculate nucleus
PPC	Posterior parietal cortex
MT/V5	Middle temporal visual area
V1	Primary visual cortex
V2	Secondary visual cortex
V3, V4	Visual areas 3, 4

CNN and Human Vision: Gabor Functions

- **Information flow when viewing an object**
  - **Retina —> LGN —> Visual Cortex V1, V2, V4, —> IT**
- **V1 cells have weights that can be described by Gabor functions**
  - **Simple cell response s(I)**

$$s(I) = \sum_{x \in \mathbb{X}} \sum_{y \in \mathbb{Y}} w(x,y) I(x,y)$$

Gabor functions

  - ***w* are Gabor functions**

$$w(x, y; \alpha, \beta_x, \beta_y, f, \phi, x_0, y_0, \tau) = \alpha \exp\left(-\beta_x x'^2 - \beta_y y'^2\right) \cos(f x' + \phi),$$

where

$$x' = (x - x_0)\cos(\tau) + (y - y_0)\sin(\tau)$$

$$y' = -(x - x_0)\sin(\tau) + (y - y_0)\cos(\tau)$$

  - **with parameters**

$$\alpha,\ \beta_x,\ \beta_y,\ f,\ \phi,\ x_0,\ y_0, \text{ and } \tau$$

Simple Cells as Gabor Functions

$$s(I) = \sum_{x \in \mathbb{X}} \sum_{y \in \mathbb{Y}} w(x,y) I(x,y)$$

Coordinate system → x_0, y_0, τ

$$w(x, y; \alpha, \beta_x, \beta_y, f, \phi, x_0, y_0, \tau) = \alpha \exp\left(-\beta_x x'^2 - \beta_y y'^2\right) \cos(f x' + \phi)$$

$$x' = (x - x_0)\cos(\tau) + (y - y_0)\sin(\tau) \qquad y' = -(x - x_0)\sin(\tau) + (y - y_0)\cos(\tau)$$

i.e., x and y are translated and rotated to form x' and y'

Simple cell will respond to:

- image features centered at (x_0, y_0)
- changes in brightness as we move along line rotated by τ radians from horizontal

Gabor Functions and Receptive Fields

$$s(I) = \sum_{x \in \mathbb{X}} \sum_{y \in \mathbb{Y}} w(x,y) I(x,y) \qquad w(x, y; \alpha, \beta_x, \beta_y, f, \phi, x_0, y_0, \tau) = \alpha \exp\left(-\beta_x x'^2 - \beta_y y'^2\right) \cos(f x' + \phi)$$

View of *w* as function of x' and y':
Function w responds to changes in brightness as we move along x' axis

$$x' = (x - x_0)\cos(\tau) + (y - y_0)\sin(\tau) \qquad y' = -(x - x_0)\sin(\tau) + (y - y_0)\cos(\tau)$$

$$w(x, y; \alpha, \beta_x, \beta_y, f, \phi, x_0, y_0, \tau) = \alpha \exp\left(-\beta_x x'^2 - \beta_y y'^2\right) \cos(f x' + \phi)$$

Gaussian – can be viewed as gating term that ensures that simple cell will respond only to:

- values near where both x' and y' are zero
  - i.e. near *center* of cell's RECEPTIVE FIELD

- Scaling factor α adjusts total magnitude of simple cell's response
- β_x and β_y control how quickly receptive field falls off

Gabor Functions and Receptive Fields

$$s(I) = \sum_{x \in \mathbb{X}} \sum_{y \in \mathbb{Y}} w(x,y) I(x,y) \qquad w(x, y; \alpha, \beta_x, \beta_y, f, \phi, x_0, y_0, \tau) = \alpha \exp\left(-\beta_x x'^2 - \beta_y y'^2\right) \cos(f x' + \phi)$$

View of *w* as function of x' and y':
Function *w* responds to changes in brightness as we move along x' axis

$$x' = (x - x_0)\cos(\tau) + (y - y_0)\sin(\tau) \qquad y' = -(x - x_0)\sin(\tau) + (y - y_0)\cos(\tau)$$

$$w(x, y; \alpha, \beta_x, \beta_y, f, \phi, x_0, y_0, \tau) = \alpha \exp\left(-\beta_x x'^2 - \beta_y y'^2\right) \cos(f x' + \phi)$$

Cosine – can be viewed as controlling how simple cell responds to changing brightness along x' axis

- f controls cosine's frequency

- φ is phase offset

Simple Cells and Gabor Functions

- ***Cartoon view* of simple cells:**
  - **Simple cell responds to specific spatial frequency of brightness**
    - **In specific direction**
    - **At specific location**
  - **Simple cells excited most when**
    - **Wave of brightness in image has same phase as weights**
    - **This occurs when**
      - **Image bright where weights are positive**
      - **Image dark where weights negative**
  - **Simple cells inhibited most when**
    - **Wave of brightness fully out of phase with weights**
      - **Image bright where weights positive and vice-versa**

Simple Cells and Gabor Functions

- ***Cartoon view* of simple cells:**
  - **Simple cell responds to specific spatial frequency of brightness**
    - **In specific direction**
    - **At specific location**
  - **Simple cells excited most when**
    - **Wave of brightness in image has same phase as weights**
    - **This occurs when**
      - **Image bright where weights are positive**
      - **Image dark where weights negative**
  - **Simple cells inhibited most when**
    - **Wave of brightness fully out of phase with weights**
      - **Image bright where weights positive and vice-versa**

CNN and Human Vision: Gabor Functions

- **Information flow when viewing an object**
  - **Retina —> LGN —> Primary Visual Cortex V1, V2, V4, —> IT**
- **V1 cells have weights that can be described by Gabor functions**
  - **Simple cell response s(I)** $s(I) = \sum_{x \in \mathbb{X}} \sum_{y \in \mathbb{Y}} w(x, y) I(x, y)$
  - **w are Gabor functions**

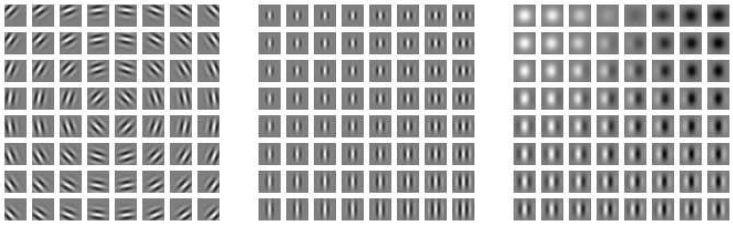
$$w(x, y; \alpha, \beta_x, \beta_y, f, \phi, x_0, y_0, \tau) = \alpha \exp\left(-\beta_x x'^2 - \beta_y y'^2\right) \cos(f x' + \phi),$$

where

$$x' = (x - x_0)\cos(\tau) + (y - y_0)\sin(\tau)$$

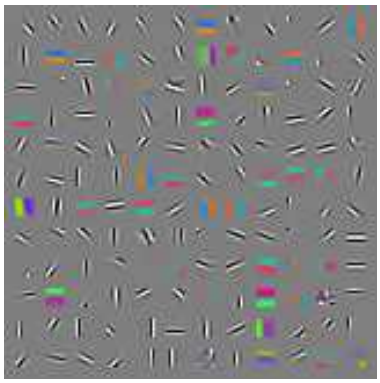
$$y' = -(x - x_0)\sin(\tau) + (y - y_0)\cos(\tau)$$

  - **with parameters** $\alpha, \beta_x, \beta_y, f, \phi, x_0, y_0,$ and τ
  - **Examples with different parameter settings**

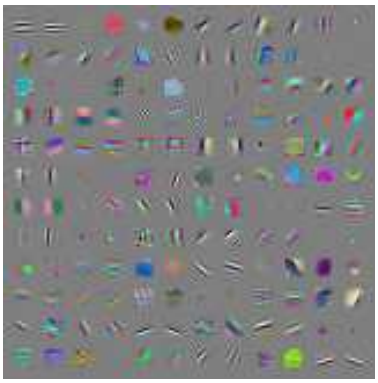


[Goodfellow 2016]

Gabor-Like Learned Kernels



Weights learned by unsupervised learning algorithm (spike and slab sparse coding) applied to small image patches



Convolution kernels learned by first layer of fully supervised convolutional maxout network (neighboring pairs of filters drive same maxout unit)

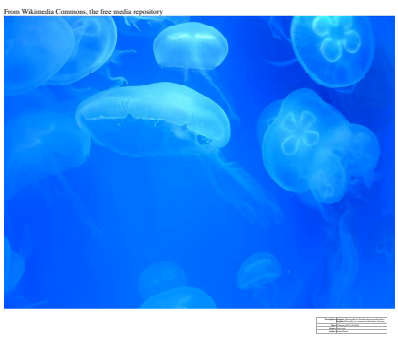
[Goodfellow 2016]

This Lecture

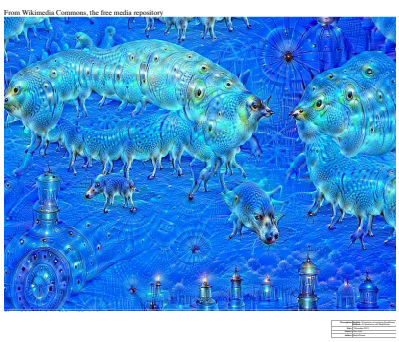
- CNNs in practice
- Gabor functions
- CNN generating inputs

CNN Generating Inputs

- Generate variants of input data
  - By activation of CNN output layer units
  - Potential utility for improving training
- DeepDream example



DeepDream



[Thoma]

Questions?