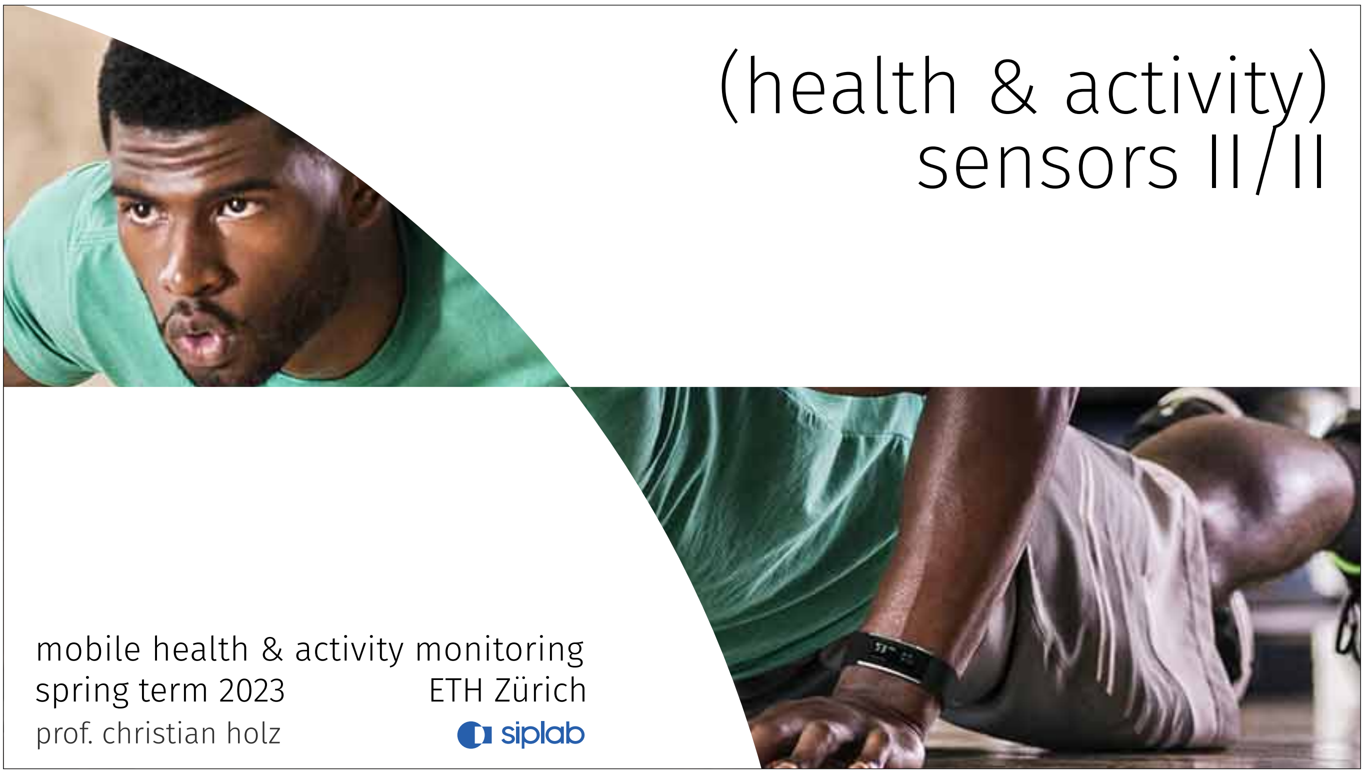


(health & activity) sensors II/II

mobile health & activity monitoring
spring term 2023 ETH Zürich
prof. christian holz
siplab

logistics

exam

end-of-term exam (Semesterendprüfung)

- presence exam, computer based
- 90 minutes

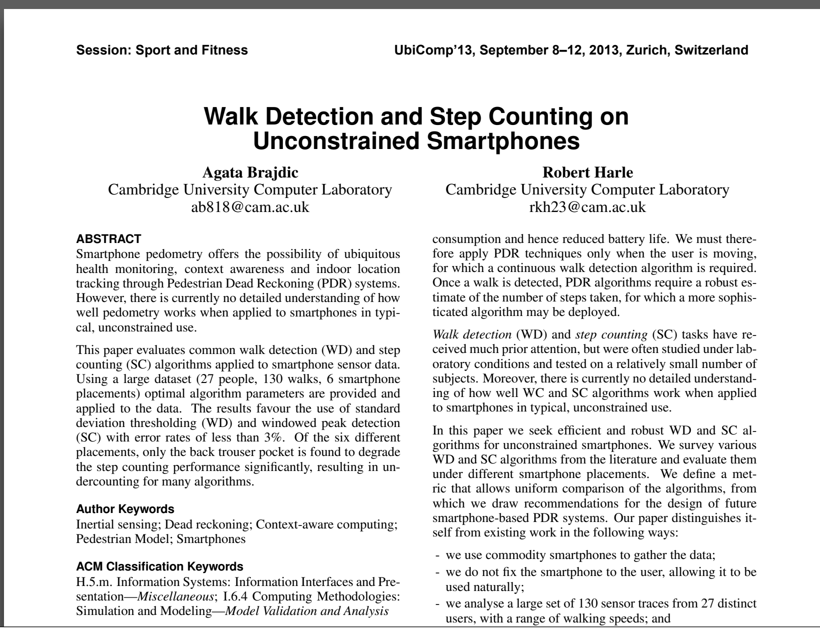
➩ counts 50% of the grade

we have a slot now

- June 12, 2–4pm
- HG G5 (pending final confirmation)
- distance exams ➩ schedule with exams office (or talk to me if it's tricky)

reading assignments

Reading Assignment 1



Session: Sport and Fitness UbiComp'13, September 8–12, 2013, Zurich, Switzerland

Walk Detection and Step Counting on Unconstrained Smartphones

Agata Brajdic
Cambridge University Computer Laboratory
ab818@cam.ac.uk

Robert Harle
Cambridge University Computer Laboratory
rkh23@cam.ac.uk

ABSTRACT
Smartphone pedometry offers the possibility of ubiquitous health monitoring, context awareness and indoor location tracking through Pedestrian Dead Reckoning (PDR) systems. However, there is currently no detailed understanding of how well pedometry works when applied to smartphones in typical, unconstrained use.

This paper evaluates common walk detection (WD) and step counting (SC) algorithms applied to smartphone sensor data. Using a large dataset (27 people, 130 walks, 6 smartphone placements) optimal algorithm parameters are provided and applied to the data. The results favour the use of standard deviation thresholding (WD) and windowed peak detection (SC) with error rates of less than 3%. Of the six different placements, only the back trouser pocket is found to degrade the step counting performance significantly, resulting in undercounting for many algorithms.

Author Keywords
Inertial sensing; Dead reckoning; Context-aware computing; Pedestrian Model; Smartphones

ACM Classification Keywords
H.5.m. Information Systems: Information Interfaces and Presentation—*Miscellaneous*; I.6.4 Computing Methodologies: Simulation and Modeling—*Model Validation and Analysis*

consumption and hence reduced battery life. We must therefore apply PDR techniques only when the user is moving, for which a continuous walk detection algorithm is required. Once a walk is detected, PDR algorithms require a robust estimate of the number of steps taken, for which a more sophisticated algorithm may be deployed.

Walk detection (WD) and *step counting* (SC) tasks have received much prior attention, but were often studied under laboratory conditions and tested on a relatively small number of subjects. Moreover, there is currently no detailed understanding of how well WC and SC algorithms work when applied to smartphones in typical, unconstrained use.

In this paper we seek efficient and robust WD and SC algorithms for unconstrained smartphones. We survey various WD and SC algorithms from the literature and evaluate them under different smartphone placements. We define a metric that allows uniform comparison of the algorithms, from which we draw recommendations for the design of future smartphone-based PDR systems. Our paper distinguishes itself from existing work in the following ways:

- we use commodity smartphones to gather the data;
- we do not fix the smartphone to the user, allowing it to be used naturally;
- we analyse a large set of 130 sensor traces from 27 distinct users, with a range of walking speeds; and

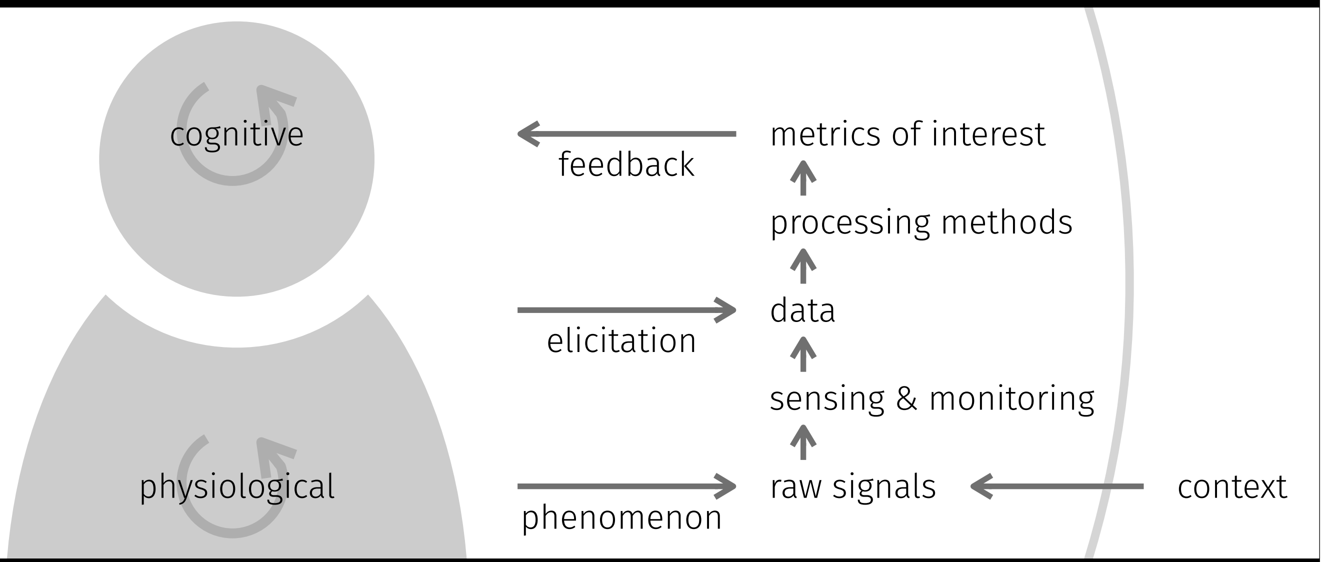
Reading Assignment 1

great intro to step counting

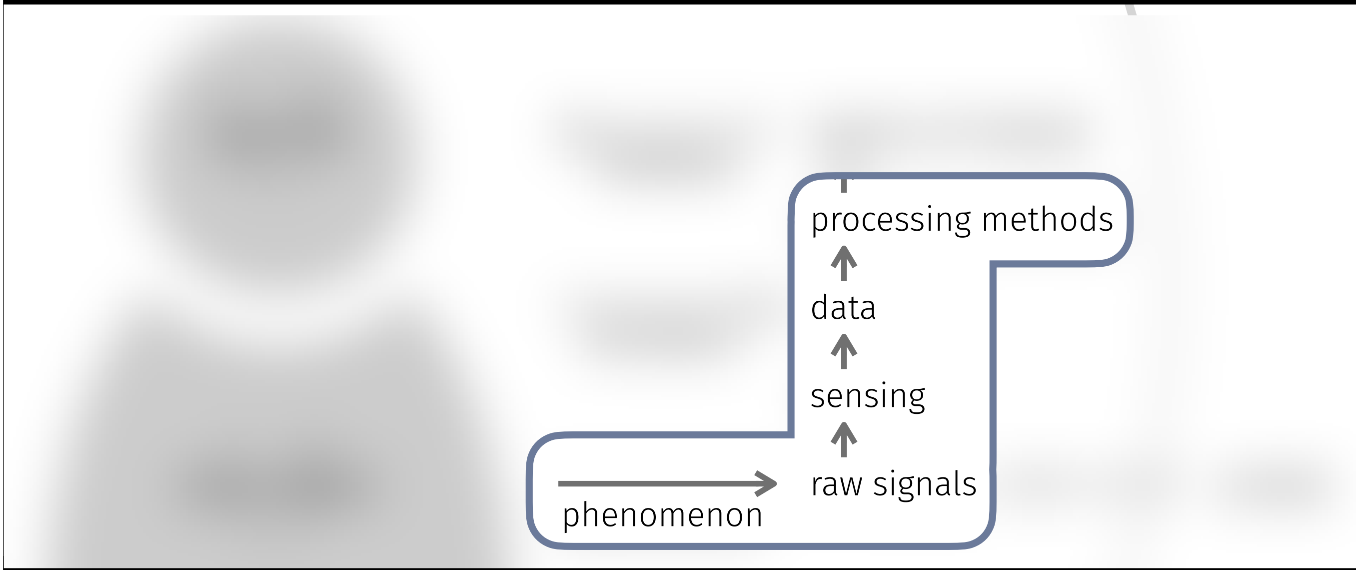
useful for solving Task 1 of the exercise

Task 1 due March 27, 2023

today’s goals

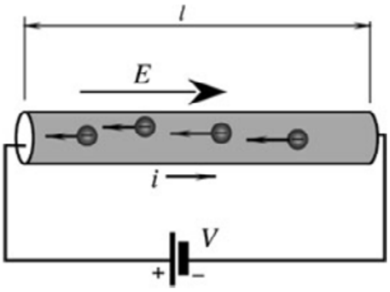


today’s goals

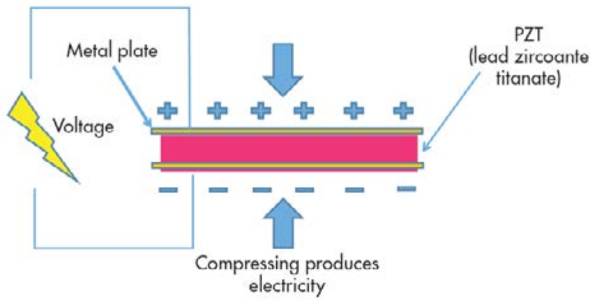


recap

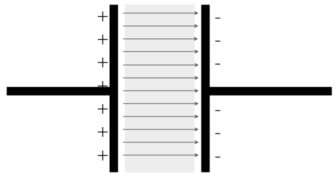
sensors



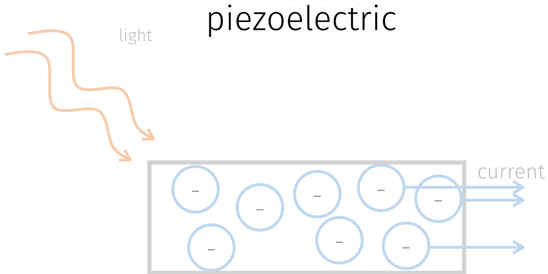
resistive



piezoelectric

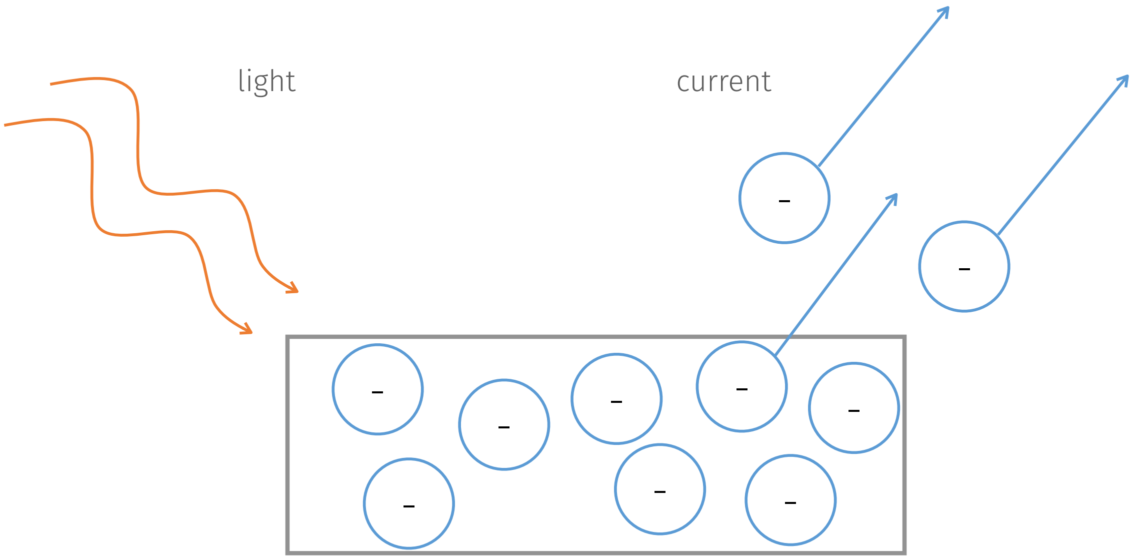


capacitive



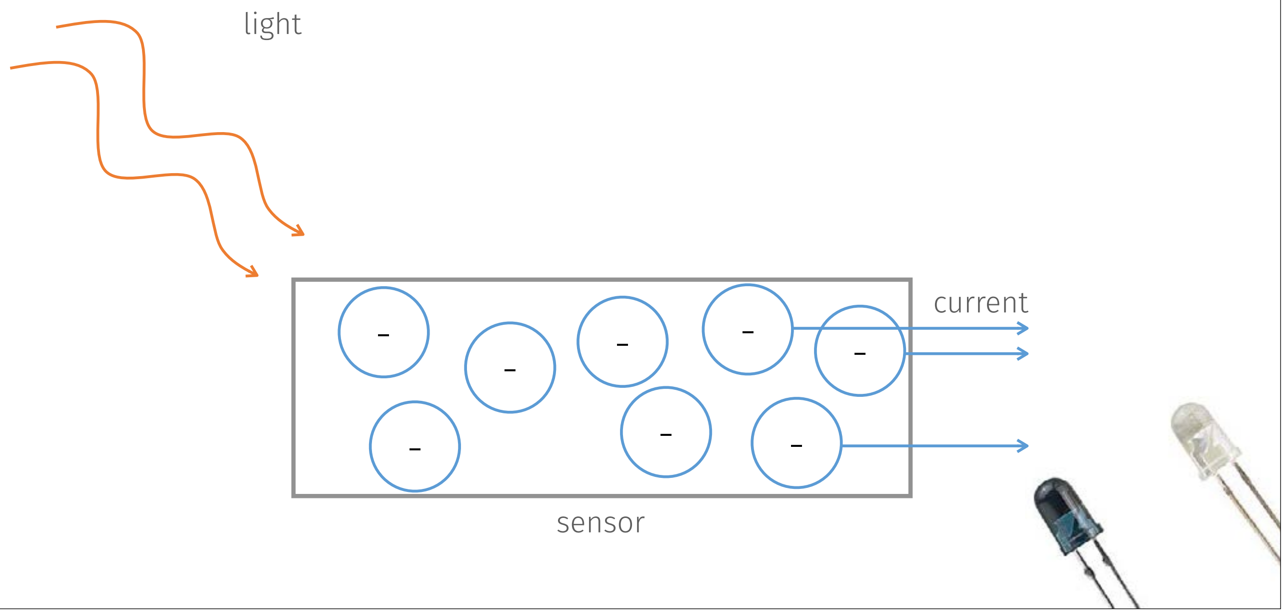
photoelectric

photoelectric: light sensors

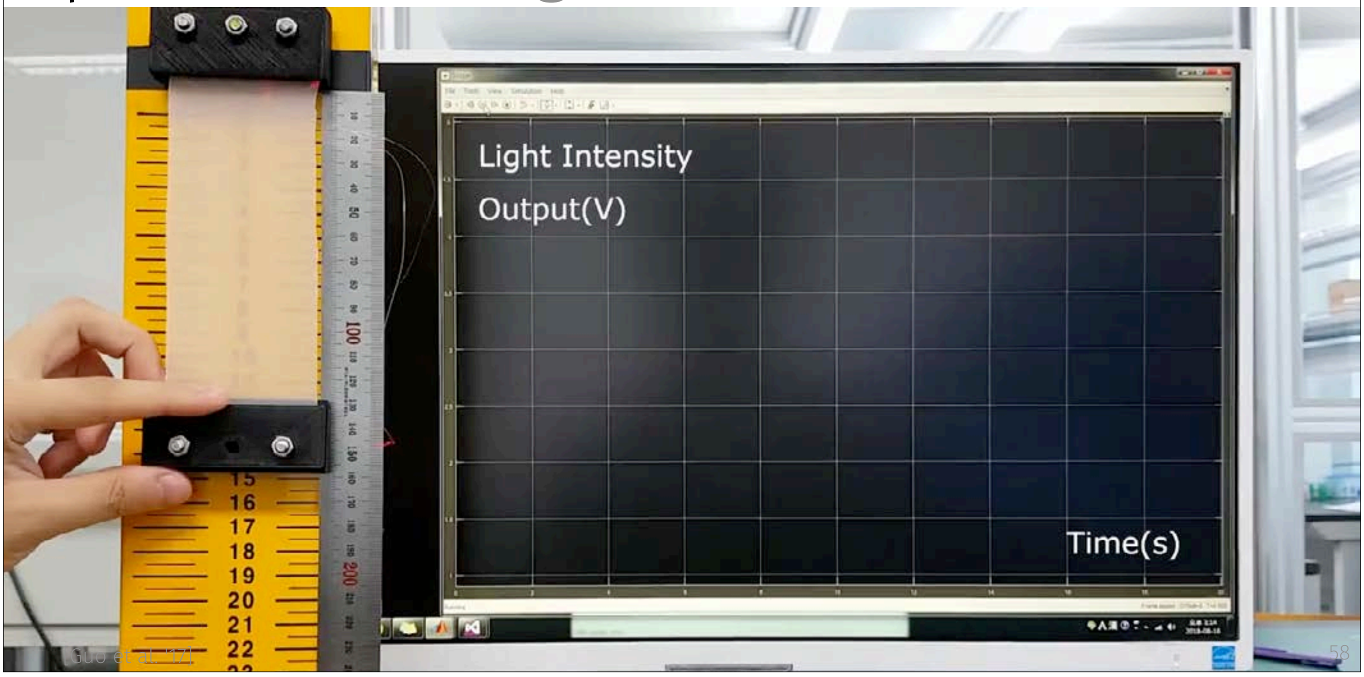


sensor

photoelectric: light sensors

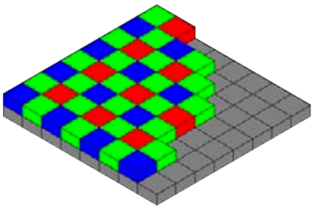


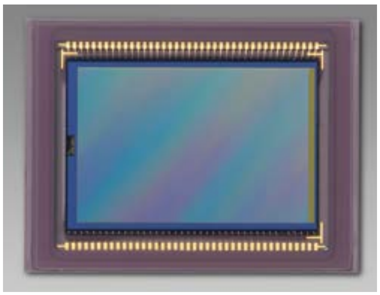
photoelectric: light sensors



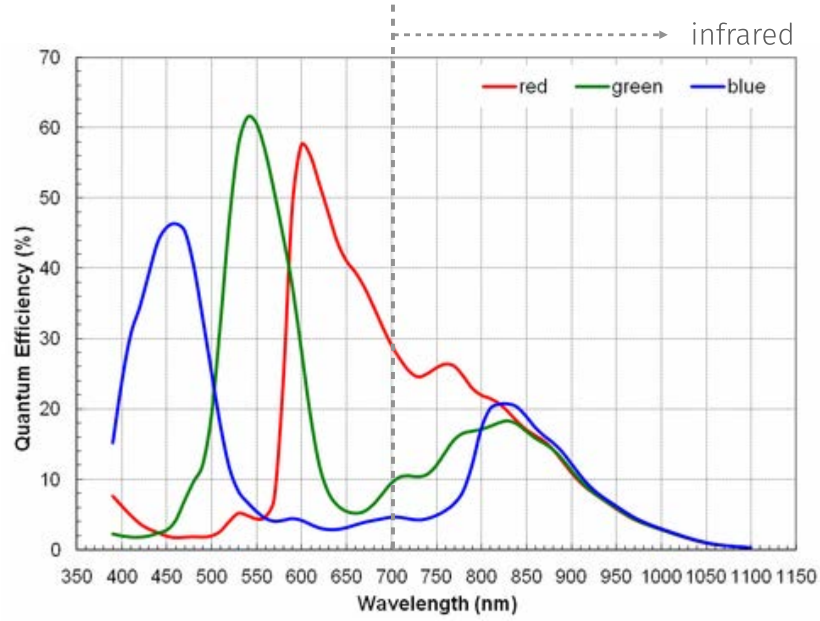
[Guo et al. '17]

photoelectric: light sensors

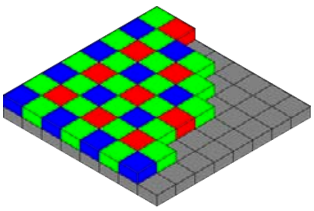


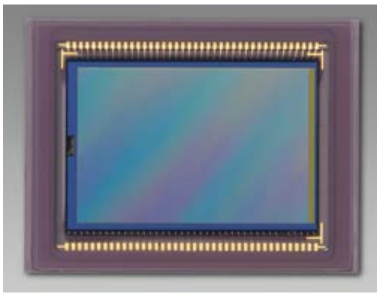


CMOS sensor
w/ Bayer filter pattern

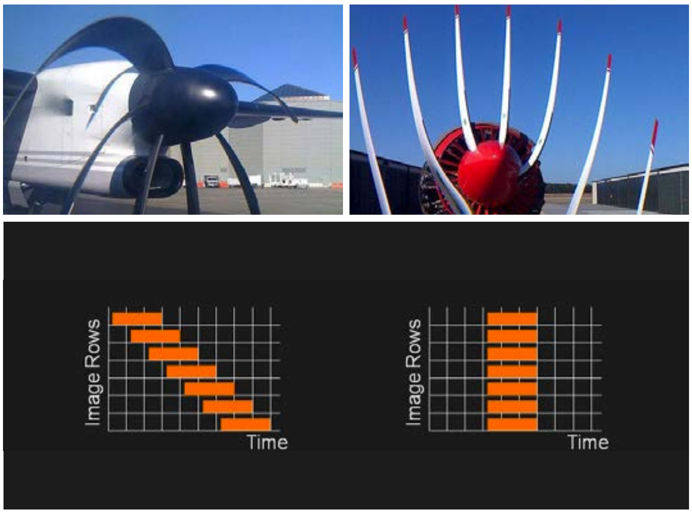


photoelectric: light sensors





CMOS sensor
w/ Bayer filter pattern



rolling shutter (CMOS)

+ fast frame rate
– noisy data

global shutter (CCD)

+ more sensitive
– slower

sensors as the human-computer **interface**

examples

example: vehicle sensors

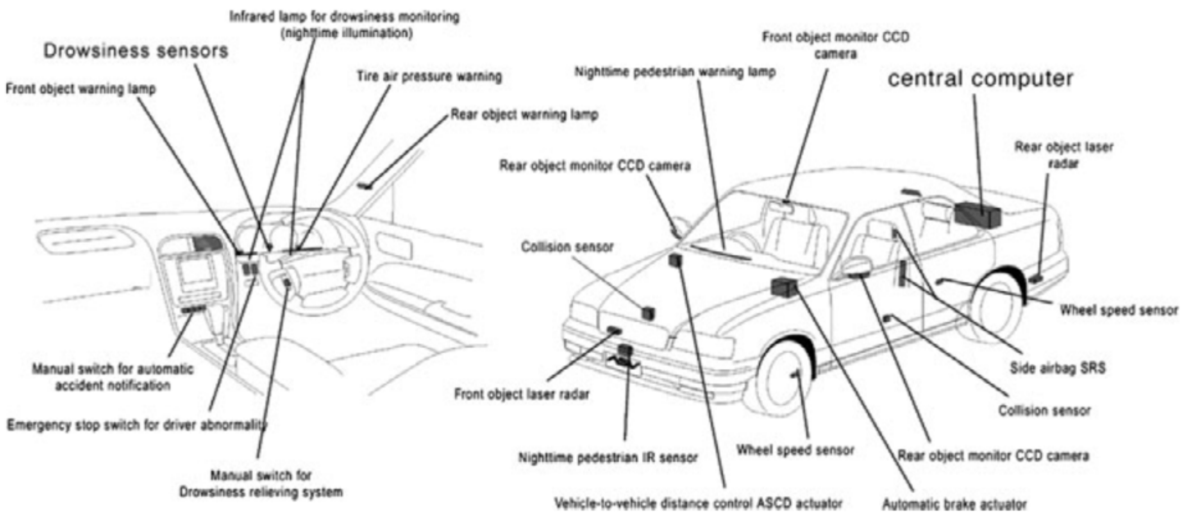


Fig. 1.4 Multiple sensors, actuators, and warning signals are parts of the advanced safety vehicle (Courtesy of Nissan Motor Company)

example: car seat, more than detection

phenomenon:

- mechanical (weight)

sensor:

- pressure sensor
- capacitive sensor

➪ different dielectric constants

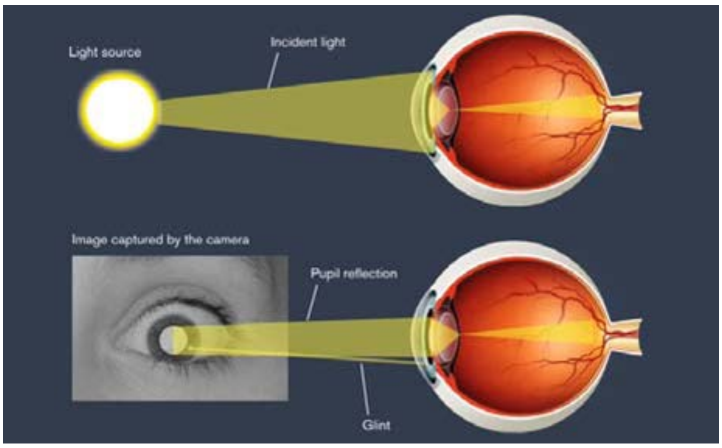
example: drowsiness in vehicle

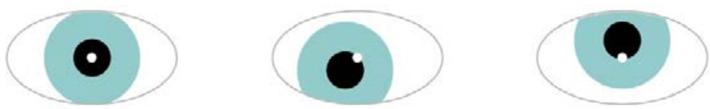
phenomenon:

- physiological

sensor:

- infrared camera





looking directly at the camera | looking down & to the right of the camera | looking above the camera

sensor considerations

environmental factors
temperature range
humidity effects
corrosions
size
ruggedness
susceptibility to EM interference
power consumption
over range protection

economic factors
cost
availability
lifetime

sensor characteristics
sensitivity
range
stability
repeatability
linearity
error
response time
frequency response

sensing **modalities**

link
exercise

modality :=

class of single independent channel
of input/output between human & device

only one implemented modality ➪ unimodal,
otherwise multimodal

multimodal input/output can be helpful to

- complement several channels of input sensing for better coverage
  ➪ overlapping modalities
- redundantly obtain or provide information

multimodal channels

equivalence	information is presented in multiple ways and can be interpreted as the same information
specialization	when a specific kind of information is always processed through the same modality
redundancy	multiple modalities process the same information
complementarity	multiple modalities take separate information and merge it
transfer	a modality produces information that another modality consumes
concurrency	multiple modalities take in separate information that is not merged

typology by [Martin et al., '01]

example modalities

common simple modalities
- button
- touch
- shake?

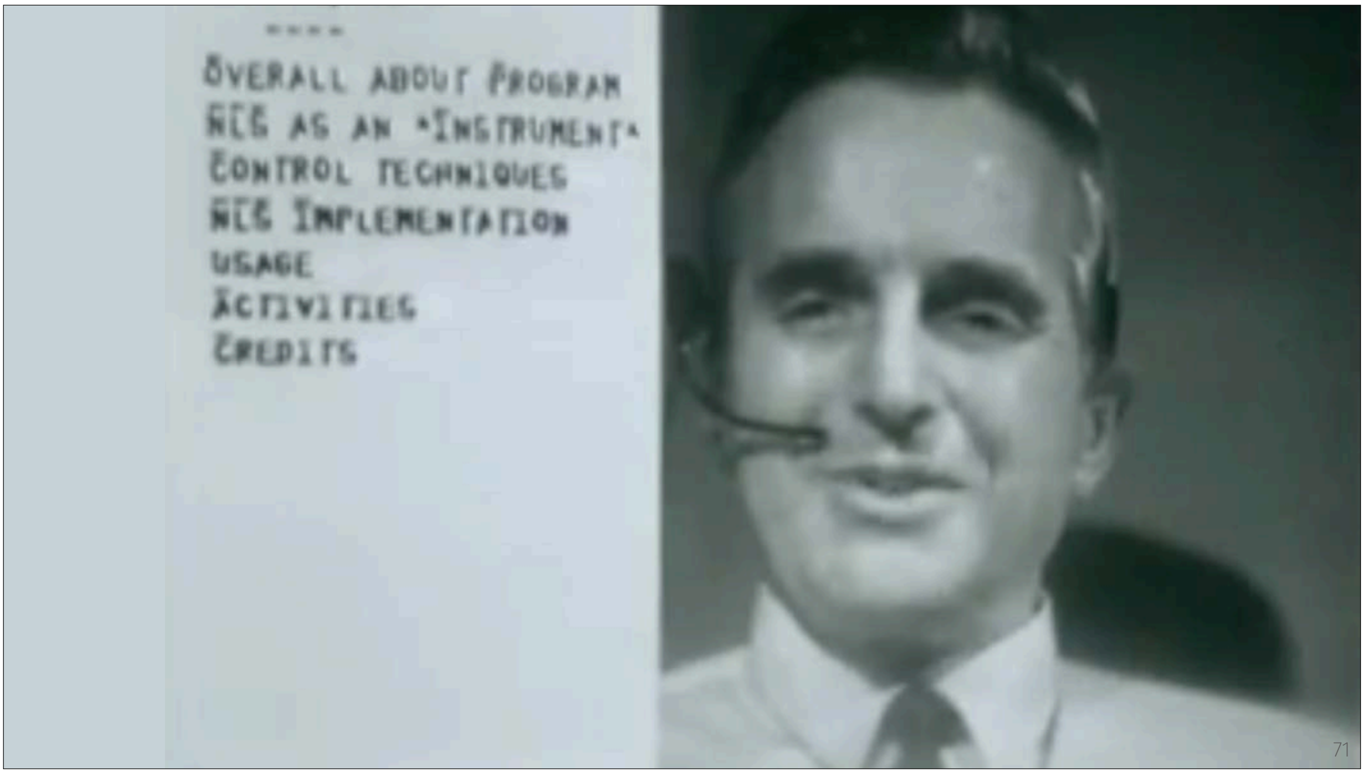
explicit

more complex modalities
- vision (e.g., for gesture, pose, activity)
- speech
- motion and activity

implicit

explicit sensing

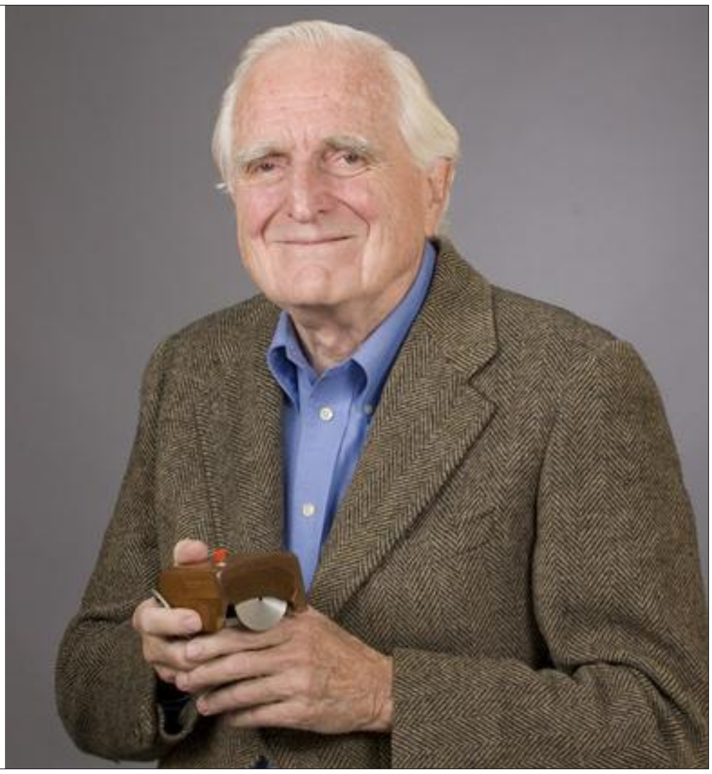
active and reliable input



Doug Engelbart

“The mother of all demos”, 1968

https://youtu.be/M5PgQS3ZBWA



explicit: Engelbart’s mouse (1961, SRI and on)

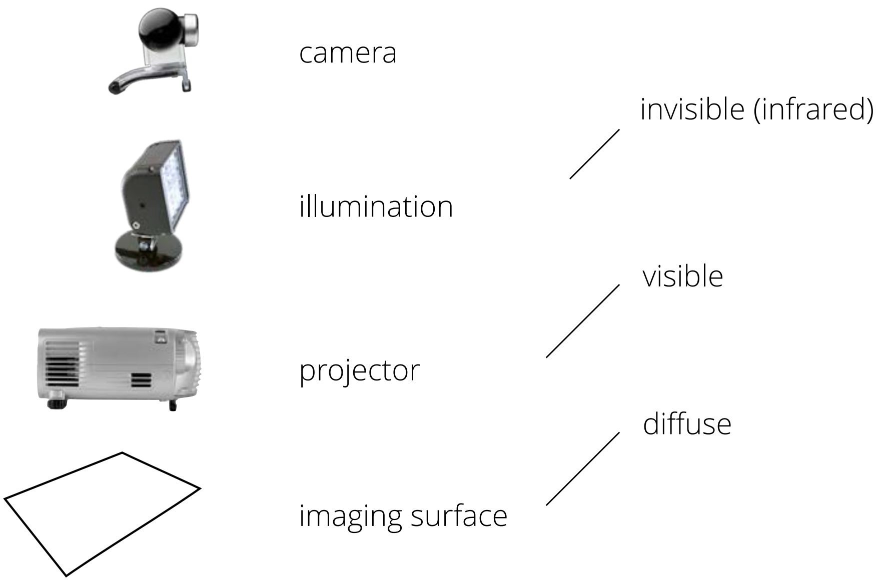

Engelbart Institute: https://www.dougengelbart.org/

implicit sensing

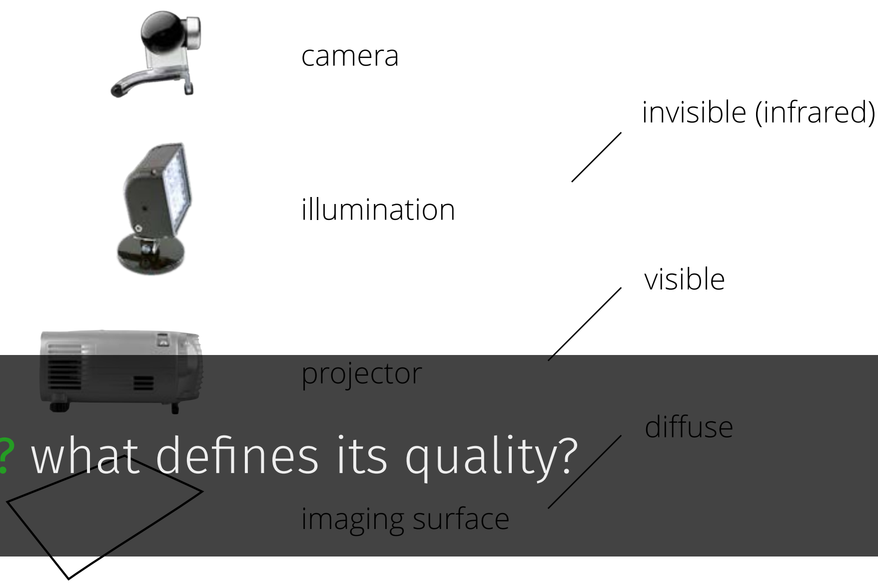
input inferred from continuous observations, thus can be passive

brief detour:
touch sensing

optical touch-input sensing

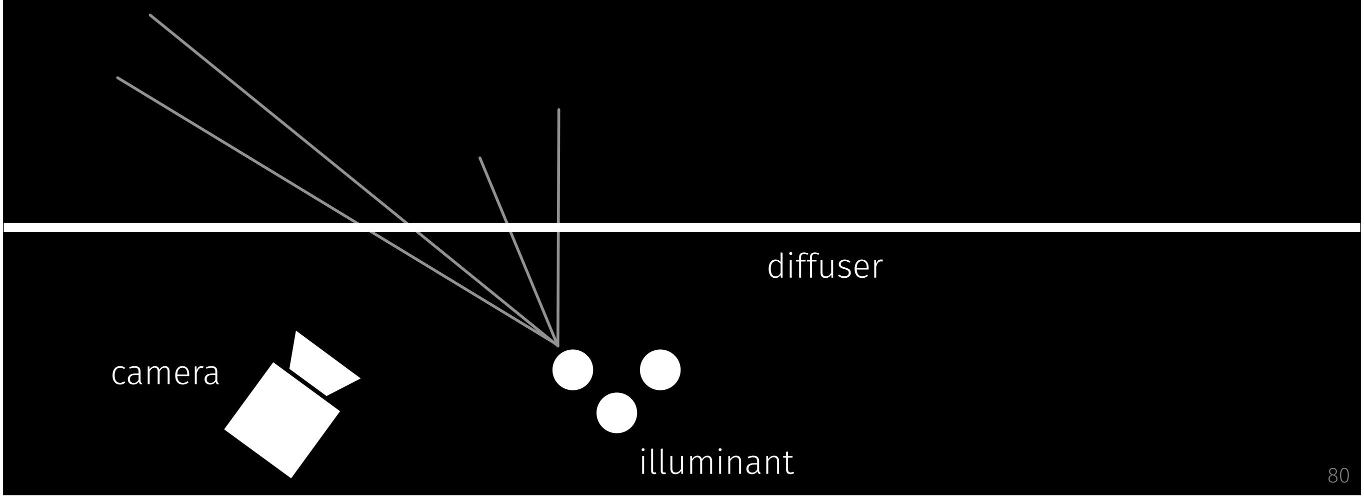


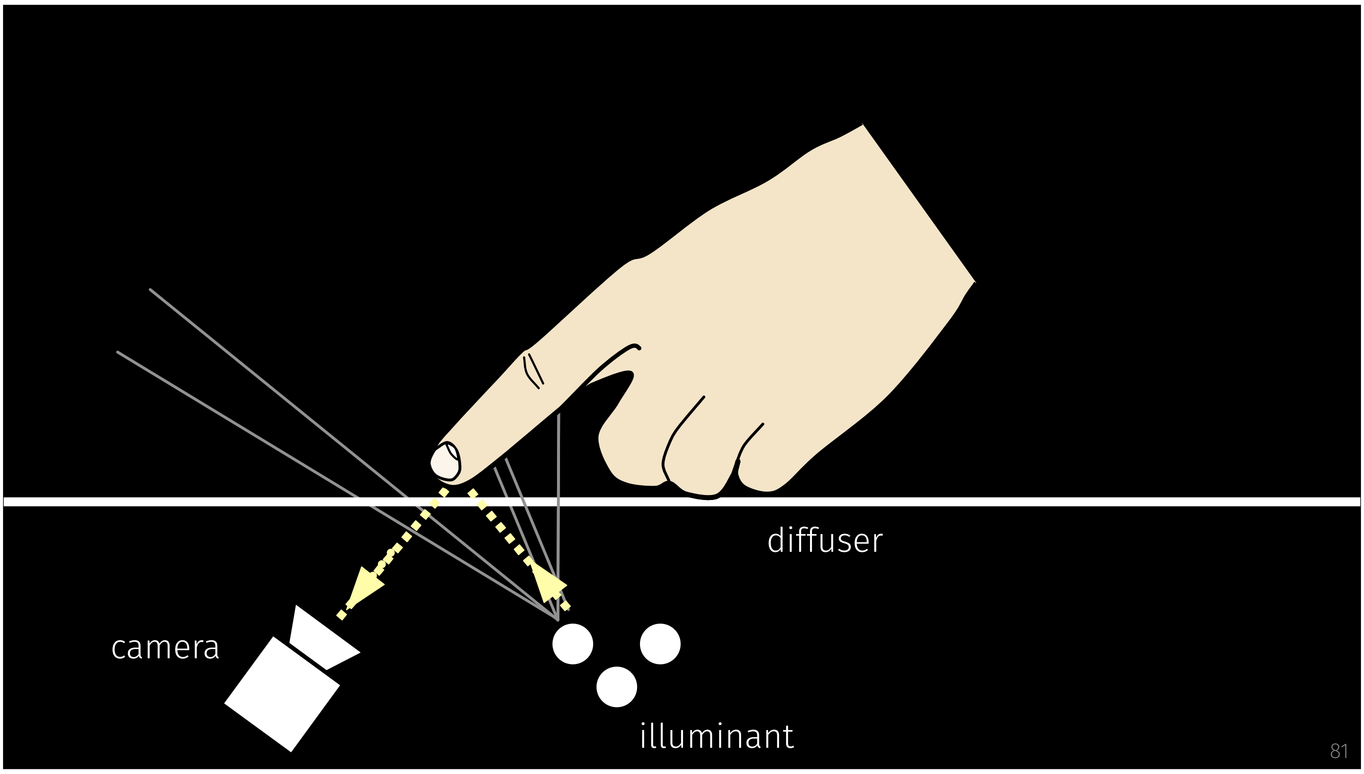
optical touch-input sensing



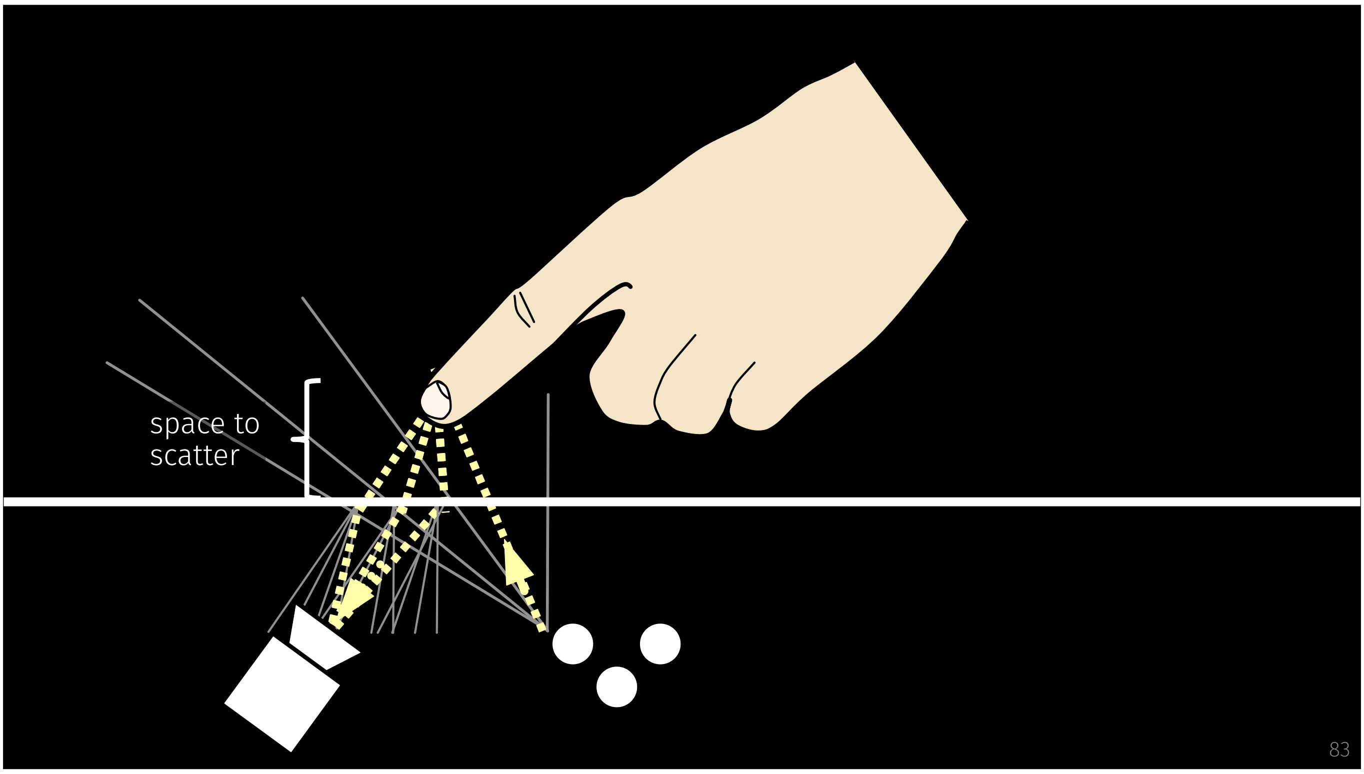
what’s a diffuser? what defines its quality?

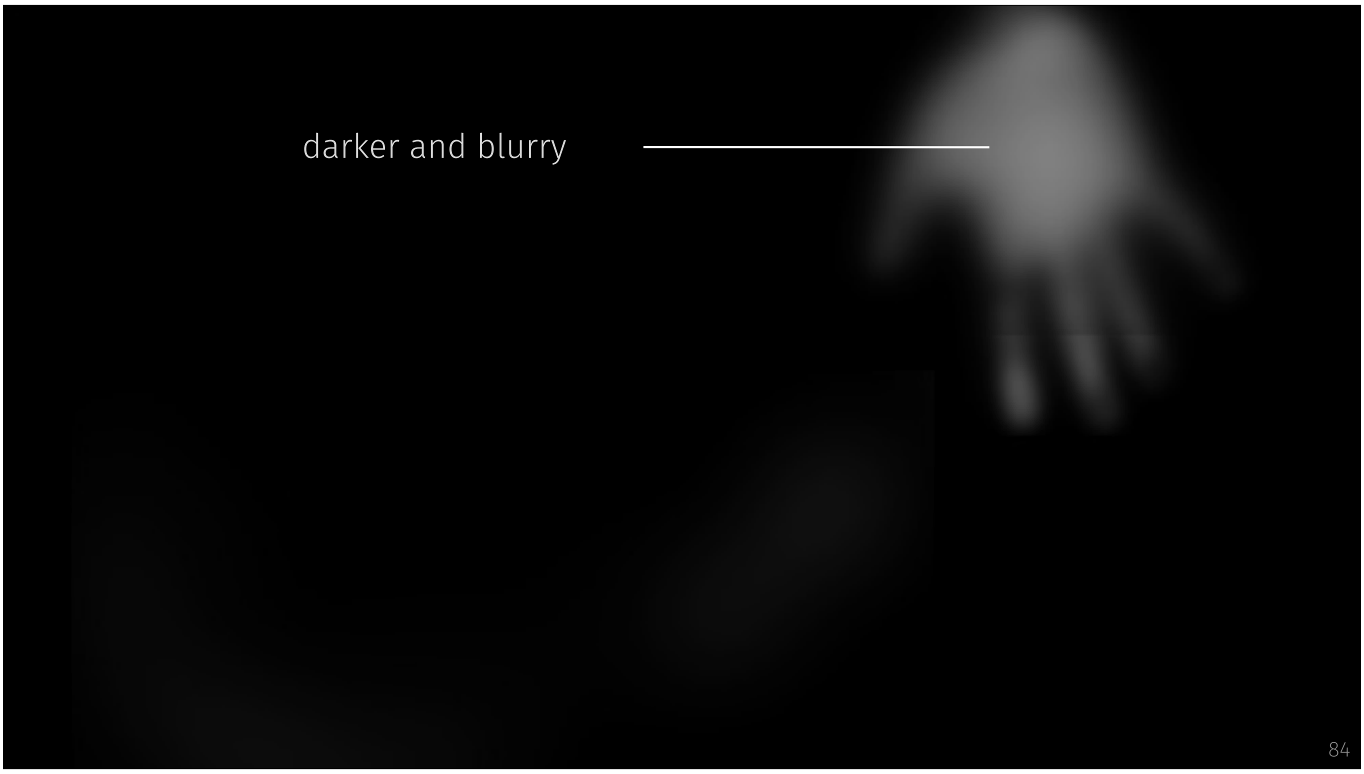
diffused illumination

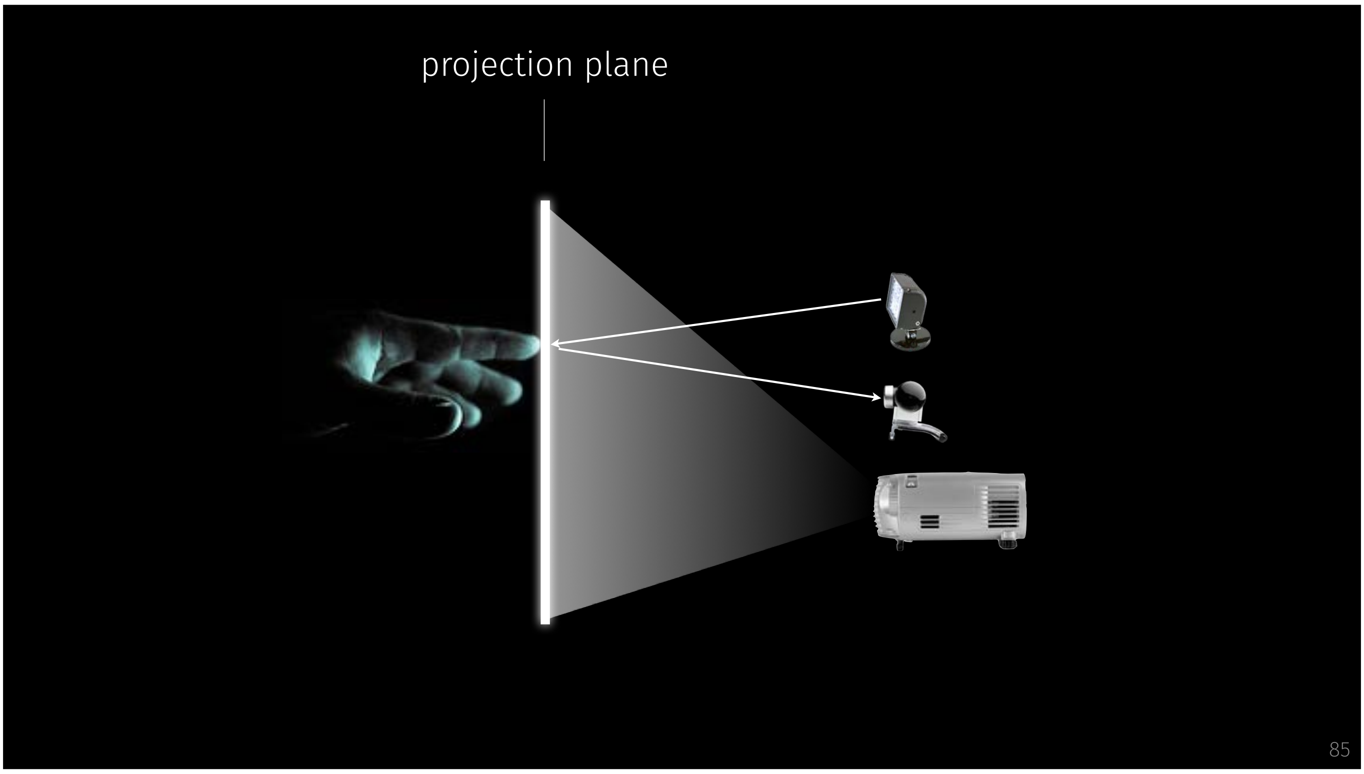



diffuser
camera
illuminant
81


82


space to
scatter
83


darker and blurry
84



86

HoloWall [Rekimoto, UIST '97]

</

end of detour
diffused illumination will be important later

example: ambient light sensors

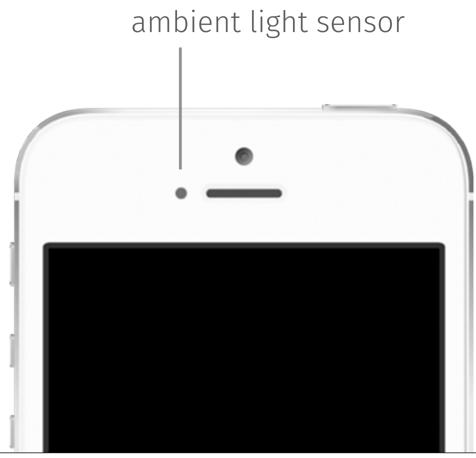
measures the light level around the device

units in SI lux

applications:

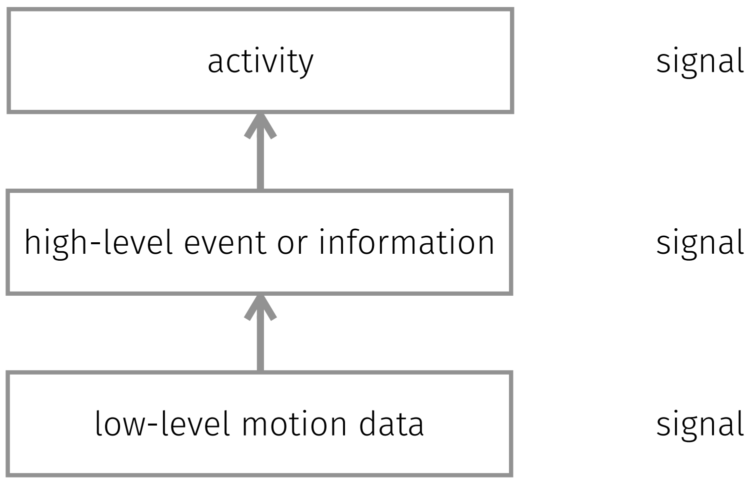
- adjust the display's brightness
- adjust image coloring
- context & activity detection

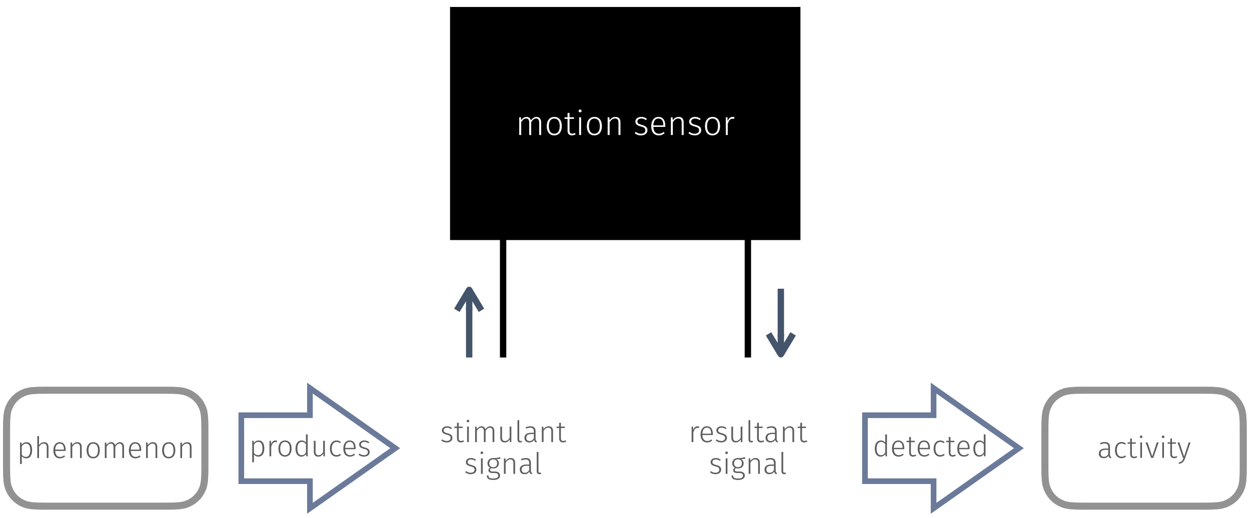
key: diffuse illumination
(≠ focused lens-based sensing)

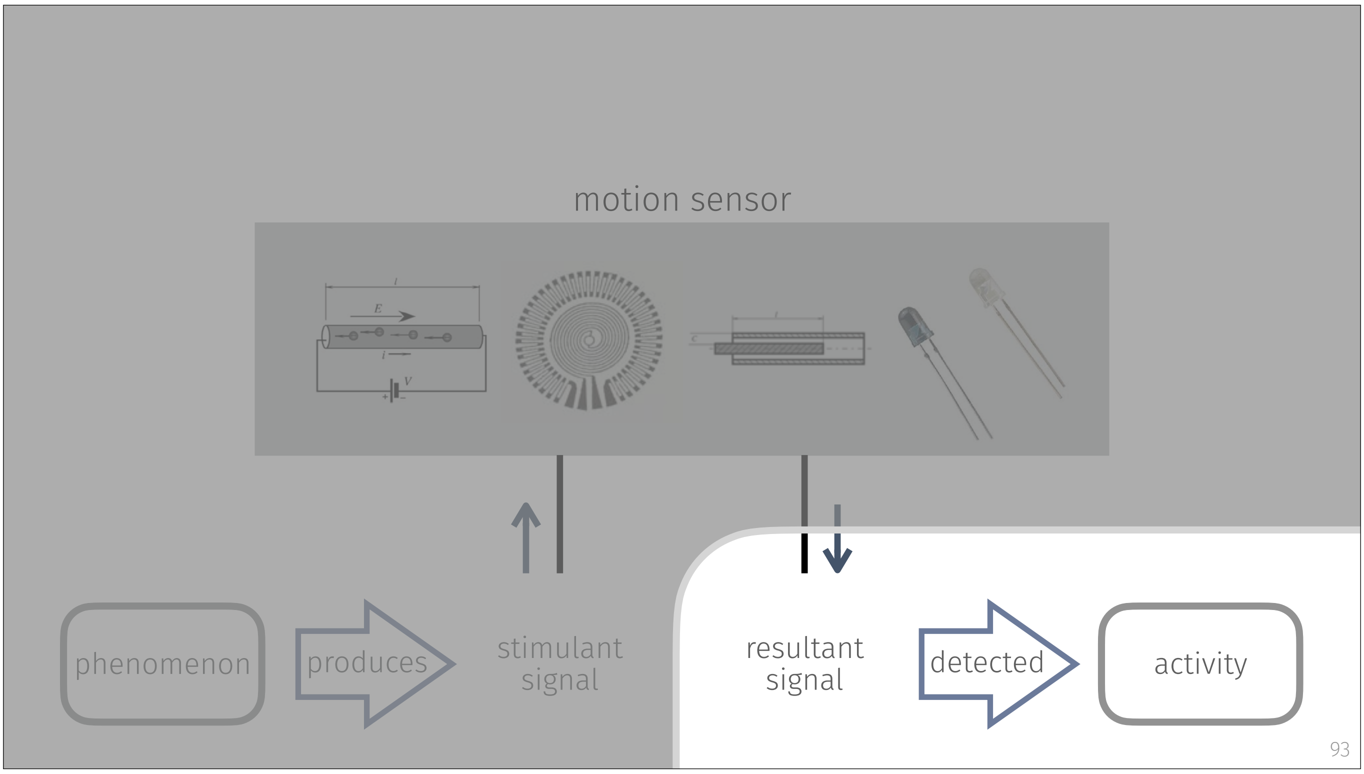


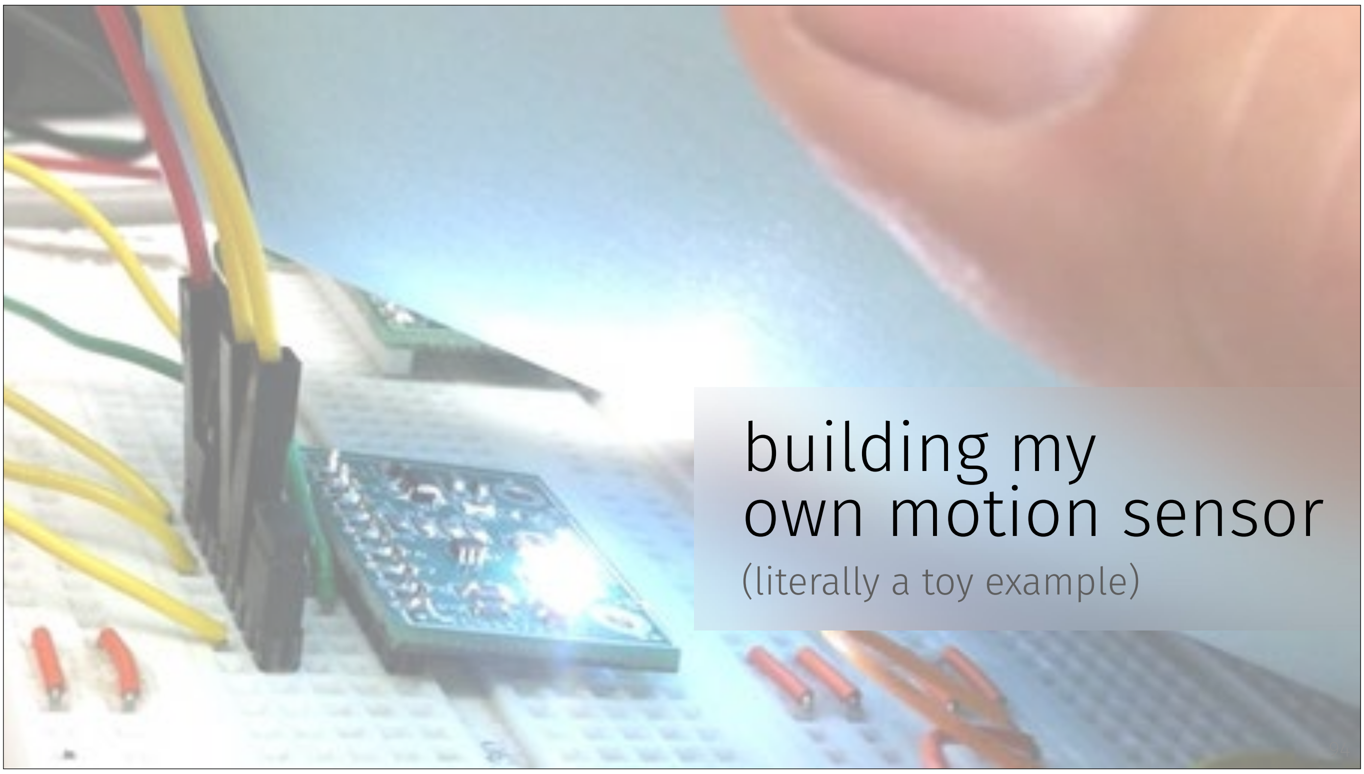
activity sensing (of motion)

abstraction levels





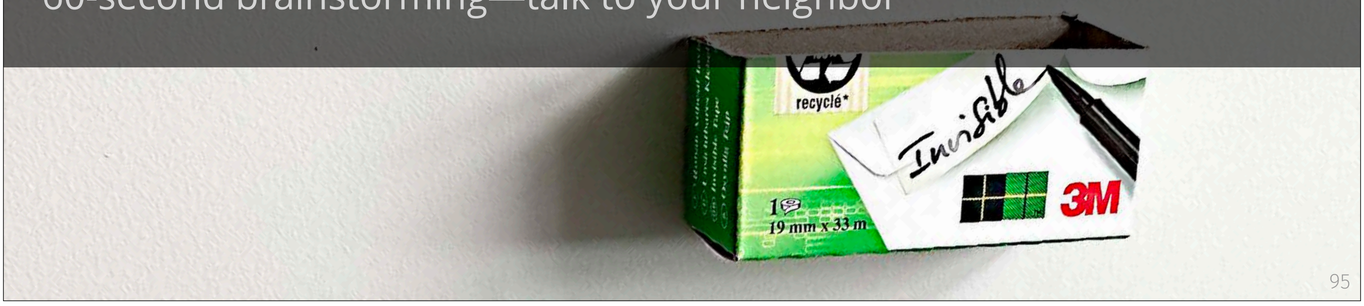

motion sensor
phenomenon
produces
stimulant signal
resultant signal
detected
activity
93


building my
own motion sensor
(literally a toy example)

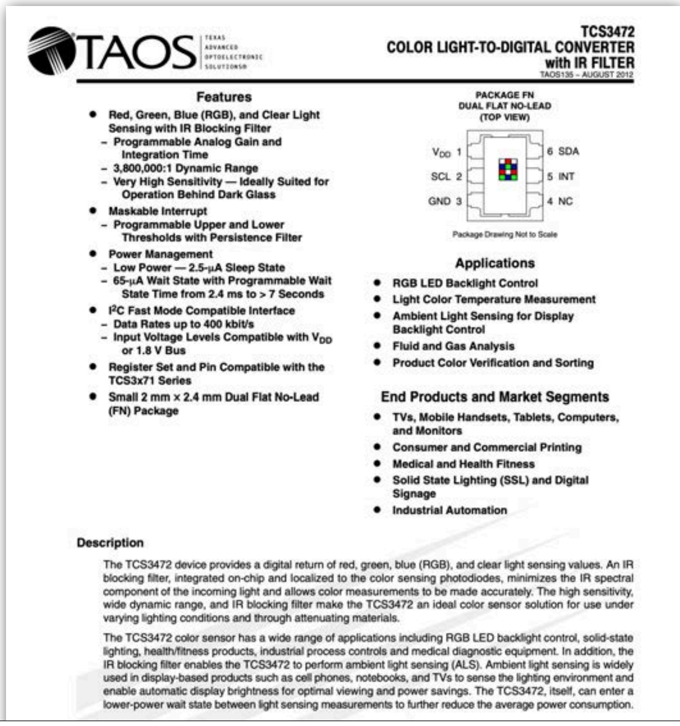
goal

build a tangible "device" that
- detects shaking
- quantifies pressure applied on the side

how would you do it?
60-second brainstorming—talk to your neighbor



TCS34725



TAOS TEXAS ADVANCED OPTOELECTRONIC SOLUTIONS®

TCS3472
COLOR LIGHT-TO-DIGITAL CONVERTER with IR FILTER
TAOS135 – AUGUST 2012

Features
- Red, Green, Blue (RGB), and Clear Light Sensing with IR Blocking Filter
  - Programmable Analog Gain and Integration Time
  - 3,800,000:1 Dynamic Range
  - Very High Sensitivity — Ideally Suited for Operation Behind Dark Glass
- Maskable Interrupt
  - Programmable Upper and Lower Thresholds with Persistence Filter
- Power Management
  - Low Power — 2.5-µA Sleep State
  - 65-µA Wait State with Programmable Wait State Time from 2.4 ms to > 7 Seconds
- I^2C Fast Mode Compatible Interface
  - Data Rates up to 400 kbit/s
  - Input Voltage Levels Compatible with V_{DD} or 1.8 V Bus
- Register Set and Pin Compatible with the TCS3x71 Series
- Small 2 mm × 2.4 mm Dual Flat No-Lead (FN) Package

PACKAGE FN
DUAL FLAT NO-LEAD
(TOP VIEW)

V_{DD} 1 — 6 SDA
SCL 2 — 5 INT
GND 3 — 4 NC

Package Drawing Not to Scale

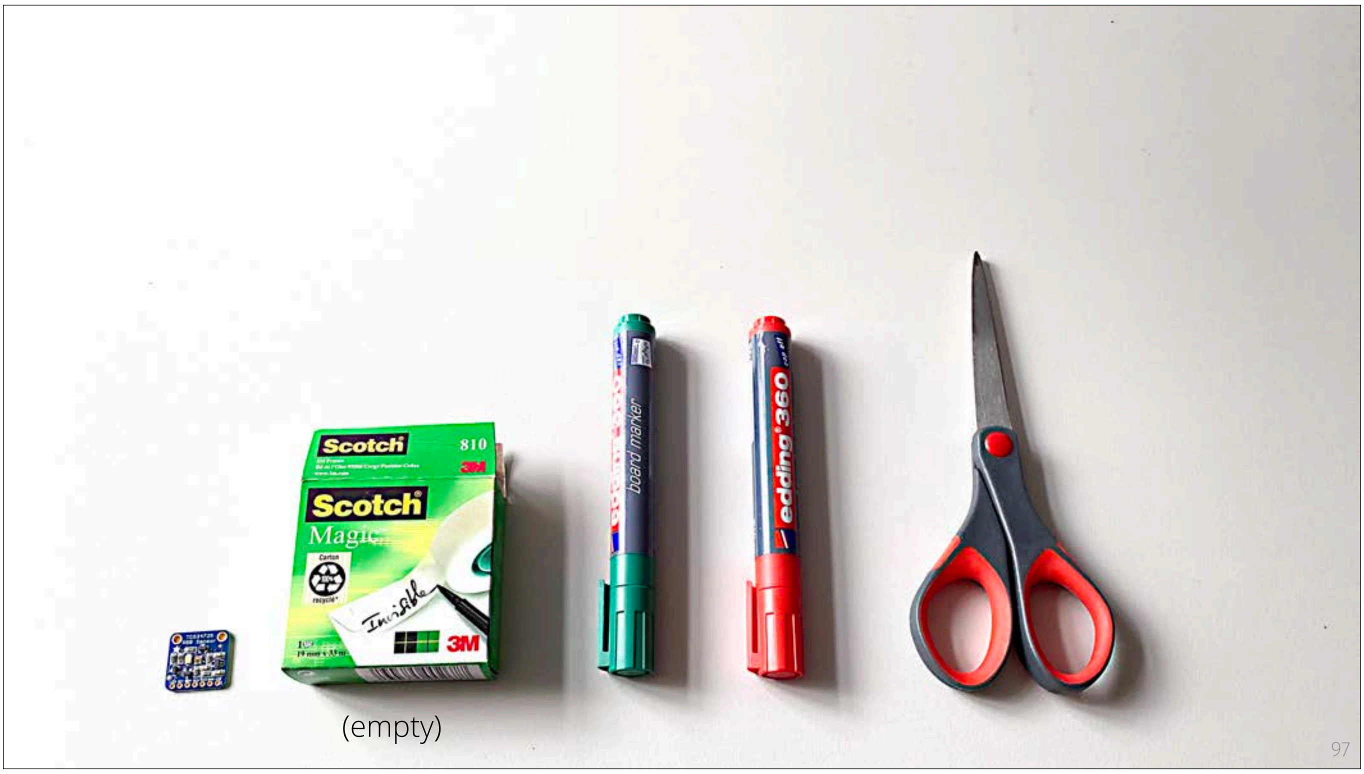
Applications
- RGB LED Backlight Control
- Light Color Temperature Measurement
- Ambient Light Sensing for Display Backlight Control
- Fluid and Gas Analysis
- Product Color Verification and Sorting

End Products and Market Segments
- TVs, Mobile Handsets, Tablets, Computers, and Monitors
- Consumer and Commercial Printing
- Medical and Health Fitness
- Solid State Lighting (SSL) and Digital Signage
- Industrial Automation

Description

The TCS3472 device provides a digital return of red, green, blue (RGB), and clear light sensing values. An IR blocking filter, integrated on-chip and localized to the color sensing photodiodes, minimizes the IR spectral component of the incoming light and allows color measurements to be made accurately. The high sensitivity, wide dynamic range, and IR blocking filter make the TCS3472 an ideal color sensor solution for use under varying lighting conditions and through attenuating materials.

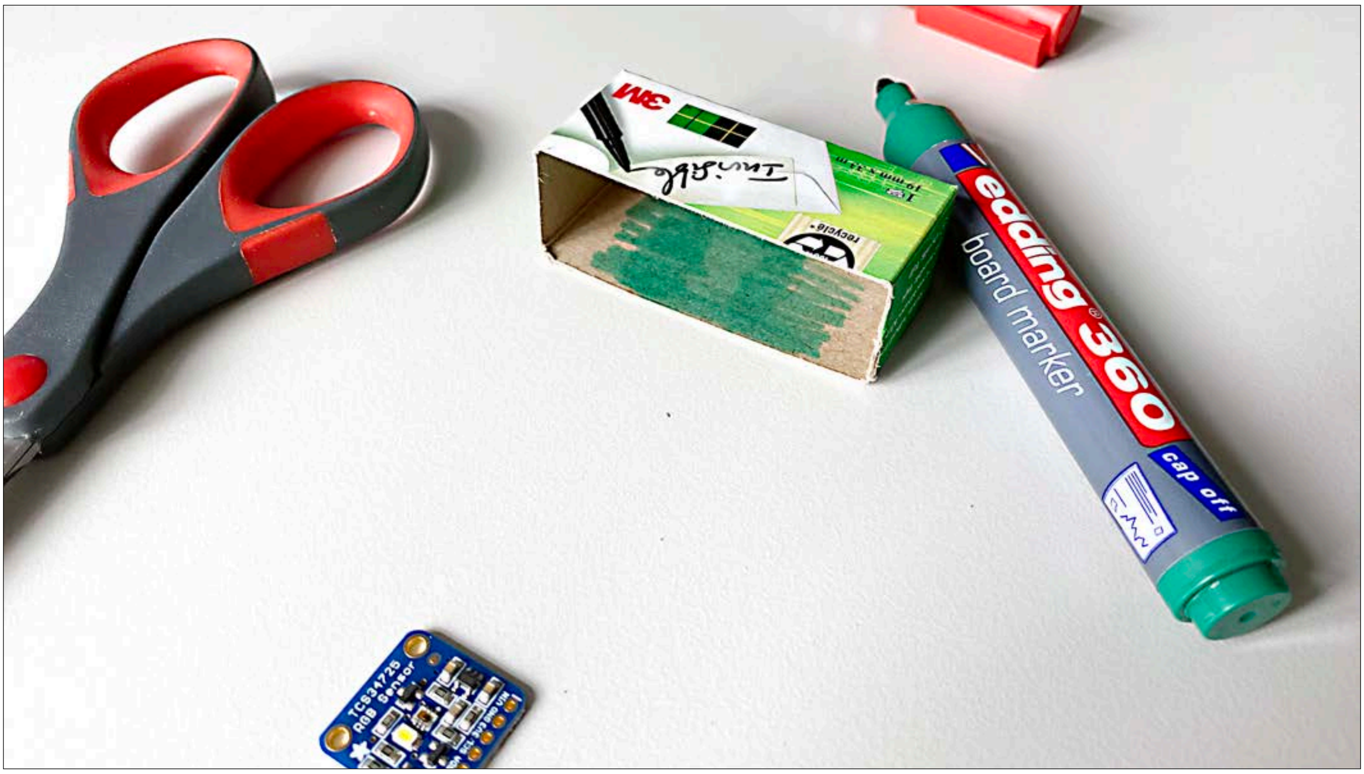
The TCS3472 color sensor has a wide range of applications including RGB LED backlight control, solid-state lighting, health/fitness products, industrial process controls and medical diagnostic equipment. In addition, the IR blocking filter enables the TCS3472 to perform ambient light sensing (ALS). Ambient light sensing is widely used in display-based products such as cell phones, notebooks, and TVs to sense the lighting environment and enable automatic display brightness for optimal viewing and power savings. The TCS3472, itself, can enter a lower-power wait state between light sensing measurements to further reduce the average power consumption.

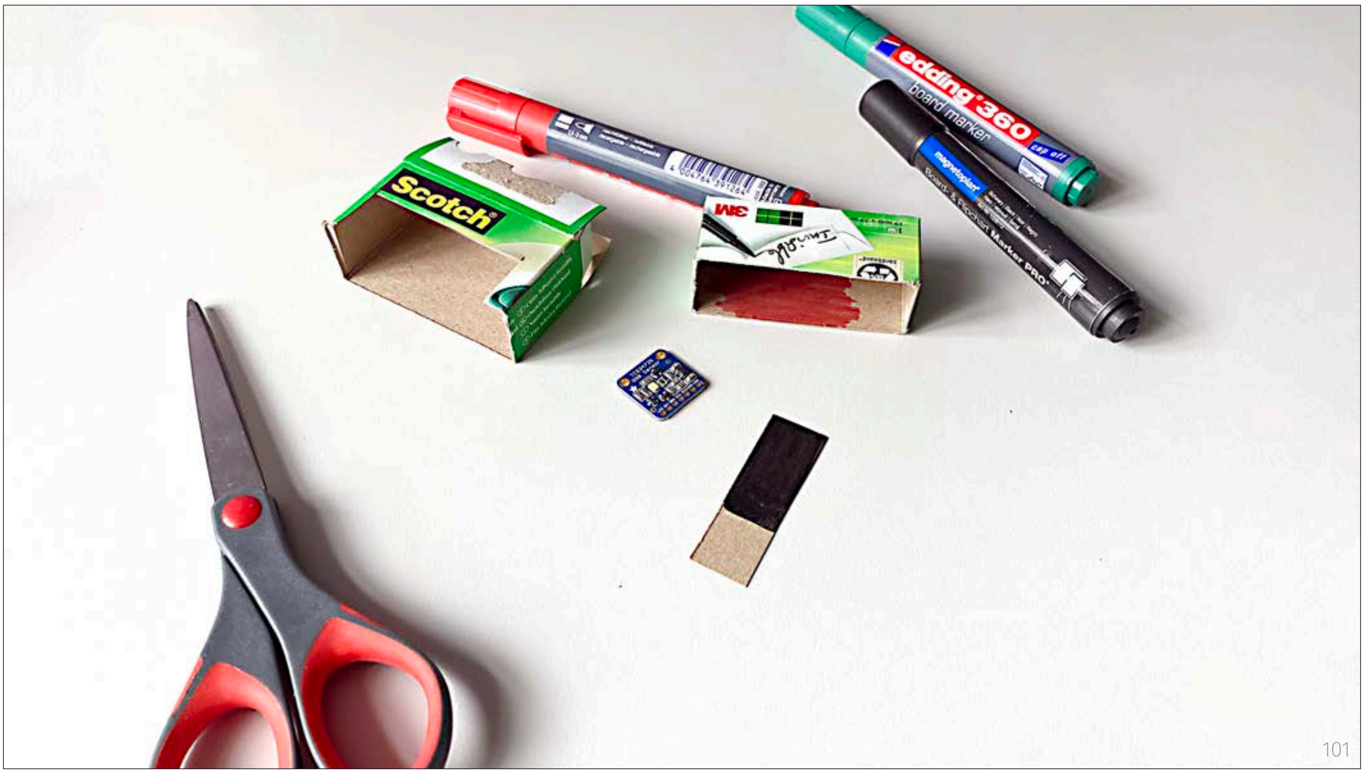

Scotch
810
3M
Scotch
Magic
board marker
edding 360
3M

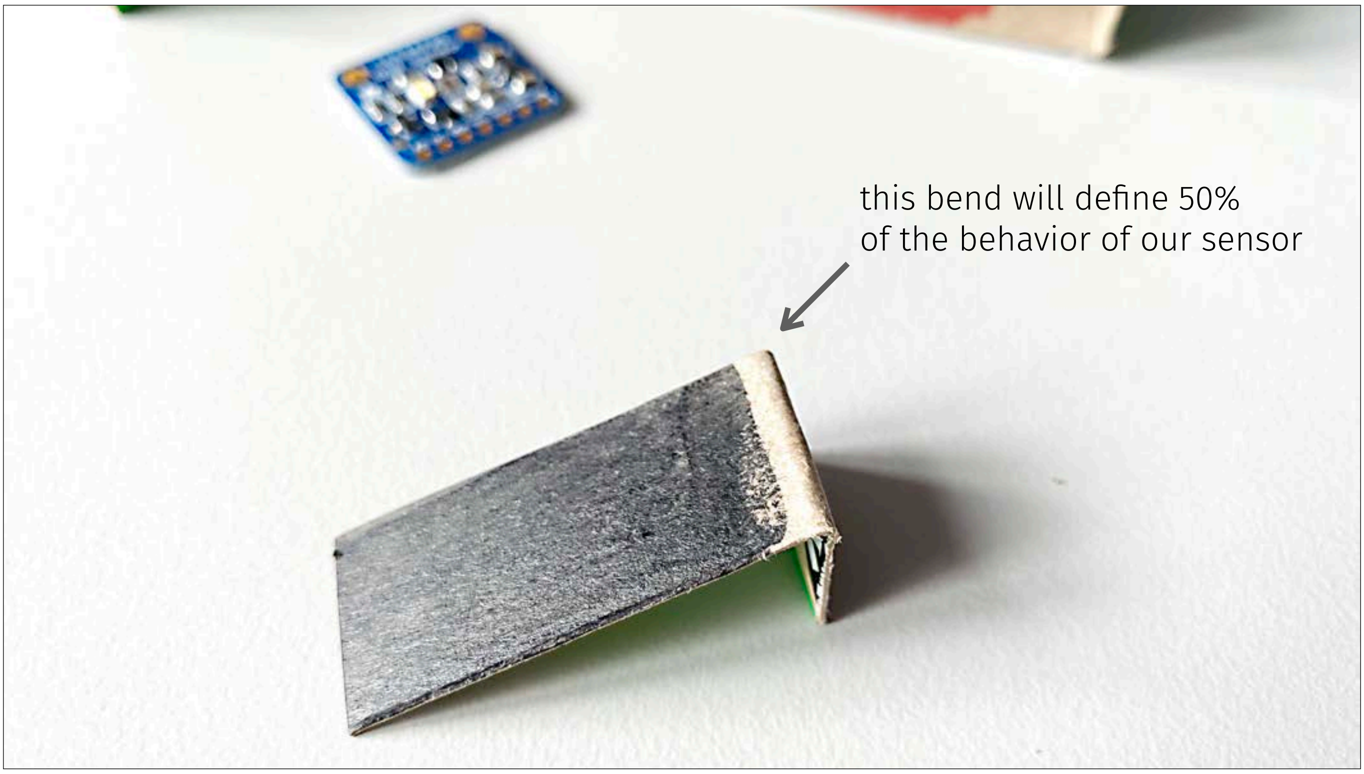
(empty)

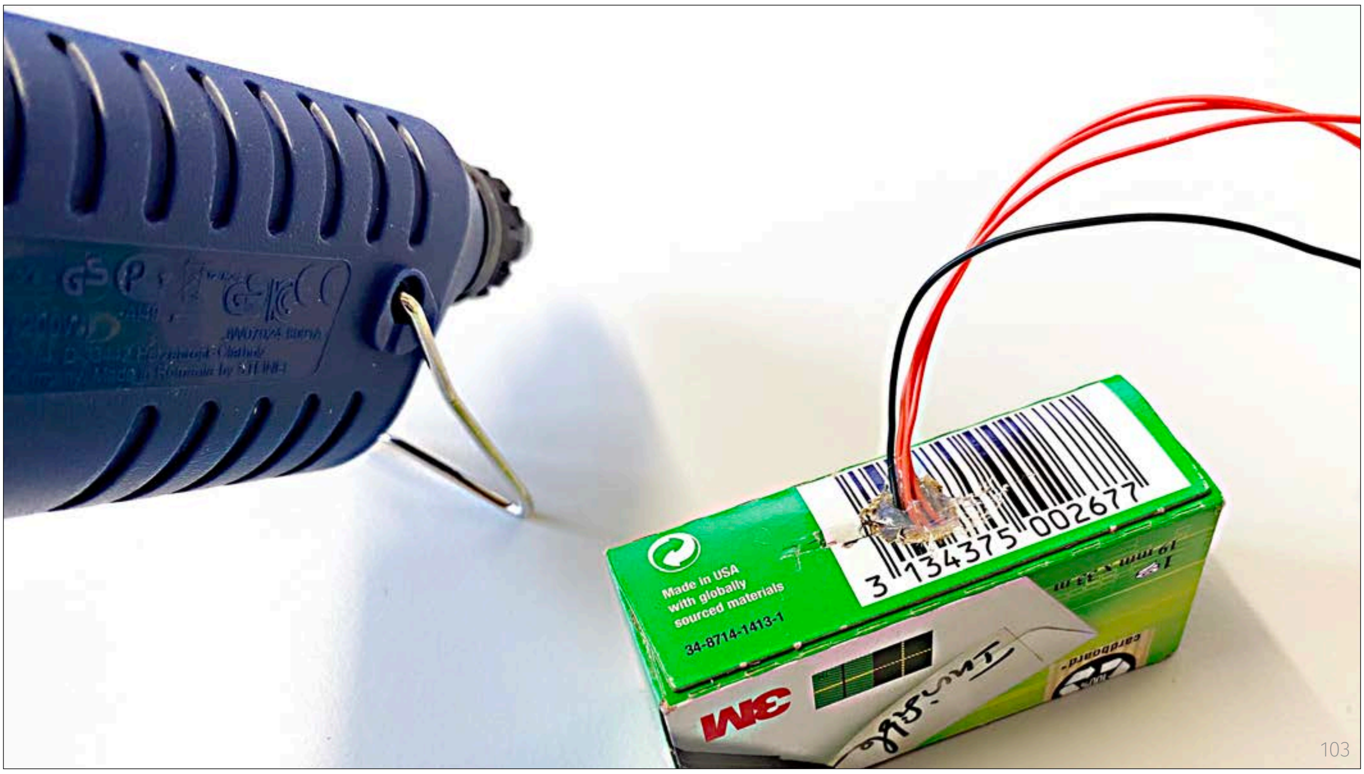

3M
edding 360
board marker

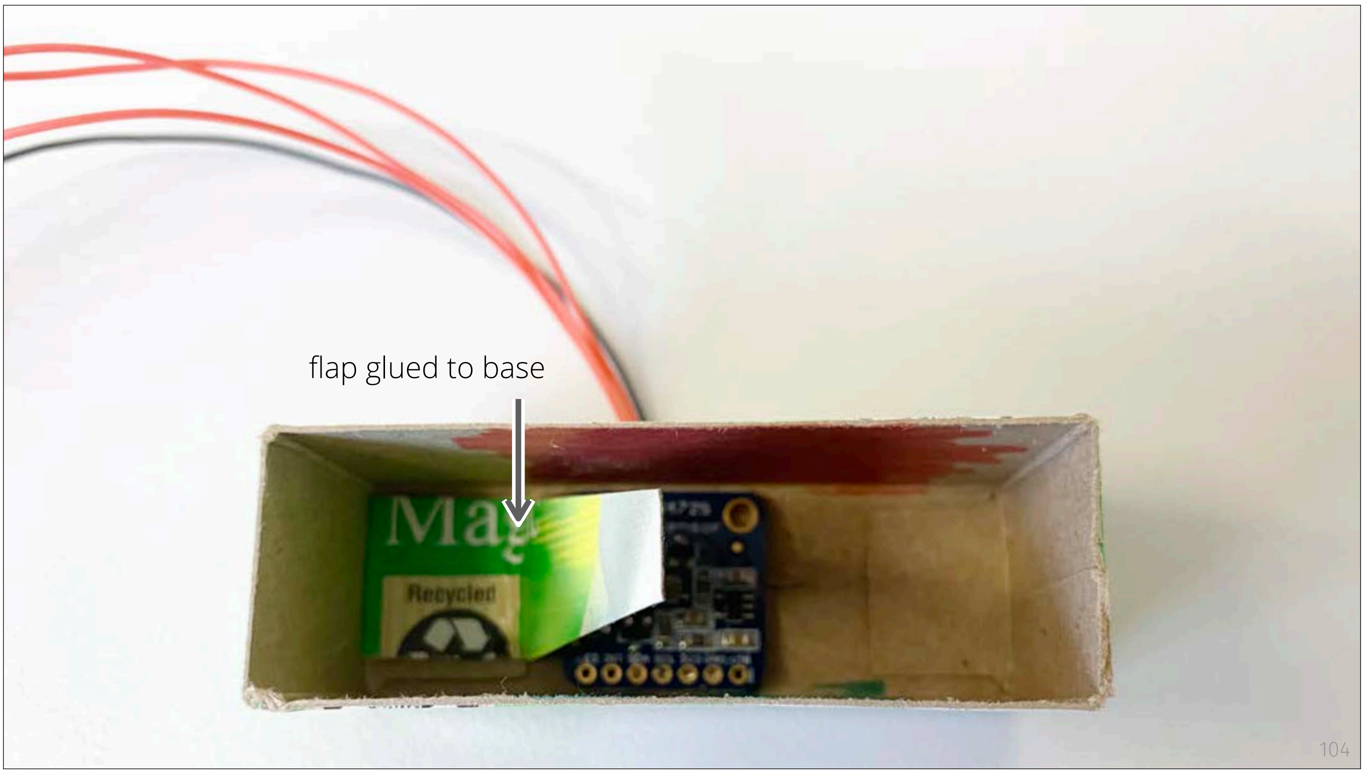

edding 360
board marker
cap off
3M


3M
edding 360
board marker
cap off
TCS34725
RGB Sensor


Scotch
edding 360
board marker

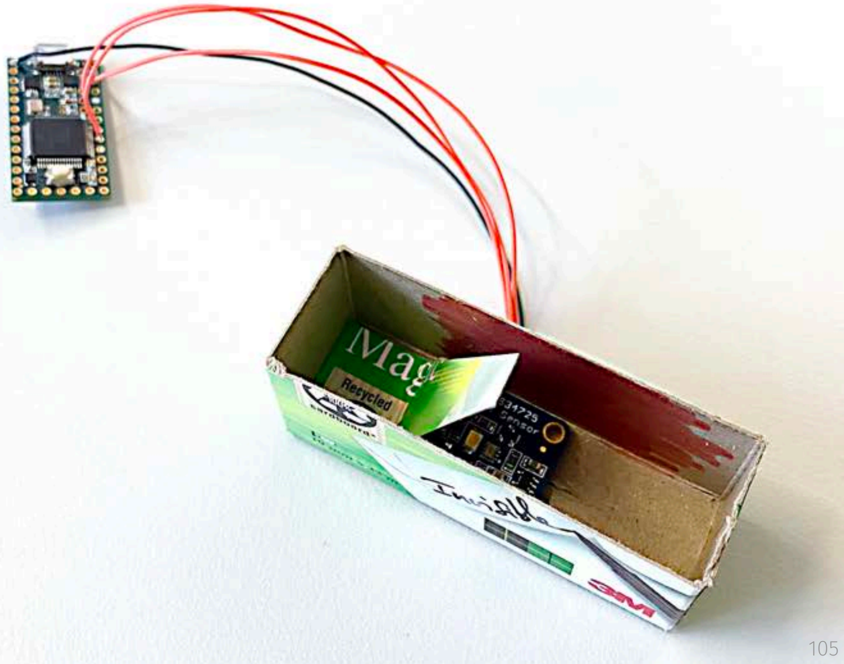

this bend will define 50%
of the behavior of our sensor


Made in USA
with globally
sourced materials
34-8714-1413-1
103

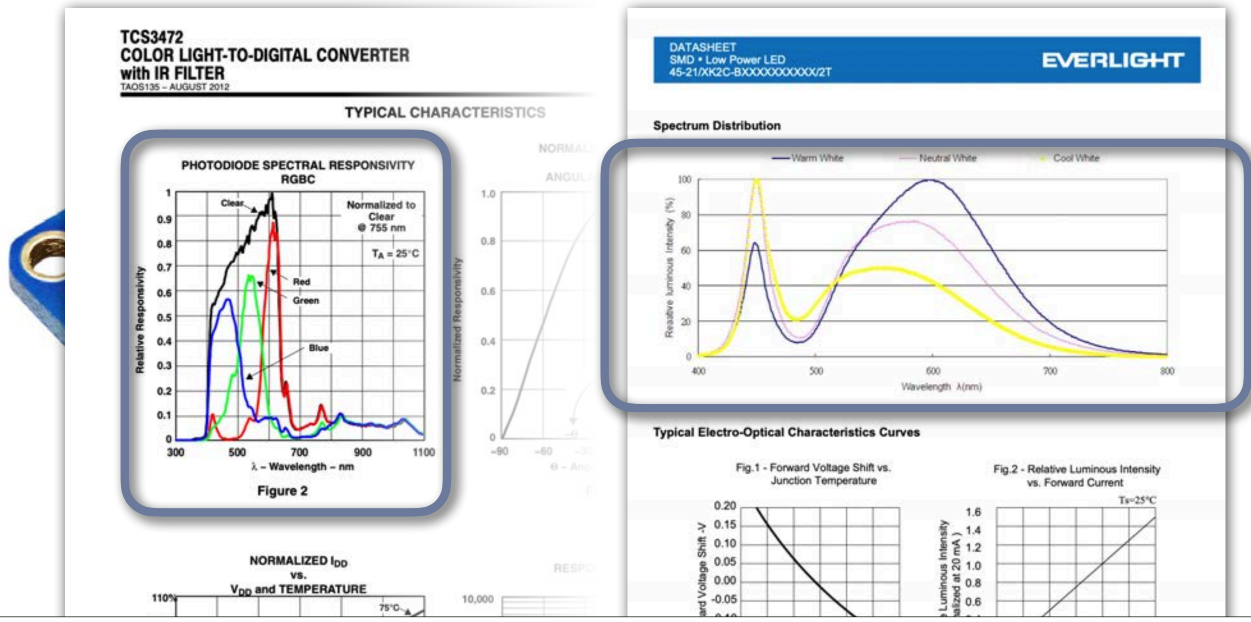

flap glued to base
Recycled
104

added a microcontroller

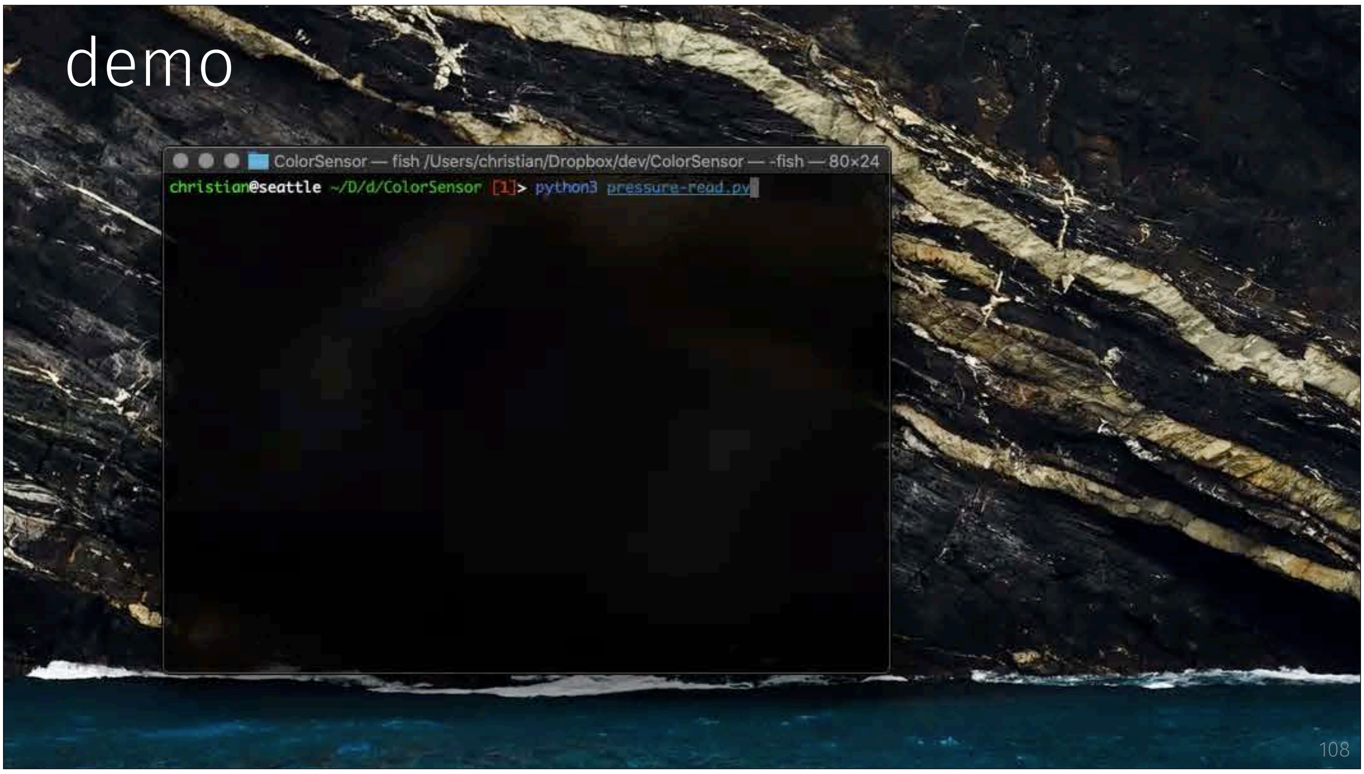
- Teensy (ARM Cortex-M4)
- Arduino-compatible



TCS34725




demo
107


demo
ColorSensor — fish /Users/christian/Dropbox/dev/ColorSensor — -fish — 80×24
christian@seattle ~/D/d/ColorSensor [1]> python3 pressure-read.py
108

motion sensors

motion sensors

responsible for measuring any kind of force
that could create motion in the x, y, and z axes of the sensor

motion can be either a linear or angular movement in any direction

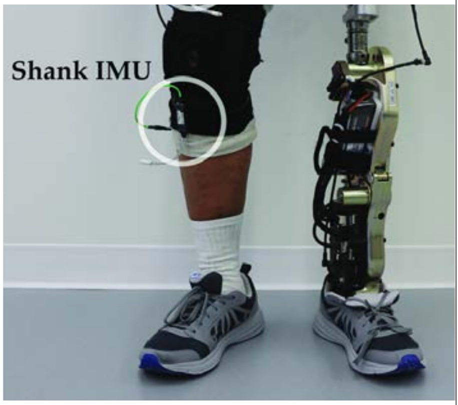
common types

- accelerometer
- gyroscope
- …but others can be used to detect motion, too

example uses

posture tracking
without cameras

fitness tracking



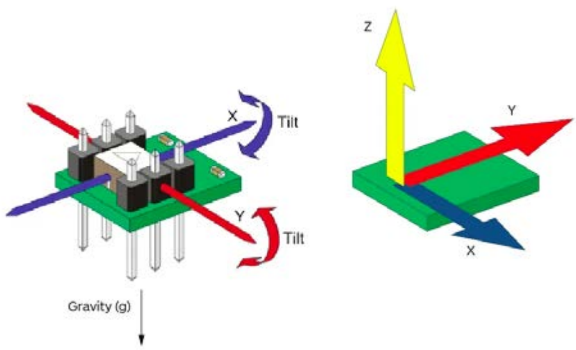
prosthetic control

inertial sensors

accelerometers

measure acceleration

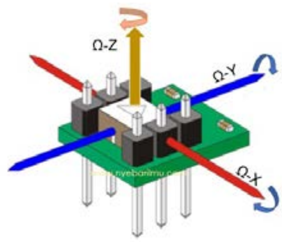
in x, y, z



gyroscopes

measure angular velocity

in yaw, pitch, and roll directions



useful for precision applications

robotics: end effector control

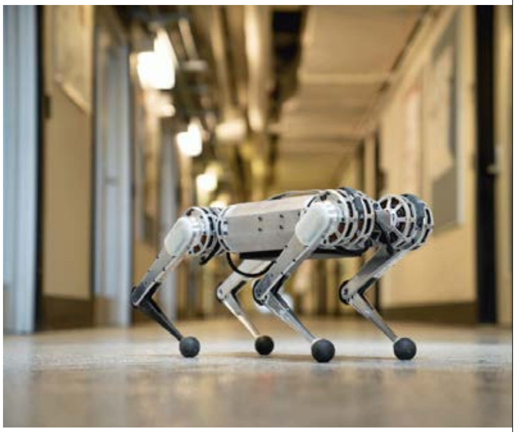
AR/VR: navigation

(space) flight

- trajectory control, avionics and subaquatic
- orientation control of satellites

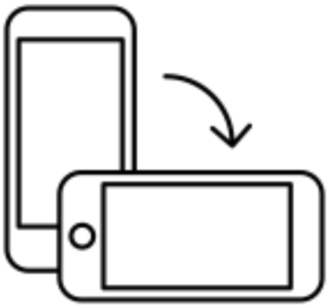
event detection

- earthquakes
- human activity

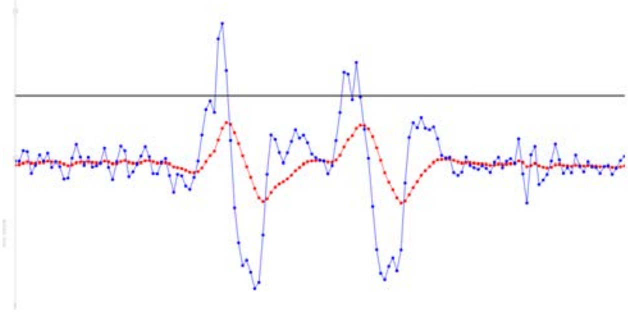


or even as simple as

phone rotation



pedometer



[https://stackoverflow.com/questions/20324356/how-to-calculate-exact-foot-step-count-using-accelerometer-in-android]

accelerometers

accelerometers

observe proper acceleration

- i.e., experienced physical acceleration relative to a free-fall
- gravitation by itself does not cause proper acceleration, since gravity acts on the inertial observer
- units: m/s^2, g

velocity (v) is speed and direction

any time there is a change in either speed or direction
➪ there is an acceleration (a)

$$v = \frac{\partial x}{\partial t} \qquad a = \frac{\partial^2 x}{\partial t^2}$$

Example	G Force
Standing on earth at sea level	1g
Bugatti Veyron from 0 to 100 km/h (2.4s)	1.55g
Space Shuttle, maximum during launch and reentry	3g
Formula 1 car, peak lateral in turns	5-6g
Death or serious injury	50g
Shock capability of mechanical Omega watches	5000g

accelerometers

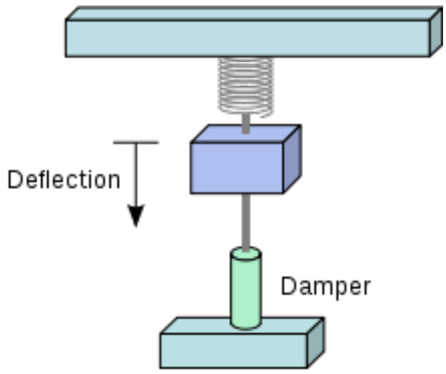
act like a spring-damper system

measure

- all external forces acting on them
- including gravity in most cases
  e.g., magnitude of 1g when stationary

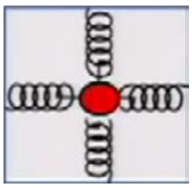
typically of interest:

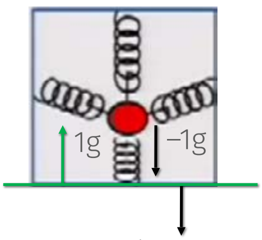
- gravity (tilt of device)
- linear acceleration (due to motion alone)
  (requires removing gravity)



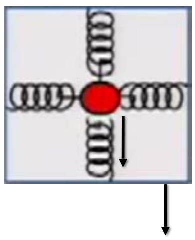
accelerometer

model: mass on spring

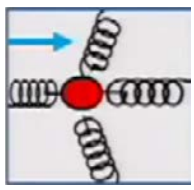




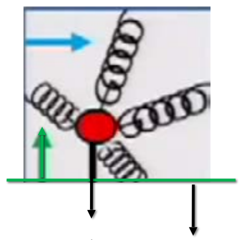
gravity
1g = 9.8m/s²



free fall

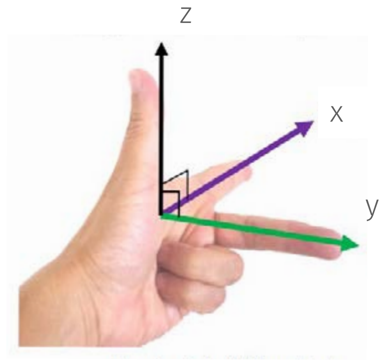


linear acceleration



linear acceleration + gravity

accelerometers

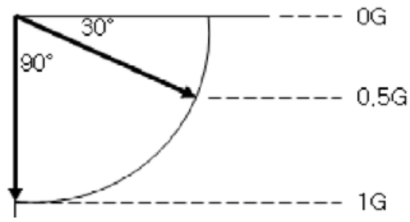


by measuring the vertical value of gravity,
tilt angle of the accelerometer can be computed

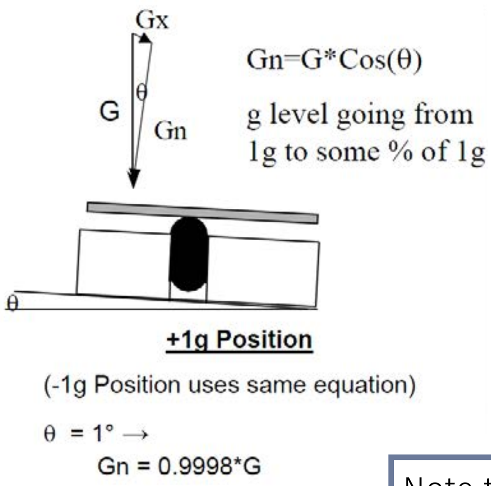
when still

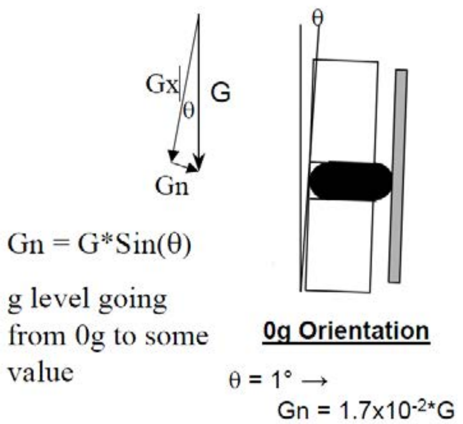
- the total magnitude of the accelerations should always be 9.81 m/s^2
- relative values of x, y, and z reveal the orientation of the accelerometer

even so, sensitive to vibration and mechanical noise



only stable near +1g or –1g positioning





Note that with a 1° error, the device is .9998/.017 = 59x more sensitive at 0g position compared to 1g position

+1g position

0g position

MEMS

Micro Electro-Mechanical Systems

MEMS sensors

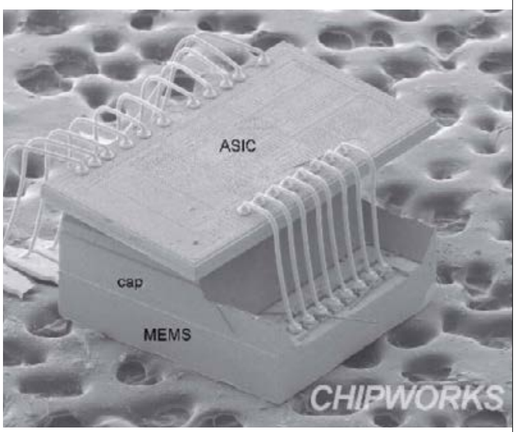
Micro Electro-Mechanical Systems

coined in 1989

creation of mechanical elements at a scale more usually reserved for microelectronics

use cavities, channels, cantilevers, membranes, … to imitate traditional mechanical systems

small enough to be integrated with electronics (micro components between $10^{-3}..10^{-1}$mm)



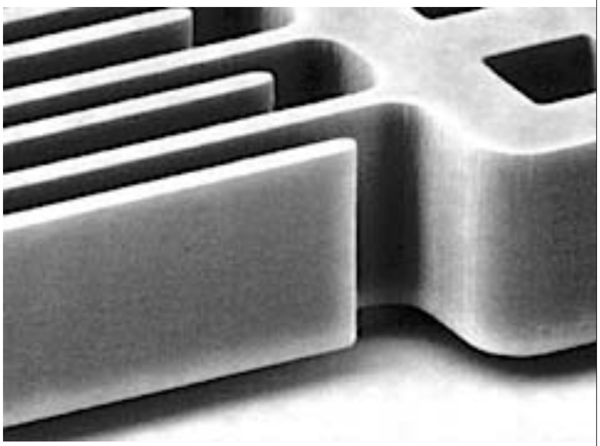
MEMS accelerometers

spring-like structure with a proof mass

damping results from residual gas

implementations

- capacitive
- piezoelectric

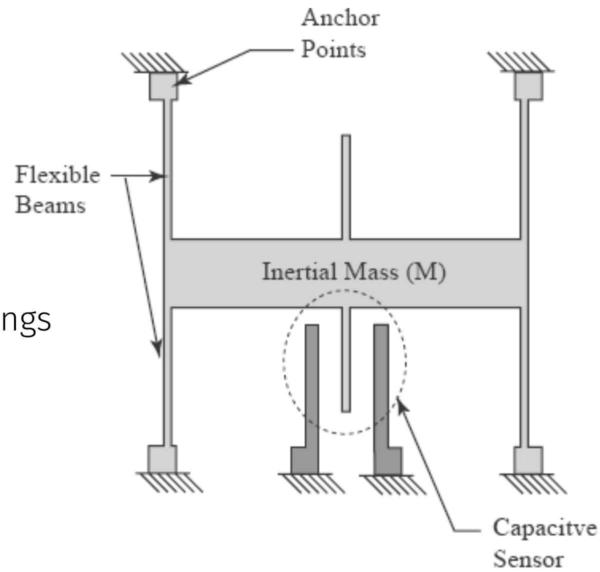


[Bernstein, J., An Overview of MEMS Inertial Sensing. Technology, Sensors 2003]

MEMS accelerometer operation I

consists of beams
and a capacitive sensor
with some anchor points

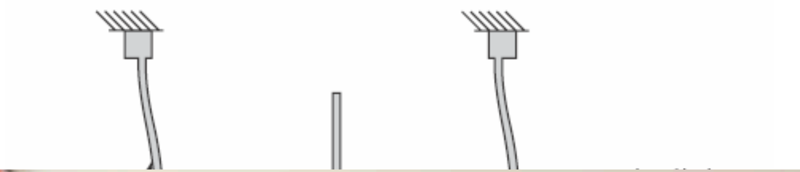
proof mass between beam springs
and a series of ‘plates’

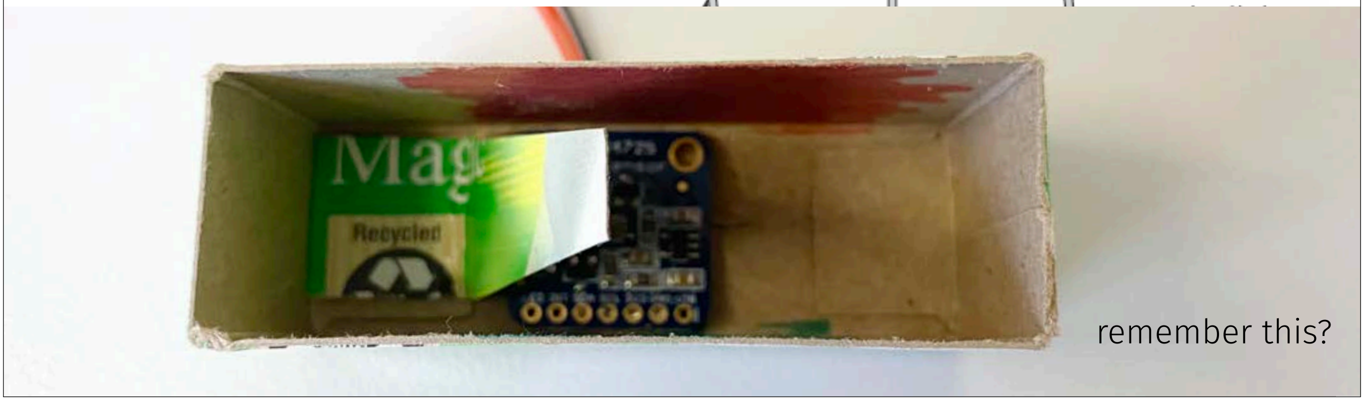


MEMS accelerometer operation II

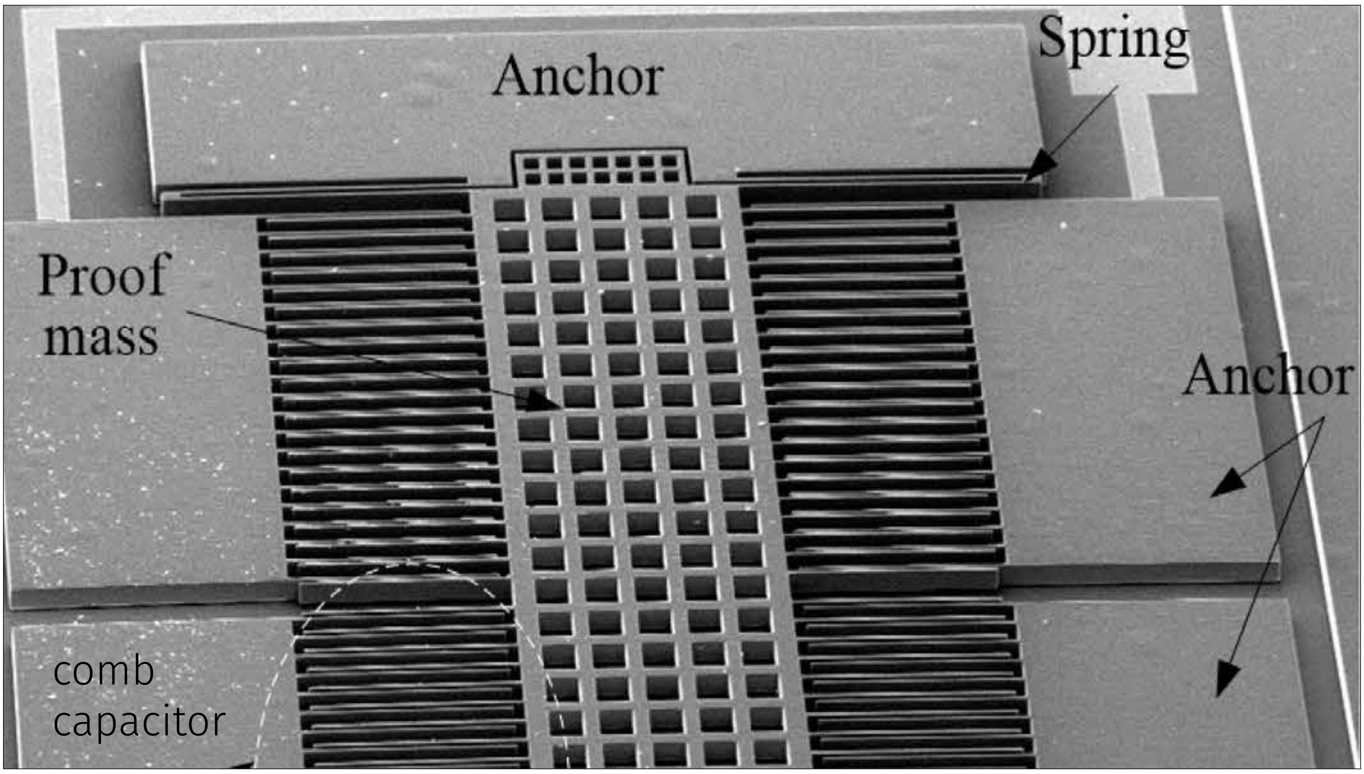
when applying acceleration

- the beams deflect





remember this?



accelerometers: dependent “sensors”

gravity “sensor”

measures the force of gravity

- along the x, y, and z axes
- unit: m/s^2

common use:

- detect tilt
- sensor motions such as free fall

gravity “sensor”

not a separate hardware component

- virtual sensor based on the accelerometer
- simply removal of real acceleration component from the reading

typically isolated through a low-pass filter

linear acceleration “sensor”

measures

- acceleration force **excluding** gravity
- along the x, y, and z axes
- unit: m/s^2

common use

- detecting activity
- tracking motion

linear acceleration “sensor”

not a separate hardware component either

- again, virtual sensor based on the accelerometer
- care about the big changes and want to avoid ‘noise’ (i.e., gravity)

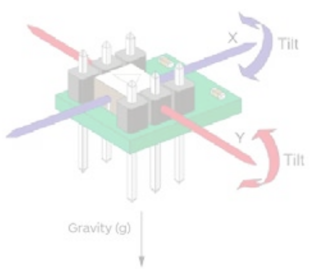
typically calculated through a high-pass filter

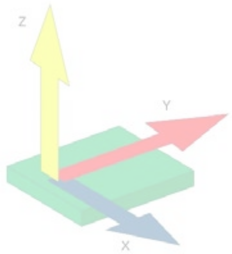
inertial sensors

accelerometers

measure acceleration

in x, y, z

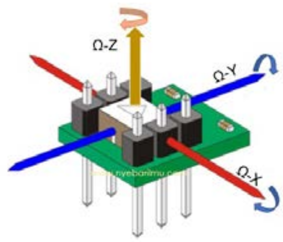




gyroscopes

measure angular velocity

in yaw, pitch, and roll directions



gyroscopes

[youtube.com/watch?v=p9zhP9Bnx-k]

gyroscopes

devices used for measuring or maintaining orientation and angular velocity

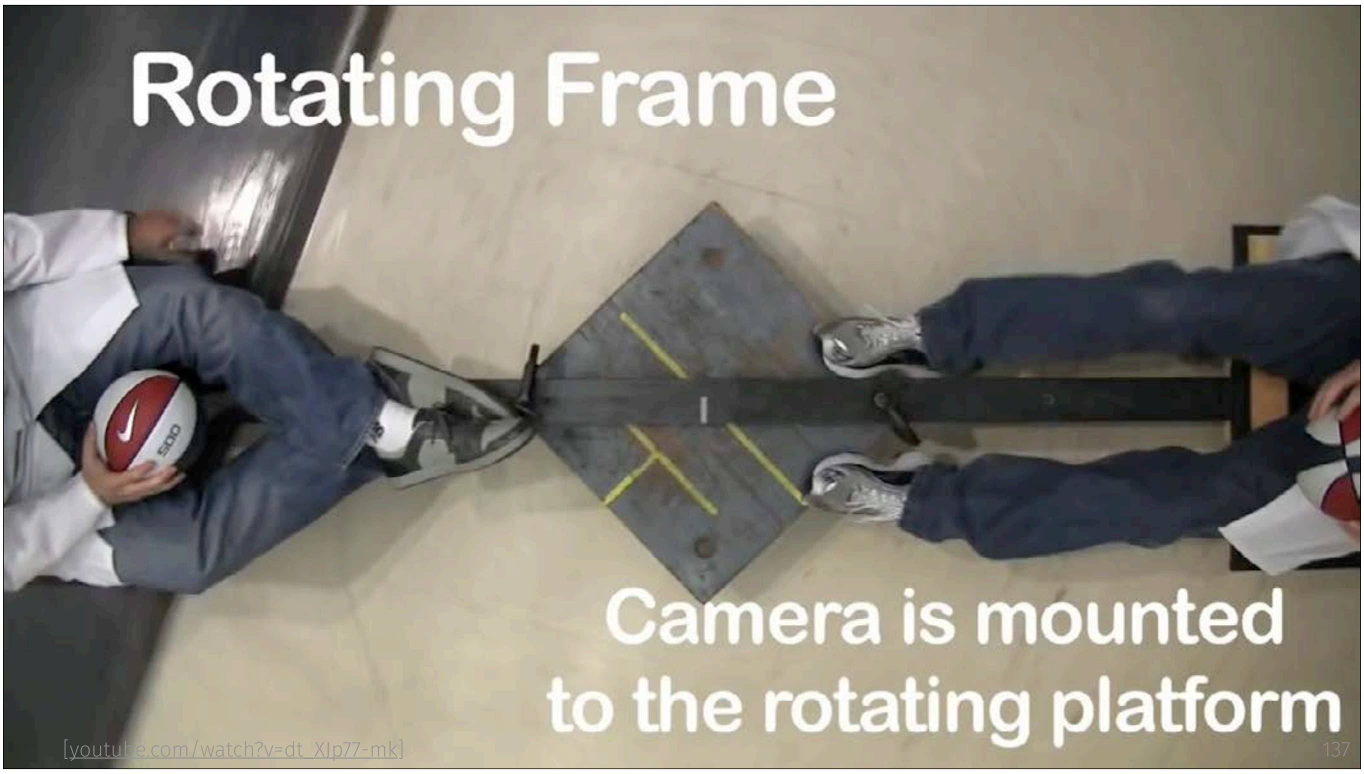
measure the rate of rotation

- typically about yaw ψ, pitch θ, and roll φ
- unit: rad/s
- with no **absolute** reference

require moving mass to leverage Coriolis effect

$$\omega = \frac{\partial \theta}{\partial t}$$

[youtube.com/watch?v=dt_XJp77-mk]

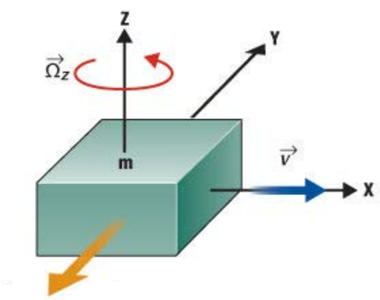


[youtube.com/watch?v=dt_XIp77-mk]

MEMS gyroscopes

based on measuring Coriolis force
as experienced by a moving object
in a rotating frame of reference

acts perpendicular to the rotation axis
and to the velocity of the body
in the rotating frame



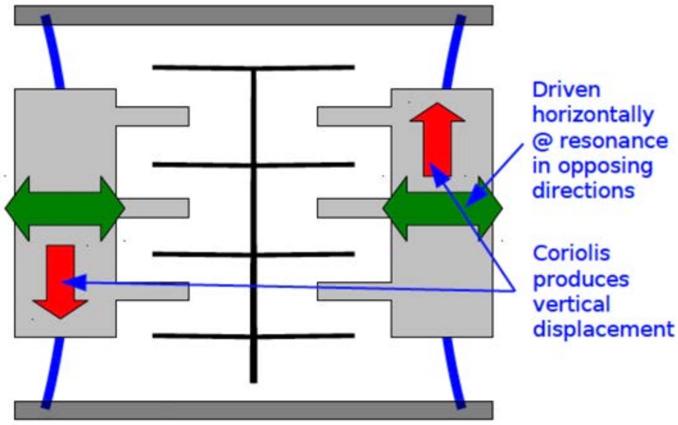
$$F_{\text{Coriolis}} = -2m(\omega \times v)$$

MEMS gyroscopes

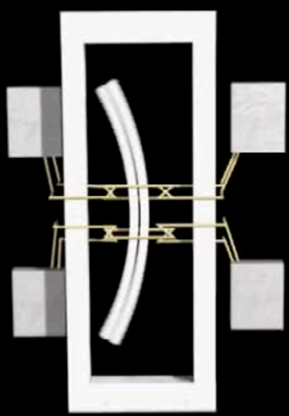
based on measuring Coriolis force
as experienced by a moving object
in a rotating frame of reference

acts perpendicular to the rotation axis
and to the velocity of the body
in the rotating frame

many implementations,
“tuning fork” is most common



➭ change in capacitance
between mass and housing



[youtube.com/watch?v=zwe6LEYF0j8]

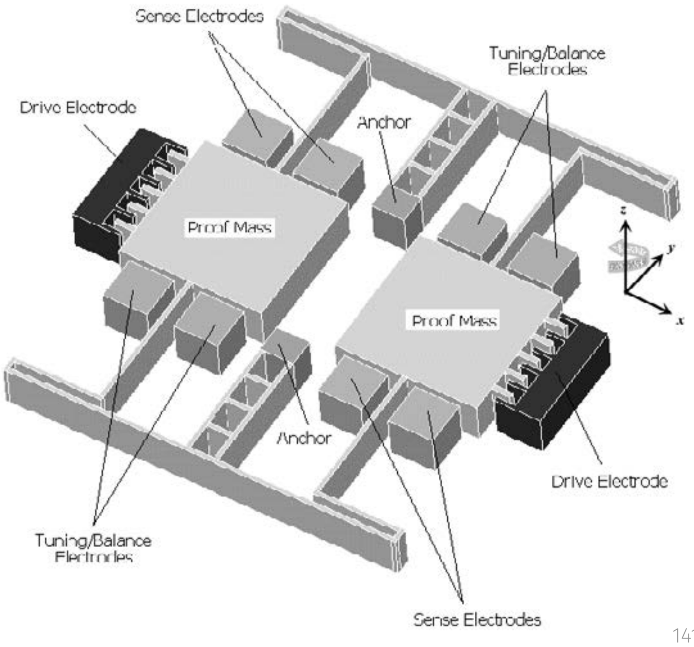
MEMS gyroscopes

based on measuring Coriolis force
as experienced by a moving object
in a rotating frame of reference

acts perpendicular to the rotation axis
and to the velocity of the body
in the rotating frame

many implementations,
“tuning fork” is most common

less sensitive to
linear mechanical movements



[Sharma et al., IEEE Sensors ‘04]

how to get to orientation?

initialize the sensor position with a known value

then measure the angular velocity (ω) around the x, y, and z axes
at measured intervals (Δt)

change in angle $= \omega \cdot \Delta t$

how to get to orientation?

initialize the sensor position with a known value

then measure the angular velocity (ω) around the x, y, and z axes
at measured intervals (Δt)

change in angle $= \omega \cdot \Delta t$

effectively, this integrates sensor readings

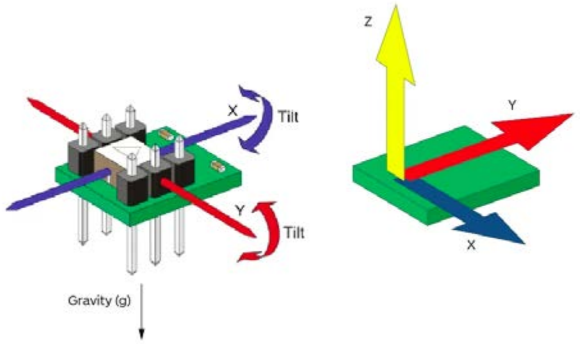
repeatedly adding increments magnifies small systematic errors over time

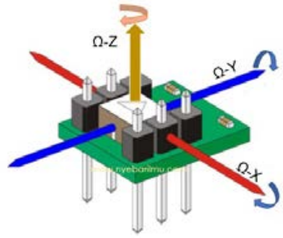
- offset errors from miscalibration
- not returning to zero-rate value on no actuation
- temperature susceptibility

gyroscopic drift is called bias (over time, gyroscope data will become inaccurate)

inertial sensors

accelerometers VS. gyroscopes





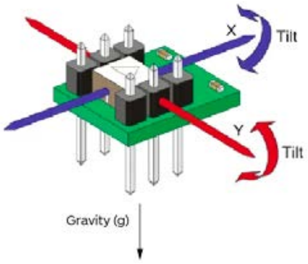
accelerometers

gyroscopes

measure acceleration

linear movement

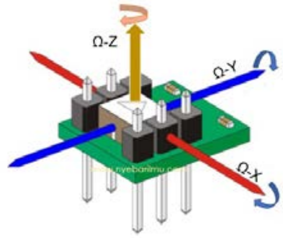
all points on a rigid body need not experience the same linear velocity



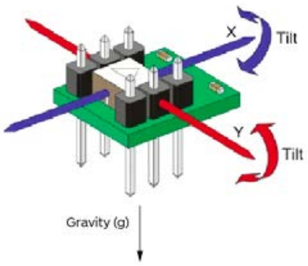
measure inertial angular velocity

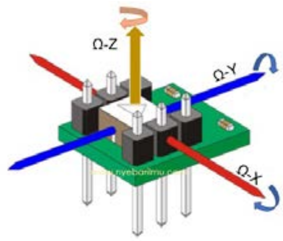
rate of change of orientation

all points on a rigid body experience the same angular velocity



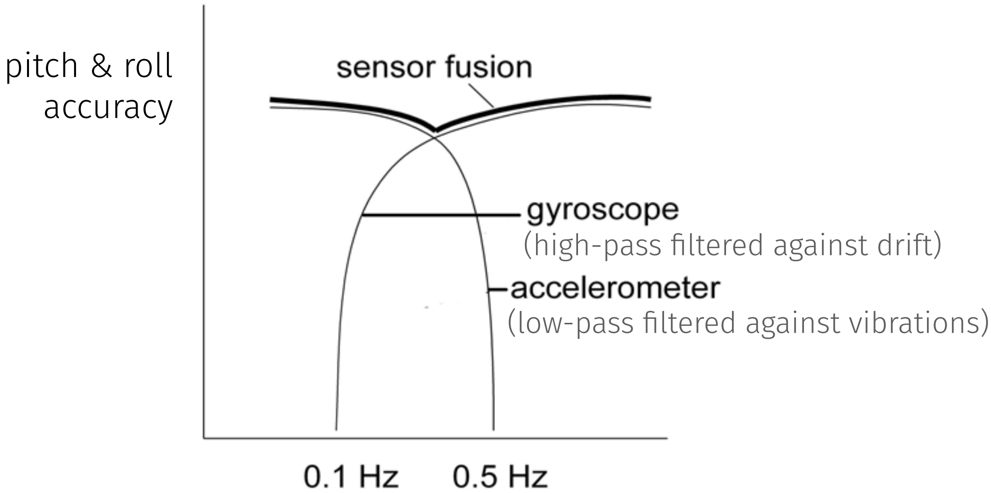
accelerometers + gyroscopes





sensor fusion: complimentary filter

accelerometers + gyroscopes



sensor fusion: complimentary filter

accelerometer and gyroscope only

- okay for tracking short term rotations
- still, small errors build up

real 3D orientation estimation needs magnetometer

➩ compensate for small drifts over long periods of time

what about position?

let's assume the orientation of the device is fixed

obtain the device position through

$$s = \int \int (a - g) dt$$

what about position?

let's assume the orientation of the device is fixed

obtain the device position through

$$s = \int \int (a - g) dt$$

error grows quadratically over time

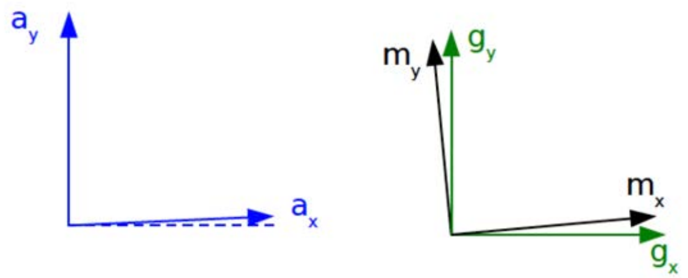
results in a fast (and unlimited) accrual of error

reference systems

sensor alignment important

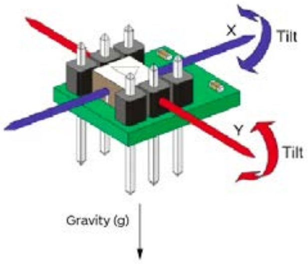
can be dangerous to assume the three sensors in a 3-D sensor are

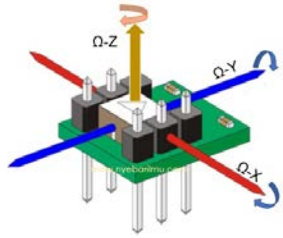
- perfectly orthogonal
- perfectly parallel to those of other sensors



putting motion sensors to use

recognizing input & activities in signals accelerometers + gyroscopes





input control & activity recognition

IMUs collect data about users and their surroundings

affords continuous input:

- tilt control
- adaptive UI support
- tracking

frequent detection-based use-cases:

- classify a user's movement (running, walking, stationary)
- detect mode of transport (foot, car, bike, bus, train, subway)

complement with other sensors (e.g., barometer, GPS, microphone for context)

input control

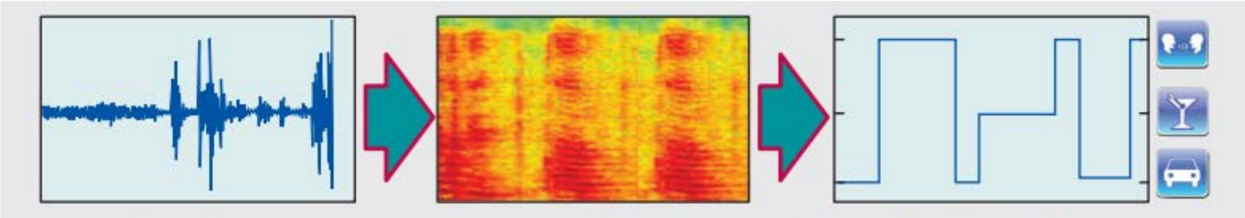
IMUs collect data about users and their surroundings

affords continuous input:

- tilt control
- adaptive UI support
- tracking

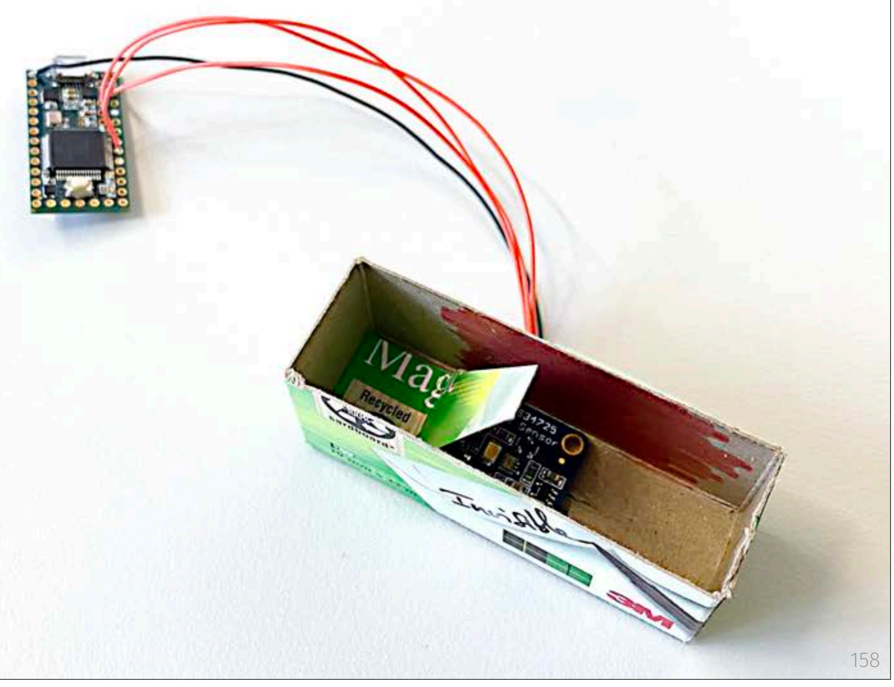
[MacBook TuxRacer 2005]

activity recognition

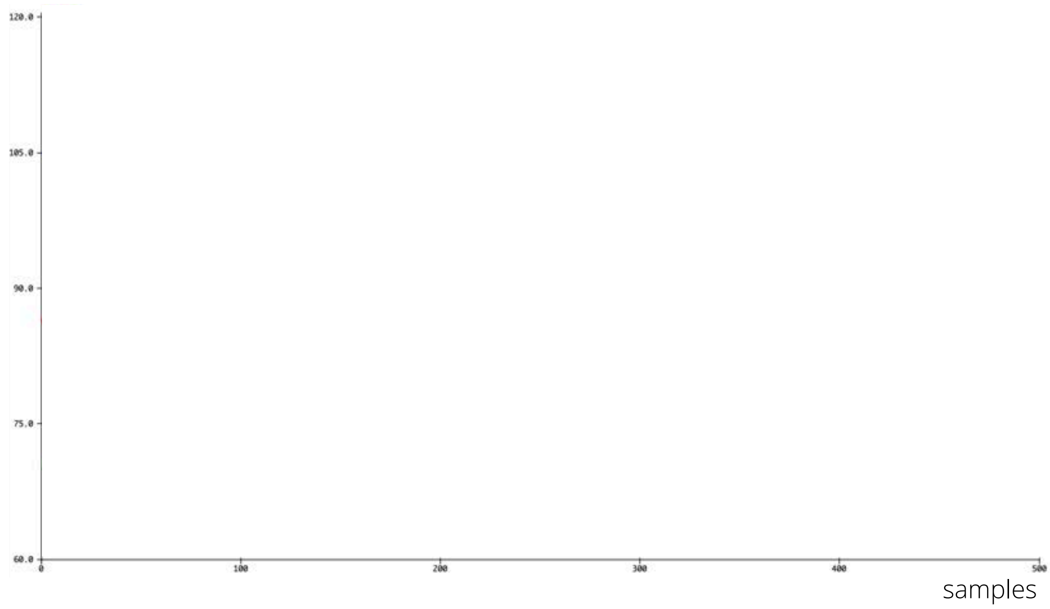


raw data extracted features classification inferences

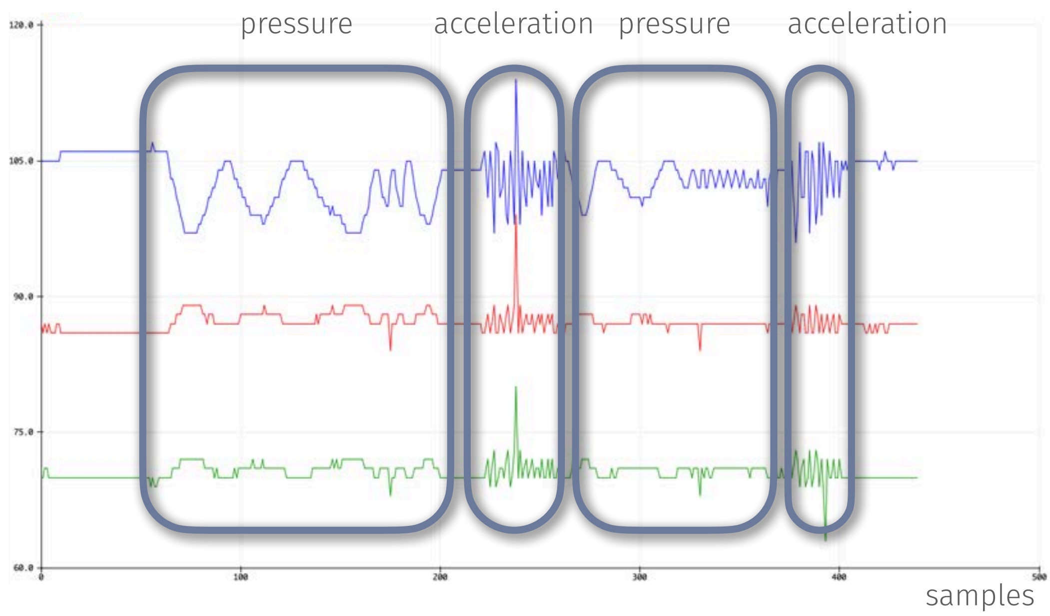
components
- microcontroller
- white light source
- color light sensor



1D plot



1D plot



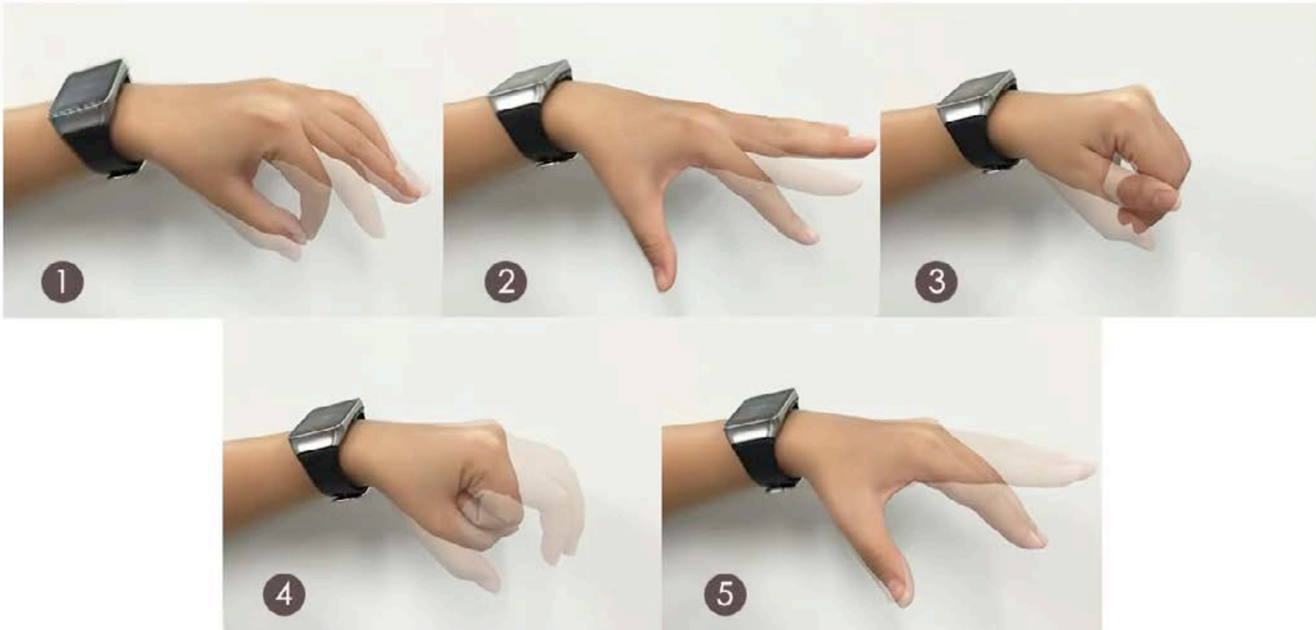
processing

```
#!/usr/bin/python
import serial
import statistics

blen = 12
allv = [0.0] * blen
allg = [0.0] * blen
zerv = -1.0
zlrn = 0.01

ser = serial.Serial('/dev/cu.usbmodem31872301')
while ser.is_open:
    l = ser.readline().decode().strip().split(',')
    r, g = int(l[0]), int(l[1])
    if zerv < 0.0:
        zerv = v[0]
    else:
        zerv = v[0] * zlrn + zerv * (1.0 - zlrn)
    v[0] = zerv - v[0]

    allv = allv[-blen + 1:] + [v[0]]
    allg = allg[-blen + 1:] + [v[1]]
    if statistics.stdev(allv) > 2.0 and statistics.stdev(allg) > 1.0:
        print("currently shaking")
    else:
        print(f"pressure mode: {max(0, v[0]):.2f}")
```

determines latency of classification

double-check if
both channels affected

[Smackbook 2005]



[Serendipity, CHI '16]

Serendipity

"To compensate for different watch orientations, we calculate the magnitude of the combined axes ($\sqrt{x^2 + y^2 + z^2}$) […]"

We calculate 7 statistical features from a 1-second sliding window:

- mean, standard deviation, max, min, 3 quantiles.
- a Fast Fourier Transform (FFT) of the window, keep the lower 10 bands […]

classifier

- supervised machine learning approach to recognize gestures
- support vector machine with best results

[Viband, UIST'16]

Viband

sensor: 3-axis accelerometer data

moving-window approach

extract features

- number of peaks
- descriptive statistics

average 20 consecutive FFT frames

- mean, sum, min, max frequencies, standard deviation
- ratios of different frequency regions

2. OBJECT RECOGNITION

Example Applications

[Viband, UIST'16]

[RecoFit, CHI'14]

activity recognition: challenges

most methods are supervised:

- need to cover variety of events
- ground-truth annotation

lab vs. in-the-wild outside controlled conditions

example modalities

common simple modalities

- button
- touch
- shake?

explicit

more complex modalities

- vision (e.g., for gesture, pose, activity)
- speech
- motion and activity

implicit

next time

signals and sampling

analog, digital

frequencies

end

prof. christian holz
mobile health & activity monitoring, spring 2023

sensing, interaction & perception lab

further reading I

Nobuyuki Matsushita and Jun Rekimoto. **HoloWall: designing a finger, hand, body, and object sensitive wall.** In *Proceedings of the 10th annual ACM symposium on User interface software and technology* (UIST ’97), 209–210. DOI: https://doi.org/10.1145/263407.263549

Mark Weiser. **The computer for the 21st century.** In *SIGMOBILE Mob. Comput. Commun.* Rev. 3, 3 (July 1999), 3–11. DOI:https://doi.org/10.1145/329124.329126

further reading II

Hui-Shyong Yeo, Juyoung Lee, Hyung-il Kim, Aakar Gupta, Andrea Bianchi, Daniel Vogel, Hideki Koike, Woontack Woo, and Aaron Quigley. **WRIST: Watch-Ring Interaction and Sensing Technique for Wrist Gestures and Macro-Micro Pointing.** In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services* (MobileHCI ’19), Article 19, 1–15. DOI:https://doi.org/10.1145/3338286.3340130

Hongyi Wen, Julian Ramos Rojas, and Anind K. Dey. **Serendipity: Finger Gesture Recognition using an Off-the-Shelf Smartwatch.** In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (CHI '16), 3847–3851. DOI:https://doi.org/10.1145/2858036.2858466

further reading III

Gierad Laput, Robert Xiao, and Chris Harrison. **ViBand: High-Fidelity Bio-Acoustic Sensing Using Commodity Smartwatch Accelerometers.** In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (UIST '16), 321–333. DOI:https://doi.org/10.1145/2984511.2984582

Dan Morris, T. Scott Saponas, Andrew Guillory, and Ilya Kelner. **RecoFit: using a wearable sensor to find, recognize, and count repetitive exercises.** In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14), 3225–3234. DOI:https://doi.org/10.1145/2556288.2557116