Overall goals for the course

• Introduce fundamental concepts in computer vision

• Enable one or all of several such outcomes
  – Pursue higher studies in computer vision
  – Join industry to do cutting-edge work in computer vision
  – Gain appreciation of modern computer vision technologies

• Engage in discussions and interaction

• This is a great time to study computer vision!
Course details

• Class webpage:
  – http://cseweb.ucsd.edu/~mkchandraker/classes/CSE152A/Fall2022/

• TAs
  – Meng Song, Mallikarjun Swamy, Rishi Chandrasekaran, Vishal Vinod: 
    {mes050, mswamy, r3chandr, vvinod}@ucsd.edu

• Tutors
  – Nick Chua, Navya Sharma, Ang Li: {nchua, n1sharma, a3li}@ucsd.edu

• Discussion section: M 3-3:50pm

• Office hours posted on course calendar

• Piazza: https://piazza.com/ucsd/fall2022/cse152a/
Self-Study Assignment

• Pick a technology area primarily driven by computer vision
  – Can pick one of these suggestions, or use anything else that you like

• Write a 1 page essay (single-spaced)
  – Can be longer (hopefully not too long)
  – Great if you include pictures (with citations)

• Example prompts (feel free to add to these or use your own):
  – How does computer vision overcome barriers or solve user needs in this technology?
  – Can you identify where knowledge of photometric or geometric image formation is used?
  – Can you identify where such knowledge does not suffice and machine learning is used?
  – Can you identify where photometric or geometric models are used along with learning?
  – What was possible in this area 10 years ago and how did computer vision advance it?
  – How do you anticipate technology in this chosen area will advance in the next 10 years?

• Due date: Nov 11, 2022
Recap
Challenges: intra-class variation

Can you design an algorithm to describe a chair?
~10,000 to 30,000
Object categorization: the statistical viewpoint

\[ p(\text{zebra} | \text{image}) \]

or

\[ p(\text{no zebra} | \text{image}) \]

• Bayes rule:

\[
\frac{p(\text{zebra} | \text{image})}{p(\text{no zebra} | \text{image})} = \frac{p(\text{image} | \text{zebra})}{p(\text{image} | \text{no zebra})} \cdot \frac{p(\text{zebra})}{p(\text{no zebra})}
\]

posterior ratio

likelihood ratio

prior ratio
Discriminative

- Direct modeling of \( \frac{p(zebra \mid image)}{p(no\ zebra \mid image)} \)

Decision boundary

Zebra

Non-zebra

CSE 152A, FA22: Manmohan Chandraker
Generative

- Model $p(image \mid zebra)$ and $p(image \mid no \ zebra)$

<table>
<thead>
<tr>
<th>$p(image \mid zebra)$</th>
<th>$p(image \mid no \ zebra)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>High</td>
<td>Low</td>
</tr>
</tbody>
</table>

CSE 152A, FA22: Manmohan Chandraker
Instance Recognition

Recognizing or retrieving specific objects from a large collection

Find these landmarks
...in these images and 1M more

Slide credit: J. Sivic
Why is it difficult?

Want to find the object despite possibly large changes in scale, viewpoint, lighting and partial occlusion.
Origin: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary  
  Salton & McGill (1983)
Bags of features for object recognition

- Works quite well for image-level classification and for recognizing object *instances*

CSE 152A, FA22: Manmohan Chandraker
Recall: matching local features

To generate **candidate matches**, find patches that have the most similar appearance (e.g., lowest SSD)

Simplest approach: compare them all, take the closest (or closest k, or within a thresholded distance)
Indexing local features

Training dataset

Test image

CSE 152A, FA22: Manmohan Chandraker
Bag of features: outline

1. Extract features
Bag of features: outline

1. Extract features
2. Learn “visual vocabulary”
Bag of features: outline

1. Extract features
2. Learn “visual vocabulary”
3. Represent images by frequencies of “visual words”
Aside: Histograms

- A statistical tool to represent properties of a data distribution
- You own a shop, want to determine whether to hand out candies to visitors
- Collect ages of visitors:
  - [9, 34, 47, 23, 15, 18, 57, 73, 89, 20, 14, 47, 67, 93, 27, 32, 75, 5, 52, 32, 12, 33, 64, 29]
Aside: Histograms

• A statistical tool to represent properties of a data distribution

• You own a shop, want to determine whether to hand out candies to visitors

• Collect ages of visitors:
  – [9, 34, 47, 23, 15, 18, 57, 73, 89, 20, 14, 47, 67, 93, 27, 32, 75, 5, 52, 32, 12, 33, 64, 29]

<table>
<thead>
<tr>
<th>Age range</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 – 20</td>
<td>7</td>
</tr>
<tr>
<td>21 – 40</td>
<td>6</td>
</tr>
<tr>
<td>41 – 60</td>
<td>4</td>
</tr>
<tr>
<td>61 – 80</td>
<td>5</td>
</tr>
<tr>
<td>81 – 100</td>
<td>2</td>
</tr>
</tbody>
</table>
Aside: Histograms

- A statistical tool to represent properties of a data distribution

- You own a shop, want to determine whether to hand out candies to visitors

- Collect ages of visitors:
  - [9, 34, 47, 23, 15, 18, 57, 73, 89, 20, 14, 47, 67, 93, 27, 32, 75, 5, 52, 32, 12, 33, 64, 29]

<table>
<thead>
<tr>
<th>Age range</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 – 20</td>
<td>7</td>
</tr>
<tr>
<td>21 – 40</td>
<td>6</td>
</tr>
<tr>
<td>41 – 60</td>
<td>4</td>
</tr>
<tr>
<td>61 – 80</td>
<td>5</td>
</tr>
<tr>
<td>81 – 100</td>
<td>2</td>
</tr>
</tbody>
</table>
Aside: Histograms

• A statistical tool to represent properties of a data distribution

• You own a shop, want to determine whether to hand out candies to visitors

• Collect ages of visitors:
  – [9, 34, 47, 23, 15, 18, 57, 73, 89, 20, 14, 47, 67, 93, 27, 32, 75, 5, 52, 32, 12, 33, 64, 29]

<table>
<thead>
<tr>
<th>Age range</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 – 20</td>
<td>7</td>
</tr>
<tr>
<td>21 – 40</td>
<td>6</td>
</tr>
<tr>
<td>41 – 60</td>
<td>4</td>
</tr>
<tr>
<td>61 – 80</td>
<td>5</td>
</tr>
<tr>
<td>81 – 100</td>
<td>2</td>
</tr>
</tbody>
</table>

- Enough visitors under 20, so give candies (anyway, you should always give candies)
Return: Bag of features

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

Regular grid
- Vogel & Schiele, 2003
- Fei-Fei & Perona, 2005
1. Feature extraction

Regular grid
  • Vogel & Schiele, 2003
  • Fei-Fei & Perona, 2005

Interest point detector
  • Csurka et al. 2004
  • Fei-Fei & Perona, 2005
  • Sivic et al. 2005
Indexing local features

- Each patch or region has a descriptor, which is a point in some high-dimensional feature space (for example, image patches or SIFT)

A 128-dimensional vector used to represent image patch (computed based on image gradients)
Indexing local features

- When we see close points in feature space, we have similar descriptors, which indicates similar local content.
Visual words: main idea

- Extract some local features from a number of images …

Example: For 11x11 image patch, each point is 121-dimensional

Example: For SIFT descriptor space, each point is 128-dimensional
Visual words: main idea

Example: For 11x11 image patch, each point is 121-dimensional.

Example: For SIFT descriptor space, each point is 128-dimensional.
Visual words: main idea

Example: For 11x11 image patch, each point is 121-dimensional

Example: For SIFT descriptor space, each point is 128-dimensional
Visual words: main idea

Example: For 11x11 image patch, each point is 121-dimensional.

Example: For SIFT descriptor space, each point is 128-dimensional.
Each point is a local descriptor for an image patch
2. Learning the visual vocabulary
2. Learning the visual vocabulary

Clustering

Slide credit: Josef Sivic
2. Learning the visual vocabulary

Clustering

Visual vocabulary

Clustering

CSE 152A, FA22: Manmohan Chandraker

Slide credit: Josef Sivic
Visual words

- Example: each group of patches belongs to the same visual word

Figure from Sivic & Zisserman, ICCV 2003
Aside: K-means clustering

- Want to minimize sum of squared Euclidean distances between points $x_i$ and their nearest cluster centers $\bar{c}_k$

$$D(X, C) = \sum_{c_k} \sum_{i \in c_k} (x_i - \bar{c}_k)^2$$

Algorithm:
- Randomly initialize K cluster centers
- Iterate until convergence:
  - Assign each data point to the nearest center
  - Recompute each cluster center as the mean of all points assigned to it

CSE 152A, FA22: Manmohan Chandraker
3. Image representation

- Histogram where each bin is a codeword
- Counts number of patches (or features) in image closest to each codeword
- Codeword = visual word = cluster center
4. Image classification

- Given the bag-of-features representations of images from different classes, how do we learn a model for distinguishing them?
Comparing bags of words

• Compare test image $q$ to each training image $d_j$ using normalized dot product between their histograms

$$sim(d_j, q) = \frac{\langle d_j, q \rangle}{\|d_j\| \|q\|}$$

$$= \frac{\sum_{i=1}^{V} d_j(i) \ast q(i)}{\sqrt{\sum_{i=1}^{V} d_j(i)^2} \ast \sqrt{\sum_{i=1}^{V} q(i)^2}}$$

for vocabulary of $V$ words
Nearest Neighbor classification

- Assign label of nearest training data point to each test data point

Black = negative
Red = positive

Novel test example closest to a positive example from the training set, so classify it as positive.
K-Nearest Neighbors classification

- For a new point, find the $k$ closest points from training data
- Labels of the $k$ points “vote” to classify

If query lands here, the 5 NN consist of 3 negatives and 2 positives, so we classify it as negative.

Source: D. Lowe
Nearest neighbors

• **Advantages:**
  – Simple to implement
  – Flexible to feature or distance choices
  – Can do well in practice with enough representative data

• **Limitations:**
  – Large search problem to find nearest neighbors
  – Storage of data
Bayesian Estimation
Skin classification techniques

Skin classifier

• Given color $X$ of a pixel: how to determine if it is skin or not?
• Nearest neighbor
  – find labeled pixel closest in color to $X$
  – choose the label for that pixel
• Probabilistic data modeling
  – fit a probability model to each class
Probability

Basic probability

- $X$ is a random variable
- $P(X)$ is the probability that $X$ achieves a certain value

Called a PDF
- Probability distribution/density function
- A 2D PDF is a surface, 3D PDF is a volume

- $0 \leq P(X) \leq 1$

- $\int_{-\infty}^{\infty} P(X)\,dX = 1$  \hspace{1cm} \text{or} \hspace{1cm} \sum P(X) = 1$
  - Continuous $X$
  - Discrete $X$

- Conditional probability: $P(X \mid Y)$
  - Probability of $X$ given that we already know $Y$

CSE 152A, FA22: Manmohan Chandraker
Probabilistic skin classification

Now we can model uncertainty

- Each pixel has a probability of being skin or not skin, given color $R$
  - $P(\sim \text{skin}|R) = 1 - P(\text{skin}|R)$

Skin classifier

- Given color $R$ at a pixel: how to determine if it is skin or not?
- Choose interpretation of highest probability
  - Set $R$ to be a skin pixel if and only if $R_1 < R \leq R_2$

Where do we get $P(\text{skin}|R)$ and $P(\sim \text{skin}|R)$?
Learning conditional PDFs

\[ P(R|\text{skin}) = \frac{\text{#skin pixels with color } R}{\text{#skin pixels}} \]

We can calculate \( P(R \mid \text{skin}) \) from a set of labeled training images

- It is simply a histogram over the pixels in the training images
  - Each bin \( R_i \) contains the proportion of skin pixels in dataset with color \( R_i \)
Learning conditional PDFs

We can calculate $P(R|\text{skin})$ from a set of labeled training images

- It is simply a histogram over the pixels in the training images
  - Each bin $R_i$ contains the proportion of skin pixels in dataset with color $R_i$

But this isn’t quite what we want

- Why not? How to determine if a pixel is skin?
- We want $P(\text{skin} | R)$, not $P(R | \text{skin})$
- How can we get it?
Bayes rule

\[ P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)} \]

In terms of our problem:

- what we measure (likelihood)
- domain knowledge (prior)

\[ P(\text{skin}|R) = \frac{P(R|\text{skin})P(\text{skin})}{P(R)} \]

what we want (posterior)

The prior: \( P(\text{skin}) \)

- Could use domain knowledge
  - \( P(\text{skin}) \) may be larger if we know the image contains a person
  - For a portrait, \( P(\text{skin}) \) may be higher for pixels in the center
- Could learn the prior from the training set. How?
  - \( P(\text{skin}) \) could be the proportion of skin pixels in training set

\[ P(R) = P(R|\text{skin})P(\text{skin}) + P(R|\sim \text{skin})P(\sim \text{skin}) \]
Bayesian estimation

- Choose label (skin or ~skin) to maximize posterior (computed with Bayes rule)
  - This is called Maximum A Posteriori (MAP) estimation

Bayesian estimation

\[ P(R|\text{skin}) \]

\[ P(R|\sim\text{skin}) \]

\[ P(R|\text{skin})P(\text{skin}) \]

\[ P(R|\sim\text{skin})P(\sim\text{skin}) \]

\[ P(\text{skin}) = 0.75 \]

\[ P(\sim\text{skin}) = 0.25 \]

= minimize probability of misclassification
Bayesian estimation

- Choose label (skin or ~skin) to maximize posterior (computed with Bayes rule)
  - This is called **Maximum A Posteriori (MAP) estimation**
- Suppose the prior is uniform: \( P(\text{skin}) = P(\sim\text{skin}) = 0.5 \)
  - In this case \( P(\text{skin}|R) = cP(R|\text{skin}) \), \( P(\sim\text{skin}|R) = cP(R|\sim\text{skin}) \)
  - Maximizing the posterior is equivalent to maximizing the likelihood
    » \( P(\text{skin}|R) > P(\sim\text{skin}|R) \) if and only if \( P(R|\text{skin}) > P(R|\sim\text{skin}) \)
  - This is called **Maximum Likelihood (ML) estimation**

CSE 152A, FA22: Manmohan Chandraker
Skin classification results

Figure 25.3. The figure shows a variety of images together with the output of the skin detector of Jones and Rehg applied to the image. Pixels marked black are skin pixels, and white are background. Notice that this process is relatively effective, and could certainly be used to focus attention on, say, faces and hands. Figure from “Statistical color models with application to skin detection,” M.J. Jones and J. Rehg, Proc. Computer Vision and Pattern Recognition, 1999 © 1999, IEEE.