COMS W4701: Artificial Intelligence

Lecture 26: Convolutional Neural Networks

Slide materials adapted from Stanford's CS231n

Tony Dear, Ph.D.

Department of Computer Science

School of Engineering and Applied Sciences

Topics

- Computer vision
- Feature extraction
- Image classification
- Convolutional neural networks

Computer Vision

- **Vision** is perception using visible light reflected by environment objects
- **Digital cameras** (e.g., color, depth, stereo) are sensors that capture light and transform them to *digital images*

- **Computer vision** is concerned with acquiring, processing, and *understanding* digital images in order to extract symbolic information

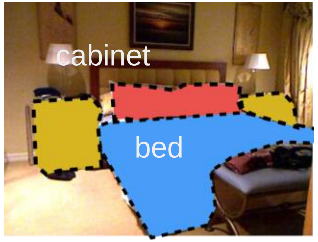
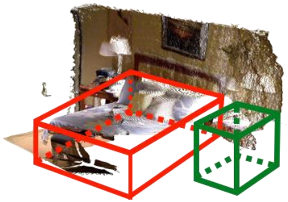
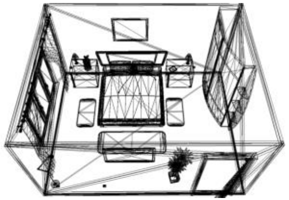


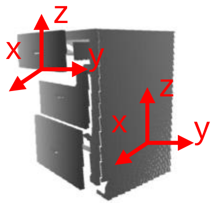
Image segmentation



3D object detection



Map - 3D reconstruction



Pose estimation

Image Representation

- A basic way to represent images is with a grid of **pixels**
- The value $I(x, y)$ of a pixel indicates light *intensity*

- For greyscale, $I(x, y)$ typically ranges from 0 (black) to 255 (white)
- For color, we may have multiple *channels* for intensities in different colors
- Common color models: RGB (red, green, blue), HSV (hue, saturation, value)

- Basic image processing applies a specified function to the values $I(x, y)$
- Ex: Brightness adjustment: $I(x, y) + \beta$; contrast adjustment: $\alpha I(x, y)$

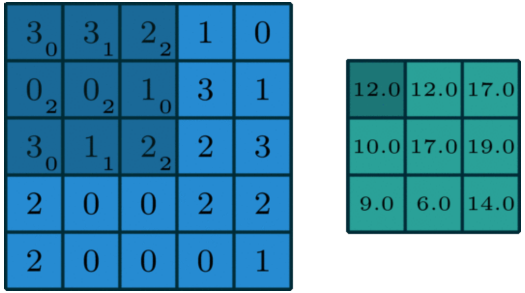
Image Kernels

- **Spatial filters** transform images using functions on pixel *neighborhoods*
- Uses: Image enhancement, information extraction, pattern detection

- Most filters can be computed using the **convolution** operation:

$$I'(x, y) = F * I = \sum_{i=-M}^{M} \sum_{j=-N}^{N} F(i, j) I(x + i, y + j)$$

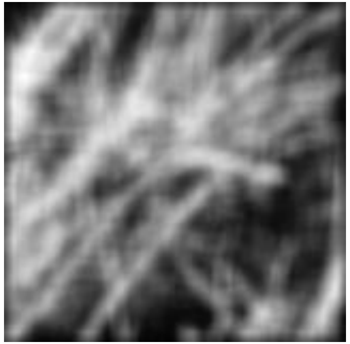
- where F is a $(2M + 1) \times (2N + 1)$ **mask** or **kernel** matrix
- Special care should be taken at grid boundaries



Examples of Filters

- *Moving average filter* $B = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

  blurs or smooths out an image
- Normalization maintains image brightness



- *Sharpening filter* $S = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$

  accentuates edges

Feature Detection

- **Features** in images are interest points that differ from the immediate neighborhood, e.g. in intensity, color, or texture

- Many geometric features like edges and corners can provide semantic content for tasks like reconstruction, estimation, and detection

- An **edge** is a region with large change in intensity along one dimension but negligible change along the orthogonal direction
- Edges can be detected using **differentiation filters** and looking for spikes

Edges and Corners

- **Sobel** kernels for edge detection:

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

- More sophisticated filters involve pre-smoothing and post-thresholding

- Corners may also be of interest, e.g. for 3D reconstruction and panorama stitching

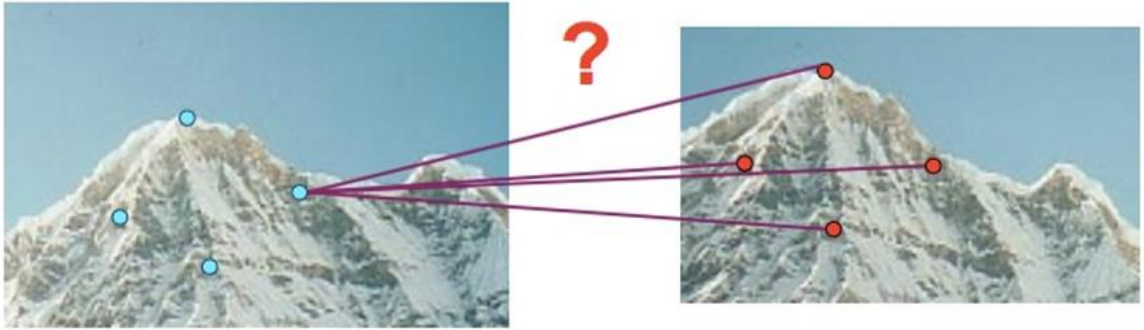


Image Descriptors

- More generally, image **descriptors** are features that can be compared across images, useful for object detection and matching
- Need to be repeatable (invariant wrt transformations) and distinctive
- Many detection algorithms, e.g., SIFT (scale-invariant feature transform)

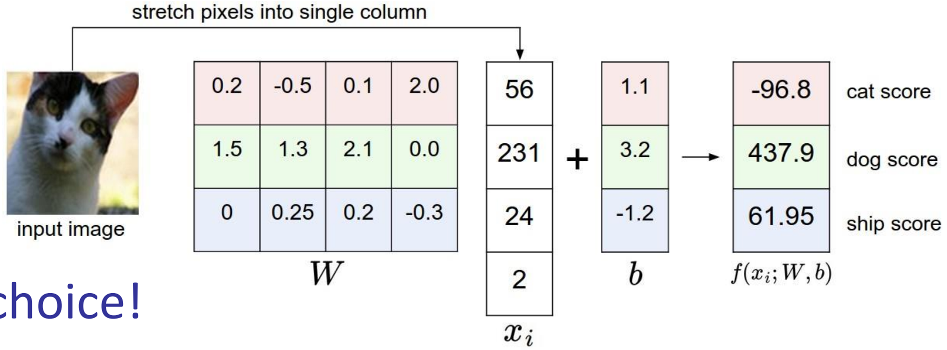


Image Classification

- **Image classification** is the task of assigning an image a class or label
- Challenges: Variation in viewpoints, scale, and lighting, deformation, occlusion, background clutter, intra-class variation

- Recall **softmax** function for generalizing logistic model to multi-class case
- Learn a set of weights for each class

$$h_i(f(\mathbf{x})) = \frac{\exp(f(\mathbf{x})_i)}{\sum_{k=1}^{K} \exp(f(\mathbf{x})_k)}$$



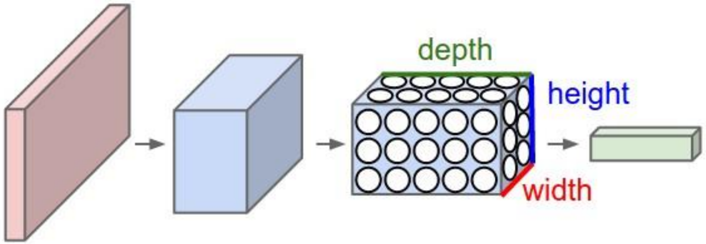
- Problem: Linear models are a poor choice!

Convolutional Neural Networks

- Performance of linear models depends heavily on the specified *features*
- Not easy to specify for general image classification beyond raw pixels

- We can use neural networks with pixels as inputs, but this representation does not preserve *spatial* information
- Fully connected networks have too many weights for a typical image size

- A **ConvNet** is a special network architecture that preserves image structure and reduces the number of parameters used

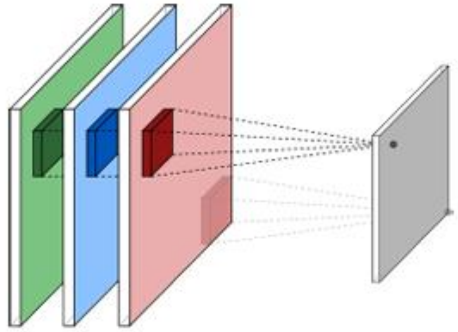


Convolution Layers

- Convolution layers apply a set of 3-dimensional filters to the input image
- As with “regular” filters, they see *spatial* relationships in pixel neighborhoods

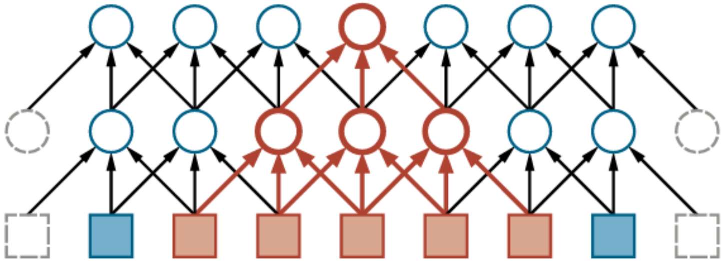
- Convolution in a neural network is a form of *parameter sharing*, since a given filter is applied to all pixel neighborhoods in the image
- Convolution is also equivalent to translation

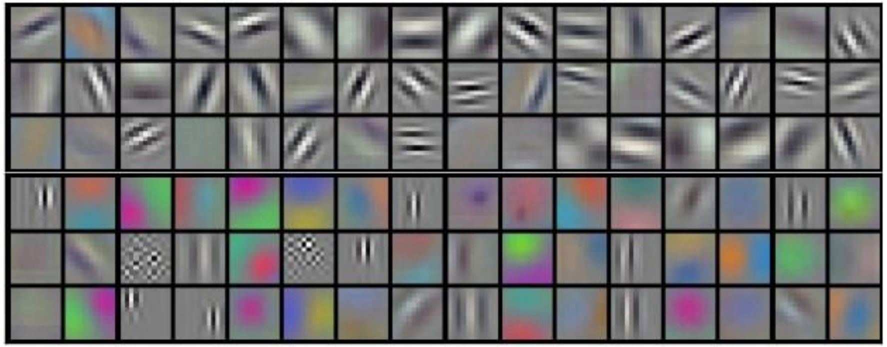
- The set of outputs from all filters make up a new image called an **activation map**



Feature Learning

- Since we have many filters, each one can detect its own "relevant" feature
- ConvNets typically have multiple convolution layers, and learned features typically progress from more primitive (edges/corners) to more high-level
- The **receptive field** of a filter also increases with each successive layer

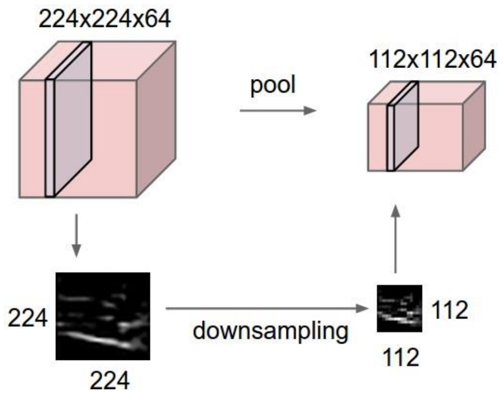




Krizhevsky et al. 2012

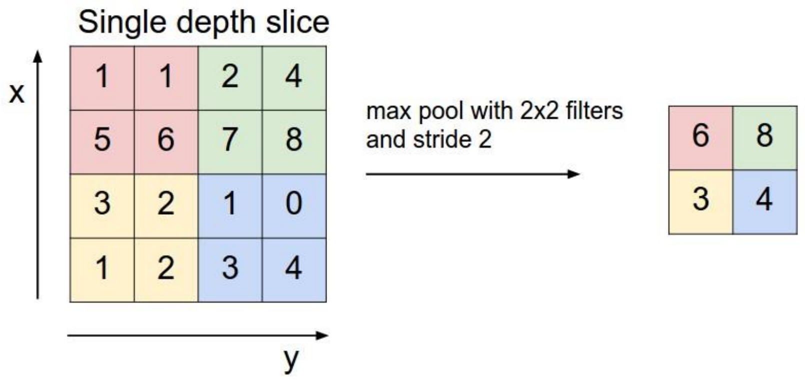
Pooling Layers

- **Pooling layers** *downsample* and shrink the image



- Reduces the number of parameters and computation in the network, and can help prevent overfitting

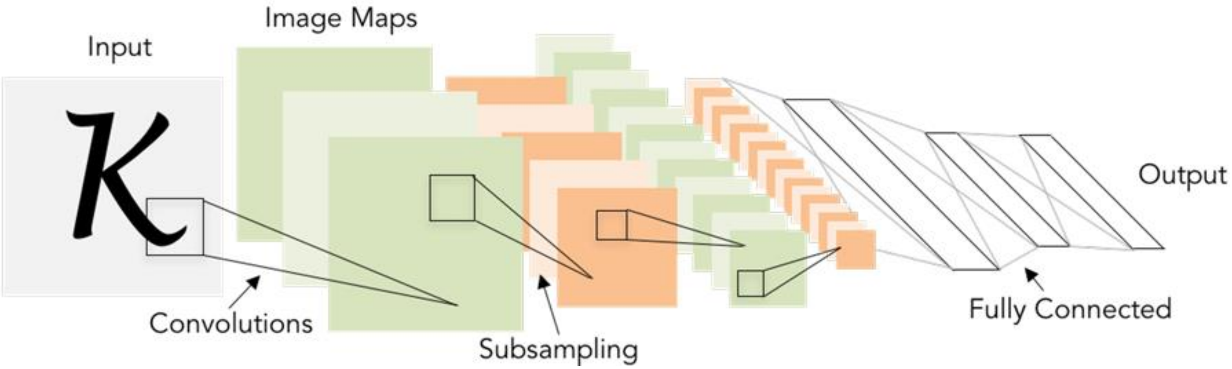
- As conv layers extract more informative features, can afford to lower resolution



- Typical pooling operations: MAX, MEAN
- Can be implemented similarly to filters

Other Layers

- Convolution layers apply *linear* operations, so it is still necessary to include *nonlinear* activation layers like ReLU
- Latter layers will act like a usual neural network, so these will typically be *fully connected* to compute flexible nonlinear functions of activation maps
- Output will be a typical softmax layer for classification



LeNet, LeCun et al. 1998

ImageNet

- ConvNets process images *end-to-end* since they perform both feature extraction and classification simultaneously

- Lots of data sets to train on, like ImageNet: over 14M images, 30k categories

- ConvNets surpassed human accuracy in annual competition in the 2010s

- Modern architectures continue to improve over classical methods

Summary

- Vision is perception that transforms light into digital images
- Computer vision gathers, processes, and analyzes image data

- One important way of understanding images is to extract features
- Hand-designed features are often insufficient for image classification

- Convolutional neural networks are especially effective to resolve these issues
- Advantages: Spatial locality, parameter sharing, known and proven architectures