CMPUT 428: Intro/Syllabus

Roderick Lan

January 9, 2024

Intro

Prof: Martin Jagersand
Can use NumPy or Matlab for operations
Labs: first lab for help/to learn, then turn in lab. Demo after

Exam 1 Feb 27 (20%)
Exam 2 Apr 2 (20%)
Project (used as Final, 20%)
Slides

1 Overview

1.1 Multiview Geometry

2D simpler, but same problem. Can carry out things, but don't always need "full models" (model only needs to consider relevant features).
Multiple points → projected onto surface. Can be done using only images. spacing

- ie. [typical processing pipeline:] take many images → figure out where cams are → project as rays, see where they intersect → create 3d model (merge point clouds) → use pictures to get texture/color
- When generating from webmining, can relate many "photo-consistent" images together by their relative position/orientation (determine where they are s.t. it minimizes error of reconstruction) → generate model
- Neural nets can learn to predict 3d maps, but won't understand geometry. (ie. trained network good for depth mapping, but not for video tracking)

1.2 Human Vision (Dorsal/Ventral Pathways)

Humans dont internalize detailed 3d maps, but we can still have representations.
Use vision as a process, direct motor responses based on it.
Object detection pathways separate from other vision pathways in brain.

1.3 Camera Geometry

Partial information used often (ie. for measurements in single img, get visual constraints, visual servoing, video tracking, rendering, etc.) spacing

> Visual servoing requires video tracking to perform well.
>
> More points/triangles → better rendering. Limited by source (?). Things like diffusion equations can be used to refine things like texture resolution of renders

Understandning (relative) coordinate systems in images is important. (ie. if you know pillars are equidistant, but not projected equiidistantly, can use that as coord sys to map image)

1.4 Geometry for Hand-Eye Coordination

Formulate as local reqs rather than global. (don't need to worry abt what the global shape is)

1.5 Intro to Image-based Visual Servoing

Given 'desired' img, sees 'initial' img. Goal is to transform initial to desired.

Can be done with just imgs, ie. take corner features and move arm until corner feature match desired spacing

> Difference b/w features formulated as error. Error minimized.
>
> RL problem does the same

Explain 1.1: Vis. Spec. Err.

Point to Point task Error: $E = [\vec{y}^* - \vec{y}_0]$

Point to Line: $E_{pl}(\vec{y}, \vec{l}) = \begin{bmatrix} \vec{y}_l \cdot \vec{l}_l \\ \vec{y}_r \cdot \vec{l}_r \end{bmatrix}$ where $\vec{l}_l = [y_2 \times y_3]_l$

1.6 Proj. Transformations

Uses rotation matrix H
Projective - 8 DOF
Affine - 6 DOF
Metric - 4 DOF
Euclidean - 3 DOF
Cannot find absolute size from imgs alone.

Example 1.1: Cut in the Middle Task

Get midpoint by drawing 2 lines from corners, finding intersection.
$x_m = (l_1 \times l_2)$
Get vanishing point x_∞ and intersect with midpoint

CV traditionally seen as converting images to models. Guiding robots more akin to what we want to do with CV in the future