Accessible Aerial Autonomy?

ARDrone ~ aerial *remote-controlled* platform

Students: Nick Berezny, Lilian de Greef, Brad Jensen, Kim Sheely, Malen Sok
Advisor: Professor Zachary Dodds
Question

Would the ARDrone make an effective robot?

Raw material:

- closed hardware
- but an open, ASCII API
- two cameras

Plan: accomplish tasks with the drone and computer vision
Several tasks tried...

(0) Flight “testing”
(1) Cooperating with the Create
(2) Navigating among landmarks
(3) Localization without landmarks
Several tasks tried...

(0) Flight “testing”
(1) Cooperating with the Create
(2) Navigating among landmarks
(3) Localization without landmarks

detect + decide  
follow  
repeat...
Task 1: *Follow that!* 

We put a ! on the Create to 
- help discern location 
- help discern orientation 

Image processing approach: 
1. threshold image to find dark regions and contours 
2. *circle?* compare region with min. enclosing circle 
3. *rectangle?* compare region with min. enclosing rect. 
4. filter noise, find centers, and construct heading line
finding

original

wall segmentation

contours
GCER cooperation demo
Lessons learned

• The ! was far from a perfect landmark

• We wanted to use something more robust that could give us more accurate pose estimation

• We decided to explore April Tags...
APRIL tags

Autonomy, Perception, Robotics, Interfaces, and Learning

Java-based landmark library from U. Michigan

an example tag in the center... provides full 6 DOF pose and scale
APRIL tags' scale range

an example tag in the center... provides full 6 DOF pose and scale ... when it's visible
Task 2: The *Hula-hoop hop*

getting from point A to point B
Hula-hop's state machine

all transitions can also be made by the keyboard
Hula-hop demo

sliding-scale autonomy is crucial
Hula-hop challenges

Drone challenges:
- *drift* ~ not easily positionable
- *connection* ~ video freezes
- *artifacts* ~ image stream noise

APRIL tag challenges:
- too narrow a field of view: height/scale tradeoffs
- call to APRIL library is slow (.5 second/image)
- *unmodifiable environments*?

Could we do **without** tags?
Localization without tags?

**SURF features**
- locally unique image patches
- fast libraries for extraction
- each SURF feature is described with a 64-dimensional vector that encodes size and local edge orientations

- in general, similar descriptor vectors are likely to be similar (or identical) image features
Localization plan

Mapping (by hand)
- collect images and positions
- extract & store SURF features

Localization
- take a new image
- extract SURF features
- match them against the map
- estimate a pose distribution

new image  map images + matches
Image-based map...

Locations with stored images == nodes in a graph

four locations in the NW corner of Sprague
Image-based map...
Image-based map...
Live localization

top three matches and their likelihood distribution plotted on the map
Comparative results

![Bar chart showing comparative results for different methods with blue and red bars. The blue bars represent Correct Coordinates and the red bars represent Correct Angle. The x-axis labels are simple, simple bidirectional, simple ratio bidirectional, scaled, scaled bidirectional, scaled ratio, and scaled ratio bidirectional. The y-axis ranges from 0.0 to 120.0.]
Verdicts?

The **AR Drone** is a capable platform

-- *as long as precise positioning is not required*

Options:
- research to improve localization
- tasks that do not require precision
Questions?