Recognition III & Motion II

Introduction to Computer Vision
CSE 152
Lecture 20
Announcements

• HW 4 due friday
• Final Exam: Wed, 6/13, 3:00PM

• Last lecture
  – Revisit PCA (Eigenfaces)
  – Fisherface
  – Bayesian Decision Theory (MAP Classifiers)
Variability: Camera position
Illumination
Internal parameters

Within-class variations

Set of Images
Appearance manifold approach

- for every object
  1. sample the set of viewing conditions
  2. Crop & scale images to standard size
  3. Use as feature vector
- apply a PCA over all the images
- keep the dominant PCs
- Set of views for one object is represented as a manifold in the projected space
- Recognition: What is nearest manifold for a given test image?

(Nayar et al. ‘96)
An example: input images
An example: basis images
An example: surfaces of first 3 coefficients
Parameterized Eigenspace
Recognition
Appearance-Based Vision: Lessons

Strengths

• Posing the recognition metric in the image space rather than a derived representation is more powerful than expected.
• Modeling objects from many images is not unreasonable given hardware developments.
• The data (images) may provide a better representations than abstractions for many tasks.
Appearance-Based Vision: Lessons

Weaknesses

- Segmentation or object detection is still an issue.
- To train the method, objects have to be observed under a wide range of conditions (e.g. pose, lighting, shape deformation). Is this feasible for all of these?
- Limited power to extrapolate or generalize (abstract) to novel conditions.
Model-Based Vision

- Given 3-D models of each object
- Detect image features (often edges, line segments, conic sections)
- Establish correspondence between model & image features
- Estimate pose
- Consistency of projected model with image.
A Rough Recognition Spectrum

Appearance-Based Recognition (Eigenface, Fisherface)

Shape Contexts

Geometric Invariants

Local Features + Spatial Relations

Aspect Graphs

3-D Model-Based Recognition

Image Abstractions/ Volumetric Primitives

Function
Motion
Structure-from-Motion (SFM)

Goal: Take as input two or more images or video w/o any information on camera position/motion, and estimate camera position and 3-D structure of scene.

Two Approaches

1. Discrete motion (wide baseline)
   1. Orthographic (affine) vs. Perspective
   2. Two view vs. Multi-view
   3. Calibrated vs. Uncalibrated

2. Continuous (Infinitesimal) motion
Epipolar Constraint: Calibrated Case

\[ \overrightarrow{O_p} \cdot [\overrightarrow{OO'} \times \overrightarrow{O'p}] = 0 \quad \Rightarrow \quad p \cdot [t \times (\mathcal{R}p')] = 0 \quad \text{with} \quad \begin{cases} p = (u, v, 1)^T \\ p' = (u', v', 1)^T \\ \mathcal{M} = (\text{Id} \ 0) \\ \mathcal{M}' = (\mathcal{R}^T, -\mathcal{R}^T t) \end{cases} \]

Essential Matrix
(Longuet-Higgins, 1981)

\[ p^T \mathcal{E} p' = 0 \quad \text{with} \quad \mathcal{E} = [t_x] \mathcal{R} \]

where \[ [t_x] = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \]
The Eight-Point Algorithm (Longuet-Higgins, 1981)

Let F denote the Essential Matrix Here

\[
\begin{pmatrix}
  F_{11} & F_{12} & F_{13} \\
  F_{21} & F_{22} & F_{23} \\
  F_{31} & F_{32} & F_{33}
\end{pmatrix}
\begin{pmatrix}
  u' \\
  v'
\end{pmatrix}
= 0
\]

Set \( F_{33} \) to 1

\[
\begin{pmatrix}
  u_1u'_1 & u_1v'_1 & u_1 & v_1u'_1 & v_1v'_1 & v_1 & u'_1 & v'_1 \\
  u_2u'_2 & u_2v'_2 & u_2 & v_2u'_2 & v_2v'_2 & v_2 & u'_2 & v'_2 \\
  u_3u'_3 & u_3v'_3 & u_3 & v_3u'_3 & v_3v'_3 & v_3 & u'_3 & v'_3 \\
  u_4u'_4 & u_4v'_4 & u_4 & v_4u'_4 & v_4v'_4 & v_4 & u'_4 & v'_4 \\
  u_5u'_5 & u_5v'_5 & u_5 & v_5u'_5 & v_5v'_5 & v_5 & u'_5 & v'_5 \\
  u_6u'_6 & u_6v'_6 & u_6 & v_6u'_6 & v_6v'_6 & v_6 & u'_6 & v'_6 \\
  u_7u'_7 & u_7v'_7 & u_7 & v_7u'_7 & v_7v'_7 & v_7 & u'_7 & v'_7 \\
  u_8u'_8 & u_8v'_8 & u_8 & v_8u'_8 & v_8v'_8 & v_8 & u'_8 & v'_8
\end{pmatrix}
\begin{pmatrix}
  F_{11} \\
  F_{12} \\
  F_{13} \\
  F_{21} \\
  F_{22} \\
  F_{23} \\
  F_{31} \\
  F_{32}
\end{pmatrix}
= 0
\]

Solve For \( F \)

Solve For \( R \) and \( t \)
Continuous Motion

• Consider a video camera moving continuously along a trajectory (rotating & translating).
• How do points in the image move?
• What does that tell us about the 3-D motion & scene structure?
The Motion Field
What causes a motion field?

1. Camera moves (translates, rotates)
2. Object’s in scene move rigidly
3. Objects articulate (pliers, humans, animals)
4. Objects bend and deform (fish)
5. Blowing smoke, clouds
Motion Field Yields 3-D Motion Information

The “instantaneous” velocity of points in an image

LOOMING

The Focus of Expansion (FOE)

Intersection of velocity vector with image plane

With just this information it is possible to calculate:

1. Direction of motion
2. Time to collision
Is motion estimation inherent in humans?

Demo
Rigid Motion and the Motion Field
Rigid Motion: General Case

Position & Orientation

\[ P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \]

\[ Z' = C_Z - Z \]

Position and orientation of a rigid body
Rotation Matrix & Translation vector

\[ \dot{p} = T + \omega \times p \]
General Motion

\[
\begin{bmatrix}
  u \\
  v
\end{bmatrix}
= f
\begin{bmatrix}
  x \\
  y
\end{bmatrix}
\]

\[
\begin{bmatrix}
  \dot{u} \\
  \dot{v}
\end{bmatrix}
= f
\begin{bmatrix}
  \dot{x} \\
  \dot{y}
\end{bmatrix}
- \frac{f \dot{z}}{z^2}
\begin{bmatrix}
  x \\
  y
\end{bmatrix}
= f
\begin{bmatrix}
  \dot{x} \\
  \dot{y}
\end{bmatrix}
- \frac{\dot{z}}{z}
\begin{bmatrix}
  u \\
  v
\end{bmatrix}
\]

Substitute \( \dot{p} = T + \omega \times p \) where \( p = (x, y, z)^T \)
Motion Field Equation

\[
\begin{align*}
\dot{u} &= \frac{T_z u - T_x f}{Z} - \omega_y f + \omega_z v + \frac{\omega_{xuv}}{f} - \frac{\omega_y u^2}{f} \\
\dot{v} &= \frac{T_z v - T_y f}{Z} + \omega_x f - \omega_z u - \frac{\omega_{yuv}}{f} - \frac{\omega_x v^2}{f}
\end{align*}
\]

- **T**: Components of 3-D linear motion
- **ω**: Angular velocity vector
- (u,v): Image point coordinates
- **Z**: depth
- **f**: focal length
Pure Translation

\[ \ddot{u} = \frac{T_z u - T_x f}{Z} - \omega_y f + \omega_z v + \frac{\omega_{xuv}}{f} - \frac{\omega_y u^2}{f} \]

\[ \ddot{v} = \frac{T_z v - T_y f}{Z} + \omega_x f - \omega_{zu} - \frac{\omega_{yuv}}{f} - \frac{\omega_x v^2}{f} \]

\[ \omega = 0 \]
Forward Translation & Focus of Expansion

[Gibson, 1950]
Pure Translation

Radial about FOE

Parallel (FOE point at infinity)

$T_Z = 0$

Motion parallel to image plane
Pure Rotation: $T=0$

\[
\begin{align*}
\dot{u} &= \frac{T u - T_x f}{Z} - \omega_y f + \omega_z v + \frac{\omega_{xuv}}{f} - \frac{\omega_{yuv}}{f} \\
\dot{v} &= \frac{T_z v - T_y f}{Z} + \omega_x f - \omega_z u - \frac{\omega_{yuv}}{f} - \frac{\omega_{xuv}}{f}
\end{align*}
\]

- Independent of $T_x \ T_y \ T_z$
- Independent of $Z$
- Only function of $(u,v)$, $f$ and $\omega$
Rotational MOTION FIELD

The “instantaneous” velocity of points in an image

PURE ROTATION

$\omega = (0,0,1)^T$
Motion Field Equation: Estimate Depth

\[
\begin{align*}
\dot{u} &= \frac{T_z u - T_x f}{Z} - \omega_y f + \omega_z v + \frac{\omega_{uv}}{f} - \frac{\omega_y u^2}{f} \\
\dot{v} &= \frac{T_z v - T_y f}{Z} + \omega_x f - \omega_z u - \frac{\omega_{uv}}{f} - \frac{\omega_x v^2}{f}
\end{align*}
\]

If \( T, \omega, \) and \( f \) are known or measured, then for each image point \((u,v)\), one can solve for the depth \( Z \) given measured motion \((du/dt, dv/dt)\) at \((u,v)\).
Pure Rotation: Motion Field on Sphere

Direction of Translation

Center of Projection

Axis of Rotation

Center of Projection
Optical Flow
Optical Flow:
Where do pixels move to?
Final Exam

• Closed book
• One cheat sheet
  – Single piece of paper, handwritten, no photocopying, no physical cut & paste.
• What to study
  – Basically material presented in class, and supporting material from text
  – If it was in text, but NEVER mentioned in class, it is very unlikely to be on the exam
• Question style:
  – Short answer
  – Some longer problems to be worked out.