Classical and Modern Recognition Techniques
Today’s outline

• We’ve covered Deep Convolutional Networks. But what did recognition techniques look like before AlexNet?
  – Bag of words models
  – Sliding window models
• What do more recent deep learning architectures look like?
How many visual object categories are there?

~10,000 to 30,000

Biederman 1987
~10,000 to 30,000
Specific recognition tasks
Scene categorization or classification

- outdoor/indoor
- city/forest/factory/etc.
Image annotation / tagging / attributes

- street
- people
- building
- mountain
- tourism
- cloudy
- brick
- ...

Svetlana Lazebnik
Object detection

- find pedestrians
Image parsing / semantic segmentation
Scene understanding?
Variability:
- Camera position
- Illumination
- Shape parameters

Within-class variations?
Within-class variations

Svetlana Lazebnik
History of ideas in recognition

- 1960s – early 1990s: the geometric era
Variability: Camera position Illumination

Alignment

Shape: assumed known

Roberts (1965); Lowe (1987); Faugeras & Hebert (1986); Grimson & Lozano-Perez (1986); Huttenlocher & Ullman (1987)
Recall: Alignment

- Alignment: fitting a model to a transformation between pairs of features (matches) in two images

\[
\sum_i \text{residual } (T(x_i), x'_i)
\]
Recognition as an alignment problem: Block world


**Fig. 1.** A system for recognizing 3-d polyhedral scenes. a) L.G. Roberts. b) A blocks world scene. c) Detected edges using a 2x2 gradient operator. d) A 3-d polyhedral description of the scene, formed automatically from the single image. e) The 3-d scene displayed with a viewpoint different from the original image to demonstrate its accuracy and completeness. (b) - (e) are taken from [64] with permission MIT Press.)

Representing and recognizing object categories is harder...

ACRONYM (Brooks and Binford, 1981)
Binford (1971), Nevatia & Binford (1972), Marr & Nishihara (1978)
Recognition by components

Biederman (1987)

Primitives (geons) → Objects

- Cube (Straight Edge, Straight Axis, Constant)
- Wedge (Straight Edge, Straight Axis, Expanded)
- Pyramid (Straight Edge, Straight Axis, Expanded)
- Cylinder (Curved Edge, Straight Axis, Constant)
- Barrel (Curved Edge, Straight Axis, End & Start)
- Arch (Straight Edge, Curved Axis, Constant)
- Cone (Curved Edge, Straight Axis, Expanded)
- Expanded Cylinder (Curved Edge, Straight Axis, Expanded)
- Handle (Curved Edge, Curved Axis, Constant)
- Expanded Handle (Curved Edge, Curved Axis, Expanded)


Svetlana Lazebnik
Generalized cylinders
Ponce et al. (1989)

Zisserman et al. (1995)

General shape primitives?

Forsyth (2000)
History of ideas in recognition

• 1960s – early 1990s: the geometric era
• 1990s: appearance-based models
Empirical models of image variability

**Appearance-based techniques**

Turk & Pentland (1991); Murase & Nayar (1995); etc.
Eigenfaces (Turk & Pentland, 1991)

- The image shows a set of facial images with different lighting, orientation, and scale conditions.
- The table below lists the experimental conditions and their corresponding correct/unknown recognition percentages:

<table>
<thead>
<tr>
<th>Condition</th>
<th>Correct/Unknown Recognition Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forced classification</td>
<td>96/0</td>
</tr>
<tr>
<td>Forced 100% accuracy</td>
<td>100/19</td>
</tr>
<tr>
<td>Forced 20% unknown rate</td>
<td>100/20</td>
</tr>
<tr>
<td>Lighting</td>
<td>85/0</td>
</tr>
<tr>
<td>Orientation</td>
<td>100/39</td>
</tr>
<tr>
<td>Scale</td>
<td>64/0</td>
</tr>
<tr>
<td>Scale</td>
<td>100/60</td>
</tr>
<tr>
<td>Scale</td>
<td>74/20</td>
</tr>
</tbody>
</table>
Color Histograms

History of ideas in recognition

• 1960s – early 1990s: the geometric era
• 1990s: appearance-based models
• 1990s – present: sliding window approaches
Sliding window approaches
Sliding window approaches

- Turk and Pentland, 1991
- Belhumeur, Hespanha, & Kriegman, 1997
- Schneiderman & Kanade 2004
- Viola and Jones, 2000

- Schneiderman & Kanade, 2004
- Argawal and Roth, 2002
- Poggio et al. 1993
History of ideas in recognition

- 1960s – early 1990s: the geometric era
- 1990s: appearance-based models
- Mid-1990s: sliding window approaches
- Late 1990s: local features
Local features for object instance recognition

Large-scale image search
Combining local features, indexing, and spatial constraints

Image credit: K. Grauman and B. Leibe
Large-scale image search
Combining local features, indexing, and spatial constraints

Philbin et al. '07
Large-scale image search
Combining local features, indexing, and spatial constraints
History of ideas in recognition

- 1960s – early 1990s: the geometric era
- 1990s: appearance-based models
- Mid-1990s: sliding window approaches
- Late 1990s: local features
- Early 2000s: parts-and-shape models
Parts-and-shape models

- Model:
  - Object as a set of parts
  - Relative locations between parts
  - Appearance of part

Figure from [Fischler & Elschlager 73]
Constellation models

Pictorial structure model

Fischler and Elschlager (73), Felzenszwalb and Huttenlocher (00)

\[
\Pr(P_{\text{tor}}, P_{\text{arm}}, \ldots | \text{Im}) \propto \prod_{i,j} \Pr(P_i | P_j) \prod_i \Pr(\text{Im}(P_i))
\]

part geometry

part appearance
Discriminatively trained part-based models

History of ideas in recognition

• 1960s – early 1990s: the geometric era
• 1990s: appearance-based models
• Mid-1990s: sliding window approaches
• Late 1990s: local features
• Early 2000s: parts-and-shape models
• Mid-2000s: bags of features
Bag-of-features models
Bag-of-features models

Object \rightarrow \text{Bag of ‘words’}
Objects as texture

• All of these are treated as being the same

• No distinction between foreground and background: scene recognition?
Origin 1: Texture recognition

- Texture is characterized by the repetition of basic elements or *textons*
- For stochastic textures, it is the identity of the textons, not their spatial arrangement, that matters

Origin 1: Texture recognition

Origin 2: Bag-of-words models

Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary  
  Salton & McGill (1983)
Origin 2: Bag-of-words models

• Orderless document representation: frequencies of words from a dictionary
  Salton & McGill (1983)

US Presidential Speeches Tag Cloud
http://chir.ag/phernalia/preztags/
Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary  
  Salton & McGill (1983)
Bag-of-features steps

1. Extract features
2. Learn “visual vocabulary”
3. Quantize features using visual vocabulary
4. Represent images by frequencies of “visual words”
1. Feature extraction

• Regular grid or interest regions
1. Feature extraction

- Compute descriptor
- Normalize patch
- Detect patches

Slide credit: Josef Sivic
1. Feature extraction

Slide credit: Josef Sivic
2. Learning the visual vocabulary

Slide credit: Josef Sivic
2. Learning the visual vocabulary

Clustering

Slide credit: Josef Sivic
2. Learning the visual vocabulary

Cluster visual vocabulary

Clustering

Slide credit: Josef Sivic
Example codebook

Source: B. Leibe
Visual vocabularies: Issues

• How to choose vocabulary size?
  • Too small: visual words not representative of all patches
  • Too large: quantization artifacts, overfitting

• Computational efficiency
  • Vocabulary trees
    (Nister & Stewenius, 2006)
But what about layout?

All of these images have the same color histogram
Spatial pyramid

Compute histogram in each spatial bin
Spatial pyramid representation

- Extension of a bag of features
- Locally orderless representation at several levels of resolution

Lazebnik, Schmid & Ponce (CVPR 2006)
Spatial pyramid representation

- Extension of a bag of features
- Locally orderless representation at several levels of resolution

Lazebnik, Schmid & Ponce (CVPR 2006)
Spatial pyramid representation

- Extension of a bag of features
- Locally orderless representation at several levels of resolution

Lazebnik, Schmid & Ponce (CVPR 2006)
Scene category dataset

*Multi-class classification results*
*(100 training images per class)*

<table>
<thead>
<tr>
<th>Level</th>
<th>Weak features (vocabulary size: 16)</th>
<th>Strong features (vocabulary size: 200)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Single-level</td>
<td>Pyramid</td>
</tr>
<tr>
<td>0 (1 × 1)</td>
<td>45.3 ±0.5</td>
<td></td>
</tr>
<tr>
<td>1 (2 × 2)</td>
<td>53.6 ±0.3</td>
<td>56.2 ±0.6</td>
</tr>
<tr>
<td>2 (4 × 4)</td>
<td>61.7 ±0.6</td>
<td>64.7 ±0.7</td>
</tr>
<tr>
<td>3 (8 × 8)</td>
<td>63.3 ±0.8</td>
<td>66.8 ±0.6</td>
</tr>
</tbody>
</table>
Caltech101 dataset


Multi-class classification results (30 training images per class)

<table>
<thead>
<tr>
<th>Level</th>
<th>Weak features (16)</th>
<th>Strong features (200)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Single-level</td>
<td>Pyramid</td>
</tr>
<tr>
<td>0</td>
<td>15.5 ± 0.9</td>
<td>41.2 ± 1.2</td>
</tr>
<tr>
<td>1</td>
<td>31.4 ± 1.2</td>
<td>32.8 ± 1.3</td>
</tr>
<tr>
<td>2</td>
<td>47.2 ± 1.1</td>
<td>49.3 ± 1.4</td>
</tr>
<tr>
<td>3</td>
<td>52.2 ± 0.8</td>
<td>54.0 ± 1.1</td>
</tr>
</tbody>
</table>
History of ideas in recognition

• 1960s – early 1990s: the geometric era
• 1990s: appearance-based models
• Mid-1990s: sliding window approaches
• Late 1990s: local features
• Early 2000s: parts-and-shape models
• Mid-2000s: bags of features
• Present trends: combination of local and global methods, context, 
  deep learning
Beyond AlexNet
These are the “VGG” networks.
“Perceptual Loss” in generative deep learning refers to these networks.
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>A-LRN</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Layers</td>
<td>11</td>
<td>11</td>
<td>13</td>
<td>16</td>
<td>16</td>
<td>19</td>
</tr>
<tr>
<td>Input</td>
<td>(224 × 224 RGB image)</td>
<td>conv3-64</td>
<td>conv3-64</td>
<td>conv3-64</td>
<td>conv3-64</td>
<td>conv3-64</td>
</tr>
<tr>
<td></td>
<td>maxpool</td>
<td>conv3-128</td>
<td>conv3-128</td>
<td>conv3-128</td>
<td>conv3-128</td>
<td>conv3-128</td>
</tr>
<tr>
<td></td>
<td>maxpool</td>
<td>conv3-256</td>
<td>conv3-256</td>
<td>conv3-256</td>
<td>conv3-256</td>
<td>conv3-256</td>
</tr>
<tr>
<td></td>
<td>maxpool</td>
<td>conv3-512</td>
<td>conv3-512</td>
<td>conv3-512</td>
<td>conv3-512</td>
<td>conv3-512</td>
</tr>
<tr>
<td></td>
<td>maxpool</td>
<td>FC-4096</td>
<td>FC-4096</td>
<td>FC-1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Number of parameters (in millions).

<table>
<thead>
<tr>
<th>Network</th>
<th>A,A-LRN</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of parameters</td>
<td>133</td>
<td>133</td>
<td>134</td>
<td>138</td>
<td>144</td>
</tr>
<tr>
<td>ConvNet config. (Table 1)</td>
<td>smallest image side</td>
<td>top-1 val. error (%)</td>
<td>top-5 val. error (%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>train ( (S) )</td>
<td>test ( (Q) )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>256</td>
<td>224,256,288</td>
<td>28.2</td>
<td>9.6</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>256</td>
<td>224,256,288</td>
<td>27.7</td>
<td>9.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>384</td>
<td>352,384,416</td>
<td>27.8</td>
<td>9.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( [256; 512] )</td>
<td>256,384,512</td>
<td>26.3</td>
<td>8.2</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>256</td>
<td>224,256,288</td>
<td>26.6</td>
<td>8.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>384</td>
<td>352,384,416</td>
<td>26.5</td>
<td>8.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( [256; 512] )</td>
<td>256,384,512</td>
<td>\textbf{24.8}</td>
<td>\textbf{7.5}</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>256</td>
<td>224,256,288</td>
<td>26.9</td>
<td>8.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>384</td>
<td>352,384,416</td>
<td>26.7</td>
<td>8.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( [256; 512] )</td>
<td>256,384,512</td>
<td>\textbf{24.8}</td>
<td>\textbf{7.5}</td>
<td></td>
</tr>
</tbody>
</table>
This is the “Inception” architecture or “GoogLeNet”

*The architecture blocks are called “Inception” modules and the collection of them into a particular net is “GoogLeNet”*
(a) Inception module, naïve version

(b) Inception module with dimensionality reduction
<table>
<thead>
<tr>
<th>type</th>
<th>patch size/strdie</th>
<th>output size</th>
<th>depth</th>
<th>#1×1</th>
<th>#3×3 reduce</th>
<th>#3×3</th>
<th>#5×5 reduce</th>
<th>#5×5</th>
<th>pool proj</th>
<th>params</th>
<th>ops</th>
</tr>
</thead>
<tbody>
<tr>
<td>convolution</td>
<td>7×7/2</td>
<td>112×112×64</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.7K</td>
<td>34M</td>
</tr>
<tr>
<td>max pool</td>
<td>3×3/2</td>
<td>56×56×64</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>112K</td>
<td>360M</td>
</tr>
<tr>
<td>convolution</td>
<td>3×3/1</td>
<td>56×56×192</td>
<td>2</td>
<td>64</td>
<td>192</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>112K</td>
<td>360M</td>
</tr>
<tr>
<td>max pool</td>
<td>3×3/2</td>
<td>28×28×192</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>inception (3a)</td>
<td>28×28×256</td>
<td>2</td>
<td>64</td>
<td>96</td>
<td>128</td>
<td>16</td>
<td>32</td>
<td>32</td>
<td>159K</td>
<td>128M</td>
<td></td>
</tr>
<tr>
<td>inception (3b)</td>
<td>28×28×480</td>
<td>2</td>
<td>128</td>
<td>128</td>
<td>192</td>
<td>32</td>
<td>96</td>
<td>64</td>
<td>380K</td>
<td>304M</td>
<td></td>
</tr>
<tr>
<td>max pool</td>
<td>3×3/2</td>
<td>14×14×480</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>inception (4a)</td>
<td>14×14×512</td>
<td>2</td>
<td>192</td>
<td>96</td>
<td>208</td>
<td>16</td>
<td>48</td>
<td>64</td>
<td>364K</td>
<td>73M</td>
<td></td>
</tr>
<tr>
<td>inception (4b)</td>
<td>14×14×512</td>
<td>2</td>
<td>160</td>
<td>112</td>
<td>224</td>
<td>24</td>
<td>64</td>
<td>64</td>
<td>437K</td>
<td>88M</td>
<td></td>
</tr>
<tr>
<td>inception (4c)</td>
<td>14×14×512</td>
<td>2</td>
<td>128</td>
<td>128</td>
<td>256</td>
<td>24</td>
<td>64</td>
<td>64</td>
<td>463K</td>
<td>100M</td>
<td></td>
</tr>
<tr>
<td>inception (4d)</td>
<td>14×14×528</td>
<td>2</td>
<td>112</td>
<td>144</td>
<td>288</td>
<td>32</td>
<td>64</td>
<td>64</td>
<td>580K</td>
<td>119M</td>
<td></td>
</tr>
<tr>
<td>inception (4e)</td>
<td>14×14×832</td>
<td>2</td>
<td>256</td>
<td>160</td>
<td>320</td>
<td>32</td>
<td>128</td>
<td>128</td>
<td>840K</td>
<td>170M</td>
<td></td>
</tr>
<tr>
<td>max pool</td>
<td>3×3/2</td>
<td>7×7×832</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>inception (5a)</td>
<td>7×7×832</td>
<td>2</td>
<td>256</td>
<td>160</td>
<td>320</td>
<td>32</td>
<td>128</td>
<td>128</td>
<td>1072K</td>
<td>54M</td>
<td></td>
</tr>
<tr>
<td>inception (5b)</td>
<td>7×7×1024</td>
<td>2</td>
<td>384</td>
<td>192</td>
<td>384</td>
<td>48</td>
<td>128</td>
<td>128</td>
<td>1388K</td>
<td>71M</td>
<td></td>
</tr>
<tr>
<td>avg pool</td>
<td>7×7/1</td>
<td>1×1×1024</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1000K</td>
<td>1M</td>
</tr>
<tr>
<td>dropout (40%)</td>
<td>1×1×1024</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1000K</td>
<td>1M</td>
</tr>
<tr>
<td>linear</td>
<td>1×1×1000</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1000K</td>
<td>1M</td>
</tr>
<tr>
<td>softmax</td>
<td>1×1×1000</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1000K</td>
<td>1M</td>
</tr>
</tbody>
</table>

Only 6.8 million parameters. AlexNet ~60 million, VGG up to 138 million
<table>
<thead>
<tr>
<th>Team</th>
<th>Year</th>
<th>Place</th>
<th>Error (top-5)</th>
<th>Uses external data</th>
</tr>
</thead>
<tbody>
<tr>
<td>SuperVision</td>
<td>2012</td>
<td>1st</td>
<td>16.4%</td>
<td>no</td>
</tr>
<tr>
<td>SuperVision</td>
<td>2012</td>
<td>1st</td>
<td>15.3%</td>
<td>Imagenet 22k</td>
</tr>
<tr>
<td>Clarifai</td>
<td>2013</td>
<td>1st</td>
<td>11.7%</td>
<td>no</td>
</tr>
<tr>
<td>Clarifai</td>
<td>2013</td>
<td>1st</td>
<td>11.2%</td>
<td>Imagenet 22k</td>
</tr>
<tr>
<td>MSRA</td>
<td>2014</td>
<td>3rd</td>
<td>7.35%</td>
<td>no</td>
</tr>
<tr>
<td>VGG</td>
<td>2014</td>
<td>2nd</td>
<td>7.32%</td>
<td>no</td>
</tr>
<tr>
<td>GoogLeNet</td>
<td>2014</td>
<td>1st</td>
<td>6.67%</td>
<td>no</td>
</tr>
</tbody>
</table>

Table 2: Classification performance.

<table>
<thead>
<tr>
<th>Number of models</th>
<th>Number of Crops</th>
<th>Cost</th>
<th>Top-5 error</th>
<th>compared to base</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>10.07%</td>
<td>base</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>10</td>
<td>9.15%</td>
<td>-0.92%</td>
</tr>
<tr>
<td>1</td>
<td>144</td>
<td>144</td>
<td>7.89%</td>
<td>-2.18%</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>7</td>
<td>8.09%</td>
<td>-1.98%</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>70</td>
<td>7.62%</td>
<td>-2.45%</td>
</tr>
<tr>
<td>7</td>
<td>144</td>
<td>1008</td>
<td>6.67%</td>
<td>-3.45%</td>
</tr>
</tbody>
</table>
ConvNet Depth

ImageNet Classification top-5 error (%)

- ILSVRC'14 GoogleNet: 6.7%, 22 layers
- ILSVRC'14 VGG: 7.3%, 19 layers
- ILSVRC'13: 11.7%, 8 layers
- ILSVRC'12 AlexNet: 16.4%
- ILSVRC'11: 25.8%
- ILSVRC'10: 28.2% shallow
Surely it would be ridiculous to go any deeper...

• To be continued with ResNet