Sept. 13th, 2016

Homework:
\[ T^w_a = \begin{bmatrix} 1 & 0.25 \\ 0 & 2 \\ 0 & 1 \end{bmatrix} \Rightarrow \text{world from } a \]

Position of A in world: \( T^w_a \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \) A's frame
\[
\begin{bmatrix}
\text{pos of light} \\
\text{in camera frame}
\end{bmatrix} = T^c_l \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \text{light frame}
\]

\( p_{\text{world}} = \text{world from } -a \times p_{-a} \)

Estimation: Average flow rate of the Mississippi
Our Guess: \( h \times w \times v \Rightarrow 5m \times 2000m \times 2m/s \)
\( \Rightarrow 20,000 \text{ m}^3/\text{s} \)

Actually about \( 17,000 \text{ m}^3/\text{s} \)

Lab:

Blob 1:
- \( Cx: 300 \)
- \( Cy: 400 \)
- Area: 1000

Blob 2:
- \( Cx \approx 250 \)
- \( Cy: 400 \)
- Area: 1000

\( \dot{x}_r = \text{constant} \)
\( \dot{\theta}_r = k_\theta (300 - \text{blob}x) \)

\( \theta \) constant of dominant blob
\( y \) angular error.
\( y \approx -320 \leq \frac{320}{-\text{blob}x} \leq 320 \)

\( \text{Bound } \dot{\theta}_r \text{ to be } \pm 1 \text{Rad/s at max.} \)

\( k_\theta \sim \frac{1}{320} \sim 0.003 \)
why \((320 - \text{blob})\) is secretly proportional to angular error.

**Projecting Points onto the camera sensor plane**

![Diagram of camera sensor plane with variables and equations]

Because of similar triangles:

\[
\frac{u}{f} = \frac{y}{x} \\
u = \frac{fy}{x} \\
u \propto \frac{y}{x} \\
u \propto \tan(\theta) \\
u \approx \theta \quad \text{small angle approx.}
\]

So \((320 - \text{blob})\) is proportional to \(\theta\). \(\text{blob} = u + 320\)

Things farther away are smaller!
Sensors
* Generally suck
* They lie to you
* Believe them at your peril.

But, your robot is useless w/o them.

Sensors are built on top of / using transducers.
Transducers convert mechanical/radiant energy to an electric signal

Sensor can be active or passive
passive: limit switches (bumper), eyes (waits/collects light), ears, camera, mic
energy comes from outside world.
active: IR/sonar rangefinder, radar, lidar, Kinect
energy comes from within.

Analog vs. Digital
Analog: produce continuous voltage or current
Digital: produce 1's and 0's.

AD converter converts analog voltage to binary #.

Side effect: saturation/range effects.
AD converter has fixed bits
=> range/precision tradeoff.
Photo interrupter - active sensor. Digital 0 or 1
0 when LED is shining through. 1 if blocked.

Rotary Encoder

Clockwise
A
B

Counter Clockwise
A
B

Uses 3-bit gray code
Only changes 1 bit at a time.

Downside: needs more sensors and larger wheel.
Use rotary encoders b/c wheels/motors aren't perfect and don't do what you ask them to do.

Limit Switch - passive sensor

On turtlebot bumper skirt

Turtlebot also has gyroscope, but not accelerometer.

Cameras, they're cool.

* Recognize objects, faces.
* Visual odometry.
* Visual servoing → important for manipulation, picking stuff up.

...they're awful

...vision is hard
...generally don't see in 3D. (except kinect, stereo, ...)
...Too much data

2 wheel encoders @ 100 Hz ~ 25 bytes/s
Bump / Cliff sensor @ 100 Hz ~ 50 bytes/s
Gyroscope @ 30 Hz ~ 40 bytes/s
Kinect video 8 bit RGB @ 30fps ~ 3.29 MB/s
Each sensor element reports a voltage proportional to light received.

Single frame $\leftarrow$ many A/D values.

In computer memory:

<table>
<thead>
<tr>
<th>R</th>
<th>G</th>
<th>B</th>
<th>R</th>
<th>G</th>
<th>B</th>
</tr>
</thead>
</table>
| 0.0 | 1.0 | 0.0 | 0.0 | 1.0 | ...

Additive color primary. Same as retina.

Row-major storage left to right, top to bottom.

3 different types of elements w/ 3 different response curves.

Light @ 450 nm: Big response in blue sensor, medium in green, tiny in red.
incoming light → aperture → exposure time → A/D Gain → auto white balance
→ red eye removal → snapshot booface → ...
→ image in memory

SI Quantities / Units Review

Base units

- length (meters)
- time (s)
- mass (kg)
- current (Amps)

Derived units

- Force (Newton = \( \frac{kg \cdot m}{s^2} \))
- Energy (Joules = Nm = \( \frac{kg \cdot m^2}{s^2} \))
- Power (Watts = \( \frac{J}{s} \))
- Torque (Nm)

\( \tau = r \times F \)

Pivot \( \downarrow \) F
\( \Rightarrow \)