Announcements
- Exercise sheet 3 will be made available today
  - Hough Transform
  - Mean-shift segmentation [last week’s topic]
  - Histogram based object recognition [today’s topic]
  - The exercise will be on Tuesday, 04.12.
  => Submit your results by Monday night.

Course Outline
- Image Processing Basics
- Segmentation
  - Segmentation and Grouping
  - Graph-Theoretic Segmentation
- Recognition
  - Global Representations
  - Subspace representations
- Local Features & Matching
- Object Categorization
- 3D Reconstruction
- Motion and Tracking

Recap: Image Segmentation
- Goal: identify groups of pixels that go together

Recap: Images as Graphs
- Fully-connected graph
  - Node (vertex) for every pixel
  - Link between every pair of pixels, (p,q)
  - Affinity weight $w_{pq}$ for each link (edge)
  - $w_{pq}$ measures similarity
  - Similarity is inversely proportional to difference
    (in color and position...)

Recap: Normalized Cut (NCut)
- A minimum cut penalizes large segments
- This can be fixed by normalizing for size of segments
- The normalized cut cost is:
  \[
  NCut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}
  \]
  \[
  assoc(A,V) = \sum_{p \in A} \sum_{q \in V} w_{pq}
  \]
- The exact solution is NP-hard but an approximation can be computed by solving a generalized eigenvalue problem.
  - J. Shi and J. Malik. Normalized cuts and image segmentation. PAMI 2000
Recap: NCuts: Overall Procedure
1. Construct a weighted graph $G=(V,E)$ from an image.
2. Connect each pair of pixels, and assign graph edge weights,
   $W(i,j) = \text{Prob. that } i \text{ and } j \text{ belong to the same region}.$
3. Solve $(D-W)y = \lambda D y$ for the smallest few eigenvectors. This yields a continuous solution.
4. Threshold eigenvectors to get a discrete cut
   (This is where the approximation is made (we’re not solving NP).
5. Recursively subdivide if NCut value is below a pre-specified value.

NCuts Matlab code available at 
http://www.cis.upenn.edu/~jshi/software/

Note: Skipping MRF Segmentation for now...
• Will present this at later point...

Topics of This Lecture
• Object Recognition
  • Appearance-based recognition
  • Global representations
  • Color histograms
• Recognition using histograms
  • Histogram comparison measures
  • Histogram backprojection
  • Multidimensional histograms
• Probabilistic Interpretation
  • Probability density estimation
  • Recognition from local samples
  • Extension: recognition of multiple objects in an image
  • Extension: colored derivatives

Object Recognition

Challenges
• Viewpoint changes
  • Translation
  • Image-plane rotation
  • Scale changes
  • Out-of-plane rotation
• Illumination
• Noise
• Clutter
• Occlusion

Appearance-Based Recognition
• Basic assumption
  • Objects can be represented by a set of images ("appearances").
  • For recognition, it is sufficient to just compare the 2D appearances.
• No 3D model is needed.

⇒ Fundamental paradigm shift in the 90’s
Global Representation

- **Idea**
  - Represent each object (view) by a global descriptor.
  - For recognizing objects, just match the descriptors.
  - Some modes of variation are built into the descriptor, the others have to be incorporated in the training data.
    - e.g. a descriptor can be made invariant to image-plane rotations.
    - Other variations:
      - Viewpoint changes
      - Translation
      - Scale changes
      - Noise
      - Clutter
      - Out-of-plane rotation
      - Occlusion

Color: Use for Recognition

- **Color:**
  - Color stays constant under geometric transformations
  - Local feature
    - Color is defined for each pixel
    - Robust to partial occlusion
  - **Idea**
    - Directly use object colors for recognition
    - Better: use statistics of object colors

Color Histograms

- **Color statistics**
  - Here: RGB as an example
  - Given: tristimulus R,G,B for each pixel
  - Compute 3D histogram
    - \( H(R,G,B) = \# \text{pixels with color (R,G,B)} \)

Color Normalization

- **One component of the 3D color space is intensity**
  - If a color vector is multiplied by a scalar, the intensity changes, but not the color itself.
  - This means colors can be normalized by the intensity.
  - Intensity is given by \( I = R + G + B \):
    - "Chromatic representation"
    - \( r = \frac{R}{R+G+B} \)
    - \( g = \frac{G}{R+G+B} \)
    - \( b = \frac{B}{R+G+B} \)

Color Normalization

- **Observation:**
  - Since \( r + g + b = 1 \), only 2 parameters are necessary
  - E.g. one can use \( r \) and \( g \)
  - and obtains \( b = 1 - r - g \)
Color Histograms

- Use for recognition
  - Works surprisingly well
    - In the first paper (1991), 66 objects could be recognized almost without errors

Topics of This Lecture

- Object Recognition
  - Appearance-based recognition
  - Global representations
  - Color histograms

- Recognition using histograms
  - Histogram comparison measures
  - Histogram backprojection
  - Multidimensional histograms

- Probabilistic Interpretation
  - Probability density estimation
  - Recognition from local samples
  - Extension: recognition of multiple objects in an image
  - Extension: colored derivatives

Recognition Using Histograms

- Histogram comparison

What Is a Good Comparison Measure?

- How to define matching cost?

Comparison Measures: Euclidean Distance

- Definition
  - Euclidean Distance (\(L_2\) norm)
  \[ d(Q, V) = \sum_i (q_i - v_i)^2 \]

- Motivation
  - Focuses on the differences between the histograms.
  - Interpretation: distance in feature space.
  - Range: \([0, \infty)\]
  - All cells are weighted equally.
  - Not very robust to outliers!
Comparison Measures: Mahalanobis Distance

- **Definition**
  - Mahalanobis distance (Quadratic Form)
  \[ d(Q, V) = (Q - V)^T \Sigma^{-1} (Q - V) \]
  \[ = \sum_i \sum_j (q_i - v_i)(q_j - v_j) \sigma_{ij} \]

- **Motivation**
  - Interpretation:
    - Weighted distance in feature space.
    - Compensate for correlated data.
  - Range: \([0, \infty]\)
  - More robust to certain outliers.

Comparison Measures: Chi-Square

- **Definition**
  - Chi-square
  \[ \chi^2(Q, V) = \sum_i (q_i - v_i)^2 \]

- **Motivation**
  - Statistical background:
    - Test if two distributions are different
    - Possible to compute a significance score
  - Range: \([0, \infty]\)
  - Cells are not weighted equally!
  - More robust to outliers than Euclidean distance.
    - If the histograms contain enough observations...

Comparison Measures: Bhattacharyya Distance

- **Definition**
  - Bhattacharyya coefficient
  \[ BC(Q, V) = \sum q_i \sqrt{v_i} \]
  - Common distance measure:
  \[ d_{BC}(Q, V) = \sqrt{1 - BC(Q, V)} \]

- **Motivation**
  - Statistical background
    - \(BC\) measures the statistical separability between two distributions.
    - Range: \([0, \infty]\)
    - (Reason for \(d_{BC}\): triangle inequality)

Comparison Measures: Kullback-Leibler

- **Definition**
  - KL-divergence
  \[ KL(Q, V) = \sum q_i \log \frac{q_i}{v_i} \]

- **Motivation**
  - Information-theoretic background:
    - Measures the expected difference (#bits) required to code samples from distribution \(Q\) when using a code based on \(Q\) vs. based on \(V\).
    - Also called: information gain, relative entropy
    - Not symmetric!
    - Symmetric version: Jeffreys divergence
      \[ JD(Q, V) = KL(Q, V) + KL(V, Q) \]

Comparison Measures: Histogram Intersection

- **Definition**
  - Intersection
  \[ \cap(Q, V) = \sum_i \min(q_i, v_i) \]

- **Motivation**
  - Measures the common part of both histograms
  - Range: \([0, 1]\)
  - For unnormalized histograms, use the following formula
  \[ \cap(Q, V) = \frac{1}{2} \left( \frac{\sum_i \min(q_i, v_i)}{q_i} + \frac{\sum_i \min(q_i, v_i)}{v_i} \right) \]

Comparison Measures: Earth Movers Distance

- **Motivation**: Moving Earth
Comp. Measures: Earth Movers Distance

• Motivation: Moving Earth
  
  Slide adapted from Pete Barnum

((distance moved) * (amount moved))

Slide adapted from Pete Barnum

• Motivation: Moving Earth

**Linear Programming Problem**

m clusters

n clusters

Q

V

V

Q

What is the minimum amount of work to convert Q into V?

Slide adapted from Pete Barnum

EMD Computation

• Constraints

1. Move “earth” only from Q to V

Q

V

Q'

V'

f_{ij} \geq 0

Slide credit: Pete Barnum

Slide adapted from Pete Barnum
EMD Computation

- Constraints

2. Cannot send more “earth” than there is

\[ \sum_{j=1}^{n} f_{ij} \leq w_{q_i} \]

3. V cannot receive more than it can hold

\[ \sum_{i=1}^{m} f_{ij} \leq w_{v_j} \]

4. As much “earth” as possible must be moved.
   - Either Q must be completely spent
   - Or V must be completely filled.

\[ \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} = \min \left( \sum_{i=1}^{m} w_{q_i}, \sum_{j=1}^{n} w_{v_j} \right) \]

Comp. Measures: Earth Movers Distance

- Motivation: Moving Earth
  - Linear Programming Problem
  - Distance measure
    \[ D_{EMD}(Q,V) = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij} \]

- Advantages
  - Nearness measure without quantization
  - Partial matching
  - A true metric

- Disadvantage: expensive computation
  - Efficient algorithms available for 1D
  - Approximations for higher dimensions...

Summary: Comparison Measures

- Vector space interpretation
  - Euclidean distance
  - Mahalanobis distance

- Statistical motivation
  - Chi-square
  - Bhattacharyya

- Information-theoretic motivation
  - Kullback-Leibler divergence, Jeffreys divergence

- Histogram motivation
  - Histogram intersection

- Ground distance
  - Earth Movers Distance (EMD)
Histogram Comparison

- Which measure is best?
  - Depends on the application...
  - Euclidean distance is often not robust enough.
  - Both Intersection and $\chi^2$ give good performance for histograms.
    - Intersection is a bit more robust.
    - $\chi^2$ is a bit more discriminative.
  - KL/Jeffrey works sometimes very well, but is expensive.
  - EMD is most powerful, but also quite expensive.
  - There exist many other measures not mentioned here
    - e.g. statistical tests: Kolmogorov-Smirnov
    - Cramer/Von-Mises
  - ...

Summary: Recognition Using Histograms

- Simple algorithm
  1. Build a set of histograms $H = \{h_i\}$ for each known object
     - More exactly, for each view of each object
  2. Build a histogram $h_t$ for the test image.
  3. Compare $h_t$ to each $h_i \in H$
     - Using a suitable comparison measure
  4. Select the object with the best matching score
     - Or reject the test image if no object is similar enough.

Topics of This Lecture

- Object Recognition
  - Appearance-based recognition
  - Global representations
  - Color histograms
- Recognition using histograms
  - Histogram comparison measures
  - Histogram backprojection
  - Multidimensional histograms
- Probabilistic Interpretation
  - Probability density estimation
  - Recognition from local samples
  - Extension: recognition of multiple objects in an image
  - Extension: colored derivatives

Localization by Histogram Backprojection

- „Where in the image are the colors we’re looking for?“
  - Idea: Normalized histogram represents probability distribution
    $$p(x|obj)$$
  - Histogram backprojection
    - For each pixel $x$, compute the likelihood that this pixel color was caused by the object: $p(x|obj)$.
    - This value is projected back into the image (i.e. the image values are replaced by the corresponding histogram values).

Color-Based Skin Detection

- Used 18,696 images to build a general color model.
- Histogram representation

Discussion: Color Histograms

- Pros
  - Invariant to object translation & rotation
  - Slowly changing for out-of-plane rotation
  - No perfect segmentation necessary
  - Histograms change gradually when part of the object is occluded
  - Possible to recognize deformable objects
    - e.g. pullover
- Cons
  - Pixel colors change with the illumination ("color constancy problem")
    - Intensity
    - Spectral composition (illumination color)
  - Not all objects can be identified by their color distribution.
Topics of This Lecture

- Object Recognition
  - Appearance-based recognition
  - Global representations
  - Color histograms

- Recognition using histograms
  - Histogram comparison measures
  - Histogram backprojection
  - Multidimensional histograms

- Probabilistic Interpretation
  - Probability density estimation
  - Recognition from local samples
  - Extension: recognition of multiple objects in an image
  - Extension: colored derivatives

Generalization of the Idea

- Histograms of derivatives
  - $D_x$
  - $D_y$
  - $D_{xx}$
  - $D_{xy}$
  - $D_{yy}$

General Filter Response Histograms

- Any local descriptor (e.g. filter, filter combination) can be used to build a histogram.

- Examples:
  - Gradient magnitude
    \[ Mag = \sqrt{D_x^2 + D_y^2} \]
  - Gradient direction
    \[ Dir = \arctan\frac{D_y}{D_x} \]
  - Laplacian
    \[ Lap = D_{xx} + D_{yy} \]

Multidimensional Representations

- Combination of several descriptors
  - Each descriptor is applied to the whole image.
  - Corresponding pixel values are combined into one feature vector.
  - Feature vectors are collected in multidimensional histogram.

Multidimensional Histograms

- Examples

Multidimensional Representations

- Useful simple combinations
  - $D_x D_y$
    - Rotation-variant
      - Descriptor changes when image is rotated.
      - Useful for recognizing oriented structures (e.g. vertical lines)
  - $Mag - Lap$
    - Rotation-invariant
      - Descriptor does not change when image is rotated.
      - Can be used to recognize rotated objects.
      - Less discriminant than rotation-variant descriptor.
Generalization: Filter Banks

• What filters to put in the bank?
  - Typically we want a combination of scales and orientations, different types of patterns.

Matlab code available for these examples:
http://www.robots.ox.ac.uk/~vgg/research/texclass/filters.html

Example Application of a Filter Bank

Scales
Orientations

Filter bank of 8 filters
Input image

8 response images: magnitude of filtered outputs, per filter

Color vs. Texture

• These images look very similar in terms of their color distributions (when our features are R-G-B)
• But how would their texture distributions compare?

Special Case: Multiscale Representations

• Combination of several scales
  - Descriptors are computed at different scales.
  - Each scale captures different information about the object.
  - Size of the support region grows with increasing \( \sigma \).
  - Feature vectors capture both local details and larger-scale structures.

Summary: Multidimensional Representations

• Pros
  - Work very well for recognition.
  - Usually, simple combinations are sufficient (e.g. \( D_x, D_y, \text{Mag-Lap} \))
  - But multiple scales are very important!
  - Generalization: filter banks

• Cons
  - High-dimensional histograms \( \Rightarrow \) lots of storage space
  - Global representation \( \Rightarrow \) not robust to occlusion

Topics of This Lecture

• Object Recognition
  - Appearance-based recognition
  - Global representations
  - Color histograms

• Recognition using histograms
  - Histogram comparison measures
  - Histogram backprojection
  - Multidimensional histograms

• Probabilistic Interpretation
  - Probability density estimation
  - Recognition from local samples
  - Extension: recognition of multiple objects in an image
  - Extension: colored derivatives
From Global To Local...

- Up to now, we have compared entire histograms.
  \[ \Rightarrow \text{Problematic if objects can be partially occluded.} \]
- Now:
  - Look at local measurements only.
  - What can we tell if we only see a single pixel of the object?

Recall: Working with Probabilities

- Random Variables:
  - $A, B$
- Probabilities:
  - $\Pr(A), \Pr(B)$
- Joint probability:
  - $\Pr(A, B)$
- Conditional probability:
  - $\Pr(A \mid B)$

Recall: Manipulation Rules

- Factorization of the joint
  \[ \Pr(A, B) = \Pr(A \mid B) \Pr(B) = \Pr(B \mid A) \Pr(A) \]
- Marginalization
  \[ \Pr(A) = \sum_i \Pr(A, b_i) = \sum_i \Pr(A \mid b_i) \Pr(b_i) = \sum_i \Pr(b_i \mid A) \Pr(A) \]
- Bayes theorem
  \[ \Pr(A \mid B) = \frac{\Pr(B \mid A) \Pr(A)}{\Pr(B)} \]

Probabilistic Derivation

- Probability of object $o_n$ given measurement $m_k$
  \[ p(o_n | m_k) \]
  - Bayes
  - Marginalization
  - \[ p(o_n | m_k) = \frac{p(m_k | o_n) p(o_n)}{\sum_j p(m_k | o_j) p(o_j)} \]
  - with
    - $p(o_n)$ the prior probability of object $o_n$.
    - $p(m_k)$ the prior probability of measurement $m_k$.
    - $p(m_k | o_n)$ the likelihood of the data given the model, i.e. the probability of the measurement $m_k$ under the model $o_n$.
  \[ \Rightarrow \text{We can read off } p(m_k | o_n) \text{ directly from the histogram.} \]
Probabilistic Recognition

- Assumption: all objects equally probable ("naive Bayes")

\[
p(o_i) = \frac{1}{N}
\]

\[
p(o_n|m_k) = \frac{p(m_k|o_n)p(o_n)}{\sum_i p(m_k|o_i)p(o_i)}
\]

\[
= \frac{1}{N} \sum_i p(m_k|o_i)
\]

\[
= \frac{1}{N} \sum_i p(m_k|o_i)
\]

value of hist. cell

sum over all objects

- Joint probability for two measurements

\[
p(o_n|m_k \land m_j) = \frac{p(m_k \land m_j|o_n)p(o_n)}{\sum_i p(m_k \land m_j|o_i)p(o_i)}
\]

- Assumption: \( m_k \) and \( m_j \) are independent

The individual probabilities can be multiplied

\[
p(o_n|m_k \land m_j) = \frac{p(m_k|o_n)p(m_j|o_n)p(o_n)}{\sum_i p(m_k|o_i)p(m_j|o_i)p(o_i)}
\]

Bayesian Recognition Algorithm

1. Build up histograms \( p(m_k|o_n) \) for each training object.

2. Sample the test image to obtain \( m_k \), \( k \in K \).
   - Only small number of local samples necessary.

3. Compute the probabilities for each training object.

\[
p(o_n|Image) = \frac{\prod_k p(m_k|o_n)p(o_n)}{\sum_i \prod_k p(m_k|o_i)p(o_i)}
\]

4. Select the object with the highest probability
   - Or reject the test image if no object accumulates sufficient probability.

Practical Issues

- Most expensive step

3. Compute the probabilities for each training object.

\[
p(o_n|Image) = \frac{\prod_k p(m_k|o_n)p(o_n)}{\sum_i \prod_k p(m_k|o_i)p(o_i)}
\]

- Notes

  - The numerator computes a score indicating how probable each object \( o_i \) in the database is.
  - This score can be used to compare the different object hypotheses.

Advantage

- Can already generate hypotheses from a small number of measurements

- Visible object portion of 10-20% may already be enough!
Practical Issues
- Most expensive step
  1. Compute the probabilities for each training object.
     \[ p(o_i|\text{Image}) = \frac{\prod_j p(m_j|o_i)p(o_i)}{\sum_i \prod_j p(m_j|o_i)p(o_i)} \]
- Notes
  - The numerator computes a score indicating how probable each object \( o_i \) in the database is.
    - This score can be used to compare the different object hypotheses.
  - The denominator is the same for all objects in the database.
    - This term is important in order to decide if we have accumulated sufficient evidence to make a decision.

Results: Probabilistic (Bayesian) Recognition
- Test database
  - 103 test objects
  - 1327 test images total
    - 607 images with scale changes and rotations for 83 objects
    - 720 images with different viewpoints for 20 objects
  - Use 6D descriptor
    - \( D_y D_x \) with \( \sigma = \{1, 2, 4\} \)
      - explicitly trained for scale changes & rotations

Experimental Evaluation
- Recognition under Partial Occlusion
  - Compare intersection, \( \chi^2 \), and probabilistic recognition
- Results
  - Intersection more robust to occlusion than \( \chi^2 \)
  - Probabilistic recognition most robust
    - 62% visibility \( \Rightarrow \) 100% recognition
    - 33% visibility \( \Rightarrow \) 99% recognition
    - 13% visibility \( \Rightarrow \) >90% recognition

Topics of This Lecture
- Object Recognition
  - Appearance-based recognition
  - Global representations
  - Color histograms
- Recognition using histograms
  - Histogram comparison measures
  - Histogram backprojection
  - Multidimensional histograms
- Probabilistic Interpretation
  - Probability density estimation
  - Recognition from local samples
  - Extension: recognition of multiple objects in an image
  - Extension: colored derivatives

Extension: Colored Derivatives
- \( YC_1C_2 \) color space
  \[ \begin{pmatrix} Y \\ C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} g_r & g_y & g_b \\ 3g_r - 2g_y & 3g_y - 2g_b & 3g_b - 2g_r \\ g_r g_y + g_y g_b + g_b g_r & g_r g_y + g_y g_b + g_b g_r & g_r g_y + g_y g_b + g_b g_r \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \]
- Color-opponent space
  - Inspired by models of the human visual system
    - \( Y \) = intensity
    - \( C_1 \) = red-green
    - \( C_2 \) = blue-yellow
- Generalization: derivatives along
  - \( Y \) axis \( \rightarrow \) intensity differences
  - \( C_1 \) axis \( \rightarrow \) red-green differences
  - \( C_2 \) axis \( \rightarrow \) blue-yellow differences
- Feature vector is rotated such that \( D_y = 0 \)
  - Rotation-invariant descriptor
Application: Brand Identification in Video

Summary

• Appearance-based Object Recognition
  ○ Using global representations

• Histograms
  ○ Color histograms
  ○ Histogram comparison measures
  ○ Multidimensional histograms

• Probabilistic Recognition
  ○ Histograms as probability density estimates
  ○ Recognition from local measurements
  ○ Recognition of multiple objects in an image

References and Further Reading

• Background information on histogram-based object recognition can be found in the following paper

• Matlab filterbank code available at
  ▶ http://www.robots.ox.ac.uk/~vgg/research/texclass/filters.html