Modern Product Engineering

2020

How “smart” products are created with sensors, algorithms and circuits

Jake Garrison

About Me

Website: jakegarrison.me

Education

Focus: *Signal Processing and Machine Learning*

- **M.S. Electrical and Computer Engineering**, University of Washington, *2018*
- **B.S. Electrical Engineering**, University of Washington, *2016*

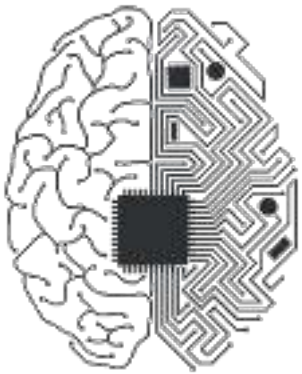
Research / Job History

- **Google Health,** Research and Software Engineering, *2018-present*
- **UbiComp Lab,** Mobile health researcher advised by Professor Patel, *2016-2018*
- **Haiku Deck,** Software developer intern for Zuru AI feature, 2016
- **Puppy.ai App,** Founder and developer of iOS app, *2015-2018*
- **General Motors EcoCar 3,** Autonomous Driving Lead, *2015 - 2017*
- **Department of Transportation,** Seattle parking load predictive modeling, *2016*
- **Tesla Motors**, Sensor Integration Intern, Autopilot and Model-X Doors, *2015*
- **Tesla Motors,** Power Electronics Intern, Supercharging and Insane Mode, *2014*
- **General Motors EcoCar 3,** Electrification Lead, *2013 - 2015*
- **Verellen Amplifiers,** Built custom analog audio gear, *2012*
- **Electric GTI Conversion**, highschool hobby project, *2010*

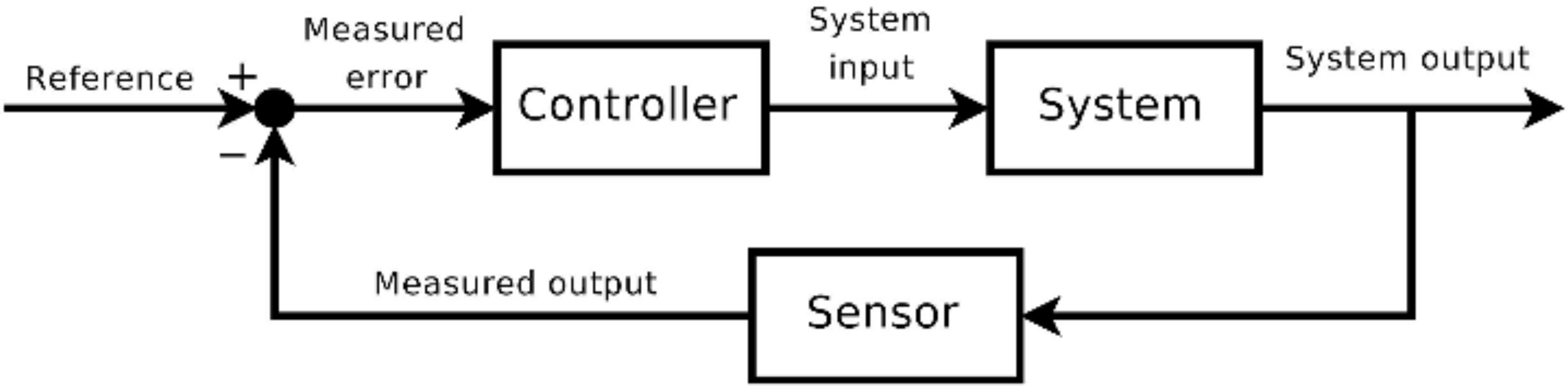
“Smart” Products

Buzzwords: smart, intelligent, AI, autonomous, robotic

Control, Sensing and Processing



Automatic Control Systems

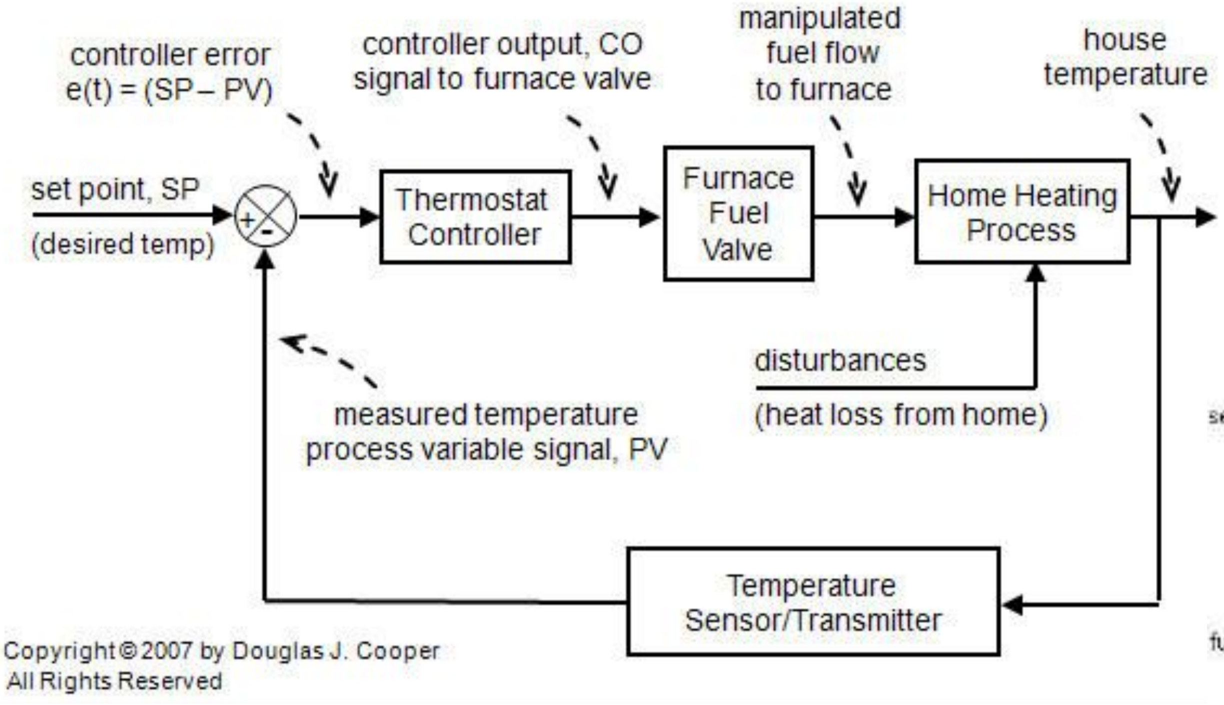


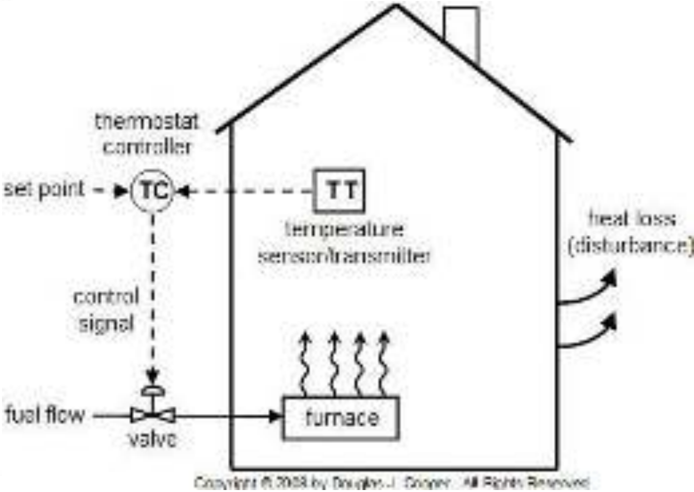
Control

Thermostat Example

Objectives:

(1) Keep temperature comfortable to occupants while *(2)* minimizing energy waste.





Control Terminology

Components

- **Final control element (FCE):** Actuator being controlled by (furnace switch)
- **Sensing element**: Device measuring current state of FCE (thermistor)
- **Processing element**: Circuit or chip responsible for creating control instruction (CO) given sensor inputs (thermostat controller)

Variables

- **PV**: Process variable (house temperature)
- **CO**: Controller output signal to FCE (thermostat instruction to furnace valve)
- **SP**: Set point (temperature set on the thermostat)
- **D**: Heat loss disturbances from changing environment (outside temp)

Closed Loop (feedback, automatic) Control

Control action from the controller is dependent on feedback from the process in the form of the value of the process variable (PV)

Automatic Controller Objectives

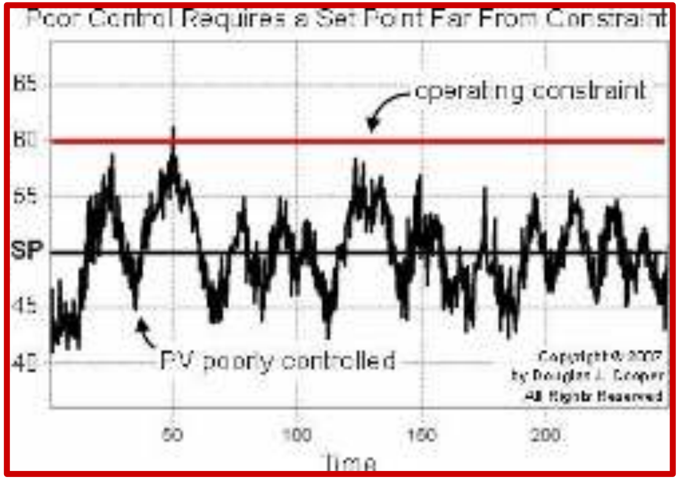
Maintain SP despite external disturbances

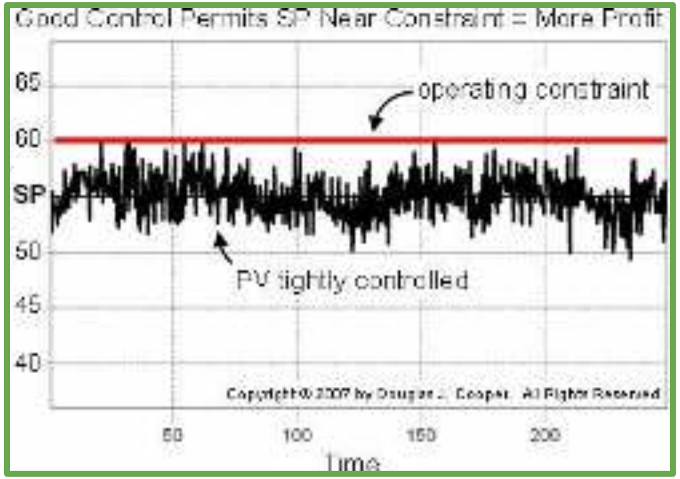
Minimize system loss: cost, energy, waste production, environmental impact, time spent

Minimize system variability: Make system more deterministic, safer.

We want: PV = SP

Keep the measured process variable (PV) at the set point value (SP) in spite of unmeasured disturbances (D) by modulating the final control element (FCE) via controller output (CO) signal.





Example Control Algorithm

Error or loss, the variable to minimize via control:

$$e(t) = SP - PV$$ ***(error = set point – measured process variable)***

- is the measured temp colder than set point ($SP - PV > 0$)? Then turn on furnace.
- is the measured temp hotter than set point ($SP - PV < 0$)? Then turn off furnace.
- What about when set point is hit ($PV = SP$, $e(t) = 0$)? Depends on algorithm complexity.

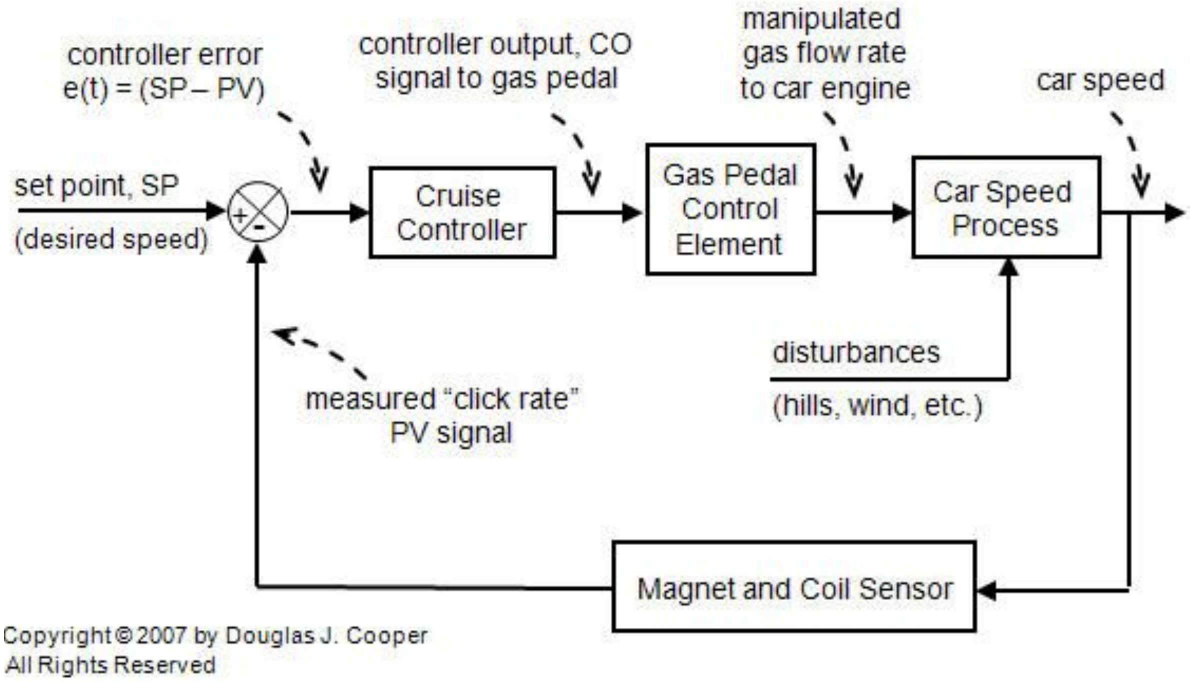
Control Considerations

- the direction to move FCE
- how far (magnitude) to move it at this moment
- how long to wait before moving it again
- whether there should be a delay between measurement and action

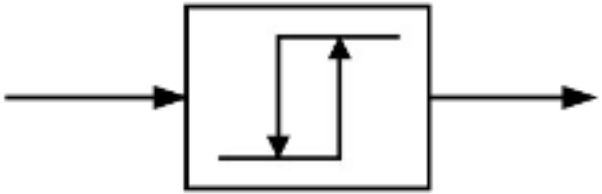
Cruise Control Example

Objectives:

(1) Keep speed of vehicle speed constant with (2) smooth braking and acceleration control.



Bang-Bang Control



Pros:

- Intuitive and easy to implement, two thresholds needed

Cons:

- Oscillates around set point, rarely resulting in 0 error
- Only works with binary switching On/Off

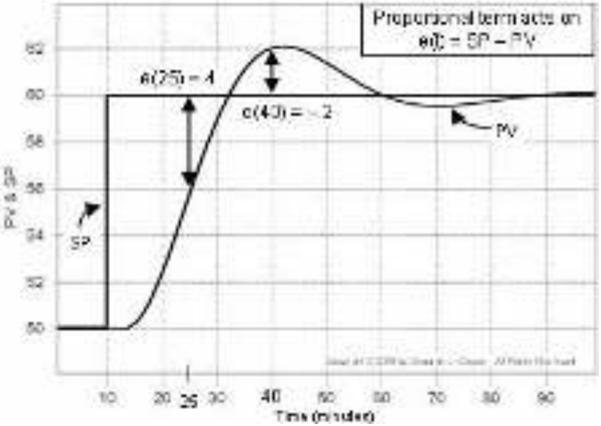
Algorithm:

If PV > PVmax: toggle Off
If PV < PVmin: toggle On
Otherwise: do nothing

Where:

- *PV = measured process variable*
- *PVmin = lower threshold, tuned*
- *PVmax = upper threshold, tuned*
- *SP = set point*

Proportional Control (P)



Pros:

- Easy to tune (single parameter)
- Easy to implement (no memory needed)

Cons:

- Constant “offset” when environment slowly deviates from expectation

Algorithm:

$$CO = CO_{bias} + \boxed{Kc \cdot e(t)}$$

Purpose of CObias? (think cruise control)

Purpose of Kc? (think response delay)

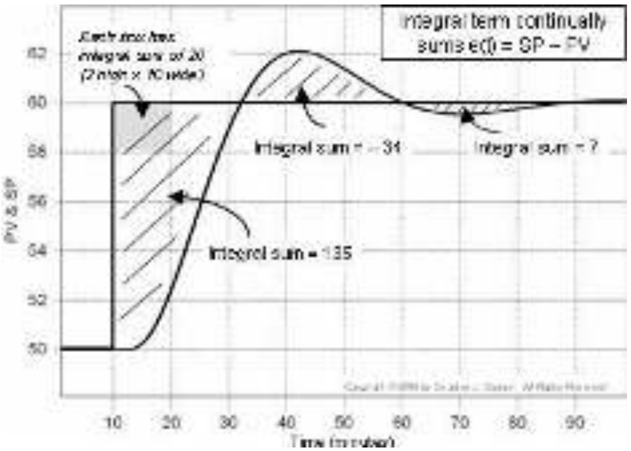
Where:

- *CObias = controller bias*
- *Kc = controller gain, tuned*
- *e(t) = controller error = SP – PV*
- *SP = set point*
- *PV = measured process variable*

Proportional-Integral Control (PI)

Pros:

- Balance between capability and complexity
- No “offset” issue thanks to integral term acting as history driven COBias autotuner



Cons:

- Two interacting parameters are harder to tune
- Poor response to disturbances (abrupt changes)

Algorithm:

$$CO = CO_{bias} + \boxed{Kc \cdot e(t)} + \boxed{\frac{Kc}{Ti} \int e(t)dt}$$

Purpose of integral term? (think error moving average)

Where:

- *CO = controller output signal*
- *CObias = controller bias*
- *e(t) = current controller error*
- *SP = set point*
- *PV = measured process variable*
- *Kc = controller gain, tuned*
- *Ti = reset time, tuned*

Proportional-Integral-Derivative Control (PID)

Pros:

- Better stability, can be used for controlling both fast and slow process variables

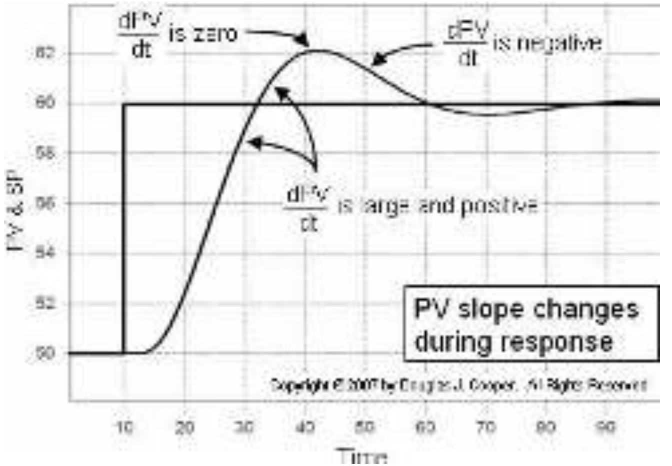
Cons:

- Many ways of configuring, can be overkill
- Harder to predict behavior
- Derivative can be unstable, requires memory

Algorithm:

$$CO = CO_{bias} + \boxed{Kc \cdot e(t)} + \boxed{\frac{Kc}{Ti}\int e(t)dt} + \boxed{Kc \cdot Td\frac{de(t)}{dt}}$$

Could also have PD which removes the integral term



Where:

- *CO = controller output signal*
- *CObias = controller bias*
- *e(t) = current controller error*
- *SP = set point*
- *PV = measured process variable*
- *Kc = controller gain, tuned*
- *Ti = reset time, tuned*
- *Td = derivative time, tuned*

When to use I or D?

Problems with D

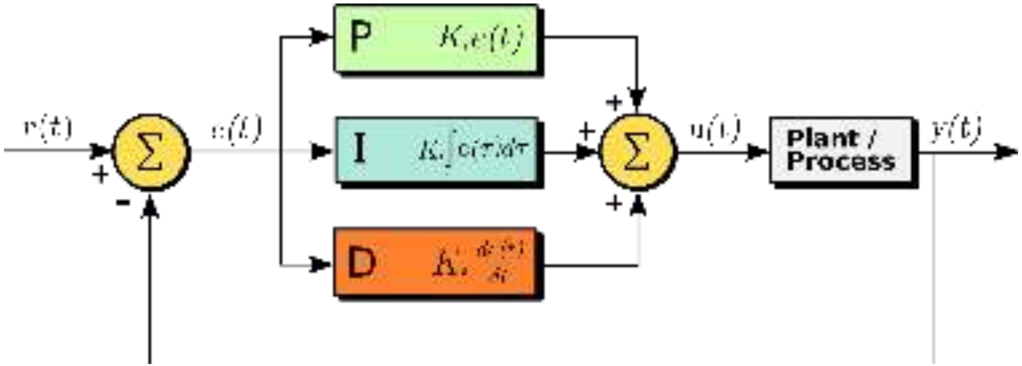
- Fast response to abrupt changes and disturbances (“kick”)
- High frequency noise entering in the system is highly amplified.

Problems with I

- Slow response to abrupt changes and disturbances (“windup”)

Which is ideal for:

- Thermostat?
- Cruise control?



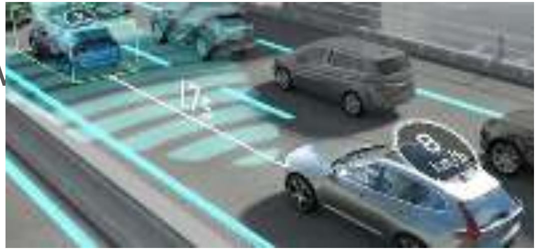
Smarter Industry Examples

Nest "Learning" Thermostat

- Source external information via wifi (local weather and grid load, user GPS)
- Learn from manual override (customer is always right)
- Learn users schedule (home vs away)
- Thermal modeling to improve temperature sensing

"Adaptive" Cruise Control

- Add additional proximity sensors to determine distance to nearby vehicles
- Adjust speed to maintain cruise setpoint, but also stay w safe proximity of nearby vehicles
- Predict live traffic and road conditions



Data Driven Controller Tuning

If representative sensor data exists with ground truth of how the controller should have responded (supervised learning), then data driven techniques can be employed to improve control!

Examples:

- Tuning with parameter search (grid, random, evolutionary, gradient descent)
- Physical modeling (thermal, propulsion)
- Statistical modeling (bayesian, probabilistic, Hidden Markov)
- Non-linear machine learning (neural networks, ensembles, random forests)

Sensing

Temperature Sensing

Thermistor: Exhibits a large, predictable, and precise change in resistance correlated to variations in temperature. As temperature increases results in resistance drops.

- Exponential rather than linear response, requires linearization (simpler than a lookup table)
- Accuracy is high from 0.05 to 1.5 °C and fast response time, operating range is -50 to 200 °C

Thermocouple: Consists of two wires of different metals connected at two points. The varying voltage between these two points reflects changes in temperature. As temperature goes up, the output voltage of the thermocouple rises.

- Thermocouples are nonlinear and often require a lookup table
- Accuracy is low, from 0.5 °C to 5 °C, medium response time, widest operating range, from -200 °C to **1750** °C.

Semiconductor: Found in side of ICs, typically two diodes with temperature sensitive voltage and current characteristics

- Linear response, so readings don't require post-processing
- Lowest accuracy, from 1 °C to 5 °C and slowest response (5s to a minute), and Narrowest temperature range, from -70 °C to 150 °C

Infrared Imaging: Infrared energy is emitted from an object as thermal radiation and is proportional to its temperature

- Non-contact and long range which is essential for some applications, but 100x more expensive
- Non point source and therefore less accurate compared to above contact based sensors

Temperature Sensing Engineering Problem

What can go wrong when a temperature sensor is packed into a “smart” consumer electronics product (like Nest Thermostat)?

Autonomous Vehicles

Vehicles have been getting more “autonomous” since the 40s...

Windshield wipers 1944

Chrysler Windsor Dartline 1958

Autonomous Vehicles Progress

(1900s) Level 0: Driver only: the human driver controls everything independently; *steering, throttle, brakes, etc*

(1970s) Level 1: Driver Assistance: Basic automatic control; *cruise control, automatic transmission*

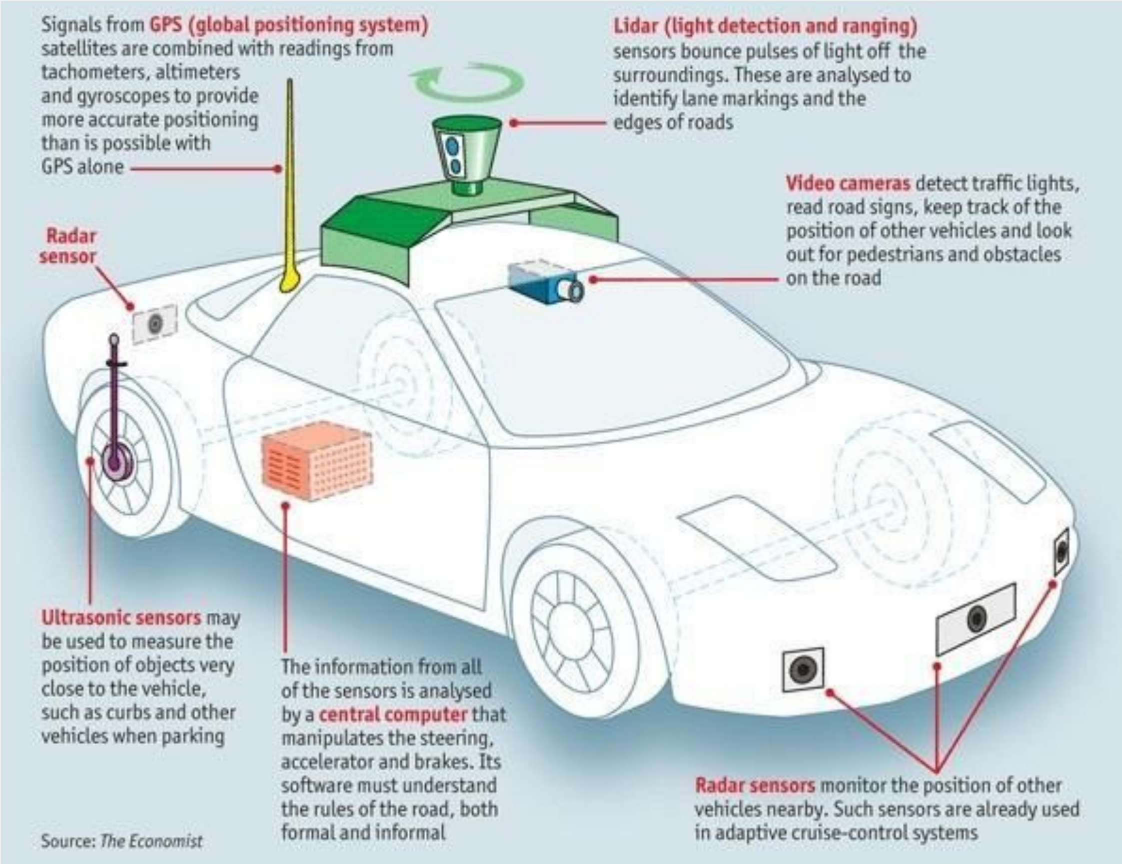
(2000s) Level 2: Partial automation: Operator must monitor at all times. At least one fully automated system; *adaptive cruise control, auto braking or lane centering*

(possible in 2010s, not legal) Level 3: Conditional automation: The operator monitors the system and can intervene. Safety-critical functions can be automated

(2020s fully legal?) Level 4: High automation: There is no monitoring required for an entire trip. Not all trips supported, driver action could be necessary.

(2030s?) Level 5: Full automation: operator-free driving.

Autopilot Perception Sensor Overview



Perception Sensors

RADAR

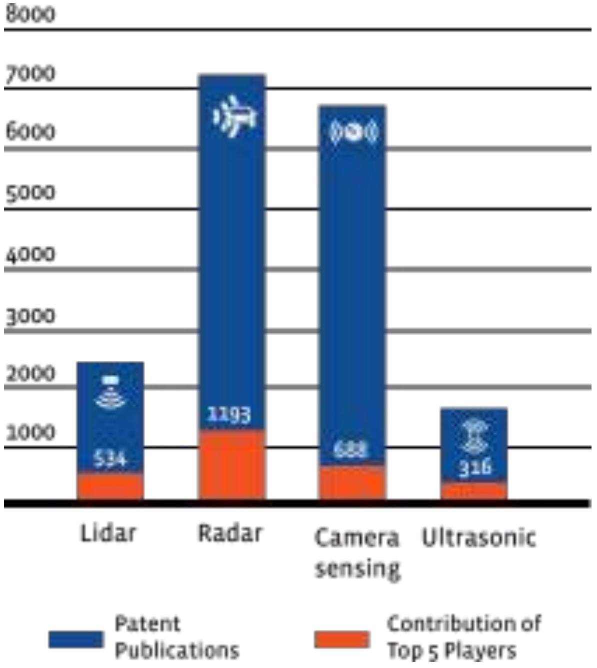
Camera (360)

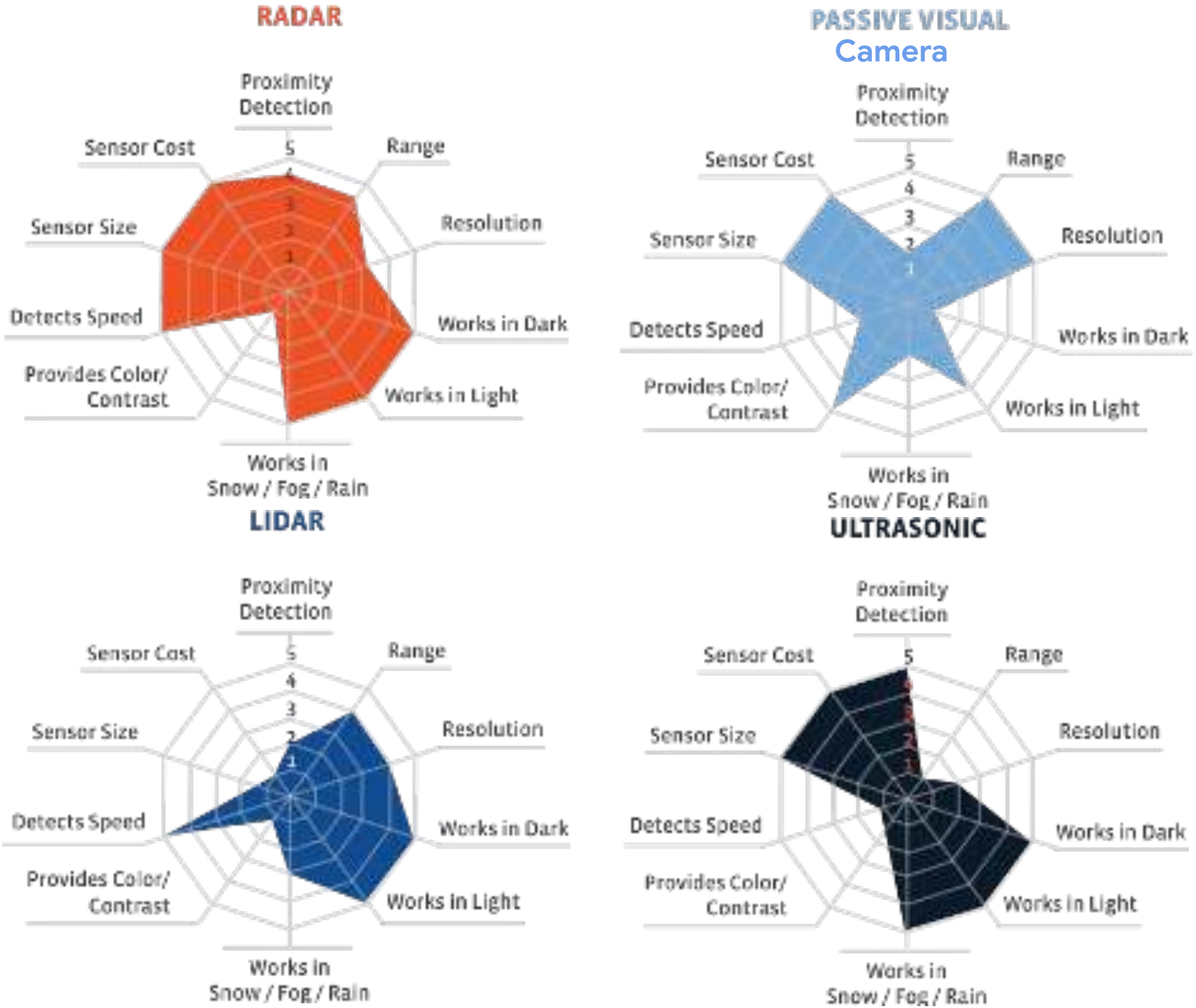
GPS

LIDAR

ULTRASONIC

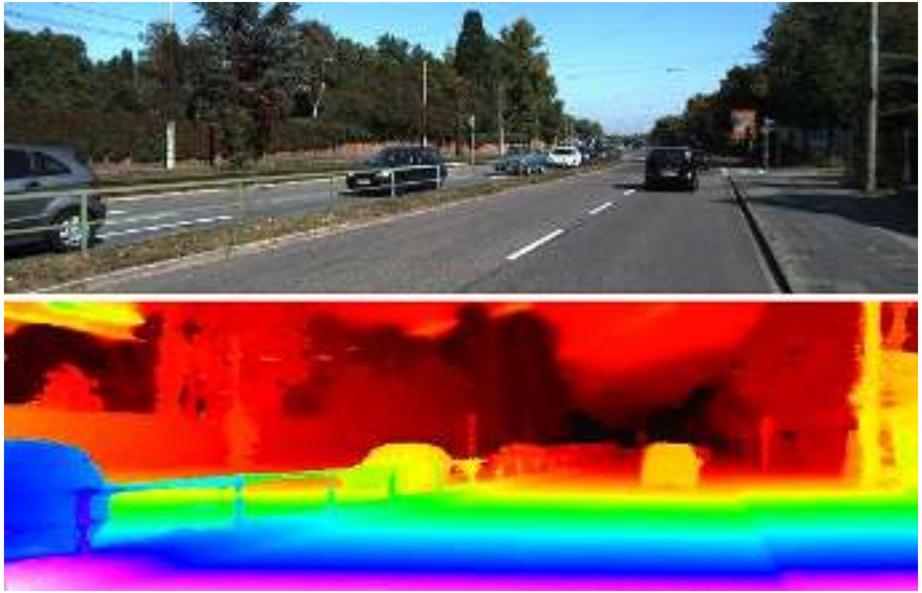
Perception Specs



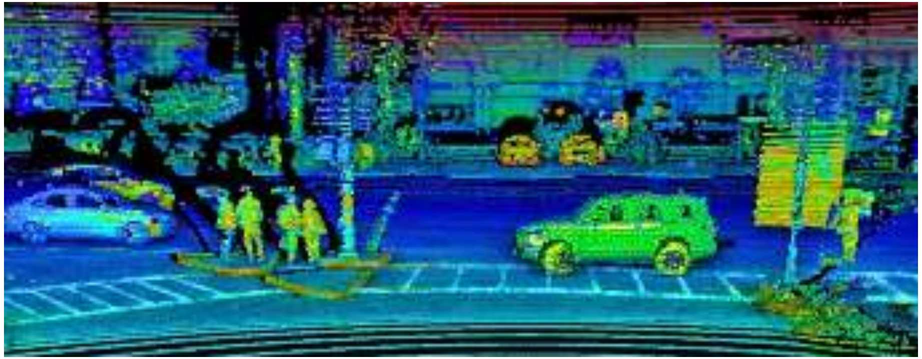


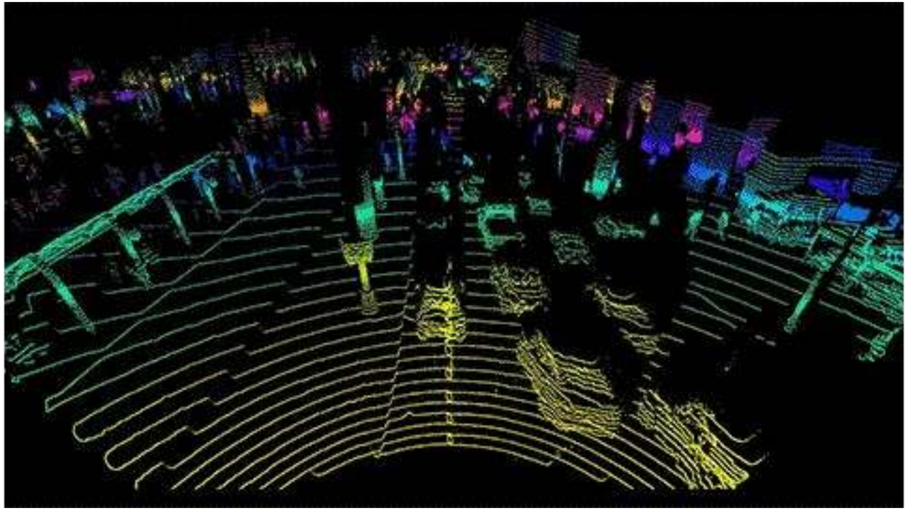
Depth Perception Task

Stereo Cameras



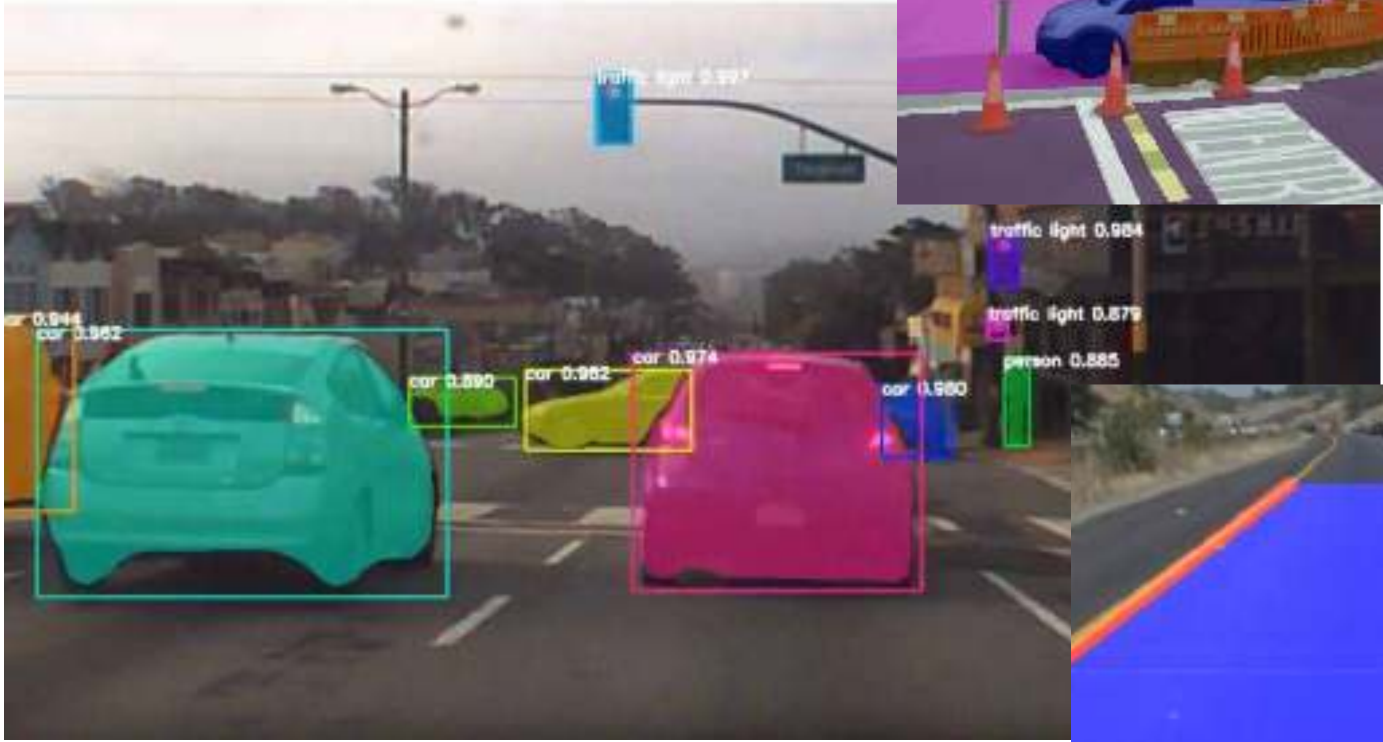
Radar / Lidar

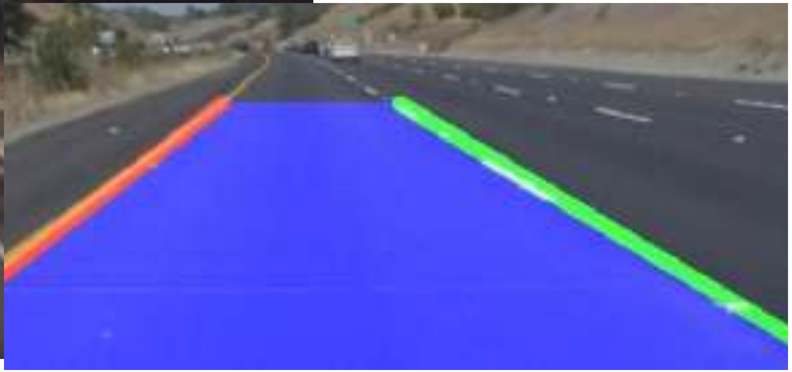




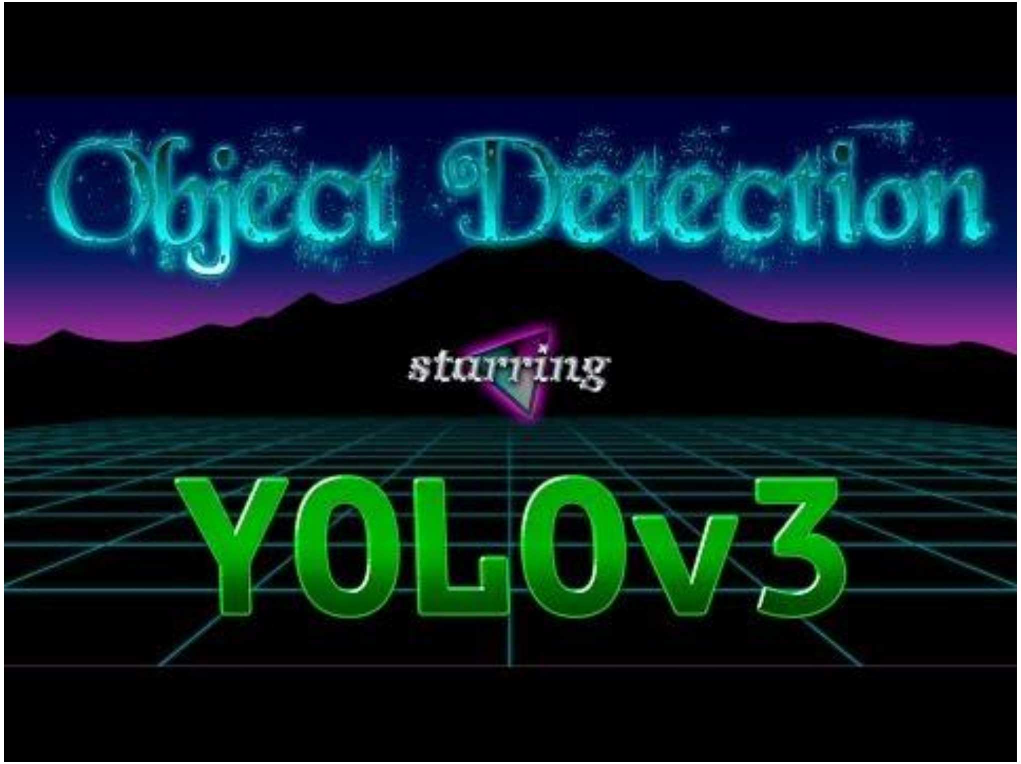
Object Detection Task

Typically cameras





Deep Learning for Camera Object Detection



OR

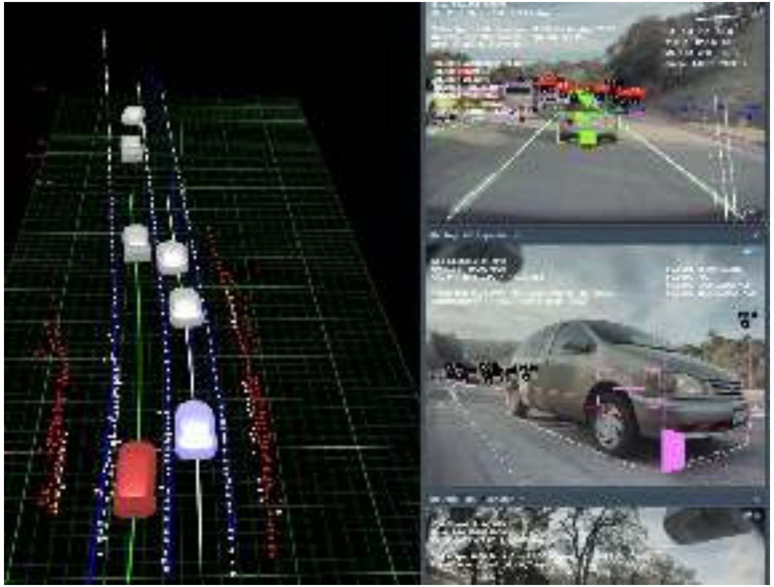
OR

OR...

OR

Tesla Autopilot

Sensor Fusion



Model X Falcon Wing Door Sensors

Sensors

- Ultrasonic
- Capacitive
- Inductive

Features

- Wall / person / obstruction avoidance
- Adaptive path planning algorithm
- Safety Overrides

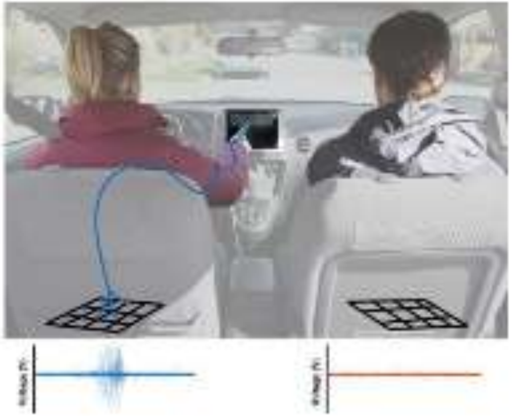
Carpacio: Repurposing Capacitive Sensors to Distinguish Driver and Passenger Touches on In-Vehicle Screens

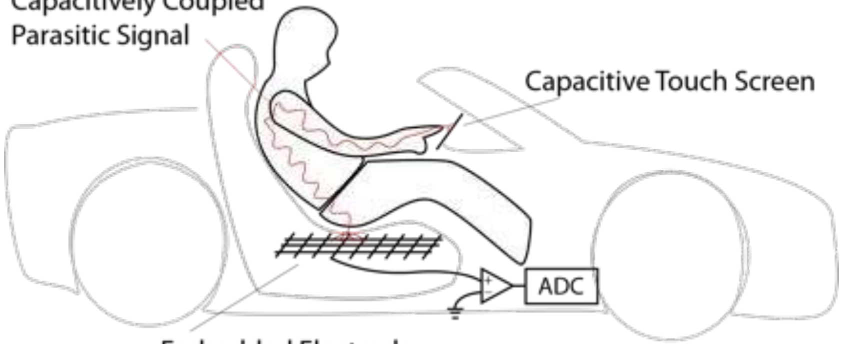
Edward Jay Wang[1,*], Jake Garrison[1,*], Eric Whitmire[2], Mayank Goel[3], Shwetak Patel[1,2]

Department of Electrical Engineering[1]
University of Washington
Seattle, WA
{ejaywang, omonoid}@uw.edu

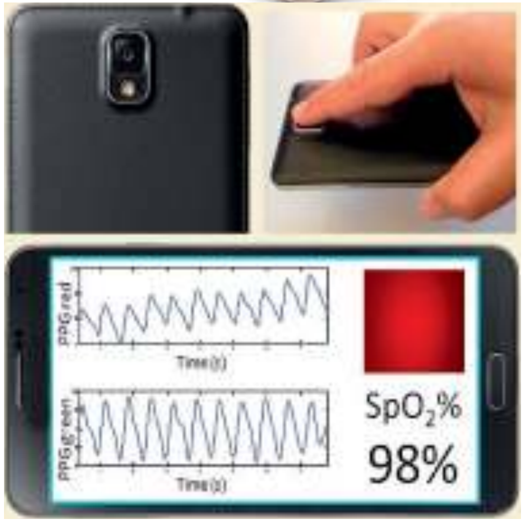
Paul G. Allen School of Computer Science and Engineering[2]
University of Washington
Seattle, WA
{emwhit, shwetak}@cs.washington.edu

School of Computer Science[3]
Carnegie Mellon University
Pittsburg, PA
mayankgoel@cmu.edu





Mobile Health Sensing



Processing

Processor Hardware

GPU

Microprocessors / FPGA

ASIC

Millimeter scale

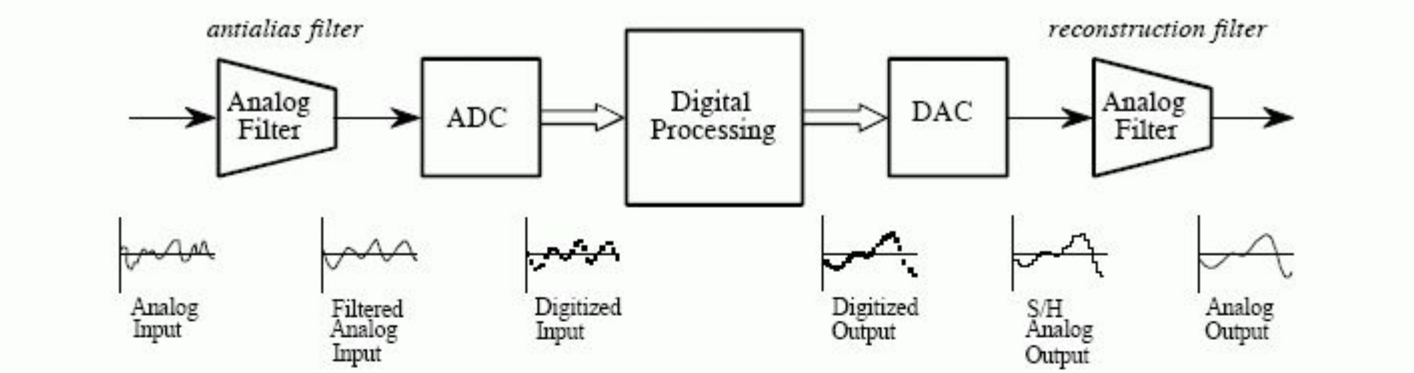
DSP

CPU

TPU

Signal Processing

- Input analog signal sensing: *microphone, antenna, transducer*
- Analog processing: *amplification, filtering, mixing*
- ADC (analog to digital converter)
- Digital signal processing, DSP: *resampling, filtering frequency and time modulation, compression, detection inference*
- DAC (digital to analog converter)
- Output analog signal: *speaker, encoded file, packet, output instruction*



Signal Processing is Everywhere

Questions / Discussion

- Signal Processing
  - Analog circuits
  - Digital signal processing + Machine learning
  - Hardware processors
- Control Algorithms
- Smart Home
  - Thermostats
  - Voice assistants
- Mobile Health Sensing
- Automotive
  - Electric vehicles
  - Autonomous vehicles
- Industry
  - Tech Startups (puppy.ai, Haiku Deck, Cryptocurrency, getting acquired)
  - Tech companies (Google, Tesla, Nest, General motors)
  - Interviewing tips

Thanks!

Website: www.jakegarrison.me

Contact: jakehgarrison@gmail.com

Sources

https://controlguru.com/the-components-of-a-control-loop/

https://www.ametherm.com/blog/thermistors/temperature-sensor-types

https://www.tesla.com/autopilotAI

https://www.unitedlex.com/news/from-google-to-tesla-its-a-war-of-lidar-or-radar