Computer Vision for HCI

Interactive Applications

1

Computer Interaction

- Human-computer interaction has not changed for decades
  - Typing, pointing, clicking
- Most work on GUIs
  - WIMP (windows, icons, menus, pointer)
- Computers are becoming
  - Smaller
  - Ubiquitous
  - Pervasive in our lives

2

2

Computer Interaction

- Need is arising for more intuitive ways of interacting with the technology
  - Pointing, clicking, and typing will not be the main interface for long
- What is needed?
  - Match to how people interact with each other and the real world

3

3

Perceptual User Interfaces (PUIs)

- Integrates UI types for more natural and intuitive interfaces
  - Perceptive
  - Multi-modal
  - Multi-media
- Enhances GUI-based apps
  - "No, *that* one", with speech, gaze, and gesture
- Enables computers as assistants
- Enables multiple styles of interaction (context)

4

4

Gesture-Based 3D Man-Machine Interaction using a Single Camera

Senthil Kumar and Jakub Segen

Note: A really <u>old</u> paper, but a really <u>good</u> one!

5

5

Main Idea

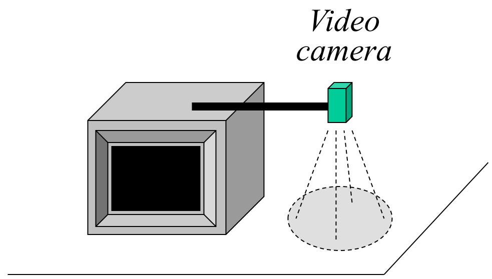
- Hand gesture input to control computer applications
- Tracks user's hand in 3D
- Three gestures interpreted as discrete computer commands
- Single camera, real-time processing
- Example interfaces
  - 3D virtual flythrough
  - Graphical scene composer
  - Video games

6

6

Camera Setup

- Camera mounted on monitor looking down at table-top

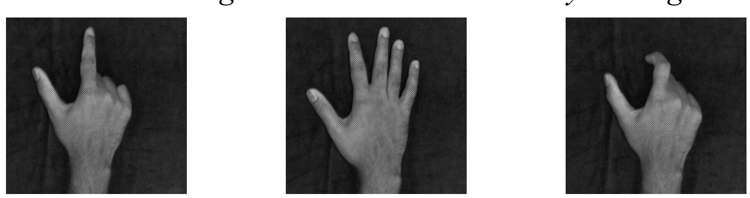


7

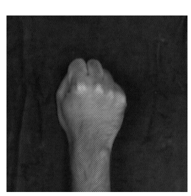
7

Gesture Classes

Static gestures — *Dynamic gesture*



"Point" "Reach" "Click"



"Ground"

(All other gestures, empty image)

8

8

Computer Vision Approach

- Based on boundary analysis of hand region
- First classify into {*Point, Reach, Click, Ground*}
- If recognize "Point", then compute fingertip
  - Position (x, y)
  - Orientation θ
  - Relative height z (from table top)

9

9

Local Feature Classifier

- Extract connected-component region
  - After background-subtraction
- Make list of boundary pixels (x_i, y_i)
  - Clockwise order
  - Perimeter length used to screen out non-hands
- Compute curvature measure at each boundary point
  - Angle between [P(i-k), P(i)] and [P(i), P(i+k)]
- Curvature extrema used to find "peaks" and "valleys"
  - Use threshold to find only large magnitudes

10

10

Local Feature Classifier

- Preliminary static gesture classification

(class = "Point") if ($N_{peaks} > T_{p1}$) and ($N_{peaks} < T_{p2}$)

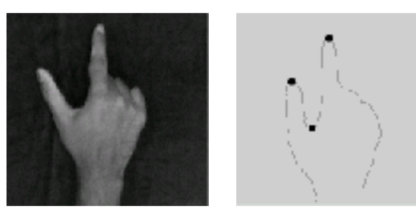
(class = "Reach") if ($N_{peaks} > T_{r1}$) or ($N_{valleys} > T_{r2}$)

- Result sets variable class to "Point", "Reach", or "Ground"

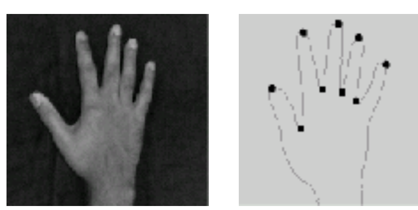
11

11

Peaks and Valleys



Point



Reach

12

12

Click Detection

- A "Click" is defined by
  - Whole hand not significantly moved from last frame, and fingertip position shifted inward toward the hand region

13

Position and Orientation

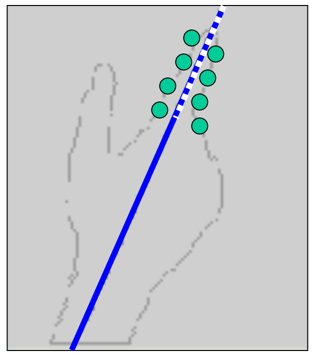
- Only computed if class found is "Point"
- From fingertip location
  - Iteratively step away (k=1..k_{max}) from peak simultaneously to left and right

    $PL_k = (xl_k, yl_k) \quad PR_k = (xr_k, yr_k)$
  - Compute midpoints down along finger

    $$Q(k) = (PL_k + PR_k)/2$$
- Compute "spine" of finger using least-squares of the midpoints $Q(k)$
  - Intersection of line with boundary gives "new" fingertip
  - Orientation of line gives orientation of finger pose

14

Line Obtained by LSQ Fit



15

15

Relative Height

- Measure proximity of hand to camera
  - Does not compute elevation angle of finger
- Measure finger width in image

  $d(k) = ||PL_k - PR_k||$

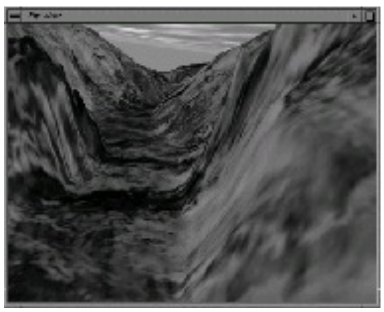
  $M = \text{average}_k(d(k))$
- Average measurement M varies smoothly with distance of finger from camera
  - Get initial M measurement (when hand first enters image and sets to z=0)
  - Deviations from M in new images gives z

16

16

Controlling Virtual Flight over Yosemite Valley

- Velocity controlled by fingertip position
  - Velocity increases as hand moves forward
- Orientation angle controls direction of flight
- Height controls elevation of flight

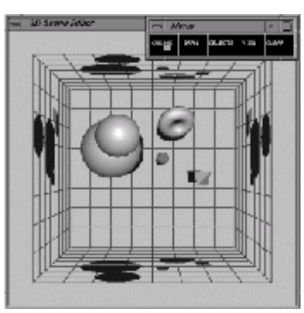


17

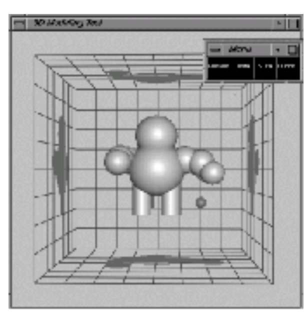
17

3D Graphical Editor

- Hand controls cursor
  - Menu mode: "Point" moves 2D cursor, "Reach" activates menu
  - Cursor mode: "Point" moves 3D cursor, "Reach-Point" selects/releases object



3D graphical editor

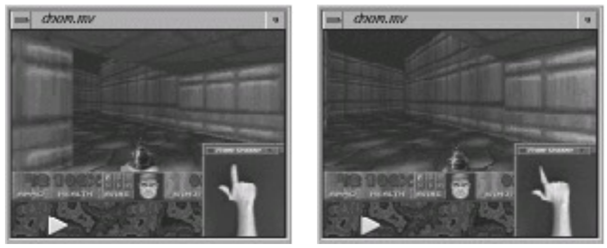


"Man" constructed

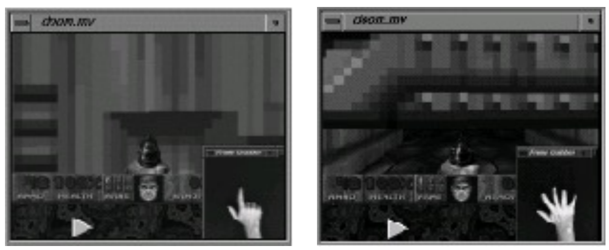
18

18

Controlling Video Games: Doom



“Point” navigation

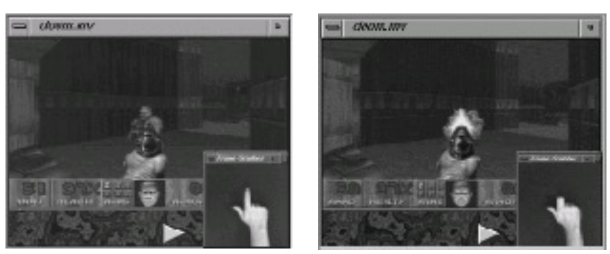


“Reach” opens the door

19

19

Controlling Video Games: Doom



“Click” fires the gun

20

20

[Video]

21

The Kidsroom: A Perceptually-Based Interactive and Immersive Story Environment

22

22

Quick Idea

- Child's bedroom
- Children enact a simple fantasy story
- Interact with virtual characters

23

23

Goals

- Action in a physical space
- Vision-based remote sensing
- Multiple people
- Use of context
- Presence, engagement, and imagination
- Children

24

24

Implementation

- Re-creates a child’s bedroom
  - 24 (W) x 18 (L) x 27 (H)
  - Real furniture: moveable bed, fixed furniture
- Two walls are video projection screens
  - Back-projected
- Four video cameras
- Theatrical lighting
- Speakers, microphone

25

25

26

26

Video Cameras

27

27

Computation

- Three workstations
  - Vision tracking, sound effects, light control
  - Action recognition (cam-1,2), MIDI commands
  - Action recognition (cam-3)
- Two workstations
  - Animation display
  - Control processes
- One Mac for music

28

28

The Story

- Interactive, imaginative adventure
- Inspired by children's stories
  - *"Where the wild things are"*
  - *"Bedknobs and broomsticks"*
- Linear narrative with interactive responses
- Story begins in child's bedroom, progresses through three other worlds
  - Forest, river, monster worlds

29

29

Bedroom World

- Children enter bedroom through door
  - One at a time, for tracker initialization
  - Whimsical music plays softly
  - Video wall screens show bedroom walls

30

30

Bedroom World

- Scavenger hunt for “magic password”
  - Furniture *speak* if child is close
  - Sends to other furniture if not have password
    - Random ordering by control system
  - Furniture calls to kids if not go to right place

31

31

Bedroom World

- Once reach furniture piece with magic password, all furniture chant the word
- A mother’s voice breaks in telling the kids to “go to bed!”

32

32

Bedroom World

- Stuffed monster doll (on video wall) comes alive and asks for the magic password to go on a big adventure
- Children shout the word
  - Loudness detector for microphone in room

33

33

Forest World

- Bedroom changes into forest land
  - Lights drop
  - Video walls change from bedroom to forest
  - Music changes
  - Narrator welcomes children

34

34

Forest World

- Children are in a deep forest
- Must follow path (around room) to river
  - Stay in a group (vision system)
  - Given hints

35

35

Forest World

- Hear monsters growing
  - Must hide behind bed (vision system)
- Continue path until get to river

36

36

River World

- Forest video projections are now of river
- Bed become a “magic boat”
- Children move bed into center of room and jump on top

37

37

River World

- Make “rowing” motions to make bed appear to go down virtual river (computer vision)

38

38

River World

- Row (sometimes) around obstacles

Liked to smash into obstacles!

(sound effects only)

39

39

River World

- Reach shoreline and put bed back
- Arrive at monster world

40

40

Monster World

- Monsters appear on video screens

41

41

Monster Dance

- Children and monsters dance together
  - Monsters teach children dance moves
    - “Y”, crouch, spin, flap
  - Then children do dance moves and monsters follow them (computer vision)

42

42

Ending

- Mother's voice tells them all to go to bed
- When children all on bed, lights change and room converts back into the bedroom
- Narrator thanks children, they exit

43

43

Computer Vision

- Object/kid tracking
- Movement detection
- Dance recognition
- Event detection

44

44

Object Tracking System

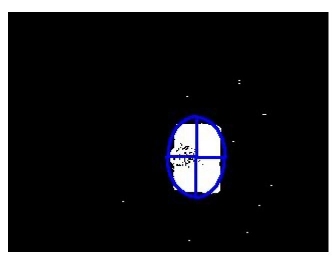
- Detect people as non-room regions
- Correspond and track through sequence

Top-down camera view of empty room (with bed)

Subtract static room from image with people

Rowing Vision System

Top view



Room subtraction

Row left

Row both

Row right

Dance Recognition

- Motion templates and pose recognition

flapping spinning 'Y' crouching 47

47

Event Detection

- "Is everyone in a group?"
- "Is everyone on the bed"
- "Is everyone on the path?"
- "Is everyone standing still?"
- "Is someone near a particular object?"

48

48

Story Control

- Narrative control
  - Scripted event loop, timers
- Music and sound control
  - MIDI and sound files
- Lighting control
  - Event controlled, coupled with vision systems
- Animation control
  - Event scripts, recognition responses
- Process control
  - Sensor servers and control program client

49

49

Achieving Project Goals

- Real action, real objects
- Remote visual sensing
- Multiple, collaborating people
- Exploiting context
- Presence, engagement, and imagination
- Children as subjects

50

50

Observations and Failures

- Perceptual limitations
  - Environment: bright lights, video screens, high camera, flooring
  - Tracking of “who” is hard, “where” is easier
  - Shadows, screens, occlusion during dance
- Perceptual expectation
  - If asked to speak/shout, then expect speech recognition systems

51

51

[Video]

52

52

KidsRoom²

By Nearlife

(was installed in London's Millennium Dome)

53

53