Projects Pitches

Why do we do projects?

- Learn to work together.
- Learn the tools to work together.
- Appreciate the system-level challenges.
- Dive-in for one favorite topic.
- ...

The plan

- We have created a **preliminary list of projects**.
- **Today**:
  - We briefly describe the projects to you.
  - Feel free to add comments to these slides.
  - We will answer clarification questions at the end, or later today.
- **Later today:**
  - You fill out a form describing your preferences.
- In the next few days, we formulate a final list of projects and allocation.
- Projects start as soon as the groups are formed.

What we will ask you on the form

- Give a **score** to each project
  - 0 = least favorite, 10 = most favorite
  - (scores are normalized)

- Other ancillary questions, such as:
  - What is your **favorite group size**?
  - Who is your **buddy** for life?
  - Do you prefer an **international project**?
    - (international projects assignments have priority)

What makes a good project?

- It's **useful to Duckietown** and becomes a permanent part of it;
- It has the potential to generate **novel research contributions**;
- It’s **challenging** and we all learn in the process;
- We have **fun** doing it.
- It can be the basis for further work.

All projects are different

- **The projects reflect the diversity of robotics.**

- Some of the dimensions:
  - **improvement** to an existing functionality *vs.* **new piece** of functionality
  - it is pure **engineering** *vs.* there is a **research** component
  - **hardware** *vs.* **software**
  - **features** *vs.* **behaviors**
  - **components** *vs.* **system / infrastructure**

- Evaluation will take into account these differences.

What makes a good project for you?

- It is **something that interests you**.
- It is the **right level of challenge** given your background:
  - *not too easy*: otherwise, you'll get bored
  - *not too difficult*: otherwise, you'll get frustrated
- It is **something that you can be proud of,** and it helps you with whatever you want to do next
  - an internship, a master's thesis, looking for a job, etc.

Attributes of great projects

- Projects adhere to **best practices in testing**
  - Unit tests (test code in isolation)
  - Regression tests (test functionality on logs)
  - Integration tests (test functionality together with other modules)
- **Rigorous evaluation**
  - "It works": It is consistently reproducible on the other side of the ocean.
  - It works on master, without breaking anything else.
  - Discussion of operating envelope and failure modes.
- Complete **documentation**
  - Technical documentation
  - Operating manual

Developing a system together

- **We are developing a system together.**
- The *gatekeepers of master* handle merges into master.
    - From time to time, some changes might be breaking.
    - Unit/regression/integration tests are what preserve functionality.
- To achieve success, groups will need to:
    - Talk with the other groups on which functionality they rely.
    - Follow closely and intervene in the the system-level discussions.
    - Merge from master daily.
- We will ask for **point of contacts** to facilitate discussions.

Team assignments

What we did in the last week

- We collected the preferences.
- We cleaned up the data.
- We removed the unpopular projects.
- We assigned people to projects.
- We clustered projects into groups, when it made sense.
  - E.g. close collaboration, boundary not clear.

Final organization

- Team organization:
  - 16 product-oriented teams
  - 1 "special ops" task force
- Work organization:
  - 20 projects (or "products")
  - 4 shared infrastructure efforts (annotation, object detection, simulator/maps, neural computation).

About each team

Name: You can change it in the next 24 hours. Choose a cool name!
Your channels/branches are going to be called #dev-TEAM-product.

Mission: This is what's important - all the rest are technical details that can change.

Motto: Your rallying cry into battle.

Members: The list of members and affiliations.

Support: Other people involved in the project: TAs, staff, semester projects, …

Mentor: This is the person that guides the technical direction of the team.

Supervisor: This is one of the instructors, making sure nothing weird happens.

Projects/products: this is what the team should create. Some large teams have 2.

Infrastructure: the thing(s) that the group is responsible to develop and/or maintain that could be useful to others

Teams and projects

Team: The Architects

Mission: “Make Duckietown a smarter city.”

Motto: OMNES VIAE ANATENSEM URBEM DUCUNT (*All roads lead to Duckietown*)

Members: Simas Glinskis, Sam Nickolay **U S**

Mentor: Jacopo Tani **C H**

Support: Marco Erni **C H**, Andrea Daniele **U S**

Supervisor: Jacopo Tani **C H**

Since pitches: de-emphasized cheapness

#devel-hw-city

Project: Smart cities

- **Motivation**: In a world of autonomous cars, cities should be smart enough to do their part. Duckietown is no different, so we shall create the infrastructure to make this possible, e.g., utilities equipped tiles and camera equipped traffic lights. While we're at it, we'll make Duckietown easier to produce and manage.

- **Possible approach**: Design modular tiles with data and power utilities. Utilities are extended by interlocking. Consider different materials for the tiles (easier storage, improved friction, reduced cost). Create city wide power and data hubs. Design and deploy camera equipped traffic lights.

- **Outcomes**:
  - ☐ **bronze**: Traffic lights generate traffic data
  - ☐ **silver**: Tiles have power and data integrated seamlessly. Modular design validated.
  - ☐ **gold**: Smart city elements ready for mass production. Prototype deployed. Cuteness factor considered.

Single robot behaviors

Team: The Identifiers

Mission: Estimate better models to make localization and control more efficient and robust.

Motto: NOSCE TE IPSUM (*Know thyself*)

Members: Manuel Dangel, Sofiane Gadi, Romeo Staub CH

Mentor: Jacopo Tani CH

Support: Selçuk Ercan CH

Supervisor: Jacopo Tani CH

Since pitches: clarified that the system id methods should make localization and control perform better.

Project: Odometry calibration / system ID

- **Motivation:** Our current odometry calibration method only identifies the ratio of model parameters for the kinematic model. With effective system ID protocols, we enable leveraging models for improved control.
- **Outcomes**:
  - ☐ **bronze**: identify kinematic model (axle length and wheel parameters independently), with and without caster wheel
  - ☐ **silver**: Identify asymmetric kinematic model. Identify dynamic model (mass and inertia) and quantify bound on disturbances (friction).
  - ☐ **gold**: Plug-and-play solution for global system ID (including camera calibration?)
- **Possible approach:** Design and deploy calibration track. Use external sensor (e.g., second Duckiebot) for validation data. Use least squares approach. Performance metrics include control performance and localization prediction accuracy.

Team: The Controllers

Mission: Make lane following more robust to model assumptions and building regulations violations

Latin motto: IN MEDIO STAT VIRTUS *(The virtue is in the middle)*

Members: Marco Stalder, Simon Muntwiler, Anna Dai, Manuel Breitenstein, Andreas Aumiller **C H**

Mentor: Miguel de la Iglesia **C H**

Support people: Nicolas Bunger **C H**

Supervisor: Jacopo Tani **C H**

Since pitches: clarified that it's about a "behavior" (perception + control)

Project: Adaptive Lane Following

- **Motivation**: The fundamental behavior of Duckiebots is driving in roads. The first step is guaranteeing following a straight road. This project targets extending the current implementation of the fast localization pipeline by introduction lane width estimation while synthesizing a better controller.
- **Outcomes:**
    - **wooden spoon:** put an I in the controller - make test cases - measure performance - latency measurement, adaptiveness
    - □ **bronze**: **straight** lane following: **localization** w/ auto tune for width + **control**
    - □ **silver**: generalize to uncertain (appearance specification) prior
    - □ **gold**: generalized perception and control with other perception + projects

Since pitches: clarified that it's about a "behavior" (perception + control)

Project: Adaptive Curvature Control

- **Motivation**: The fundamental behavior of Duckiebots is driving in roads. The second step is guaranteeing following a curved road. This project aims at extending the current functionalities to explicitly account for road curvature estimation, and designing a controller taking into account that curvature.

- **Outcomes:**
  - ☐ **bronze**: **curved** lane localization + **curvature estimation** + control
  - ☐ **silver**: expand perception to take into account patterns
  - ☐ **gold**: generalized perception and control approach with other perception + control projects (e.g., road curvature based approach)

Team: The Saviors

Mission: "Save the little duckies that want to cross the road"

Motto: IMPERARE SIBI MAXIMUM IMPERIUM EST *(Controlling oneself is the greatest form of command)*

Members: Fabio Meier, Fabrice Oehler, Julian Nubert, Niklas Funk **CH**

Support: Dzenan Lapandic **CH**, Khurana Harshit **CH**, Selçuk Ercan **CH**

Mentor: Jacopo Tani **CH**

Supervisor: Jacopo Tani **CH**

Since pitches: do not use Duckie detector - try instead obstacle field (reproject texture on local map, then avoid yellow).

Project: Obstacle avoidance pipeline

- **Motivation**: Ensuring safety is a paramount concern for the citizens of Duckietown. The city has therefore commissioned the implementation of protocols for guaranteed obstacle detection and avoidance.

- **Outcomes:**
  - ☐ **bronze**: obstacle avoidance using obstacle field (reprojected texture to local map and avoid yellow)
  - ☐ **silver**: change reference trajectory with motion primitive or variational method
  - ☐ **gold**: generalization = reference trajectory; additional tracking functionality (e.g. we look at duckie on vehicle)

Team: The Navigators

Mission: "Make Duckiebots reliably navigate intersections"

Motto: EXPERTO CREDE (*Trust the experts*)

Members: Theodore Koutros, Flin Höpflinger, Giovanni Cioffi, Dario Brescianini C H

Mentor: Jacopo Tani C H

Support: Selçuk Ercan C H, Julien Kindle C H, Thomas Ackermann C H

Supervisor: Jacopo Tani C H

Since pitches: clarified methods

Project: Intersection Navigation

- **Motivation**: The fundamental behavior of Duckiebots is driving in cities. Our current feedback derives from detecting lines, which are not present on intersection tiles. As a result, Duckiebots currently navigate intersections in open loop. This project will change this.

- **Outcomes:**
  - □ **bronze**: closed loop behavior for intersection navigation
  - □ **silver**: evaluate different approaches and quantify differences
  - □ **gold**: integrate approach with other perception + control projects
- **Possible Approach**: intersection navigation using: (a) line localization, (b) visual odometry or (c) april tags (d) ad-hoc solutions

Team: Parking

Mission: “Make parking lots a functional component of Duckietown”

Motto: DUCK EX MACHINA

Members: Nils Funk, Brett Stephens, Samuel Nyffenegger **C H**

Mentor: Andrea Censi **C H**

Support: Khurana Harshit **C H**, Julien Kindle **C H**

Supervisor: Andrea Censi **C H**

Since pitches: clarified that also the spec for the parking lots should be developed.

Project: Parking and Parking Lots

- **Motivation**: Parking lots are a recent tile addition to Duckietown and are still underspecified. The objective of this project is to formalize the specifications for and successfully implement parking maneuvers.

- **Outcomes:**
  - ☐ **bronze**: appearance specs for parking lots, and straight parking using RRT*, localization in lots
  - ☐ **silver**: develop appearance specs in harmony with Duckietown specs, and implement parallel parking
  - ☐ **gold**: all the above and parking lot entrance/exit coordination

Multi-robot behavior

Team: The Coordinators

Mission: "Coordinate intersection navigation safely through explicit communication"

Latin: IN HOC SIGNO VINCES (*With this sign(al), you will win*)

Members: Su Yilmaz, Victor Jiao **U S**,

Robin Deuber, Valentina Cavinato, Nicolas Lanzetti, Giulia Zobrist, Gioele Zardini **C H**

Mentor: Jacopo Tani **C H**

Support: Mauro Salazar **C H**

Supervisor: Jacopo Tani **C H**

Since pitches: expanded on potential approaches

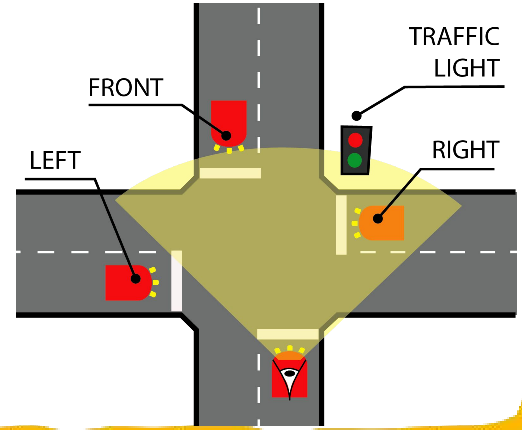
Project: LED-based communication

- **Motivation:** Duckiebots need to communicate with each other to express fleet level behaviors. Currently, this is achieved through the blinking of LEDs at specific frequencies. Traffic light encode messages in the same way (not based on color).
- **Outcomes**:
  - ☐ **bronze**: benchmark existing solution (time, accuracy, false positive/negative)
  - ☐ **silver**: reduce interpretation time and improve performance
  - ☐ **gold**: determine optimal signal to transmit to maximize successful interpretation subject to detection time.
- **Possible approach:** change the signals that are transmitted, improve computational aspects of current implementation, switch from sampling to extended exposure detection, include color in message discrimination, use natural language signals, detect motion of upcoming/departing vehicles through smear extent

No change since pitches

Project: Coordination algorithm

- **Motivation**: Intersections are nasty places where many things can go wrong. Either in presence of a traffic light or in absence of a centralized coordinator, vehicles need to engage intersections while providing safety guarantees (no collisions). High throughput and robustness to rogue vehicles are important too.
- **Outcomes**:
  - ☐ **bronze**: benchmark existing implementation
  - ☐ **silver**: improve throughput and robustness while guaranteeing safety
  - ☐ **gold**: only use the "natural" signals (blinking lights, headlights); allow an RC car to participate, handle rogue vehicles
- **Possible approaches**: "formal methods" for control.



Team: Formations and implicit coordination

Mission: "Formation keeping and collision avoidance using implicit communication"

Latin motto: ALIIS VIVERE (*Live for others)*

Members: Manfred Diaz, Jonathan Arsenault

Nicolas Schmid, Lukas Schönbächler, Sandro Losa, Michael Klemmer, Kornel Eggerschwiler **C H**

Mentor: Liam Paull, Jacopo Tani **C H**

Support: Selcuk Ercan **C H**, Dzenan Lapandic **C H**

Supervisor: Liam Paull, Jacopo Tani **C H**

Infrastructure: Annotation framework

Since pitches: extracted formation control from project with larger scope.

Project: Formation keeping

- **Motivation**: When multiple Duckiebots are navigating Duckietown it is natural to encounter traffic. In order to do so, Duckiebots must detect other vehicles and maintain relative distance, i.e., keep a formation. This project deals with implementing formation control of Duckiebots.

- **Outcomes:**
  - □ **bronze**: drive in traffic (e.g., red traffic light), follow the leader using April tags/other fiducial markers
  - □ **silver**: formation keeping, show traffic waves
  - □ **gold**: “closed” formation keeping, avoid traffic waves
- **Possible Approach**: Detect vehicles first with fiducial markers, through onboard duckie, or directly through Duckiebot vehicle detection. Estimate distance from obstacle and control at relative distance.

Since pitches: extracted formation control from project with larger scope.

Project: Coordination with implicit communication

- **Motivation**: Not all Duckiebots are alike. Allowing for different hardware configurations (e.g., with or without LEDs) poses challenges for intersection coordination in absence of a centralized coordinator (traffic light). This project builds on detection, tracking and anticipation for regulating intersection navigation in absence of explicit communication (LEDs).

- **Outcomes:**
  - □ **bronze**: Intersection coordination without communication
  - □ **silver**: Track other vehicles, stop when collision is expected
  - □ **gold**: Formalize conventions for which it is always clear who has precedence and guarantee safety

Team: Distributed estimation

Mission: "Make Duckiebots create a map of the world collaboratively"

Motto: UBI UNUM IBI OMNES (*Where there is one, there is everybody*)

Members: Marcel Kaufmann, Philippe Laferriere, Samuel Laferriere □,

Luca Somm, Napat Karnchanachari, Leonie Traffelet **C H**

Mentors:

Comms: Francesco Ferrara **C H**, Antoine Patru **C H**,

SLAM: Liam Paull □

Support: Yanjun Cao □

Supervisor: Andrea Censi **C H**

Infrastructure: Communication layer

Since pitches: removed "cooperative localization", joined with multi-SLAM project.

Project: Fleet wireless communication

- **Motivation:** A fleet of autonomous cars needs to communicate so that they can coordinate picking up passengers.
- **Outcomes**:
  - ☐ **bronze**: Reliable physical network layer (centralized duckietown network) that allows broadcasting of pose network data, broadcasting of status, health, cooperative planning to serve requests.
  - ☐ **silver**: There is no centralized point of failure; nodes can join and leave.
  - ☐ **gold**: Demonstrate cooperative localization. Ad-hoc networking.

No change since pitches

Project: SLAM – Multi-robot

- **Motivation**: SLAM is important if not necessary for autonomy. While Duckiebots (and robots in general) can perform SLAM on an individual basis, there are several advantages to performing localization and mapping jointly across several vehicle (i.e., multi-robot SLAM). Such an approach can take advantage of robot-robot measurement and exploit scenarios when a subset of the team has an accurate pose estimate
- **Possible approach**: Develop a pose graph formulation of the multi-vehicle SLAM problem (using least squares or nonlinear optimization) that exploits robot-robot and world-robot observations.
- **Outcomes**:
  - □ **bronze**: Centralized least squares solution to SLAM.
  - □ **silver**: Centralized nonlinear optimization formulation of SLAM.
  - □ **gold**: Decentralized least squares or nonlinear formulation of SLAM.

Team: Fleet-level planning

Mission: "Create the mobility-on-demand service for Duckietown."

Motto: VICTORIA CONCORDIA CRESCIT (*Victory through harmony*)

Members: Sandro Meier, Sebastian Ratz, Benjamin Hahn, Nicolas Scheidt **C H**

Mentor: Claudio Ruch **C H**

Support:

Supervisor: Andrea Censi **C H**

No change since pistches

#devel-fleet-planning

Project: Fleet-level planning

- *Note: we didn't have the fleet planning lectures yet.*
- **Motivation**: It is much more difficult to plan for a fleet than for a single car.
- **Proposed approach:** implement state-of-the-art fleet-level planning algorithms.
- **Outcome:**
  - □ **bronze**: coordinated task assignment
  - □ **silver**: provably optimal fleet planning with rebalancing
  - □ **gold**: distributed, fault-tolerant planning

- Bronze: Being able to locate all duckiebots in duckietown and visualize them on a map on the computer screen.

Silver: Being able to send a reference location command to all duckiebots in duckietown. The duckiebots then travel to their assigned destination.

- Gold: Computing this set of reference locations as the m-stochastic mean locations of some artificial travel demand. This allows to implement the m-stochastic queue median policy paper. See here: https://dspace.mit.edu/openaccess-disseminate/1721.1/65388 page 5.

- Platinum: On the computer visualization using the mouse create a customer request which is then assigned to one duckiebot that travels from its location to the request origin, then to the request destination and then back to its wait position (the k-stochastic man location) above.

We could offer to the students to use our visualizer as well as our code for the dispatchers. It is in JAVA. If they prefer to write themselves they can easily do that, because it is a very limited amount of code.

Team: Supervised learning

Mission: Understand which component(s) in a complex system can/should be replaced by a learning-based component.

Motto: NULLIUS IN VERBA *(on the word of no one)*

Members: Alex Lamb, Rithesh Kumar, Michael Noukhovitch □,

Tianlu Wang, Shaohui Yang C H

Mentor: Liam Paull □

Support: Florian Golemo □, Julian Zilly C H, Amazon?, Nick Wang (NCTU)

Supervisor: Liam Paull □

Infrastructure: Deep Learning HW/SW infrastructure

Some changes, not reflected in slides yet.

Project: What to learn?

- **Motivation**: The Duckiebot is currently implemented as composition of blocks (sensor processing, perception, estimation, control, planning, actuation, etc). Any of these components or combination of these components could be replaced with a supervised learning approach. Let's assume you have a fixed dataset budget, which components or combination of components should be replaced.

- **Proposed approach:**

- **Outcome:**
  - □ **bronze**: End-to-end behavior imitation. Replace all components with supervised learning and evaluate.
  - □ **silver**: Replace several other components with supervised learning approaches (requires data annotation).
  - □ **gold**: Formalization of the resource constrained optimization problem with justification for experimental results observed.

Team: Transfer learning

Mission: *Transfer to physical hardware what is learned in simulation*

Motto: DOCENDO DISCITUR *(You learn by teaching)*

Members: Chip Schaff **U S** , Ruotian Luo **U S** ,

Nithin Visisth □, Thomas George □

Mentor: Maxime Chevalier-Boisvert□, Liam Paull□

Support people: Yanjun Cao□, Florian Golemo□

Supervisor: Liam Paull□

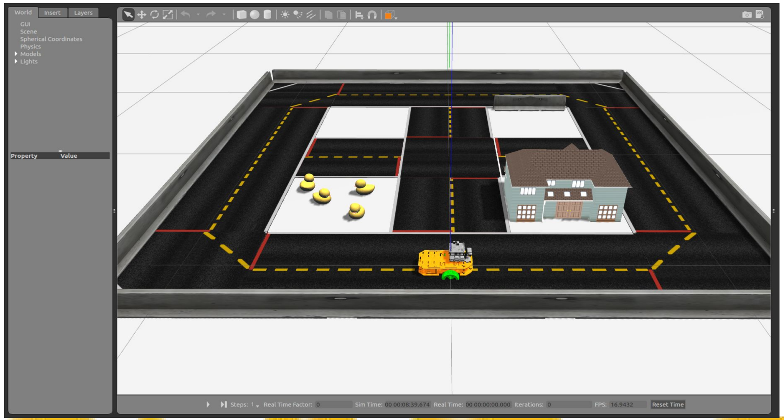
Infrastructure: The simulator

Project: Transfer Learning

- **Motivation**: We wish to teach a robot how to drive by allowing to train in a realistic simulator to reduce the amount of testing/training required on the real physical system.

- **Proposed approach:** This approach supports using deep reinforcement learning for training the agent.

- **Outcome:**
  - ☐ **bronze**: Train an agent in simulator on a simulated environment with no intersections and deploy in the real Duckietown and evaluate (and possibly re-train).
  - ☐ **silver**: Train an agent in simulator on a simulated environment with intersections and deploy in the real Duckietown and evaluate (and possibly re-train).
  - ☐ **gold**: Meta-learning - state of the art.
    Train on randomly generated environments in simulation and deploy on real Duckietown map that the robot has never previously seen



Team: Neural SLAM

Mission: To “learn” the map

Motto: COGITO ERGO SLAM

Members: Adrien Ali Taiga □, Sai Krishna Gottipati □

Mentor: Liam Paull □

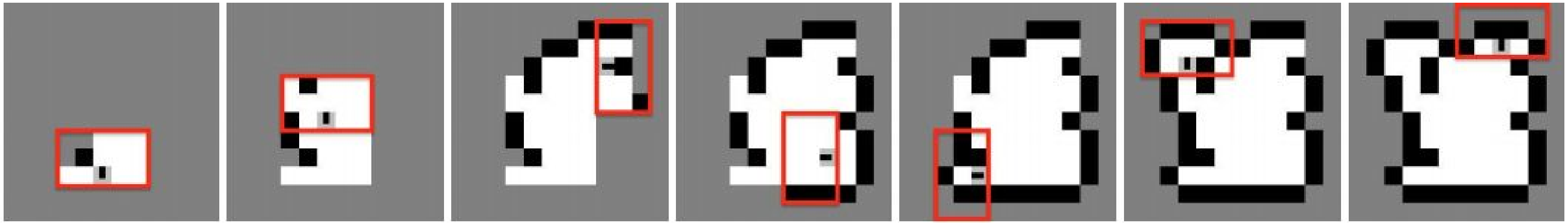
Support people: Florian? Tristan?

Supervisor: Liam Paull □

Project: Neural slam

- **Motivation**: Recent work in SLAM is calling into question whether we need an explicit map at all. We will investigate a "neural turning machine" style representation for learning a map representation.

- **Outcomes:**
  - ☐ **bronze**: Develop a neural map representation for a Duckietown map
  - ☐ **silver**: Build and update the map in realtime
  - ☐ **gold**: Distribute the map building process across several Duckiebots



Team: Anti-instagram

Mission: Make the Duckiebot robust to illumination variation.

Motto: SEMPER VIGILANS (*Always vigilant*)

Members: David Yunis, Shengjie Lin **U S**

Milan Schilling, Christoph Zuidema **C H**

Mentor: Matthew Walter **U S** , Andrea Censi **C H**

Support: Steve Basart **U S**

Supervisor: Matthew Walter **U S**

No changes since pitches

Project: Online continuous anti instagram

- **Motivation:** We want to have thousands of Duckiebots everywhere. Therefore, we need the code to be robust to environment changes. The *Anti-Instagram* project was a first step towards obtaining illumination invariance using machine learning. We need it to be more robust.
- **Outcomes**:
  - ☐ **bronze**: better UI, health measures, better tests; measurable effect on lane localization accuracy.
  - ☐ **silver**: nonlinear adaptivity, better lines detection
  - ☐ **gold**: running online continually
- **Possible approach:** extend from linear to nonlinear model; change the way that line detections are computed.

Team: Single-robot SLAM

Mission: *Make the Duckiebot create the map themselves.*

Motto: ALIS VOLAT PROPRIIS *(It flies by its own wing)*

Members: Devshi Mehrotra **U S**, Ryan Teehan **U S**, Casper Neo **U S**

Mentor: Srinadh Bhojanapalli **U S**

SLAM – Single-robot

- **Motivation**: In order for a Duckiebot to reason globally (e.g., global planning), it must have an estimate of its pose and a map of Duckietown in a global coordinate frame. We an formulate this problem of joint estimation of the map and robot pose in the context of SLAM.
- **Possible approach**: Develop a feature-based SLAM algorithm, e.g., an EKF or Rao-Blackwellized particle filter (analogous to FastSLAM) that estimates the map and robot pose for each robot separately.
- **Outcomes**:
  - ☐ **bronze**: Feature-based SLAM using the AprilTags as landmarks; evaluation of accuracy and consistency relative to ground-truth map.
  - ☐ **silver**: Feature-based SLAM using the lane segments as landmarks; evaluation of accuracy and consistency relative to ground-truth map; and visual comparison between rendered and reference maps.
  - ☐ **gold**: Feature-based Rao-Blackwellized SLAM using line segments.

Team: Visual odometry

Mission: *Enable the robots to track its pose using a single monocular method*

Motto: TBD

Members: Mohammadreza Mostajabi **U S** , Yunfan Tang **U S**, Igor Vasiljevic **U S**

Mentor: Mohammadreza Mostajabi **U S**

Supervisor: Matt Walter **U S**

Visual odometry/SLAM: classical and learning-based

- **Motivation**: Localization and mapping is one of the key challenges for Duckiebots navigating Duckietown. Several promising approaches have been proposed recently, including the approach of Zhou et al. (https://arxiv.org/abs/1704.07813), which performs monocular depth and pose estimation using raw video data.

- **Possible approach**: Adapt the recent implementation of (Zhou et al) to run on the compute-constrained Raspberry Pi, and then incorporate filtering to build a full SLAM system
- **Outcomes**:
  - □ **bronze**: Implement existing algorithm on resource-constrained Raspberry Pi.
  - □ **silver**: Same as bronze with the addition of filtering.
  - □ **gold**: ???

Shared infrastructure projects

- **Simulator / map tools** - maintained by the transfer learning and fleet planning teams.
- **Image/video annotation system** - many groups need this.
- **Communication layer** - Created by Distributed estimation team, but also used by fleet planning.
- **Deep learning HW/SW**
- **April Tags detection and database**

Team: The Heroes

Mission: "Make everything work together"

Motto: E PLURIBUS UNUM (*From many, one*)

Members: Lucy **U S** ,

Breandan Considine □,

Sonja Brits **C H**

Mentors/Supervisors: Andrea Censi **C H**, Liam Paull □

No changes since project pitches.

Integration heroes *special task force*

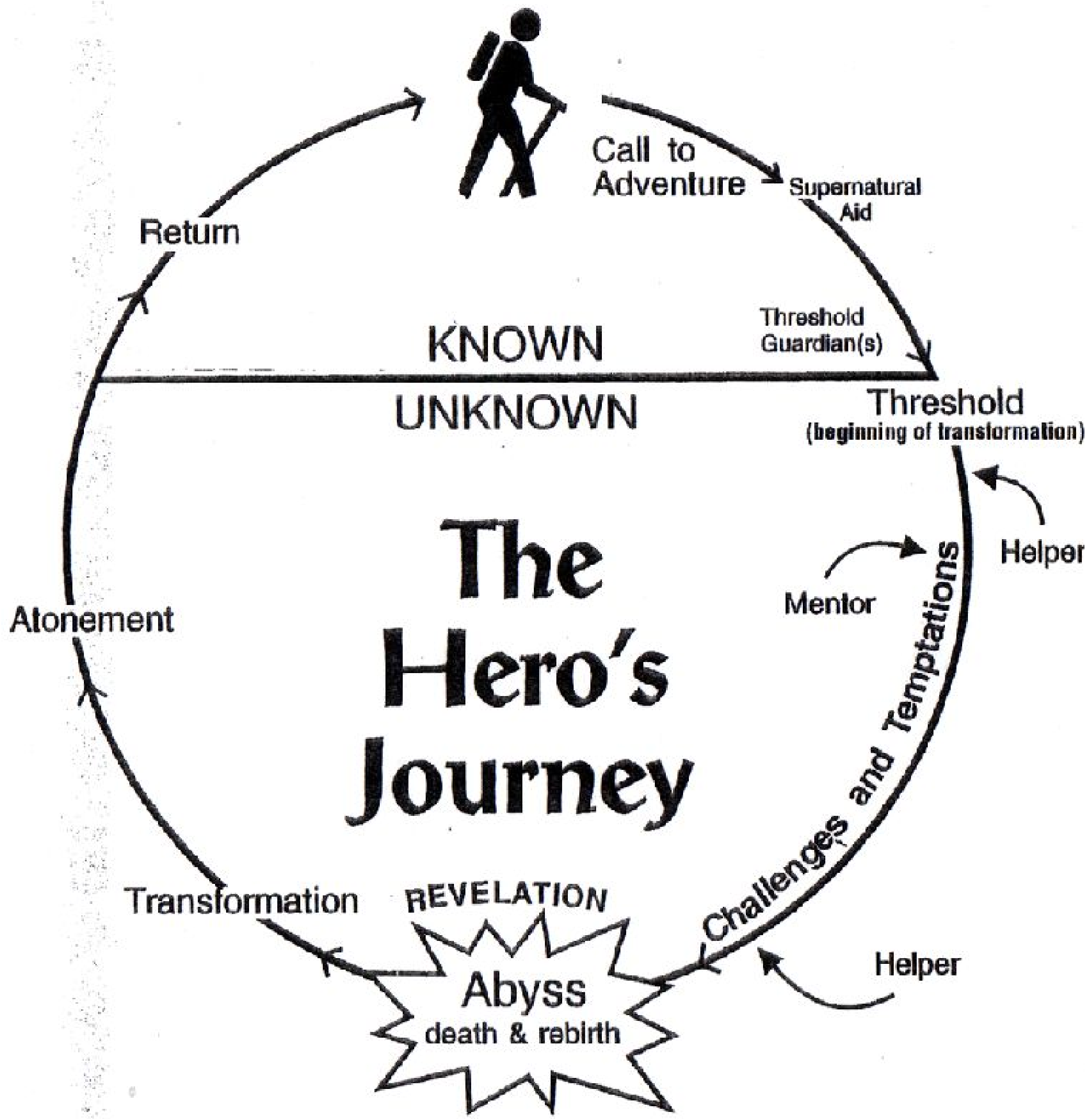
- *Note: This is a special project, in which rather than creating a new feature, you will help in making the system as a whole better. Development will be based on the needs to be identified, and might involve working both on infrastructure as well as "glue" between projects.*
- **Motivation**: The difficult part is putting everything together!
- **Outcomes:**
  - □ **bronze**: everything works together
  - □ **silver**: everything works together and is maintainable
  - □ **gold**: we are proud of the elegance of the system

No changes since project pitches.

Knowledge heroes *special task force*

- **Motivation**: We want to have impeccable documentation.
- **Outcomes:**
  - □ **bronze**: improve organization and format of the Duckiebook
  - □ **silver**: improve the code behind the Duckiebook
  - □ **gold**: create the magic Duckiebook that adapts to each reader.

Projects Kickoff


Call to Adventure
Supernatural Aid
Threshold Guardian(s)
KNOWN
UNKNOWN
Threshold
(beginning of transformation)
Return
The Hero's Journey
Helper
Mentor
Challenges and Temptations
Atonement
Helper
Transformation
REVELATION
Abyss
death & rebirth

Projects phases

- Phase 0: Preferences/assignments (Call to adventure).
- **Phase 1: Kick-off. (Beginning of transformation)**
- Phase 2: Development. (Abyss, death and rebirth)
- Phase 3: Demos, presentations (Atonement/return)

Phase 1: kickoff

- Get to know your team.
- Supervisors will organize first meeting, then self-organize
- Choose a name for your team (current are temps)
- Understand logistics:
  - Google Docs, Duckuments, Github, chat = Zoom
- Find what's already implemented. Try all the basic demos that already exist.
- Preliminary design:
  - Define the scope / goal of the products.
  - Decide performance measures, and how to evaluate.
  - Define initial approach.
  - Make sure that your plan is compatible with all the other plans.
- **Deliverable: Preliminary design document. Deadline: Next Monday**

Projects Philosophy

- Supervisors: just to make sure that things run smoothly
- Mentors: last word on technical direction, though found by consensus
- In general, **teams** need to self-organize meetings etc. after the first one.
- **You can not assume anything will be provided to you**: go find the persons that are supposed to provide what you need, and make sure your plans are compatible.
- **Helping others will be rewarded.** Examples:
    - taking logs for data collection
    - making code reviews
    - giving advice
    - ...

Groups and global organization

- Global coordination necessary:
  - sw
  - integration
  - docs
  -

Learning opportunities

Group: neural SLAM

Mission: **SLAM, but neural?**

Motto: COGITO ERGO SLAM

Members: XXX XXX **U S** , XXX XXX □

Mentor:

Support people:

Supervisor:

Finish slide

Neural slam

- **Motivation**:

- **Outcomes:**
    - □ **bronze**:
    - □ **silver**:
    - □ **gold**:

- **Possible Approach**:

Finish slide

Team: transfer learning

Mission: *Transfer to physical hardware what is learned in simulation*

Motto: DOCENDO DISCITUR *(You learn by teaching)*

Members: XXX XXX **U S** , XXX XXX □

Mentor:

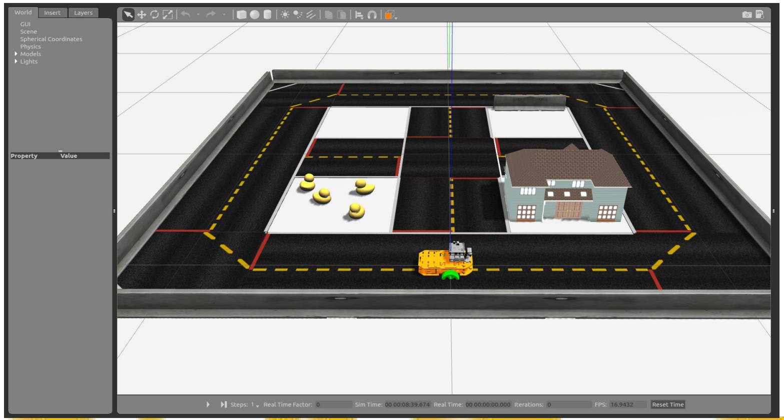
Support people:

Supervisor:

Finish slide

Transfer Learning

- **Motivation**: We wish to teach a robot how to drive by allowing to train in a realistic simulator to reduce the amount of testing/training required on the real physical system.
- **Proposed approach:** A team (led by Florian, Maxime, and Yanjun) has been working on developing the simulator and infrastructure required to support this project. It involves using OpenAI Gym and Gazebo as a simulator. This approach supports using deep reinforcement learning for training the agent.
- **Outcome:**
  - ☐ **bronze**: Train an agent in simulator on a simulated environment with no intersections and deploy in the real Duckietown and evaluate (and possibly re-train).
  - ☐ **silver**: Train an agent in simulator on a simulated environment with intersections and deploy in the real Duckietown and evaluate (and possibly re-train).
  - ☐ **gold**: Meta-learning - state of the art.
    Train on randomly generated environments in simulation and deploy on real Duckietown map that the robot has never previously seen



Advanced Perception

Group: Single-robot SLAM

Mission: *Make the Duckiebot create the map themselves.*

Motto: ALIS VOLAT PROPRIIS *(It flies by its own wing)*

Chicago:

Zurich:

Montreal:

Support people:

Supervisor:

Finish slide

SLAM – Single-robot

- **Motivation**: In order for a Duckiebot to reason globally (e.g., global planning), it must have an estimate of its pose and a map of Duckietown in a global coordinate frame. We an formulate this problem of joint estimation of the map and robot pose in the context of SLAM.
- **Possible approach**: Develop a feature-based SLAM algorithm, e.g., an EKF or Rao-Blackwellized particle filter (analogous to FastSLAM) that estimates the map and robot pose for each robot separately.
- **Outcomes**:
  - ☐ **bronze**: Feature-based SLAM using the AprilTags as landmarks; evaluation of accuracy and consistency relative to ground-truth map.
  - ☐ **silver**: Feature-based SLAM using the lane segments as landmarks; evaluation of accuracy and consistency relative to ground-truth map; and visual comparison between rendered and reference maps.
  - ☐ **gold**: Feature-based Rao-Blackwellized SLAM using line segments.

Team: Visual odometry

Mission: *Enable the robots to track its pose using a single monocular method*

Motto:

Members:

Support people:

Supervisor:

Finish slide

Visual odometry/SLAM: classical and learning-based

- **Motivation**: Localization and mapping is one of the key challenges for Duckiebots navigating Duckietown. Several promising approaches have been proposed recently, including the approach of Zhou et al. (https://arxiv.org/abs/1704.07813), which performs monocular depth and pose estimation using raw video data.

- **Possible approach**: Adapt the recent implementation of (Zhou et al) to run on the compute-constrained Raspberry Pi, and then incorporate filtering to build a full SLAM system
- **Outcomes**:
  - ☐ **bronze**: Implement existing algorithm on re
    Raspberry Pi.
  - ☐ **silver**: Same as bronze with the addition of f
  - ☐ **gold**: ???

Add gold standard

Other ideas

ADD FLAGS HERE (NATIONAL vs INTERNATIONAL PROJECT)

#devel-automap

Map tools / Graphical Map Embodied Programming

- **Motivation**: One advantage we have in Duckietown is our ti
of the environment. Tiles are one of N types and the placem
road markings are all tightly constrained. As a result, we sho
specify a map and then autogenerate both metric and topolo
representations
- **Proposed approach:** Make a drag-and-drop GUI where ma
designed and then a process for automatically verifying the
generating navigation and localization maps
- **Outcomes:**
  - ☐ **bronze**: Automated metric map generation
  - ☐ **silver**: Automated navigation map generation
  - ☐ **gold**: Make the GUI for dragging and dropping

+ navigation

Finish slide
(harmonize with last min
additions)

Duckiebot Hardware redesign

- **Motivation:** The current hardware design of Duckiebots is perfectible and the cost relatively high. Moreover, reducing the size of Duckiebots would allow for deployment of bigger fleets on given Duckietown, enhancing the research value of the platform.
- **Possible approach:** Electronics: (a) merge LED and PWM HATs; (b) evaluate external contribution for addressable LEDs; Mechanical: (c) redesign chassis; (d) evaluate smaller batteries; Cuteness: (e) explicitly consider user customization in design (e.g., spot for duckie(s), shells, license plate, …) measure battery level (voltages)
- **Outcomes**: while keeping current functionalities:
  - ☐ **bronze**: drive LEDs with a cheaper solution than PWM + LED hat
  - ☐ **silver**: redesign chassis to minimize axle length, reduce overall cost to <125 CHF, minimize custom produced solutions (maximize off-the shelf)
  - ☐ **gold**: produce functional DB18 prototype, maximize cuteness factor

Learning Latent Representations

-How to find latent features that impact the performance of the system?

- e.g. 1 million duckiebots, 100,000 where things break: why?

  - answer:

    50,000 -> lighting condition

    30,000 -> the map was incorrect

    20,000 -> not calibrated

- disentangled features

- https://arxiv.org/abs/1703.07718

- cloud-based learning

- leverages the diversity of our dataset in some way? to find nuisance parameters etc

Liam needs a little more time to think about this one

Julian

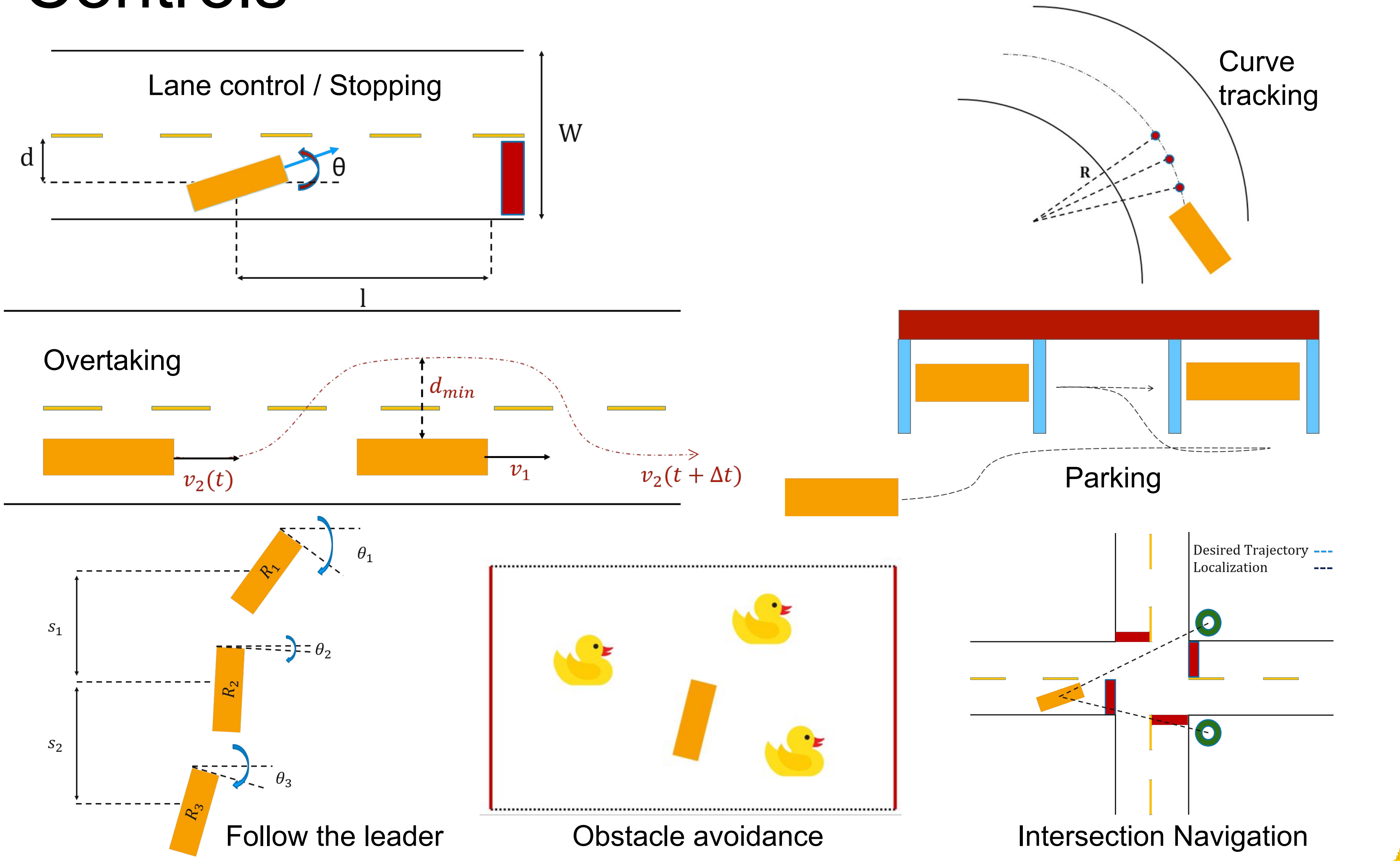
~~Improved low-latency lane localization~~

- **Motivation:** Fast, accurate and robust pose estimation is critical for controlling Duckiebots well. The current lane-filter implementation uses hard-coded appearance specifications and suffers from visual clutter disturbances. We need to improve its robustness.
- **Possible approach:** use improved DB model in lane filter, reduce visual clutter restricting field of view, evaluate particle filter approach, estimate lane and tape width as well as road curvature.
- **Outcomes**:
  - □ **bronze**: benchmark current lane-filter implementation, estimate distance from obstacle (stop line, Duckiebot, duckies)
  - □ **silver**: improved robustness
  - □ **gold**: estimation of world model parameters

Controls

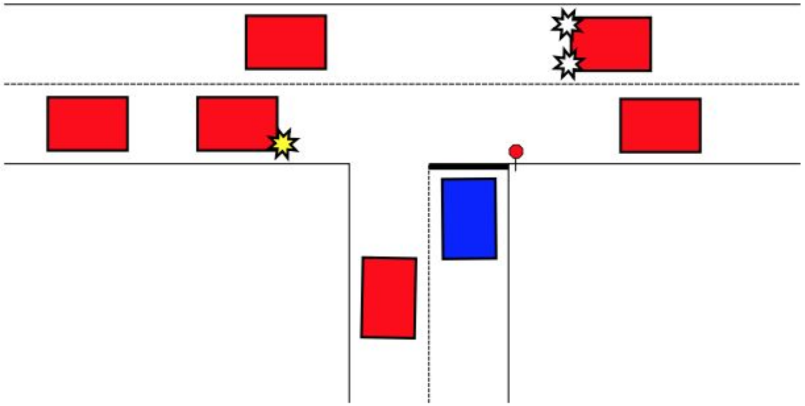
- **Motivation**: Whatever we plan, controllers must execute. The current implementation is a P control based on a linearized kinematic model. We can do better than that.
- **Proposed approach:** Consider several tasks: (1) lane control, (2) curve control, (3) stopping at red line, (4) intersection navigation, (5) obstacle avoidance, (6) follow the leader, (7) parking and (8) overtaking. For each, benchmark when applicable, create baseline when not. Analyse PID performance systematically. Use nonlinear model; Pure pursuit controller, evaluate nonlinear control approaches (feedback linearization, MPC).
- **Outcomes:**
  - ☐ **bronze**: Baseline and benchmarking for each task
  - ☐ **silver**: Improve tracking performance and robustness for most tasks
  - ☐ **gold**: Unified controller approach, different solutions with trade-off analysis implementable by changing a single parameter

Controls


Lane control / Stopping
d
θ
W
l
Curve tracking
R
Overtaking
d_{min}
$v_2(t)$
v_1
$v_2(t + \Delta t)$
Parking
θ_1
R_1
s_1
θ_2
R_2
s_2
θ_3
R_3
Follow the leader
Obstacle avoidance
Desired Trajectory
Localization
Intersection Navigation

~~Detection, Tracking, and Prediction~~

- **Motivation**: One of the biggest problems plaguing self-driving vehicles is the ability to infer and predict what is about to happen in the near future. In practice, there are not very many "likely" options and we can infer these options by knowing the road topology
- **Proposed approach:** Implement state-of-the-art object detection and tracking methods. Once complete, use time-series analysis and hybrid discrete-continuous estimation to estimate the likely future actions of other Duckiebots. To achieve this we will leverage both geometric and semantic information.
- **Outcome:**
  - ☐ **bronze**: Detection
  - ☐ **silver**: Tracking
  - ☐ **gold**: Prediction



Automated Algorithm Selection

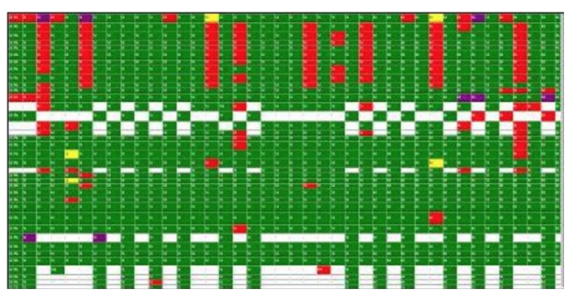
- **Motivation**: How can we have hundreds to thousands of people contribute algorithms to Duckietown? Perhaps by having an automated system that given an algorithm, automatically evaluates it, and if better than the current one, it adds it to the baseline configuration.
- **Outcomes**:
  - □ **bronze**: completion of regression test framework, including both functionality and performance
  - □ **silver**: All algorithms in Duckietown are integrated in the test framework. We can easily tweak parameters and be sure that the performance is improved.
  - □ **gold**: cloud-based system where everybody can submit their work.

What-the-ducking-duck

- **Motivation**: as we have seen, much of the difficulty is in configuring the system and especially have uniform configuration.
- **Proposed approach:** turbo-charge what-the-duck + remote dashboard
- **Outcomes:**
  - □ **bronze**: Scalable system to have remote configuration management, and automated tools (installable via apt-get)
  - □ **silver**: It is possible to upload logs automatically, especially for diagnosing problems;
  - □ **gold**:
    - It is possible to remote SSH into any Duckiebot in the world;
    - it is possible to visualize ROS topics remotely;
    - one-click "share to Facebook" functionality

Chatbot - a personalized debugging assistant

- **Motivation**: 1) We want to massively scale this project up. 2) The vast majority of questions that we have received are answerable from either a) the Duckiebook (although the content on there changes rapidly), b) looking at the output of the "What-the-duck" or c) looking at the "Remote Dashboard" (if the robot is on and connected)
- **Proposed approach:** Build a chatbot that is "physically aware"
- **Outcome:**
    - ☐ **bronze**: Implement a basic chatbot by using the Slack API and referring people to the appropriate section in the book.
    - ☐ **silver**: Connect the chatbot to the output of "What the Duck"
    - ☐ **gold**: Enable Chatbot to interact with remote dashboard and recommend actions to the user and observe the output



Formal co-design of Duckiebot platforms

- **Motivation**: We want to create a platform that is cheap, easy to use and assemble everywhere in the world, while maintaining certain functionality and performance guarantees.
- **Proposed approach:** Model the problem using the theory of co-design. Optimize at the same time over hardware components, including sensing, computing, and actuation, algorithms, and algorithm parameters.
- **Outcome:**
  - ☐ **bronze**: Formalize the choices that we have made so far, by obtaining performance functionality/resource curves for all components (e.g. current-to-speed for actuators, resolution to flops for CV processing)
  - ☐ **silver**: Either show that we made the optimal choices, or show what choices could have been made differently.
  - ☐ **gold**: Create an automated system that creates the optimal Duckiebot given budget constraints and functionality goals. Optimize over Duckietowns and Duckiebots at the same time.

#devel-self-calibration

~~Self-Supervised Calibration~~

- **Motivation**: Calibrating the robot (odometry and camera) is time consuming and fragile operation.
- **Proposed approach:** Use the existing elements of the system to bootstrap the calibration using self-supervised learning. For example, use the entropy of localization as a reward signal to automate the learning of the camera calibration. Consider algorithms such as pix2pix neural network.
- **Outcome:**
    - □ **bronze**: Solid self-awareness of calibration quality.
    - □ **silver**: Automated self-supervised self-calibration for one component.
    - □ **gold**: Automated self-supervised self-calibration for multiple components under the same framework..