Today

- Some logistics
- Overview lecture on recognition models
- Discussion of bag-of-words and constellation model approaches

Schedule

Demo guidelines
Implement/download code for a core idea in the paper and show us toy examples:
- Experiment with different types of (mini) training/testing data sets
- Evaluate sensitivity to parameter settings
- Show (on a small scale) an example in practice that highlights a strength/weakness of the approach

- Want to consider illustrative example, not a system

Demo presentation format

- Give algorithm, relevant technical details
- Describe scope of experiments
- Present the experiments, explain rationale for outcomes
- Conclude with a summary of the messages

Timetable for presenters

- By the Wednesday the week before:
  – email slides to me, schedule time to meet and discuss.
- Week of:
  – refine slides, practice presentation, know about how long each part requires.
- Day of:
  – send me final slides as PDF file

For Feb 1 and Feb 8 presenters: by upcoming Wednesday and Friday
Reviews

- Submit one review per week unless you are presenting (but read all assigned papers)
- Evaluation:
  0 none
  1 “check –”: little effort/reflection
  2 “check”, good review
  3 “check+”, very good review

Possible levels of recognition

<table>
<thead>
<tr>
<th>Categories</th>
<th>Specific objects</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>butterfly</td>
<td>Wild card</td>
<td></td>
</tr>
<tr>
<td>butterfly</td>
<td>Tower Bridge</td>
<td></td>
</tr>
<tr>
<td>building</td>
<td>Bevo</td>
<td></td>
</tr>
<tr>
<td>building</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Recognition questions

- How to represent a category or object
- How to perform the recognition (classification, detection) with that representation
- How to learn models, new categories/objects

Representations

- Model-based
- Appearance-based
- Parts + structure
- Multi-view
- Bag of features

Learning

- What defines a category/class?
- What distinguishes classes from one another?
- How to understand the connection between the real world and what we observe?
- What features are most informative?
- What can we do without human intervention?
- Does previous learning experience help learn the next category?
Learning situations

- Varying levels of supervision
  - Unsupervised
  - Image labels
  - Object centroid/bounding box
  - Segmented object
  - Manual correspondence (typically sub-optimal)

Inputs/outputs/assumptions

- What input is available?
  - Static grayscale image
  - 3D range data
  - Video sequence
  - Multiple calibrated cameras
  - Segmented data, unsegmented data
  - CAD model
  - Labeled data, unlabeled data, partially labeled data

Inputs/outputs/assumptions

- What is the goal?
  - Say yes/no as to whether an object present in image
  - Determine pose of an object, e.g. for robot to grasp it
  - Categorize all objects
  - Forced choice from pool of categories
  - Bounding box on object
  - Full segmentation
  - Build a model of an object category

Outline

- Overview of recognition background
  - Model-based
  - Appearance-based
  - Local feature-based
    - Features and interest operators
    - Bags of words
    - Constellation models/part-based models

Model-based recognition

- Which image features correspond to which features on which object model in the "modelbase"?
- If enough match, and they match well with a particular transformation for given camera model, then
  - Identify the object as being there
  - Estimate pose relative to camera
Hypothesize and test: main idea

- Given model of object
- New image: hypothesize object identity and pose
- Render object in camera
- Compare rendering to actual image: if close, good hypothesis.

How to form a hypothesis?

Given a particular model object, we can estimate the correspondences between image and model features

Use correspondence to estimate camera pose relative to object coordinate frame

Generating hypotheses

We want a good correspondence between model features and image features.

- Brute force?

Brute force hypothesis generation

- For every possible model, try every possible subset of image points as matches for that model’s points.
- Say we have \( L \) objects with \( P \) features, \( N \) features found in the image

Generating hypotheses

We want a good correspondence between model features and image features.

- Brute force?
- Prune search via geometric or relational constraints: interpretation tree
- Pose consistency: use subsets of features to estimate larger correspondence
- Voting, pose clustering

Pose consistency / alignment

- Key idea:
  - If we find good correspondences for a small set of features, it is easy to obtain correspondences for a much larger set.
- Strategy:
  - Generate hypotheses using small numbers of correspondences (how many depends on camera type)
  - Backproject: transform all model features to image features
  - Verify
2d affine mappings

- Say camera is looking down perpendicularly on planar surface

- We have two coordinate systems (object and image), and they are related by some affine mapping (rotation, scale, translation, shear).

\[
\begin{bmatrix}
    x \\
    y \\
    t_x \\
    t_y \\
\end{bmatrix}
= \begin{bmatrix}
    m_1 & m_2 & x \\
    m_3 & m_4 & y \\
    0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
    u \\
    v \\
\end{bmatrix}
\]

Solving for the transformation parameters

- Rewrite in terms of unknown parameters

Alignment: backprojection

- Having solved for this transformation from some number of detected matches (3+ here), can compute (hypothesized) location of any other model points in the image space.

\[
\begin{bmatrix}
    u \\
    v \\
\end{bmatrix}
= \begin{bmatrix}
    m_1 & m_2 & x \\
    m_3 & m_4 & y \\
    0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
    x \\
    y \\
    t_x \\
    t_y \\
\end{bmatrix}
\]

Alignment: verification

- Given the backprojected model in the image:
  - Check if image edges coincide with predicted model edges
  - May be more robust if also require edges to have the same orientation
  - Consider texture in corresponding regions?
Pose clustering (voting)

- Narrow down the number of hypotheses to verify: identify those model poses that a lot of features agree on.
  - Use each group’s correspondence to estimate pose
  - Vote for that object pose in accumulator array (one array per object if we have multiple models)

Application: Surgery

- To minimize damage by operation planning
- To reduce number of operations by planning surgery
- To remove only affected tissue

Problem
  - ensure that the model with the operations planned on it and the information about the affected tissue lines up with the patient
  - display model information supervised on view of patient
  - Big Issue: coordinate alignment, as above
Summary: model-based recognition

- Hypothesize and test: looking for object and pose that fits well with image
  - Use good correspondences to designate hypotheses
  - Limit verifications performed by voting
- Requires model for the specific objects
  - Searching a modelbase
  - Registration tasks
- Requires camera model selection

Outline

- Overview of recognition background
  - Model-based
  - Appearance-based
  - Local feature-based
    - Features and interest operators
    - Bags of words
    - Constellation models

Limits of model-based recognition?
Global measure of appearance
- vector of pixel intensities
- grayscale / color histogram
- bank of filter responses, …

Global measure of appearance
- e.g., Color histogram

Color Histograms
Off-line, for each image
create histogram with a bin for each color
initialize each bin counter = 0
for each pixel in image:
increment bin counter corresponding to pixel color
end

On-line, use histograms in image similarity measure:
Euclidean, dot product, histogram intersection, etc.

Images Classified as Sunsets using Overall Color Histograms

Global measure of appearance
- e.g., responses to linear filters
- Applying filter = taking a dot-product between image and some vector
- Filtering the image is a set of dot products
- Insights
  - filters look like the effects they are intended to find
  - filters find effects they look like

Learning with global representations
- In addition to sorting images based on nearness in feature space, can learn classifiers
Learning with global representations

- In addition to sorting images based on nearness in feature space, can learn classifiers.

Windowed search

- Windowed correlation search: to find a fixed scale pattern.

Windowed search

- In general, simple way to check the global measure of appearance when the test image has clutter; search over scales, orientations,…

When are “global” representations (and window-based detection) appropriate?

Limitations of global representations

- Success may rely on alignment
- All parts of image impact description

Outline

- Overview of recognition background
  - Model-based
  - Appearance-based
  - Local feature-based
    - Features and interest operators
    - Bags of words
    - Constellation models
Local image features

- Illumination
- Object pose
- Clutter
- Occlusions
- Intra-class appearance
- Viewpoint

Classes of transformations

- **Euclidean/rigid**: Translation + rotation
- **Similarity**: Translation + rotation + uniform scale
- **Affine**: Similarity + shear
  - Valid for orthographic camera, locally planar object
- **Photometric**: affine intensity change
  - \( I \rightarrow aI + b \)

Invariant local features

Subset of local feature types designed to be invariant to
- Scale
- Translation
- Rotation
- Affine transformations
- Illumination

1) Detect distinctive interest points
2) Extract invariant descriptors

[Mikolajczyk & Schmid, Matas et al., Tuytelaars & Van Gool, Lowe, Kadir et al.,...]

History of local invariant features...

Dense correspondence search

For each epipolar line
- For each pixel / window in the left image
  - compare with every pixel / window on same epipolar line in right image
  - pick position with minimum match cost (e.g., SSD, correlation)

Adapted from Li Zhang
History of local invariant features…

Sparse correspondence search

• Restrict search to sparse set of detected features
• Rather than pixel values (or lists of pixel values) use feature descriptor and an associated feature distance
• Still narrow search further by epipolar geometry

Wide baseline stereo

• 3d reconstruction depends on finding good correspondences
• Especially with wide-baseline views, local image deformations not well-approximated with rigid transformations
• Cannot simply compare regions of fixed shape (circles, rectangles) – shape is not preserved under affine transformations

Interest points:

From stereo to recognition

• Feature detectors previously used for stereo, motion tracking
• Now also for recognition
  – Schmid & Mohr 1997
  • Harris corners to select interest points
  • Rotationally invariant descriptor of local image regions
  • Identify consistent clusters of matched features to do recognition
Matching with features

• We need to match (align) images

[These slides are from Darya Frolova and Denis Simakov]

Matching with Features

• Detect feature points in both images

Matching with Features

• Detect feature points in both images
• Find corresponding pairs

Matching with Features

• Problem 1:
  – Detect the same point independently in both images

  no chance to match!

  We need a repeatable detector

Matching with Features

• Problem 2:
  – For each point correctly recognize the corresponding one

  We need a reliable and distinctive descriptor
(Good) invariant local features

- Reliably detected
- Distinctive
- Robust to noise, blur, etc.
- Description normalized properly

Exhaustive search
A multi-scale approach

Key idea of invariance

We want to extract the patches from each image independently: features should adapt their shape, covariant with the affine transformation relating them.
Scale space (Witkin 83)

- First derivative peaks
- Contours of $f' = 0$ in scale-space

Gaussian filtered 1d signal $x$

Scale space insights:
- Edge position may shift with increasing scale ($\sigma$)
- Two edges may merge with increasing scale (edges can disappear)
- An edge may not split into two with increasing scale (new edges do not appear)

Adapted from Steve Seitz, UW

Scale Invariant Detection

- Consider regions of different sizes around a point
- At the right scale, regions of corresponding content will look the same in both images

Scale Invariant Detection

- The problem: how do we choose corresponding circles independently in each image?

- Solution:
  - Design a function on the region (circle), which is "scale invariant" (the same for corresponding regions, even if they are at different scales)
    - Example: average intensity. For corresponding regions (even of different sizes) it will be the same.
  - For a point in one image, we can consider it as a function of region size (circle radius)

Scale Invariant Detection

- Common approach:
  - Take a local maximum of this function
  - Observation: region size, for which the maximum is achieved, should be invariant to image scale.

Important: this scale invariant region size is found in each image independently!

Slide credit: Darya Frolova and Denis Simakov

Slide credit: Darya Frolova and Denis Simakov
Scale Invariant Detection

[Images from T. Tuytelaars]

Automatic scale selection

Lindeberg et al., 1996

Following example was created by T. Tuytelaars, ECCV 2006 tutorial
Scale Invariant Detection

- A "good" function for scale detection: has one stable sharp peak

- For usual images: a good function would be a one which responds to contrast (sharp local intensity change)

Scale selection principle

- Intrinsic scale is the scale at which normalized derivative assumes a maximum -- marks a feature containing interesting structure. (T. Lindeberg '94)

\[ \text{Maxima/minima of Laplacian} \]

Scale invariant detection

Requires a method to repeatably select points in location and scale:

- Only reasonable scale-space kernel is a Gaussian (Koenderink, 1984; Lindeberg, 1994)
- An efficient choice is to detect peaks in the difference of Gaussian pyramid (Burt & Adelson, 1983; Crowley & Parker, 1984)
- Difference-of-Gaussian is a close approximation to Laplacian
SIFT: Key point localization

- Detect maxima and minima of difference-of-Gaussian in scale space
- Then reject points with low contrast (threshold)
- Eliminate edge responses (use ratio of principal curvatures)

Candidate keypoints: list of (x, y, σ)

SIFT: Example of keypoint detection

Threshold on value at DOG peak and on ratio of principle curvatures

(a) 233x189 image
(b) 832 DOG extrema
(c) 729 left after peak value threshold
(d) 536 left after testing ratio of principle curvatures

Scale Invariant Detection: Summary

- Given: two images of the same scene with a large scale difference between them
- Goal: find the same interest points independently in each image
- Solution: search for maxima of suitable functions in scale and in space (over the image)

Affine Invariant Detection

- Intensity-based regions (IBR):
  - Start from a local intensity extrema
  - Consider intensity profile along rays
  - Select maximum of invariant function f(t) along each ray
  - Connect local maxima
  - Fit an ellipse

- Maximally Stable Extremal Regions (MSER)
  - Threshold image intensities: I > I₀
  - Extract connected components (“Extremal Regions”)
  - Seek extremal regions that remain “Maximally Stable” under range of thresholds

Point Descriptors

- We know how to detect points
- Next question: How to describe them for matching?

Point descriptor should be:
1. Invariant
2. Distinctive

Rotation Invariant Descriptors

- Find local orientation
  Dominant direction of gradient

- Rotate description relative to dominant orientation

Scale Invariant Descriptors

- Use the scale determined by detector to compute descriptor in a normalized frame

SIFT descriptors: Select canonical orientation

- Create histogram of local gradient directions computed at selected scale
- Assign canonical orientation at peak of smoothed histogram
- Each key specifies stable 2D coordinates (x, y, scale, orientation)

SIFT descriptors: vector formation

- Thresholded image gradients are sampled over 16x16 array of locations in scale space
- Create array of orientation histograms
- 8 orientations x 4x4 histogram array = 128 dimensions

SIFT properties

- Invariant to
  - Scale
  - Rotation
- Partially invariant to
  - Illumination changes
  - Camera viewpoint
  - Occlusion, clutter

2 D. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. Accepted to IJCV 2004
Revisiting model-based recognition with more powerful features:

Recognition with SIFT [Lowe]

1) Index descriptors (distinctive features narrow possible matches)
2) Hough transform to vote for poses (keypoints have record of parameters relative to model coordinate system)
3) Affine fit to check for agreement between model and image (approximates perspective projection for planar objects)

Planar objects

- Model images and their SIFT keypoints
- Input image
- Recognition result

Model keypoints that were used to recognize, get least squares solution.

Value of local (invariant) features

- Complexity reduction via selection of distinctive points
- Describe images, objects, parts without requiring segmentation
  - Local character means robustness to clutter, occlusion
- Robustness: similar descriptors in spite of noise, blur, etc.

3D objects

- Background subtract for model boundaries
- Objects recognized, though affine model not as accurate.
- Recognition in spite of occlusion

Local representations

Describe component regions or patches separately

Local features will be something we can match across images...
What possible models for objects and categories can be formed with local descriptors as the basis?
Outline

• Overview of recognition background
  – Model-based
  – Appearance-based
  – Local feature-based
    • Features and interest operators
      • Bags of words
      • Constellation models

Analogy to documents

Of all the sensory impressions proceeding to the brain, the visual experiences are the dominant ones. Our perception of the world around us is based essentially on the messages that reach the brain from our eyes. For a long time it was thought that the retinal image was transmitted point by point to visual centers in the brain; the cerebral cortex was a movie screen, so to speak, upon which the image in the eye was projected. Through the discoveries of Hubel and Wiesel we now know that behind the origin of the visual perception in the brain there is a considerably more complicated course of events. By following the visual impulses along their path to the various cell layers of the optical cortex, Hubel and Wiesel have been able to demonstrate that the retinal image falling on the retina undergoes a stepwise analysis in a system of nerve cells stored in columns. In this system each cell has its specific function and is responsible for a specific detail in the pattern of the retinal image.

China is forecasting a trade surplus of $90bn (€51bn) to $100bn this year, a threefold increase on 2004’s $33bn. The Commerce Ministry said the surplus would be created by a predicted 30% jump in exports to $750bn, compared with a 18% rise in imports to $660bn. The figures are likely to further annoy the US, which has long argued that China’s exports are unfairly helped by a deliberately undervalued yuan. Beijing agrees the surplus is too high, but says the yuan is only one factor. Bank of China governor Zhou Xiaochuan said the country also needed to do more to boost domestic demand so more goods stayed within the country. China increased the value of the yuan against the dollar by 2.1% in July and permitted it to trade within a narrow band, but the US wants the yuan to be allowed to float freely. However, Beijing has made it clear that it will take its time and tread carefully before allowing the yuan to rise further in value.

1. Feature detection and representation

• Regular grid
1. Feature detection and representation

- Regular grid
- Interest point detector

2. Codewords dictionary formation

- Compute SIFT descriptor
  - [Lowe'99]
- Normalize patch
- Detect patches
  - [Mikolajczyk and Schmid '02]
  - [Matas et al. '02]
  - [Sivic et al. '03]

1. Feature detection and representation

- Regular grid
- Interest point detector
- Other methods
  - Random sampling
  - Segmentation based patches

2. Codewords dictionary formation

- Vector quantization
Extract some local features from a number of images ...
3. Image representation

**Visual words = textons**

- Previous use of local feature quantization:
  - Texton = cluster center of filter responses over collection of images [Leung & Malik, 1999; Varma & Zisserman 2002]
- Represent texture or material with histogram of texton occurrences (or prototypes of whatever feature type employed)

**Today’s papers: two general ways to build a representation from local features**

- Bag of words
- Constellation models
**Next time: visual vocabularies**

- Interest operators, sampling strategy
- Quantization strategy
- Search, indexing structures

---

**Next time**

- Topic: visual vocabularies
- Presenter: Joseph
- Demo: Xin
- Papers to read (review one):