CS 378: Autonomous Intelligent Robotics (FRI)

Dr. Todd Hester

Are there any questions?

Logistics

- Post for teammates on Piazza
  - Project topics, skills
- BWI Lab - GDC 3.414B
  - Office Hours
- Greg Dudek talk this morning

Texas Tech REU

- Robotics and Autonomy
- Mohan Sridharan
- Deadline: March 8
- Program: June 3 - August 8
- http://www.cs.ttu.edu/~smohan/Outreach/Docs/2013/TTU_REU2013.pdf

Today

- Assignment
- Kalman Filters

Assignment

- Goals
  - Make the robot follow an orange ball
  - Learn how ROS code works
  - Get some experience with the robots
- Work in groups of 2-3
- Lab Access
  - 10-6 Monday-Friday (For now)
- Robot reservations
- Robot names
- Looking at the code

```
int main(int argc, char **argv){

    ros::init(argc, argv, "follower");
    ros::NodeHandle n;

    // advertise that we will publish cmd_vel messages
    velocity_pub = n.advertise<geometry_msgs::Twist>("cmd_vel", 1000);

    // subscribe to blob messages and call blobCallback when they are received
    ros::Subscriber sub = n.subscribe("blobs", 1000, blobCallback);

    ros::Rate loop_rate(10);

    ros::spin();


    return 0;
}
```

```
// This method is called whenever a blob message is received
void blobCallback(const cmvision::Blobs::ConstPtr& msg){
  // This is the output velocity that we will publish
  geometry_msgs::Twist output;

  // first, we can check if any blobs were found
  if (msg->blob_count > 0){

        // we may want to access / look at multiple blobs in the array
        for (int i = 0; i < msg->blob_count; i++){

        // another example print with some blob info
        std::cout << "Detected blob " << i << " with area " << msg->blobs[i].area << std::endl;

        // some things to look at
        msg->blobs[i].area;      // blob area
        msg->blobs[i].x;         // blob center x
        msg->blobs[i].y;         // blob center y
        msg->blobs[i].left;      // blob left x
        msg->blobs[i].right;     // blob right x
        msg->blobs[i].top;       // blob top x
        msg->blobs[i].bottom;    // blob bottom x
```

```
// This method is called whenever a blob message is received
void blobCallback(const cmvision::Blobs::ConstPtr& msg){
  // This is the output velocity that we will publish
  geometry_msgs::Twist output;

  // first, we can check if any blobs were found
  if (msg->blob_count > 0){

        // we may want to access / look at multiple blobs in the array
        for (int i = 0; i < msg->blob_count; i++){

        }

        // TODO: decide what velocities to publish based on blob info

        output.linear.x = 0; // TODO: fill in this with some number for fwd velocity (meters/sec)
        output.angular.x = 0; // TODO: fill this in with some angular velocity (radians/sec)

        velocity_pub.publish(output); // publish this velocity message that we filled in
  }

}
```

Assignment

- Other files
- Running the robots
- Start early!!!!
  - It will take longer than you think
  - There are limited resources / robots
  - Robots can and will break

Kalman Filter

Next week

- Monday night
  - Reading response
- Tuesday
  - New wiki paper
- Thursday
  - Robot assignment