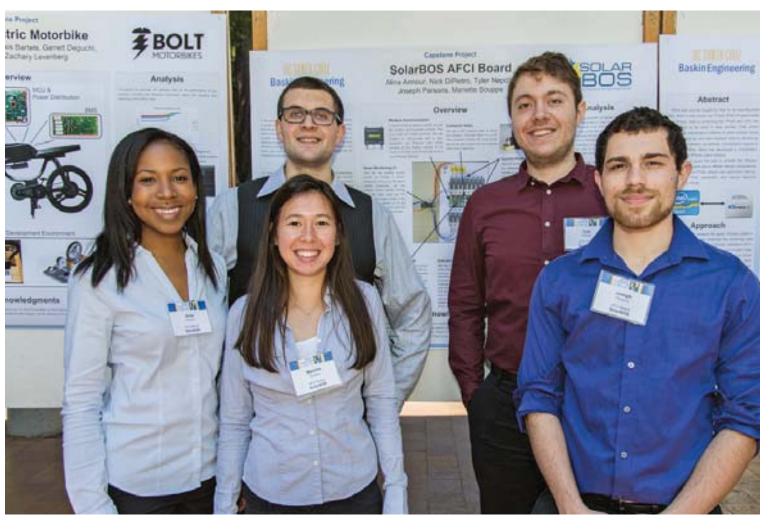
Baskin Engineering | UC SANTA CRUZ

PARTNERS' DAY

CORPORATE SPONSORED
SENIOR PROJECTS

2019 PROGRAM



Introduction

I am delighted to present the **2018-19 Corporate Sponsored Senior Projects Program (CSSPP)** at the Baskin School of Engineering. This publication includes a selected group of this year's 60 capstone projects carried out by student teams in Computer Science, Computer Engineering and Electrical Engineering. These immersive projects represent truly collaborative efforts initiated by students and their faculty mentors.

CSSPP embodies the essential engineering enterprise – working collaboratively to create new technologies that address real-world challenges. Throughout the academic year, students interact closely with teammates; some making visits to their corporate sponsor's worksite, and all experiencing the opportunity to solve problems along the way.

Working with mentors at corporate partner companies, students begin to understand what it means to be a professional engineer. For this, we deeply appreciate our corporate sponsors and their support of CSSPP at Baskin Engineering.

On behalf of our generous corporate sponsors and my colleagues on the faculty and staff at Baskin Engineering, I congratulate our extraordinary students, who have worked hard and enriched our lives through their energy, intellect, and determination. I know you will be impressed by their work.

Alexander L. Wolf
Dean
Baskin School of Engineering

Baskin Engineering UC SANTA CRUZ

Acknowledgments

We would like to acknowledge and thank the faculty, teaching assistants, and staff who have been so instrumental in the Corporate Sponsored Senior Projects Program:

SENIOR DESIGN FACULTY
CORPORATE SPONSORED SENIOR PROJECT PROGRAM 2018-2019

Patrick E. Mantey
Professor of Electrical & Computer Engineering
Director, Senior Design Capstone
Director, Information Technologies Institute

Stephen Petersen
Teaching Professor & Undergraduate Director, Electrical & Computer Engineering

Gabriel Elkaim
Professor, Electrical & Computer Engineering

Max Dunne
Lecturer, Electrical & Computer Engineering

Tela Favaloro
Lecturer, Electrical & Computer Engineering

Richard Jullig
Lecturer, Computer Science & Engineering

TEACHING ASSISTANTS & GRADUATE RESEARCH ASSISTANTS
CORPORATE SPONSORED SENIOR PROJECT PROGRAM 2018-2019

Joseph Adamson

Morteza Behrooz

Eric Cao

Emily Evers Enlow

David Kooi

Alec Reid

Dylan Rothfeld

Roy Shadmon

Alexandra Stambaugh

Reihaneh Torkzadeh Mahani

Baskin Engineering UC SANTA CRUZ

CORPORATE SPONSORED SENIOR PROJECTS PROGRAM STAFF

Alexander Wolf
Dean, Baskin School of Engineering

Jeffrey Jue
Special Agreements Officer, Industry Alliances & Technology Commercialization

Frank Howley
Senior Director of Corporate Development, Baskin School of Engineering

Lynne Sheehan
Network Administrator, Facilities/Machine Room, Baskin School of Engineering

Bob Vitale
Director of Laboratories & Facilities, Baskin School of Engineering

Tony Sabala
Facilities Manager, Baskin School of Engineering

Russell Evans
Senior Development Engineer, Baskin School of Engineering

Kelsey Evans Detwiler
Development Assistant, Baskin School of Engineering, University Relations

Emelye Neff
Administrative Assistant, Baskin School of Engineering

Sponsors

SPECIAL THANKS TO OUR SPONSORS for your generous support of our Corporate Sponsored Senior Projects Program. Your time, experience and financial support were beneficial to our students and the success of their Senior Design Projects.

Accolade

AUTOSENSE International

CISCO™

Continental

DEEPMAP

Hewlett Packard Enterprise

poly

skyTran®

Accolade Master Data Management

Ruchi Sheth, Paulo Engelke, Leon Pham,
Shruti Jain, Jazmin Maravilla, Khoa Hoang

Approach

I Given data records of client healthcare activities, the content of a record is organized into a set of attributes. In a table, each row represents one person record.

II In order to find out which records belong to the same client, we had to compare each row with every other row. That resulted in a table holding every possible pair of rows.

III For each comparison, we used a similarity measure based on the data value in the attribute, such as: binary value for numbers, discrete distance values for locations, etc.

IV The comparison results are placed into a list which is fed into the random forest machine learning model. The model trains on the list and predicts whether two rows belong to the same client. The prediction is compared with the client ID match of those records.

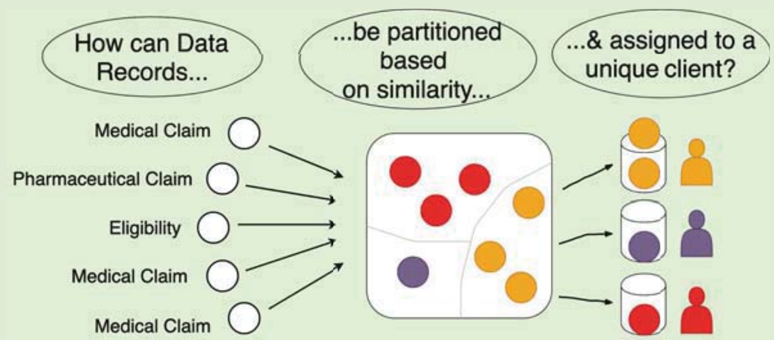
Acknowledgements

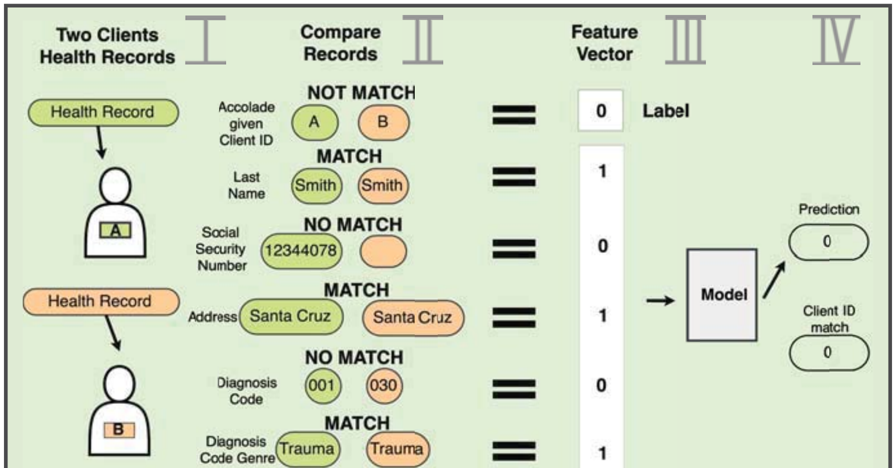
UCSC: Dr. Richard Jullig, Roy Shadmon, Dylan Rothfeld, Patrick Mantey, Emelye Neff
Accolade: Bilal Nemutlu, Michael Distasio, Ashutosh Dube, Jaff Kosar, Ankita Jain, Eric Salerno, Mike Hilton

Overview

Accolade provides a healthcare concierge service to employees of client companies. The clients and partners of Accolade provide health care records and personal information. Records can be filled out poorly, have incorrect information or share very similar personal information across individuals. This makes it difficult to map data to a single person.

Goal: As a proof of concept, we utilized the random forest machine learning model to classify if two health care records belong to the same person.





Results

Model: The Random Forest Classifier library in Python does not allow for incremental training. Therefore, we needed to keep all the data in memory and switch our classifier in scope.

Speed: Initially, we compared records row by row. Doubling the data, exponentially increased the time. We built our own Dictionary Based Comparison Algorithm. This was 60 times faster. This led to memory problems.

Memory: The resources we utilized to program the above solution was not enough for the data we wanted to process in terms of available memory. We modified our comparison algorithm with native Pandas methods. This gave us more functionality and consistency.

Accuracy: Our Accuracy is close to a 100%. This is due to a limited training set. Cases that are harder to match are rare and have to be filtered out systematically.

Test: This model will be used by Accolade to predict whether two records are the same. The given matches of attributes will provide judgement in whether there is a match.

Baskin Engineering UC SANTA CRUZ

Accolade

Capstone Project

Modular Data Pipeline

Juan Andreas, Emily Madigan, Christian Ortiz,
Ryan Bautista, Wellford Chan, Egan Bisma

Abstract

Accolade changes how company employees and their family members receive the health programs they need for happier and more productive lives. By combining technology and human expertise in new ways, Accolade provides a personalized health and benefits solution for employers, health plans, and their members.

The main goal of this project is to build an Extract-Transform-Load (ETL) data pipeline that produces a featured data repository. Using this, the secondary goal is to create an encounter prediction model. The model's purpose is 1) to validate the output of the pipeline and 2) to predict future client encounters. Our pipeline will improve the availability of data and help teams plan for future services.

Approach

To handle the large amounts of data required, this project leverages PySpark, a distributed cluster-computing framework; and AWS S3, a cloud storage system, to construct the training dataset. Data curation in the pipeline is composed of four phases:

1. The **Extract Phase** automatically collects client data recorded in data sources by Health Assistants.
2. The **Transform Phase** transforms the extracted data from a raw categorical form into a numerical representation that machine learning models can train on. Using PySpark, the pipeline executes custom scripts and functions to transform multiple datasets.
3. The **Load Phase** writes the transformed data, as partitioned data files, back into AWS S3.
4. The **Combine Phase** combines unique, transformed datasets into one training dataset.

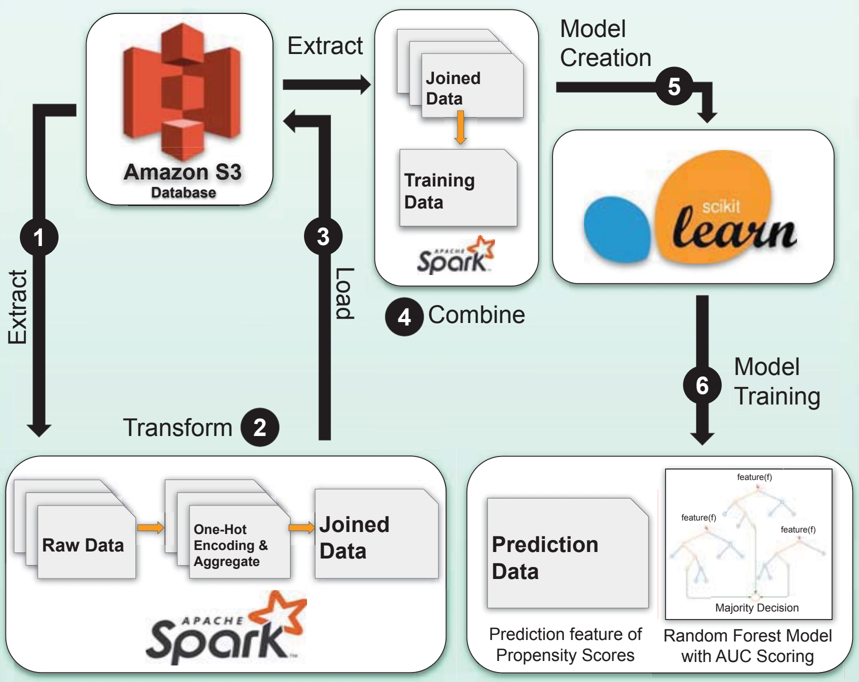
The pipeline trains a model for encounter prediction with this dataset using Scikit-Learn, a machine learning Python library. This is composed of two phases:

5. The **Model Creation Phase** implements a Random Forest Regressor. The regression problem this model solves is predicting the probability of a client having a future encounter with Accolade.
6. The **Model Training Phase** trains the model on the training dataset. The effectiveness of the model is measured by the Area Under the Curve (AUC) score and produces a new dataset-feature from the prediction as propensity scores.

Overview

Data engineering is a time consuming process, but it is paramount and essential for training machine learning models. The Modular Data Pipeline provides a framework to automate the cleaning of data for machine learning to help Accolade provide a better personalized experience for their clients.

Architecture



Acknowledgments

We would like to thank our Accolade Inc. sponsors Michael Distasio and Adi Anandkumar for giving us this opportunity. We would like to thank Dr. Richard Jullig for helping us enforce best practices for software engineering. Finally, we would also like to thank our TAs Reihaneh Torkzadehmahani and Morteza Behrooz for their guidance.

Design

Goal
To create a reusable end-to-end data pipeline for industrial machine learning.

Difficulties
This project originally utilized AWS Glue, a cloud service, to automate parts of the ETL pipeline. However, two challenges with AWS Glue included the service not being able to fulfill the project needs, and a lack of permissions to build new infrastructure. To solve this issue, Accolade recommended local development instead of using this cloud service.

Results

Aggregation Library
Data Scientists can modify data using modular functions. For example,

- Data Scientists are able to create new features by aggregating information from extracted data.
- Data Scientists are able to one-hot encode categorical values into nominal binary values.

Automated ETL Data Pipeline
Data Scientists can construct different building blocks of a training dataset for machine learning models in one go. By combining Docker; a virtualization software, with Bash scripts, each phase is sequentially executed.

Encounter Prediction Model
Data Scientists can use the training dataset produced by the pipeline to predict the probability that a customer will have another Accolade encounter in the future.

Conclusion

This project creates a reusable tool to assist Data Scientists at Accolade with automated data curation of raw data sets. The Modular Data Pipeline will increase Accolade's availability of data prepared for machine learning usage. As we continue to iteratively improve our machine learning models, we will be able to more accurately predict which services Accolade's customers need while also reducing overhead in Accolade's productivity.

Capstone Project

Autosense Cloud-Connected BAIID

Alysia Tran, David Li, Mehtab Badh, Samin Akter

Abstract

In 1986, Autosense International developed a user-friendly Breath Alcohol Ignition Interlock Device (**BAIID**) which is required for drivers who have been convicted of a DUI offense.

The BAIID prevents the vehicle from starting when the driver is intoxicated. It uses a host of sensors to correctly determine breath alcohol concentrations and to detect tampering. The BAIID comes with cellular connectivity, making it possible to connect to an online server.

Approach

The project was divided into two parts:

(1) the device to cloud communication between the BAIID, proxy server, and database.

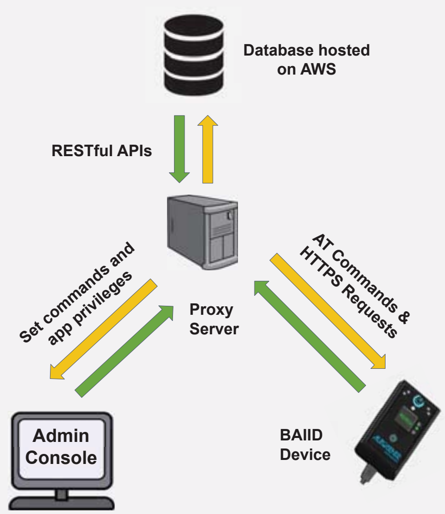
(2) the Admin user interface through which BAIID commands can be configured and BAIID data can be downloaded.

The device to cloud part of the system was implemented using HTTP POST and GET RESTful APIs for the data exchange between BAIID and cloud-based proxy server, instead of the previous SOAP model. The data exchange between the proxy server and cloud-based database was implemented via RESTful APIs written in Python. The client side of the Admin Web application was implemented with HTML5, CSS, and BootStrap. The server side was implemented in PHP and MySQL for the database queries and dynamic Web page support.

Overview

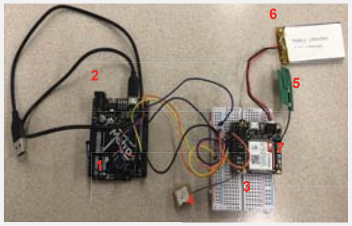
The goal of this project is to streamline the data transfer between BAIIDs (Breath Alcohol Ignition Interlock Devices) and Autosense's servers and web applications. This involves designing and implementing the software architecture for a cloud-connected BAIID, through which the device can be made to execute specific commands and upload vital device operational data to the cloud. In addition, a cloud-based admin console application has been built for service center technicians, distributors, and administrators to monitor, control, and collect sensor data of the BAIID Devices.

Architecture



BAIID Simulator

The Arduino Logic Board which simulates the BAIID is wired to a cellular phone module with a SIM card, two antennas, and a battery. The SIM card and antennas are used for GPS network connectivity (internet).



1. Arduino Logic Board
2. USB connection to computer
3. Adafruit FONA GSM (cellular phone module)
4. GPS Antenna
5. GSM/Cellular Antenna
6. Battery
7. SIM Card (below FONA device)

Acknowledgments

We would like to thank our sponsors Dr. Kishan Jainandunsing and Teresa Barrientez from Autosense International. We would also like to thank Richard Jullig, Patrick Mantey, Morteza Behrooz, and Reihana Torkzadehmahani for their guidance and mentorship.

User Application Hierarchy

At the top of the hierarchy is **Autosense International.**

Autosense International sends out its products to **distributors**.

Distributors then distribute them to **service centers**.

Service centers send out **technicians** to install the BAIID unit into people's vehicles.

Results

- The proxy server authenticates the Arduino.
- The Arduino receives the commands from the proxy server and reports back the execution status to the server.
- The Arduino uploads data to the database via the proxy server.
- The ACAP sets up commands, displays active and inactive commands and downloads data from the database to the user's PC.

Conclusion

This project brought together the different aspects of designing the software architecture of a cloud -connected device. Some future goals include implementing more features and extending the web application onto iOS and Android platforms.

Capstone Project

Cloud-Native

Anthony Campos, Barbara Moretto, Gavin Moy,
Katelyn Stone, Sabrina Tsui, Kevin Velasquez

Abstract

As demand increases for web applications, developers have had to create new internal tools and customer facing services to keep up with the rapid growth. Cisco Systems Inc. has created a set of open source tools, such as frameworks and plugins, to aid development for cloud-native applications. The Cloud-Native project consists of two components: Osseus, a tool for internal developers to generate template code when creating a new applications using existing plugins, and BGP application, a new feature that provides Border Gateway Protocol (BGP) functionality to a virtual network.

Approach

Osseus Development Tool

Osseus' development strategy was to split its production into separate efforts: building a user interface (UI), REST API and a code generator. The **UI** provides a way for the user to configure an agent setup consisting of chosen plugins. The **REST API** facilitates the communication between the UI and the data store (**Etcd**), which takes the user selected plugins and stores them. The **Generator** then produces boilerplate code based on the user's configuration and stores this back in Etcd where the UI webhook can retrieve the boilerplate code for the user to download.

Border Gateway Protocol Application

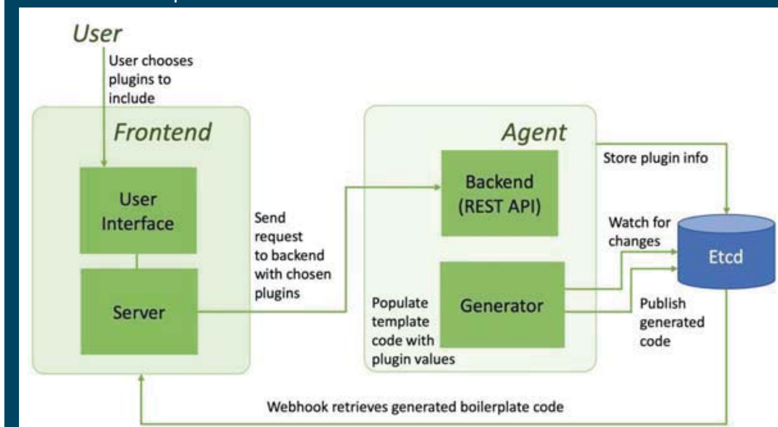
Kubernetes (k8s) is a container management system for cloud applications. K8s generates a cluster with a **master node** for managing the cluster, and **worker nodes** that run different instances of an application, each with its own **subnet**. The job of the **Border Gateway Protocol** (BGP) **Application** is to advertise the subnet information of a worker to any **BGP Peer.** This allows these peers to communicate with the different subnets. A **configuration file** is sent to the K8s API which specifies BGP configuration about the master and any peers. This is used to advertise subnet information to the peers. Using the K8s **API**, our **BGP Custom Resource Definition** (CRD) is able to store the configuration from the config file into **Etcd** (our data store). The BGP application watches for BGP configurations and advertises subnet updates to the BGP peers.

Overview

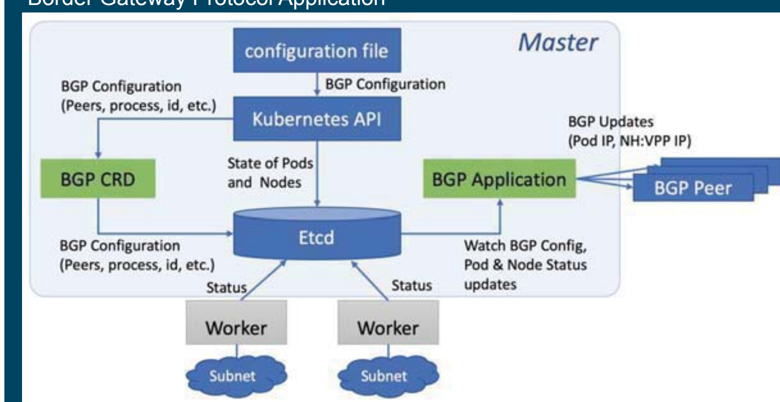
The Cloud-Native project automates two aspects of cloud-native applications: building template cloud applications and providing external communication into an existing virtual cluster. The Osseus internal tool provides a user interface to generate template code for engineers to avoid inconsistencies in code structure when manually writing boilerplate code. The Border Gateway Protocol (BGP) application automatically advertises the availability of resources (subnets) inside of a cluster to peers outside a cluster, allowing external users to access those internal resources shared by BGP.

Architecture

Osseus Development Tool



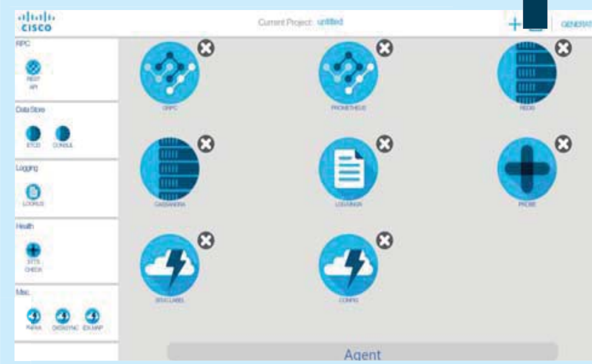
Border Gateway Protocol Application



Project Flow

The BGP app adds functionality to a k8s cluster and is an additional feature available to operators of virtual networks. The Osseus tool generates template code for cloud-native applications, such as BGP.

BGP Application



Results

Together, the Cloud-Native Project helps engineers by providing extra resources to aid in cloud development.

- The Osseus tool enables a mode of simple startup of applications and ensures a greater level uniformity among separate cloud-native projects.
- The BGP application allows cloud networking operators to quickly expose all the cloud-native applications in a cluster to external peers without the need to advertise each subnet separately.

Acknowledgements

We would like to thank our sponsors Nikos Bregiannis and Jan Medved for their support, as well as Pr. Mantely, Pr. Jullig and the CMPS 117 teaching staff.

Capstone Project

IoT Intrusion Detection for Traffic Security

Alexis Flores, Harshaan Krishnakumar, Ryan Hamamoto

UC SANTA CRUZ Baskin Engineering

CISCO

Abstract

With the emerging development of autonomous vehicles, the need for secure, robust software becomes an important factor in determining vehicle security. Failing to do so can lead to vulnerabilities that would allow potential hackers to tamper with vehicle safety features, send false information to roadside units, or violate traffic laws. Open source software such as Simulation for Urban Mobility (SUMO) can simulate realistic traffic scenarios involving autonomous vehicles but lacks the network infrastructure for communicating data between vehicles and roadside units, as well as a means of simulating potential threats. This project extends SUMO by providing a new vehicle class that mimics a compromised smart vehicle, new device modules that generate safety messages for wireless communications, and a means of collecting vehicle information to be used to detect malicious intruders.

Approach

The project focused on extending the existing software (mentioned above) to encompass more use cases and address future situations we may encounter with the integration of smart cars into our roadways. The design was implemented using:

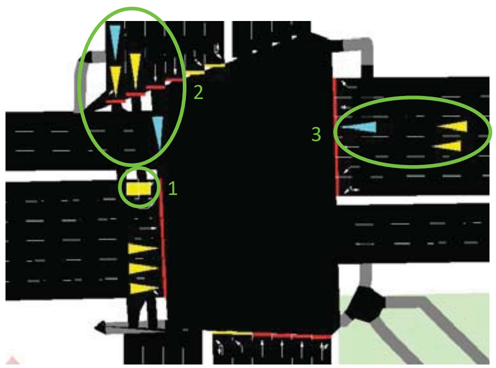
- ❖ A vehicle class to assign rogue car behavior to any vehicle within a given simulation.
- ❖ A library of callable python functions to interface with the simulation using a TCP based client/server architecture.
- ❖ A device class attachable to vehicles to collect and packetize relevant simulation data.
- ❖ A roadside unit object class to receive vehicle packets, parse the data, and detect rogue vehicles.

Overview

Cisco Security Business Group is interested in modeling vehicles that violate traffic laws within a simulation, capture vehicle behavioral data, and communicate said data via packets throughout a network of vehicles and roadside units within the simulation. The project utilized the following tools:

- ❖ **Simulation of Urban Mobility (SUMO)** to generate car mobility and road network.
- ❖ **OMNeT++** an event-based communication networks framework to generate simulations.
- ❖ **Vehicles in Network Simulation (Veins)** to execute vehicular models via **OMNeT++** while interacting with the road traffic simulator **(SUMO)** through a TCP socket and python scripts.

Simulation Frame



1. A roadside unit collecting all vehicle behavioral data to be parsed by a threat detection engine.
2. A rogue vehicle running a red light, while another rogue vehicle must stop to avoid creating a collision.
3. A rogue vehicle spoofing 5 different vehicles on the roadway.

- Normal Vehicle
- Rogue Vehicle
- Roadside Unit (RSU)
- Red Light
- Yellow Light
- Green Light

Dedicated Short Range Communications (DSRC)

Dedicated Short Range Communications (DSRC) is a protocol for short-range wireless communication channels specifically designed for automotive use. It provides the foundation for a variety of applications such as automated tolling, enhanced navigation, and traffic management. The DSRC standard proposes various safety messages such as Basic Safety Message (BSM). The goal of this project was to extend an existing vehicle mobility simulator (SUMO) to generate fields relevant to various types of DSRC safety messages. In doing so, we have created new software modules and vehicle classes that could capture fields relevant to BSM. This information can be processed by a roadside unit (RSU) located at an intersection, for example, to detect any traffic abnormalities.

Results

- ❖ Simulation testbed can declare and dynamically affect a vehicle that tailgates, runs red lights, merges frequently, speeds, and spoofs multiple vehicles.
- ❖ Implemented a DSRC Device module that is attached to a vehicle, and transmits real-time vehicular information such as speed, acceleration, wheel angle, position, and other parameters defined in Basic Safety Messages to roadside units that store and parse this data.
- ❖ Users can execute a script that randomly generates a simulation with all contributed components and vehicles such as potholes, roadside units, rogue cars, and DSRC devices.

Conclusion

The global self-driving cars and trucks market is anticipated to reach over 4 million units by 2030. We achieved the goal of granting users the ability to simulate smart cities with autonomous vehicles in a larger variety of scenarios to better prepare for the future.

Acknowledgements

We would like to thank the following people for their help throughout the last two quarters. Their contributions were vital to the development and improvement of the project:

Nancy Cam-Winget (Cisco SBG Sponsor)

Subharthi Paul (Cisco SBG Sponsor)

Saman Taghavi Zargar (Cisco SBG Sponsor)

Dr. Richard Jullig (Software Design Lecturer)

Morteza Behrooz (Teaching Assistant)

Reihaneh Torkzadehmahani (Teaching Assistant)

Capstone Project

Project GUPPI

Grand Unified Platform for Process Interactions

Joseph Aronson, Kate Miller, Japheth Frolick

Abstract

The goal of Project GUPPI is to create a unified platform for data scientists to easily access all the services they need.

Project GUPPI enables data scientists to maximize their productivity by providing them with convenient access to their most valuable tools, such as cloud services and collaboration platforms, all from within Jupyter Notebook.

Motivation

In today's technological market, companies have been collecting loads of information about users, internal services, and more. Interpreting and visualizing all this data has become a serious challenge. To combat this growing problem, a new branch of computer science called data science has emerged.

A data scientist writes software and builds models to help visualize and understand the massive amounts of collected data. Due to the field's recent emergence, the tools lack some essential features. Our goal is to create a tool for data scientists to migrate data, build and deploy models, and effectively communicate with their team all in the same place.

Jupyter Notebook

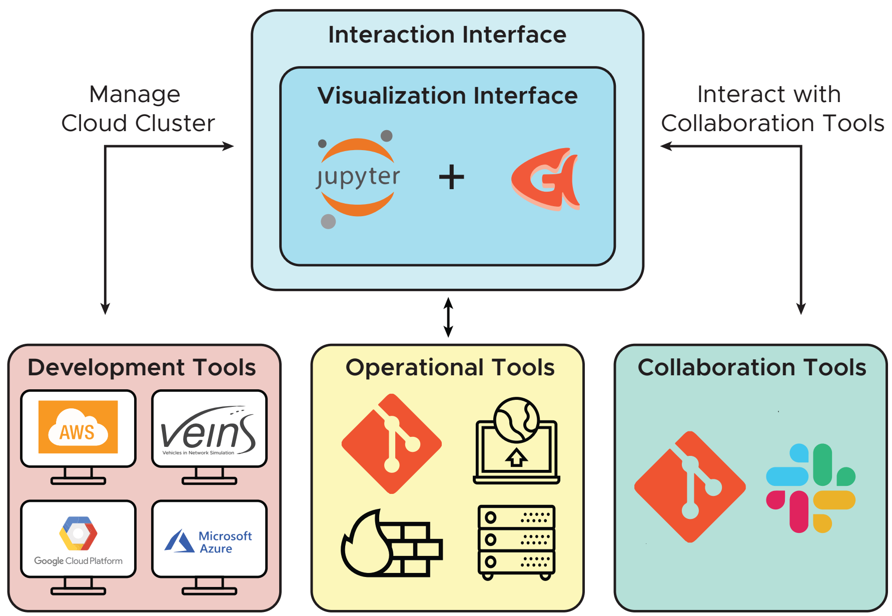
Jupyter Notebook is an open source web application used to build and share documents that contain text, visuals, and live code. It is utilized by data scientists to visualize data from various sources and share their findings.

Overview

Project GUPPI extends Jupyter Notebook, a tool that is commonly used by data scientists. This extension features new commands that can be run within Jupyter to complete otherwise tedious tasks, such as managing cloud instances, running models using secure remote login, and collaborating with your team.

Project GUPPI is a devops environment that allows users to interact with our new commands without the use of any external software.

Architecture



Acknowledgments

Thanks to our instructors Richard Jullig, Patrick Mantey, Reihaneh TorkzadehMahani, Morteza Behrooz, and our sponsors from the CISCO Security Business Group, Nancy Cam-Winget, Subharthi Paul, Saman Taghavi Zargar, and Oleg Bessonov.

Challenges

The toolset of a data scientist is constantly changing
Project GUPPI had to be designed to make it easy for developers to add new services. To accomplish this, the extension defines abstract classes and dynamically adds commands when a new service is added. This simplifies the process of creating new plugins

Limited by constraints of Jupyter
The process of building the interface was made convoluted by Jupyter Notebook's limited set of predefined interface elements.

Results

Project GUPPI features:

- An interface to create and manage instances over multiple cloud services
- An advanced tagging system to interact with groups of machines
- Remote command line access concurrently across multiple cloud computing services
- Integration with various collaboration tools such as Slack and GitHub
- An open source design making it easy for developers to add additional tools and services.

Conclusion

With Project GUPPI, data scientists can now migrate data from various cloud services, build and deploy models, and effectively communicate with their team all in the same place. We plan on continuing to work on this project and release it to open source.

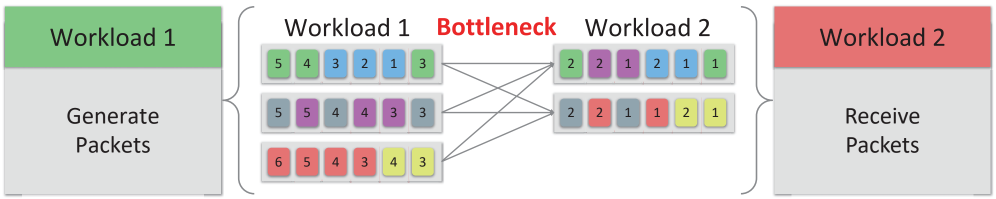
Capstone Project

Cisco Optimized Packet Passing

Reginald Chand, Cody Cunningham, Lito Pineda,
Anand Prabhakar, Ronuel Benjamin Tan, Alex Widmann

Introduction

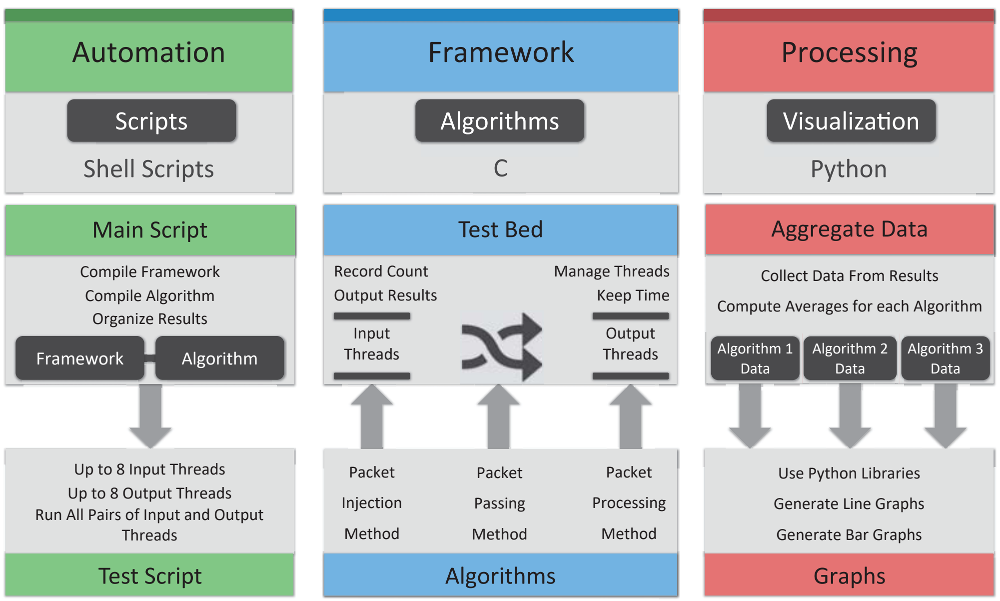
Virtualization is the process of using software to emulate what is normally accomplished with hardware, allowing better utilization of physical resources. Cisco has been moving towards virtualization in order to improve network efficiency and reduce the total cost of operation. However, Cisco discovered that passing packets between two sets of threads (workloads) within a virtual system is a bottleneck. The purpose of this project is to research various algorithms and optimization techniques that might reduce this bottleneck, increasing throughput between workloads. The aim is to surpass the industry hardware standard of 10Gbs, which is about 14 million packets per second.



Approach

Rather than working with existing solutions, to avoid any inherent limitations the system was built from the ground up using C and Assembly Language. The framework was built to define an interface for algorithms to insert their packet generation method and packet processing method, allowing algorithm performance to be tested in a standardized environment. After each run, the framework generates performance metrics and python scripts generate easily comparable graphs for further analysis. For efficient testing, scripts were created to automatically run various combinations of tests cases for any number of algorithms.

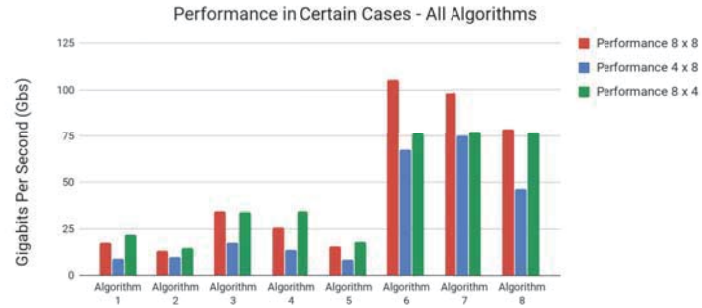
Architecture



Optimizations

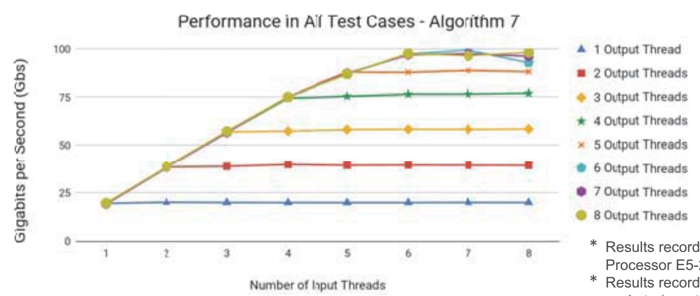
Each algorithm contains a combination of various optimizations	1	2	3	4	5	6	7	8
Flow Hashing - Redirecting packets to a certain output thread based on flow	✓				✓			
Bit Masking - An efficient method of hashing flows by bit masking packet headers		✓						✓
Queue Partitioning - Partition output queues so threads can write to it concurrently		✓	✓					✓
Divide and Conquer - Fixed assignments of input threads to output threads			✓	✓	✓	✓	✓	
Vector Passing - Instead of passing one by one, pass a group of packets each time			✓			✓	✓	✓
Cached Buffers - Exploit fast cache access by working in exclusive local buffers						✓	✓	✓
Contiguous Memory - Write packets back to back rather than in fixed size blocks						✓	✓	✓
Double Buffering - Split the shared queues in two for concurrent reads and writes			✓				✓	✓

Algorithms



All Algorithms Performance Graph

This graph shows cases where the number of input threads are equal to, less than, or greater than the number of output threads. The data is averaged from multiple runs where the payload size is the network standard of 64B.



Single Algorithm Performance Graph

The graph to the left shows the performance result for all 64 test cases and shows trends as the number of input/output threads change. The data is averaged from multiple runs.

* Results recorded on a dual socket system with 2 Intel® Xeon® Processor E5-2660 v2
* Results recorded with input and output threads on separate sockets (worst case)

Results

We found multiple optimization techniques and implemented them in various algorithms to test the effects they had on throughput. Analyzing the graphs of our data allowed us to measure their effectiveness. Optimizations such as double buffering, local buffers, contiguous memory, and divide and conquer techniques proved the most beneficial in improving throughput.

Acknowledgements

We would like to thank Ian Wells and Kyle Mestery, distinguished engineers at Cisco, for all of their work and involvement in this project. As well as professors Richard Jullig, Patrick Mantey, and Morteza Behrooz for all of their time and assistance.

Capstone Project

CRML Fusion

Doug Korody, Enes Yazgan, Ian Feekes, Lily Nguyen, Ruihong Yu, Volha Hancharova

Abstract

The advent of smart vehicles has opened up new paths to safer cars. Vehicles understand the environment by mapping their location and surroundings to detect potential hazards. This is usually done with LIDAR sensors, which can cost up to $70,000 per sensor. Using a far cheaper combination of RADAR and camera sensors as a substitute makes this process accessible and dependable.

Approach

We use the Robot Operating System (ROS2) environment to process multiple data streams concurrently. We display our RADAR data by converting the information to point clouds in the RViz visualizer tool. Our simultaneous localization and mapping algorithm (SLAM) uses the synchronized RADAR and odometry data to allow vehicles to localize themselves relative to static objects in offline scenarios.

Initial testing rig - equipped with RADAR, camera, and odometry

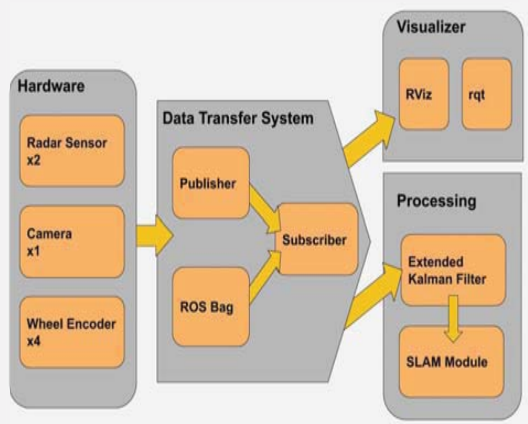
Overview

The ROS2 environment synchronizes data from the RADAR and camera sensors on our testing rig. After visualizing this fused data in RViz, we apply the Extended Kalman Filter to remove noise and then process this refined data in our SLAM algorithm. Using the filtered data and vehicle odometry, this algorithm maps the vehicle's surroundings and tracks the vehicle's position within the map.

Analysis

To reduce errors and noise in our RADAR data we use an Extended Kalman FIlter. This allows vehicle location and exterior objects to be estimated with higher degrees of accuracy over time. This filter allows us to generate a functional SLAM algorithm for our vehicle.

Architecture



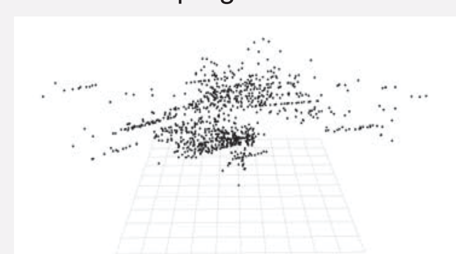
Acknowledgments

We would like to thank our sponsors at Continental Shuhei Takahashi and Dirk Remde, and our teaching staff Richard Jullig, Patrick Mantey, Morteza Behrooz, and Roy Shadmon for all of their help and guidance.

Final testing rig - equipped with similar sensors to the initial testing rig

Results

By applying RADAR data to our SLAM algorithm, we can create an updating map of our vehicle's surroundings. We also created a visualizer for our sensors' outputs. We have established a strong foundation for a large project; future work can augment the SLAM algorithm (adding loop-closure detection, for example) and verify the new work using our visualization program.



RADAR coordinates as a point cloud in the RViz visualization tool

Continental: Smart Intersection

Kai Huang, Alex Blumer, Michael Shen, Yutaka Tokunaga, Won Ko

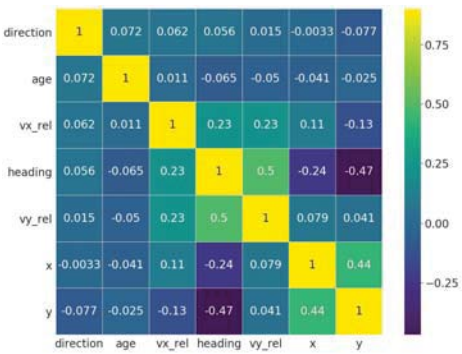
University of California, Santa Cruz

Motivation

With 2.5 million accidents in intersections occurring annually, this corporate sponsored project tackles the problem by predicting critical scenarios that may lead to collisions and pedestrian injuries. The project takes advantage of sensor/radar data and camera footage, to best represent all objects in the intersection.

Feature Engineering (Sensors)

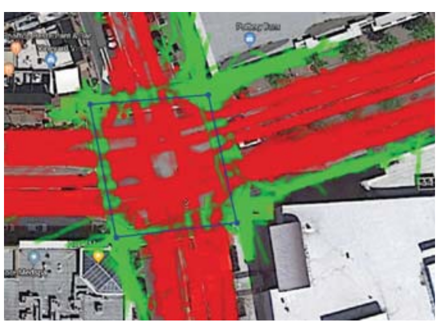
- Input: **x**, **y**, (**vx_rel**, **vy_rel**), (**heading**)
- Output prediction: **direction = left, right, straight**



Value directly proportional with relation metric

Dataset Preprocessing (Sensors)

- Drop non-complete trajectory of vehicle instance
- Drop instances with less than 20 data-points
- Label the turn direction and origin of each vehicle
  - Line segment intersection
  - Start and destination crosswalk
- Separate into two different datasets:
  - Full path of vehicle (clean)
  - Path of vehicle before intersection (test)
- Normalize distribution of prediction classes
- Z-transform to normalize values of input variables



Training - Vehicle Trajectory Prediction (Sensors)

- Given an instance of vehicle's path over its lifetime in the intersection, output turn prediction for straight, left, or right
- Temporal data analysis, best results came from Recurrent Neural Networks (LSTM)

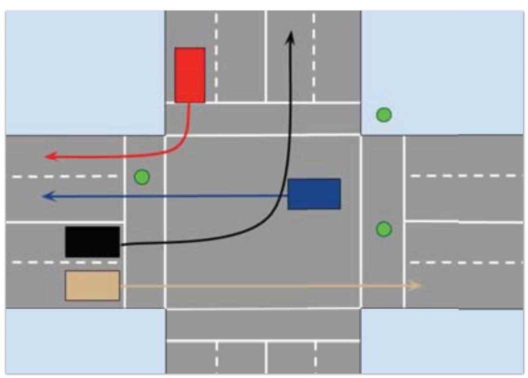
Model	Accuracy (before)	Accuracy (after)
Linear Regression	44.1%	57.3%
MLP	59.7%	67.5%
RNN (LSTM)	**98.6%**	**89.5%**

Model	Accuracy (before)	Accuracy (after)
1-Layer LSTM (30% dropout)	70.3% with SGD	66.3% with RMSprop
3-Layer LSTM (20% dropout)	71.5% with RMSprop	58.5% with SGD
3-Layer LSTM (no dropout)	**98.6% with RMSprop**	**89.5% with RMSprop**

- Pure sequence classification into turns (left, right, straight)
- Integration of scikit-learn's Grid Search into Keras models for tuning hyperparameters
- Data Reshaping to force consistency in instance timesteps
  - Variable length data points for each vehicle needed to be truncated or padded to reach values of 500 and/or 1000
- Variation in batch_size and epochs used
  - Batch size of **5-10** and epoch of **100-150**
- Optimization function
  - **SGD**, **RMSprop**, Adagrad, Adadelta, Adam, Adamax, Nadam

Critical Scenario Detection (Sensors)

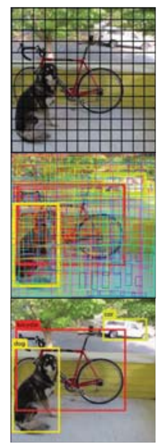
- Uses origin and predicted turn direction to identify vehicle path
- Vehicle-Vehicle collisions
  - Compare own path to those of vehicles in the intersection. Check for intersections and same final direction.
- Vehicle-Pedestrian Collisions
  - Checks which crosswalks a pedestrian is currently in
  - Each vehicle checks if its path crosses a crosswalk that contains a pedestrian



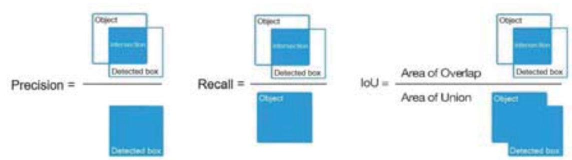
Pedestrian in immediate crosswalk:	Blue, Tan, Black
Pedestrian on other side of intersection:	Red, Tan
Vehicle crossing path:	Black
Vehicle potentially entering same lane:	Red

Dataset Preprocessing and Image Processing

- KITTI (cars, vans, trucks, person, sitters, cyclists, trams, misc)
- COCO (80 object classes)
- Custom (person, car, motorbike, bus, truck)
  - Footage provided by Continental from ground-truth camera
  - Automated labeling using filtered classes of YOLOv3-COCO
  - Deleted all video frames with no predictions
  - Split dataset into training set & testing set
    - Training Set: 80% (16,800 frames)
    - Testing Set: 20% (4,200 frames)
- YOLOv3 which uses darknet-53, pretrained on COCO used as a precursor for further training
  - Transferred learning for 12000 epochs, 9000 early-stop



	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			



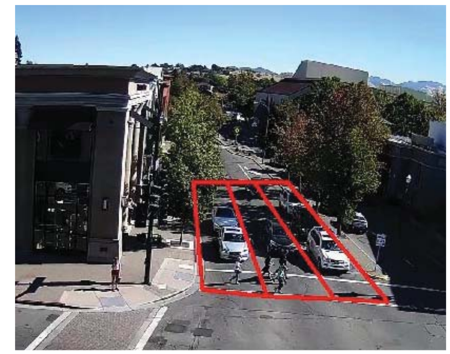
Dataset	IoU	mAP	FPS
YOLOv2 (KITTI)	70.12	33.51	8-10*
YOLOv3 (KITTI)	56.26	45.53	6-7*
YOLOv3-tiny (KITTI)	47.01	42.54	18-21*
YOLOv3 (COCO)	65.17	55.13	5-6*
YOLOv3-tiny (COCO)	53.53	33.15	20-25*
YOLOv3 (COCO + Custom)	**77.16**	**63.63**	**23-28**

Object Classes	Average Precision (AP)
Person	72.65
Car	95.99
Motorbike	23.66
Bus	94.93
Truck	94.31

* FPS performance via NVIDIA Jetson TX2

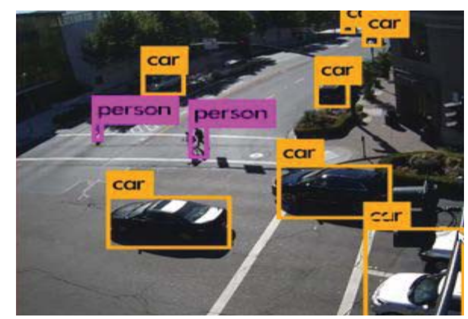
Critical Scenario (Computer Vision)

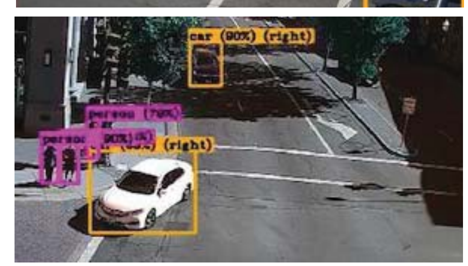
- Fixed camera perspective, so lane coordinates can be predefined into multiple static polygons
- >50% of overlapping area of vehicle's bounding box to the lane polygon determines vehicle's occupied lane and prediction of lane trajectory



- Alerts when detection of cars overlap (> 70%) with **critical regions** (sidewalks, buildings)
- When a pedestrian is detected to be within the trajectory of a vehicle
- Threat levels (alert, warning, critical) depending on the distance between a pedestrian and car

Object Classification Demo





Capstone Project

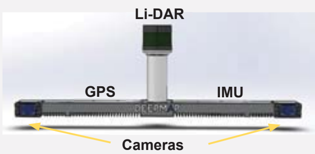
DeepMap Demonstration Robot

Harsh Bhakta , Talin Hallajian, Jian Hao Miao, Anil Celik Maral,
Sage Somers, Yuxuan Zhang

Objective

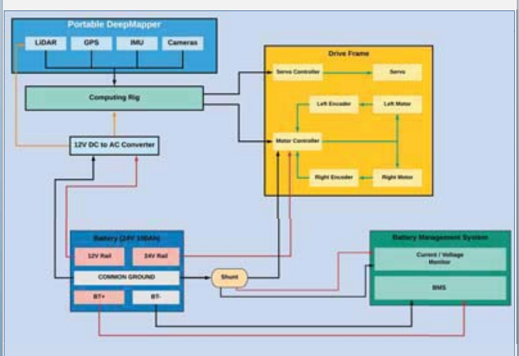
DeepMap is a company whose goal is to provide localization and mapping services to be used for autonomous navigation. Using their Portable DeepMapper, we were asked to create a robot with the following specifications:

- autonomous
- holds the DeepMapper
- battery life of at least 45 mins
- speed of 3 miles per hour
- weighs less than 200lbs
- showcases their localization and mapping capabilities
- can navigate within a map while maintaining safe driving skills



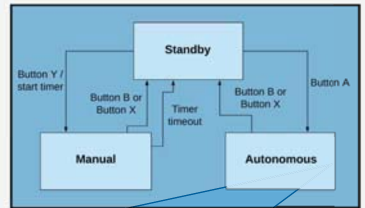
Electrical Overview

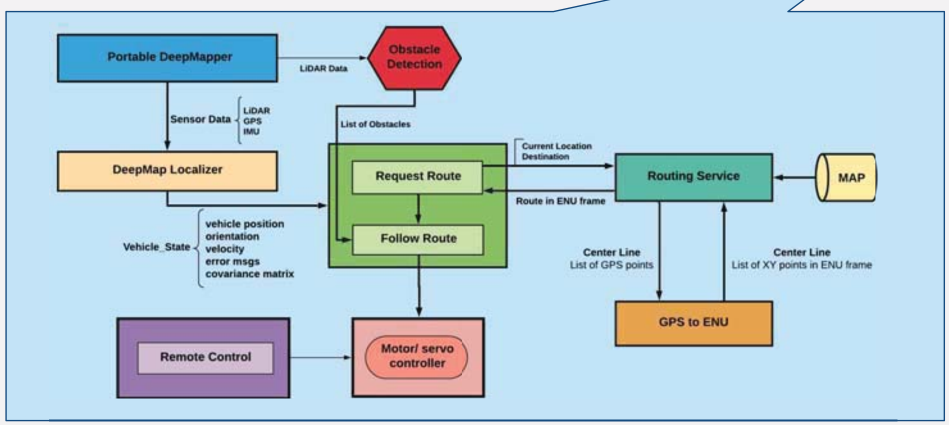
- Our entire system is powered by a 25.6V (eight cells at 3.2V each) battery monitored by a Battery Management System for regulating battery cell voltage and protecting the battery from over and under charging.
- The 12V rail of our battery is used to power the main computer as well as LIDAR, the BMS system, and the servo while the 24V rail of the battery is used to power the motors and the main driving system
- A variety of fuses are put throughout our system to prevent the overdrawing of current from the different components while a main emergency stop switch connected through common ground is used to cut power from the entire system if needed.



Software Architecture

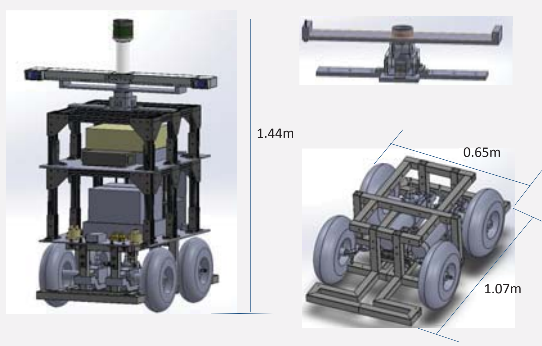
- We chose to use the ROS framework since it provides a robust modular environment with support for many different features.
- To use DeepMap's API we wrapped it into the ROS framework so it could be used within nodes for easier communication
- The main mode of interfacing with the robot is through an Xbox controller. This allows us to change states and move the robot manually.
- The robot has three modes controlled by a state machine: standby, manual, and autonomous
  - depending on the state, only certain commands will be accepted from the controller





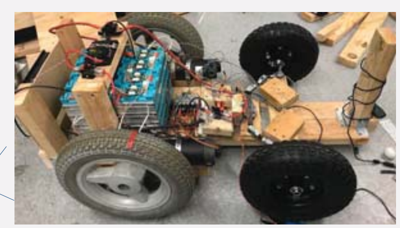
Mechanical Design

- The robot chassis was chosen to be made of aluminum because it is light and durable.
- Due to manufacturing costs this design was not implemented, so we created a prototype instead.
- The main objective of the mechanical design was to build a structure that can withstand the weight of our robot and can safely secure the Portable DeepMapper.



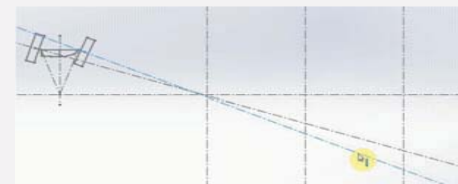
Prototype

- Due to long manufacturing times and high costs, we decided to build a wooden prototype for this project.
- We chose to use wood as it is cheap, readily available, and can easily and quickly be put together.
- Wood is also durable enough such that it can support the weight of our electrical components.



Ackerman Steering

- Rear wheel driven with Rack and pinion steering system.
- Front steering with Ackerman geometry to:
  - Minimize tire slippage.
  - Sweep around curves smoothly.
- Steering Angle: -15 to 15 degrees.
- Torque required to move the Rack is 50 kg-cm.



Conclusion

While we were unable to get the robot to move as autonomously as we would have liked due to various issues encountered within our time constraint, our robot still serves as a showcase of DeepMap's software and sensor rig and will be useful for demos and for testing purposes.

Acknowledgments

We would like to thank the following for their help and support:

Prof. Max Dunne, Prof. Gabe Elkaim, David Kooi, BELS, David Bulnes and the rest of the team at DeepMap

Sponsored By

DEEPMAP

Parallel ML

Kenny Luu, Rohit Falor, Simran Chawla,
Phil John, Kirby Choy, Avery Gascon

Abstract

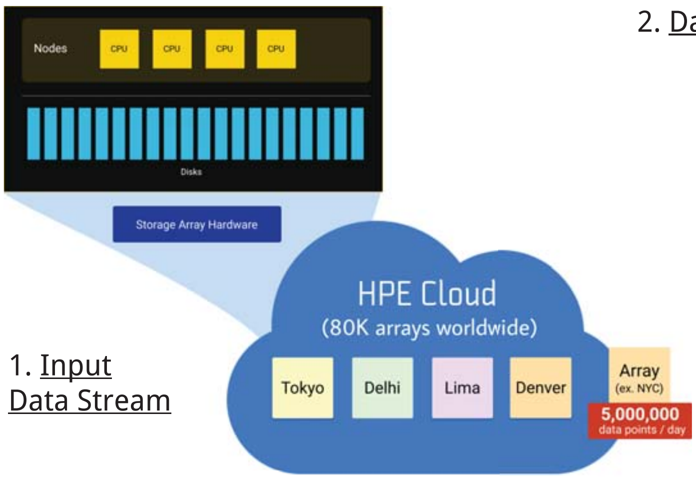
HPE has tens of thousands of 3PAR storage arrays that send back system data. Using this data, we were able to forecast and create predictions about storage usage and identify low performing storage arrays. This allows HPE to better serve their customers, including banks and hospitals, and opens up opportunities for HPE to offer their customers upgrades that better suit their needs.

Overview

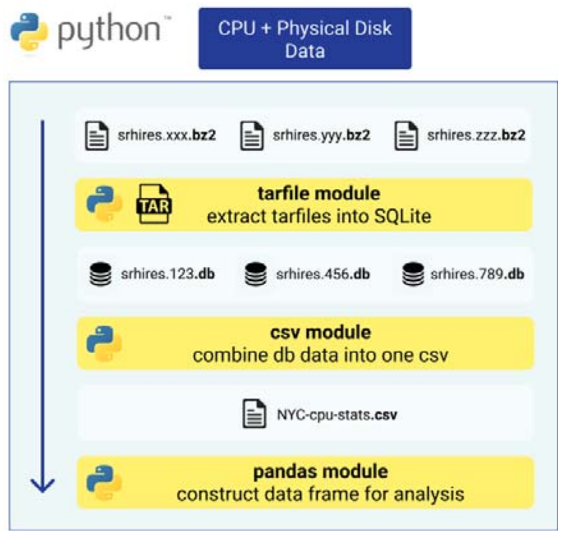
The two problems we analyzed were storage usage and CPU performance. 3PAR customers benefit from early warnings, thus we utilized these two key metrics. For CPU performance, we identified poorly utilized cores in the controller, so that customers can be alerted and upgrade if needed. For storage usage, we observed free / unused space over time, and if a customer is trending towards zero free space available, we can alert them as to when they will run out of space.

Architecture

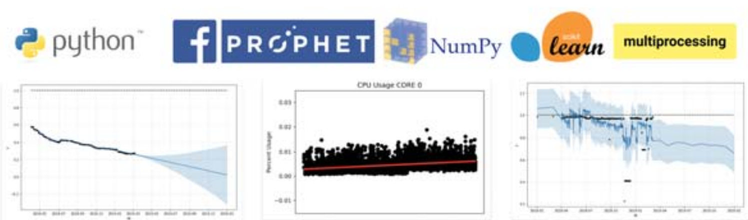
1. Input Data Stream



2. Data Processing and Cleanup



3. Parallel Machine Learning



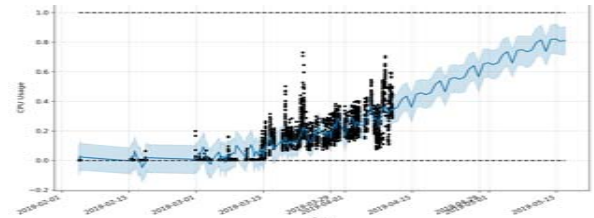
Linear Regression (PD) *Linear Regression (CPU)* *Logistic Model (CPU)*

Approach

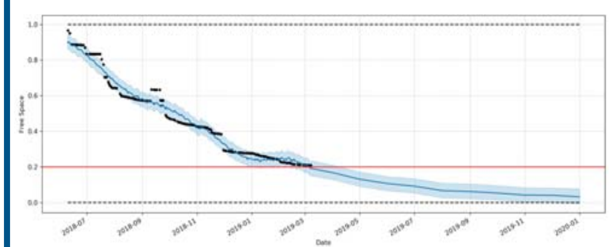
We approached the prediction problems as data scientists. First, the **input data stream** collects system data from around the world. The next step is the **data processing and cleanup**, where we process the data into a usable format. Then, using Jupyter Notebooks, we experimented with and manipulated the data to identify the factors of poor CPU usage and storage usage. Once we identified these factors, we used **parallel machine learning** to isolate the storage arrays that stood out. After these storage arrays were identified, we used Prophet, a data forecasting tool, to create forecasts on storage and CPU usage.

Results

CPU Usage Analysis:
By using Prophet forecasting, we can see how the CPU usage trends for the next few months. This was done by taking the average CPU usage of a storage array at each timestamp and computing a Linear Regression trendline through the data. We filtered out CPUs with less than 50% average usage over time and ensured there was enough samples in order to create a more accurate forecast.



Storage Space Analysis:
Similar to the process above, we used various storage space metrics such as free space and disk type to create a projection of the percentage of free space of a server in the coming months. We filtered out servers which will not reach below 20% usage within a year in order to get servers with concerning storage space usage.



Acknowledgments

Sponsors (HPE):
Ayman Abouelwafa, Bryan Carroll, Brandon Yates
UCSC:
Professor Richard Jullig, Morteza Behrooz, Reihaneh Torkzadehmahani

Capstone Project

PCBA Automated Robotic Manager(A.R.M.)

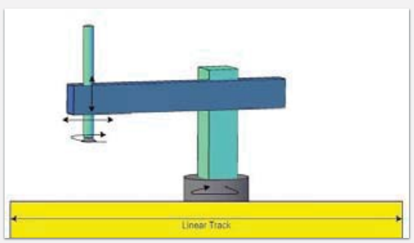
Miguel Flores, Stephen Kemp, Brandon Lake, Tianshu Ji
Aaron Diep, Javier Estrada, Xiaotian Zhang

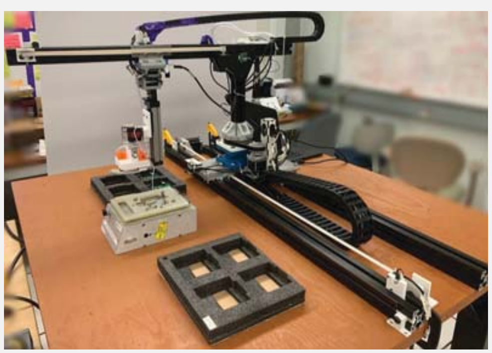
Poly's Need

- Poly is sponsoring the creation of a robotic system to automate their PCBA-testing process, which is currently done manually.
- Poly wants us to produce a proof-of-concept device to investigate the effectiveness of automation for their existing manual PCBA-testing process. Their goal is to have a faster system than the manual process, to achieve higher test throughput and to maximize operator time efficiency by having an unmanned test station.

Our Solution

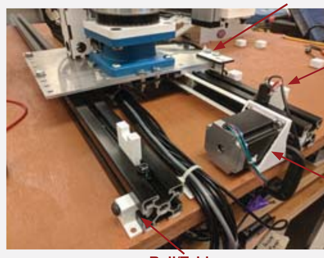
- The A.R.M. is a pick and place robot designed specifically for handling PCBAs.
- The A.R.M. picks up a single PCBA, places it into the test fixture and pings the testing computer to initiate the testing process. When the test is complete, the A.R.M. picks up the PCBA from the test fixture and places it into a Pass or Fail tray depending on the outcome of the test.
- The A.R.M. is a cylindrical robot moving in five degrees of freedom
  - Main linear track
  - Rotational base
  - End effector reach
  - End effector height
  - End effector rotation



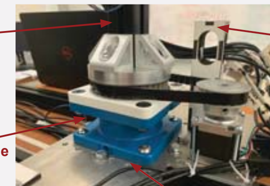


Mechanical Design Overview

1 *Main Linear Track*



2 *Rotational Body*

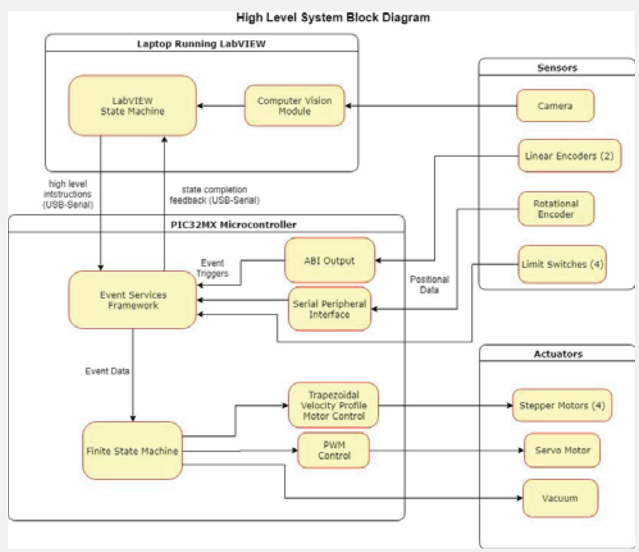


3 *End-Effector Reach*

4 *End-Effector Design*

System Flow

- Computer running LabVIEW sends instructions to PIC32MX µController using USB Serial Communication Protocol
- µController runs an Events/Services (E/S) framework checking for events and sending state transition requests to the finite state machine
- µController controls stepper motors using trapezoidal velocity profiles
- µController reads feedback from encoders using SPI and ABI protocols as events
- LabVIEW reads PCBA barcode data from Camera

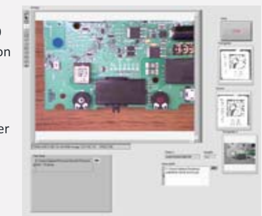


Hardware Used

- Sensors
  - 12 bit linear encoders using ABI output
  - 14 bit rotary encoder using SPI
- Actuators
  - 4 stepper motors of varying NEMA profiles and torques
  - 24V LewanSoul servo motor
  - 12V vacuum pump
- Control
  - Microchip MAX32 microcontroller

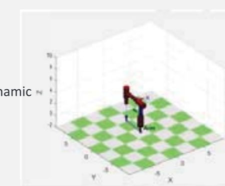
LabView Computer Vision

- Camera details:
  - National Instruments IMX259
  - Interfaces through USB3 Vision
  - Controlled using Labview software
- Camera uses :
  - Operates as a barcode scanner
  - Potentially used for PCBA orientation feedback



Simulation

- Linear Track
  - Linear Track Movement Algorithm
  - Linear Track Algorithm
- Rotational Arm
  - Rotation base Finite Element Analysis(FEA) in Solidworks
  - Rotation Arm and End Effect dynamic in Matlab
- Allows for
  - Optimal Pathing Simulations
  - Complete Test Time Estimates



Conclusion and Further Steps

We have developed a system that can operate at a high degree of precision in each axis. With the right programming and adaptation, the A.R.M. has the potential to operate with a variety of PCBAs or even as a general pick-and-place robot.

With the physical system and basic software framework complete, further time would be spent developing the high-level software to be used to manage large numbers of PCBAs and to calculate optimal pathing. Additionally, further feedback programming could be developed using the mounted camera with computer vision to increase the accuracy and potential applications of the system.

Acknowledgments

We would like to thank the following for their help and support:

Prof. Gabriel Hugh Elkaim, Prof. Michael Wehner, Prof. Maxwell Dunne, Mark Brinkerhoff of Fusion Design, Rick Frier, Carl Thompson, Warren Kruth, Shannon Thompson, and the rest of the Poly Test Engineering team!

Capstone Project

Matthew Kuzara, Tyler Gelzleichter, Andrew Pang, Jonathan Alvarado

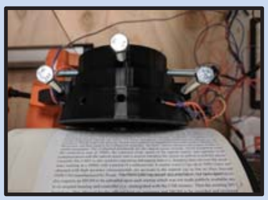
High Bandwidth Non-Contact Velocity and Distance Sensors

Objective

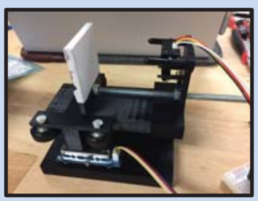
The project aims to create a non-contact high bandwidth sensor for tracking velocity and position along major axes.

The sensor is designed to be housed within skyTran's high speed autonomous transit vehicle and integrate with its internal control system via a feedback loop.

Mounts and Tests



Optical Sensor Mount: Designed to house the relensed optical chip and multiple IR LEDs. Holds both a fixed distance above the tracking surface.

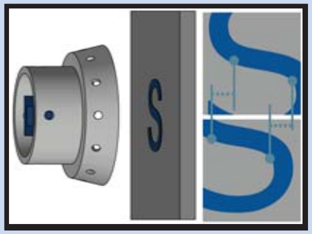


Distance Sensor Test Mount: Tested on a sliding rig that could hold two different sensors and move a flat surface back and forth with millimeter precision.

High Speed Test Bench: PVC spins in lathe to simulate linear speeds up to 25 m/s.

Verification: Encoder attached to the spinning axle of the lathe to confirm velocity. Sliding potentiometer connected to the moving surface of the distance sensor test mount to verify actual distance.

Optical Sensor

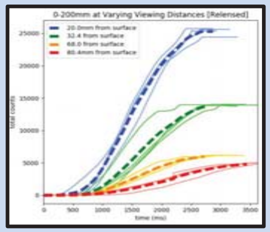


PMW3366:
- -Used in optical gaming mice
- -Removed from mouse and used in sensor to detect movements across planar surface.

CMOS Camera detects feature movement on surface to produce a positional displacement.

Displacement converted to meters via ratio dependent on the distance from viewable surface.

Velocity determined by reading the displacement at a fixed polling rate.



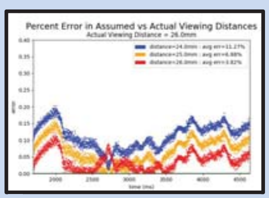
CMOS FOV limits max recognizable displacement between frames
- -limited movement can be tracked between two consecutive frames.
- -Increasing FOV increases max trackable velocity.

Captured CMOS frames with original lensing (left) vs. relensed (right). FOV changes from a less than 1 mm to ~3.5 mm.

Sensor relensed to increase CMOS FOV through addition of a second lens to the system.

Created two major drawbacks:
1. Reduced lighting
2. Decreased level of detail

Resolved by:
1. Adding brighter IR illumination
2. Adding pattern to surface



The average percent error increases as the assumed viewing distance deviates from the actual distance.

Distance Sensor

Goal: Looking for a way to measure short distance, with millimeter precision and high bandwidth.

Method: Two different distance sensors, combined with a complementary filter.

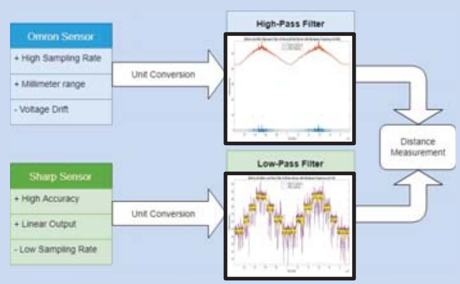
Omron Light Convergent Reflective Sensor: Capable of supporting high bandwidth and millimeter range. Distance output measurement drifts over time, creating consistency issues.

Synchronous Sampling: To account for lighting differences and ambient voltage output, the Omron sensor was sampled synchronously.

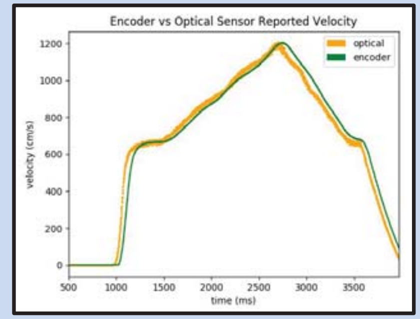
Distance: Output from the sensor is analog voltage, read into an analog-to-digital converter on our microprocessor, then converted to millimeters using a model of best fit based on our data.

Sharp Sensor: Much lower bandwidth, but higher accuracy with less drift. Used with our original sensor to stabilize readings.

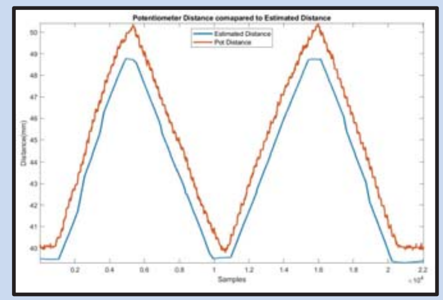
Complementary Filtering: High frequency signals from the Omron sensor and low frequency signals from the Sharp sensor were combined to create the output signal.



Results



The optical sensor has been tested against speeds of up to 20 m/s and can report the velocity at 1kHz with < 5% avg. error.



Within a ten millimeter range, the distance sensors are capable of measuring gap distance to millimeter precision at a frequency of 500Hz.

Acknowledgements

Max Dunne, David Kooi, Gabe Elkaim: for guidance and support.
Robert Baertsch and skyTran: for presenting the opportunity and making this possible
And **BSOE and BELS** for giving us a home

We are very pleased to include posters for the Senior Design Projects that were done without industry sponsors. Some of these projects were instigated and/or sponsored by research at the Baskin School of Engineering while others were created by students with the assistance of faculty mentors and TAs. We have selected three of these projects for presentation in the program, and all were invited to display their posters that summarize their projects.

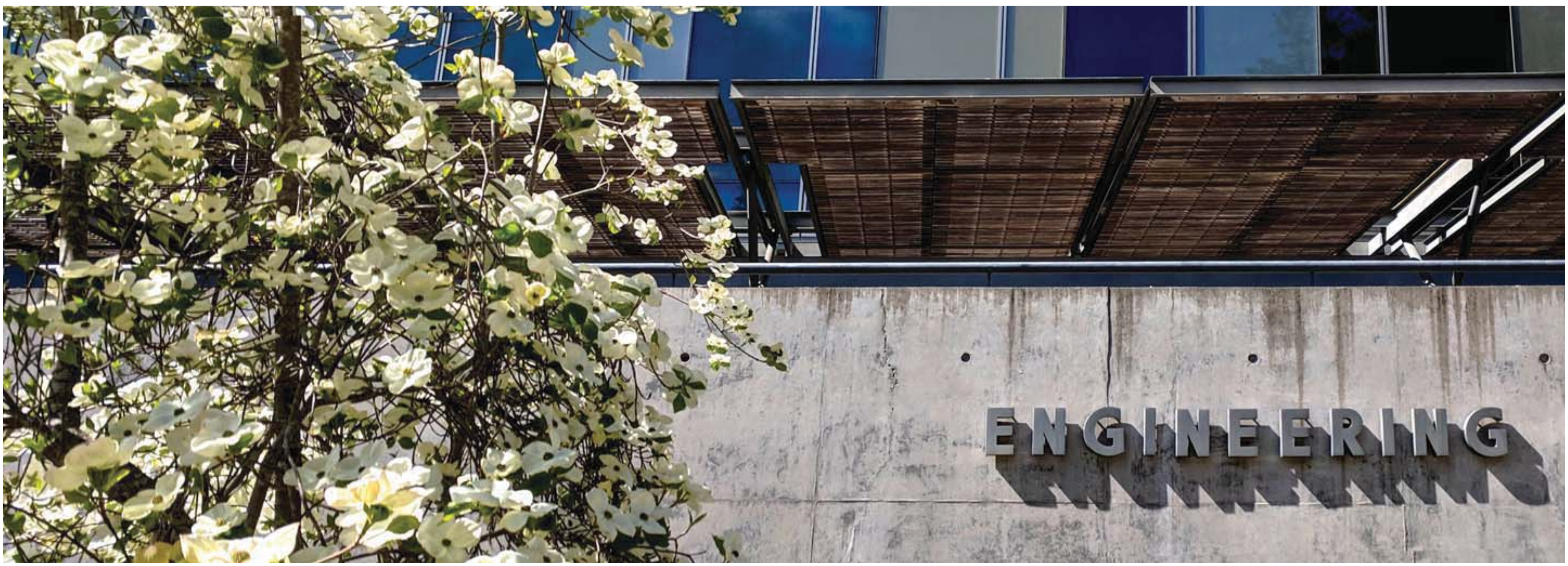
Hydroponic Automation

Low-Cost Whale Tracker

Passive Cleaner Boat Bot

SEADSConnect

UpVoice



Baskin School of Engineering Capstone Project

Hydroponic Automation

Barron Wong, Jaden King,
Eric Lery, Andrew Choi

UC SANTA CRUZ Baskin Engineering

Abstract

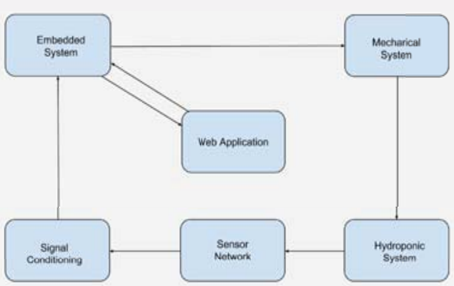
Although 70% of the earth's surface is covered in water, only 3% of that is freshwater. Traditional farming methods account for 80% of freshwater usage. As the population increases, efficient water usage becomes a necessity. Hydroponics is a method that can lead to 90% saving in the amount of water need for food production. However, hydroponics comes with an increase in cost in the form of labor and technical complexity. This project aims to automate some of these tasks, making it more feasible.

Objectives

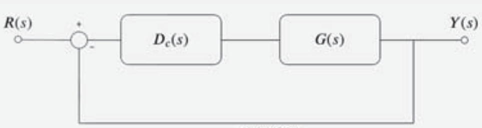
This system aims to control important aspects of the growing environment. Users will have an easy way to set and maintain parameters of the system. Results will be measured by being able to maintain the proper habitat for growing spinach.

- **Water Level**
  - User Defined Target ± 5%
- **Water Flow**
  - 1.5 Liters Per Minute ± 5%
- **pH**
  - 5.8 ± 5%
- **EC**
  - 1300µS ± 100µS
- **Alert System**
  - Notifies user of low supply levels
- **User Experience**
  - Hardware Interface
  - Web Application

System Overview



Feedback Control

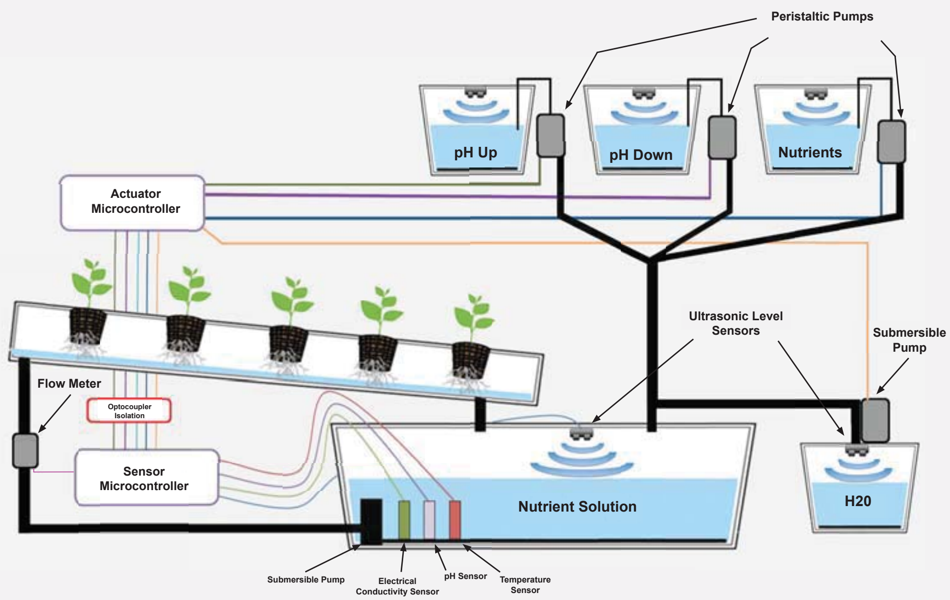


$$Y(s) = \frac{D_c(s)G(s)}{1 + D_c(s)G(s)}R(s)$$

$$E(s) = R(s) - Y(s)$$

$$E(\infty) = \lim_{s \to 0} sR(s) - Y(s) = 0$$

Mechanical Structure



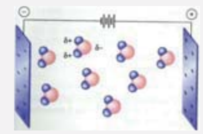
User Interface and Experience

- **Set target levels via a web app or the physical hardware**
- **Common plant requirements are pre-loaded, and custom levels can be saved as well.**

- **Track resource usage over time with online data logging**
- **Monitor systems conditions remotely from any internet connected computer or phone.**

Conductivity Sensor

Two electrodes with a voltage differential measures the resistance of water based on the space between them. An alternating current is used to prevent water polarization.



To read the conductivity, we need a circuit that:

- **Generates** an AC signal (LM555 + Bias)
- **Measures** the resistance across the probe (Reduced Wheatstone)
- **Rectifies** the output (Full Bridge Rectifier + Differential Amp)
- **Isolates** from the other circuits (Optocoupling + DC-DC transformer)

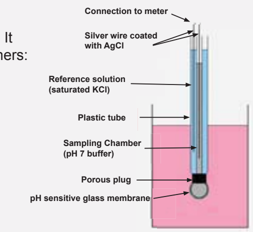
pH Sensor

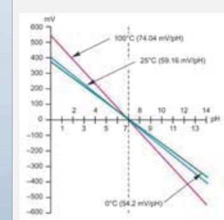
The pH probe is electrochemical cell. It has two electrodes in isolated containers:

- Reference Chamber
- Sampling Chamber

The electrodes are silver chloride, causing redox half-reactions:

$$Ag^+ + e^- \longleftrightarrow Ag(s)$$
$$AgCl(s)^+ + e^- \longleftrightarrow Ag(s) + Cl^-$$
$$AgCl(s) + Ag(s) + e^- \longleftrightarrow Ag(s) + Ag^+Cl^- + e^-$$





Within pH [0 - 13.5], the probe is a voltage source with the *Nernst Equation*:

$$V = E^0 - \frac{RT}{F}ln(a_{Cl^-})$$

Since pH [a(cl)] is logarithmic, the voltage is linear.

Signal conditioning for PSOC:

- **Biased** into the positive range
- **Low-pass** filtered
- **Buffered** to remove effects of external loading
- **Amplified** to scale from 0V to 5V.

Results

By controlling three key metrics: pH, nutrient level, and water flow, an optimal growing environment can be established for a range of plant types. The water flow is controlled to 5% of the target. Our pH remains within 5% of the setpoint and the nutrient level within 8%. This is achieved while consuming up to 90% less water than conventional growing methods. Sensor calibration is automated with a three point calibration for the pH and a one point calibration for the EC. Resource consumption data is recorded, giving users information that can be used to further tune growing conditions.

Acknowledgements

We would like to thank the following people for their assistance and support of this project:
Professor Max Dunne, Joseph Adamson, Russell Evans and the entire staff at BELS

Capstone Project

Low-Cost Whale Tracker

Kohl Grunt, Chris Hoblet, Mitchell Wright

Abstract

In addition to other marine animals, whales are commonly entangled in fishing nets as well as other ocean debris. Often times when agencies such as California Whale Rescue are notified of a trapped whale, by the time the agency is able to get to the last known coordinates, the whale is gone and not able to be assisted. Our motivation for our project stems from the need to create a network of boaters and hobbyists that can easily and inexpensively deploy a tracking device that will attach to the entangled lines of a trapped whale. California Whale Rescue will then be able to have a live flow of latitude and longitude coordinates which correspond to the whale's location.

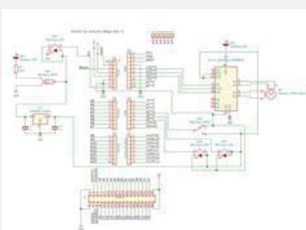
Approach

Our approach can be boiled down into two main sections. First, we have the electrical side, which is responsible for giving the capsule the ability to communicate with the outside world. Second, we have the mechanical side, which connects the capsule to the trailing lines of the whale.

Gripping Mechanism

Gripping Mechanism: A 1 D.O.F. gripper is a modified 4-bar linkage system that was designed in SolidWorks and 3D printed with SLA resin to provide a proof of concept for the design.

Motor Controller: An ATMega 2560 receives button input from the user to control the motion of the motor. A TFT display shows relevant information regarding the battery life of its two 18650 cells and motor position of the gripper.



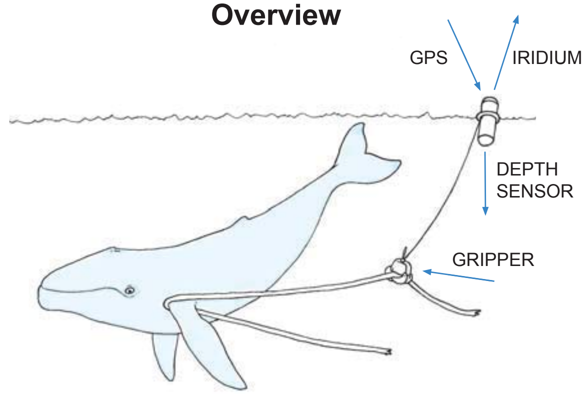
Motor Controller Wiring Diagram

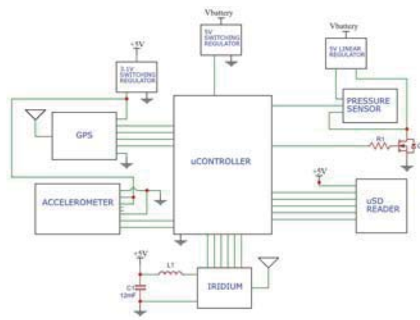
Motor Controller Prototype

Detachment Mechanism: The prototype detachment mechanism prototype was fabricated out of quick-disconnect fittings, an 8mm hex rod and socket, and a shifter cable. The housing is made of ABS plastic fittings.

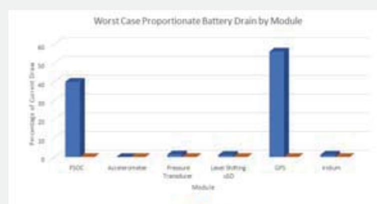
Linear Actuator: The prototype is driven by a four-start T8 lead screw with an 8mm pitch and 2mm lead was used in conjunction with a 1.7A 40Ncm Nema 17 stepper motor driven by a DRV8825 driver module. Four 18650 batteries provide 14.8V to the driver to supply 1..07A to the motor.

Overview





Power Distribution: The power is distributed using a small array of voltage regulators. The primary regulators were switching to allow for maximum efficiency of about 90%. The pressure transducer uses its own linear regulator for increased precision. The iridium uses its own bank of capacitors to accomodate large spikes in current. The GPS module contains its own backup battery to maintain its ephemeris data for up to 5 days.

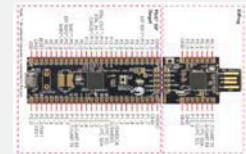


The figure above displays each module and the percentage of the battery that they will drain over the course of the journey.

Battery: This system is designed to use a set of 4 $LiSOCl_2$ C batteries. This will provide the system with 18 Ah, which would allow it to run at least 46 days. This chemistry was chosen for its energy density and low yearly capacity loss.

$$Batt_{life} = \left(\frac{Batt_{capacity} \cdot Batt_{efficiency} \cdot Reg_{efficiency}}{I_{inactive}+I_{active}}\right)/24_{hours}$$

uController: The microcontroller that we chose to use is the PSoC 5LP from Cypress Semiconductor. Not only is this a board that we were comfortable with, it provides great signal processing methods as well as decent low power current draw.

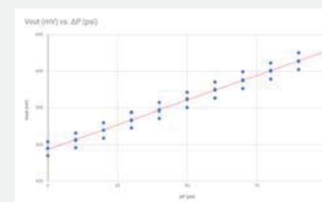


Above is a picture of the PSOC 5LP Development Board

Iridium: The Iridium module provides a link from the beacon back to the rescue crew. The GPS coordinates are sent to an constellation of satellites which then deliver the position information by email to the end users.

Above is a picture of the Iridium and its antenna

Pressure transducer: The pressure transducer module is crucial for determining the depth that the capsule is at. This information is not only useful for research purposes, but also for determining when to try and send and receive the capsules coordinates.



The figure above displays the ratio of the input pressure to the output voltage of the pressure transducer. Note the linearity and low sensitivity

GPS: The GPS module serves the very important of role trying to attain a fix given the depth that the capsule is at. This information is then stored and will be sent out as soon as possible.

Above is a GPS test where we drove south to Capitola. The areas where there are no data points are where we covered the GPS module in tin foil so that it couldn't get a fix.

LS uSD Card Breakout: This module provides the ability to log the data coming from the external sensors in a readable format. To be able to view the depth and coordinates of the whale with time stamps provides valuable data for researchers to sift through and make sense of it.

Special thanks for the support through this project go to our corporate sponsor, DOER, the BELS staff, Jose De Jesus Fuentes Grajales, Thomas J. Nolan, Prof. Max Dunne, and Joseph Adamson. An extra special thanks to our families and significant others for their unconditional love, support, and patience through the entirety of this project.

Passive Cleaner Boat Bot

Department of Electrical Engineering* and Department of Biomolecular Engineering**
Richard Alves*, Michael Choi*, Adam Grose*, Selina Guan*, Dan Li*, Rafael Ramirez*, Arina Romanova*, Ricardo Rodriguez**

Aquatic waste continues to harm the world's ecosystems. Current technology aims to slowly remove the trash already polluting the deep sea, but no plans exist for the new debris originating from river runoff. The PCBB team proposes a localized, autonomous boat and net system to passively guard coastlines and collect trash that flows from rivers and streams before it reaches the open ocean and decomposes into microscopic particles.

PCBB Vessel Assembly

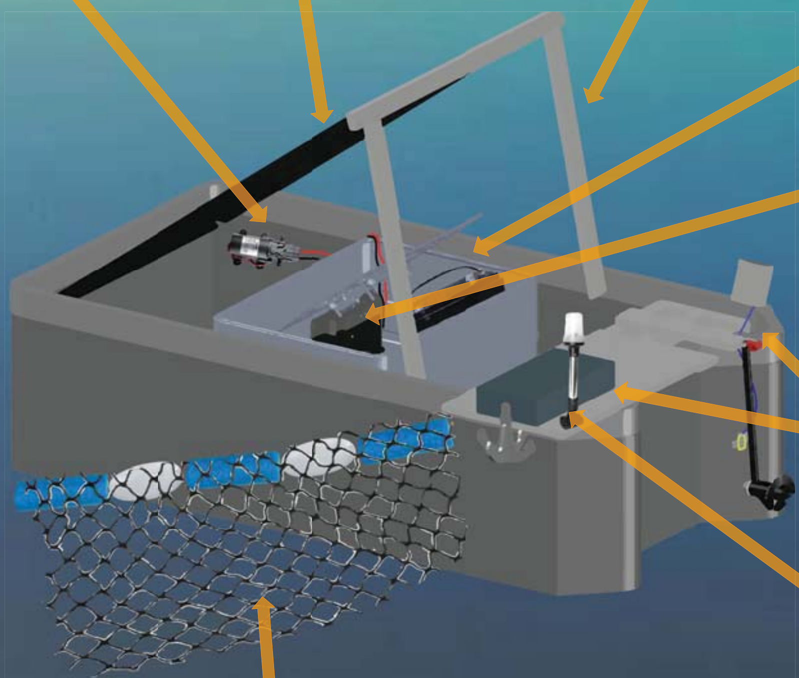
Water Pump
Removes excess water

Solar Panel
350W panel charges the battery

Frame
Wooden Frame used to hold solar panel in place

Battery
Powers the motors, lights, and autonomous system

Safety
Power system connected with fuses and breakers

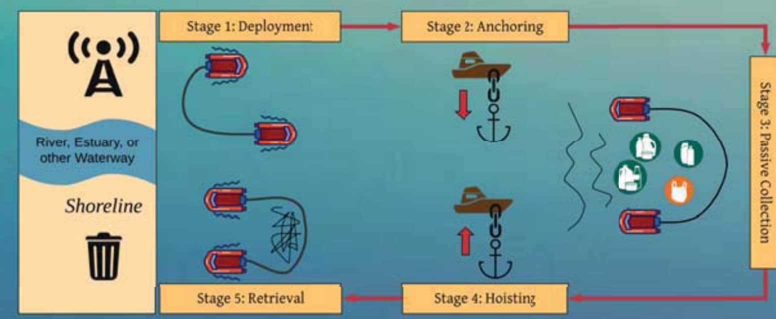


Motors
- Propels the vessels
- Lifts and drops the anchor

Beacon Light
- Follows navigation light requirements
- Indicates emergency state

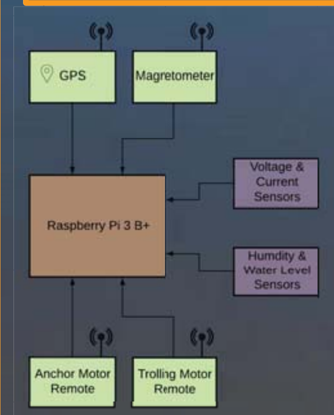
Net
Connected between two boats to catch macro-debris

PCBB Operation



PCBB vessels operate on a five stage cycle every time they are used for cleanup:
Stage 1: **Deployment** - the boat is sent out from the shore to its aquatic destination
Stage 2: **Anchoring** - after settling at its stop, the boats anchor themselves firmly
Stage 3: **Passive Collection** - currents move debris into the net until full
Stage 4: **Hoisting** - once done collecting, the boats release anchor, destined for shore
Stage 5: **Retrieval** - with trash ready for processing, the boats return to their hub

Autonomous Navigation and Self-Maintenance



Safety Sensors
- Humidity and water level sensors operate a pump and shut down the autonomous system if there is excess water onboard
- Voltage and current sensors monitor battery charge and power usage

Navigational Sensors
- GPS and Magnetometer for guidance
- Wireless remotes for the trolling and anchor motors for manual operation

Future Work

In future generations of PCBB, obstacle detection functionality will be added using technology such as LiDAR. A second boat will be built with a transceiver and receiver in each boat to communicate with each other and navigate the ocean as a combined pair. An improved waterproof frame will be constructed using metal instead of wood.

Acknowledgements: Tela Favaloro, Stephen Petersen, Keith Corzine, Marco Rolandi, Joan R Mccallum, Bill Alderson, ECE Department, Giving Day, Colleges Nine & Ten, Crown College, and Oakes College

Capstone Project

SEADSConnect

Ari Berkson, Austin Harmon, Qizhi Liu,
Priya Padmanaban, Jordan Hui

Abstract

SEADSConnect is a real-time energy usage tracking application. One of our goals is to display easy to read power consumption analytics by tracking power usage from a user's home through the use of a **SEADS** (Smart Energy Analytic Disaggregation System) device. Through SEADSConnect's user friendly interface, users can learn how much power individual appliances in their home use and when to run their appliances to better optimize points earned during an **"OhmHour."** By giving users simple tools to optimize their OhmHours, we can help make saving energy and money more convenient.

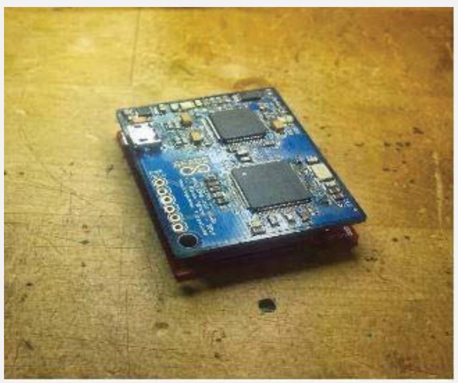


Figure 1. A picture of an early version of the SEADS panel device without its case on. This is just one of the types of SEADS devices in production

Acknowledgements

Richard Jullig | Pat Mantey (Instructors)
Max Dunn (OhmConnect Sponsor)
Dylan Rothfeld (Teaching Assistant)
Alec Reid | Olexiy Burov | Eric Cao (Grad Students)
Peter Wargo (Graphic Design)

Overview

A major problem OhmConnect users face is knowing how much power they consume during an OhmHour. Before SEADSConnect, users had to estimate which appliances would consume the most power and omit using them for the duration of their OhmHour to achieve their target power usage. If a user did not meet their target power consumption, the only way for users to learn how much energy they used is from a report generated by PG&E two days after the OhmHour. On the other hand, the **SEADS** (Smart Energy Analytic Disaggregation System) device faces the problem of lacking a centralized hub where users can easily read and digest the data it collects from a user's fuse box. SEADSConnect provides a solution to these two problems by connecting the two platforms together in a web app that provides easy to read real-time power consumption analytics, real-time alerts when users consume power that will lead to exceeding their target power usage during OhmHours, and user specific tips that can help reduce their power bill.

Method

1. Users connect a SEADS device into a fuse box within their house.
2. Users can review and monitor energy usage in real-time.
3. SEADSConnect downloads information such as the time of an upcoming OhmHour for a user and user specific power consumption goals from OhmConnect.
4. Users evaluate data from SEADSConnect to choose when appliances can be switched on or off to optimize points earned from OhmConnect.

OhmConnect

OhmConnect is a service that works with various power companies in the U.S. and Canada. Their goal is to have users reduce their energy consumption when the grid is stressed. By having users reduce energy when the grid is stressed, OhmConnect can help reduce the need for utility companies to start up expensive peaker power plants. To get users to reduce, OhmConnect assigns "**OhmHours**" to users. When users reduce their energy during these hours, they will earn points based on various factors like how much they reduced, their power consumption history, and their community's energy usage. Points can be used to get cash back and other prizes.

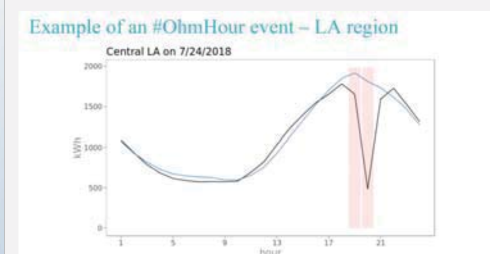


Figure 2. Black line is actual usage, and blue line is estimate usage. Red bars are the time of OhmHour

Conclusion

At the current state of this project, SEADSConnect has the ability to track the real time energy usage of a home with a SEADS device installed. SEADSConnect also allows its users to monitor the energy consumption of one appliance compared to the rest of their home. However, SEADSConnect does not have full OhmConnect integration because the test API provided by OhmConnect is limited to one specific user.

The next steps for SEADSConnect would include full integration of OhmConnect for all their users, along with displaying time of use energy pricing to show the benefits of reducing power during peak hours of the day.

UpVoice

https://upvoice.app

Dhillon, Bobby
Dumitrean, Marius-Paul
Fronteras, Sean
Manseau, Ryan
Pham, Thomas

Special Thanks
Jullig, Richard
Leung, Joshua
Shadmon, Roy
Rothfeld, Dylan

Abstract

Do you want to get paid to participate in assemblies? UpVoice is a mobile platform that encourages participation in public assemblies by giving participants cash.

Examples of assemblies include:

- Protests
- Business Openings
- Conferences

UpVoice allows any person in the US to: *create a cause* for a gathering with a time and place, *pool money* into the gathering via crowdfunding, and *receive money* for taking pictures of the assembly.

Approach

Android & iOS Mobile Applications

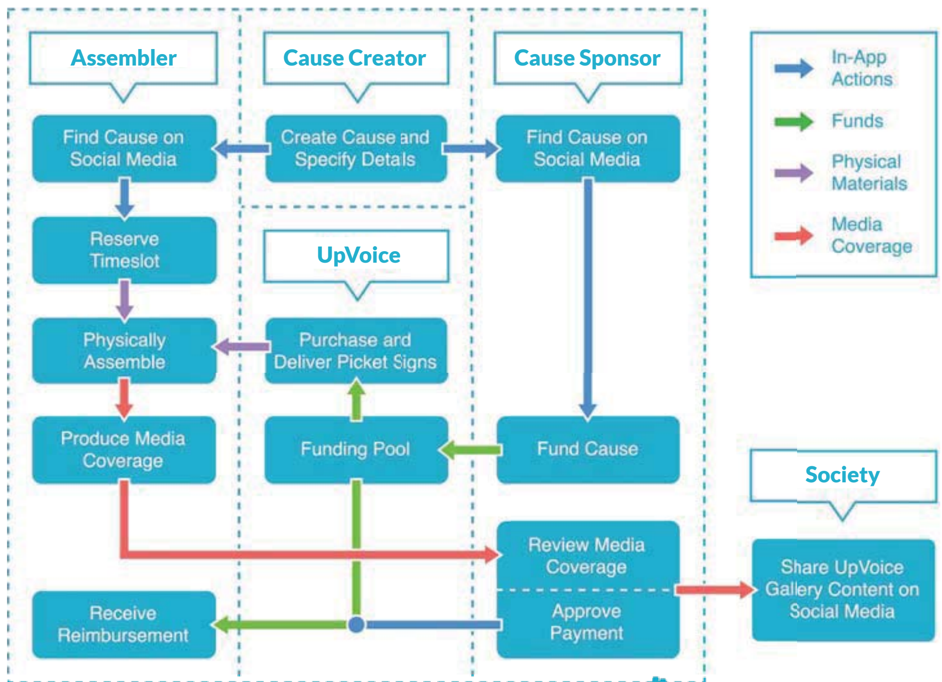
- Allow anyone to create an assembly.
- Easily find all assemblies near you.
- Simplifies the consolidation of event photo media from assemblers.

Web Application

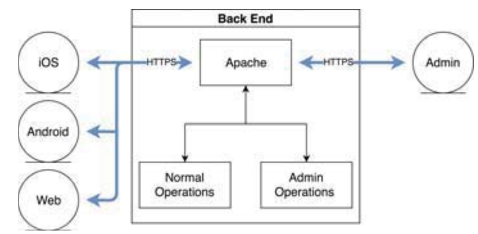
- Allow photos from events to be easily reviewed and shared on social media.
- Allow funding page for causes to be easily shared.

Overview

Outlined below is a per-role overview of our platform, including interactions between roles. The main actors in our platform are *cause creators* who initiate the process, *cause sponsors* who fund the assemblers to generate media, *assemblers* who show up to events, *UpVoice* which organizes and facilitates all of the interactions, and *Society* which can view and share assembly media from a centralized location.



Architecture



Backend	LAMP Stack
Android	Native Java
iOS	Native Swift
WebApp	Vue.js

The user applications are simple clients, wrapping RESTful API calls to interact with a SQL based LAMP backend.

Conclusion

As an idea, UpVoice has generated mixed reactions, especially in regards to protesting. Many expressed interest in our platform's ability to support individuals involved in social activism, yet others think the project is immoral for providing a platform that could be used to promote viewpoints they detest.

We are optimistic that these strong reactions indicate real potential for UpVoice to achieve its goal: Promoting necessary discourse on issues that are often left unaddressed by society.

Contact us

Patrick E. Mantey
Professor of Electrical & Computer Engineering
Director, Senior Design Capstone
Director, Information Technologies Institute
mantey@ucsc.edu

Frank Howley
Senior Director of Corporate Development
fhowley@ucsc.edu

Baskin Engineering
UC SANTA CRUZ

Join us on Facebook
facebook.com/BaskinSchoolofEngineering

Visit our website
csspp.soe.ucsc.edu