Liteturn: Automated, Gesture-Controlled Cyclist Turn Lights Using Cheap and Efficient Consumer Devices

Kevin Yeh

CS 386W: Fall 2014
Wireless Networking

What's Wrong with Hand Signals?

- Visibility: At wide angles and in low-lit conditions, hand signals are not visible to motorists and other cyclists.
- Mobility: Hand signals are impractical and unsafe when both hands need to be on the handlebars.
- Utility: Narrow use; no physical mechanism for indicating acceleration/deceleration or hazard status to motorists behind you.

Current Solutions

- DORA, LumaHelm: LED-embedded helmets

Bulky, requires handlebar-mounted devices
Conceptual products with no practical implementations

- Zackees: Embedded gloves with metal rivets

Requires physical, prolonged metal rivet connection to enable lights
Fails to improve visibility when hands are on handlebars

- MIT Labs Research: Sensor-embedded jackets
- HelSTAR Helmets: Hooks into motorcycle-based navigation systems

Current Solutions

All of these products have major accessibility issues!

Requires novel accessory clothing to be worn/washed by user

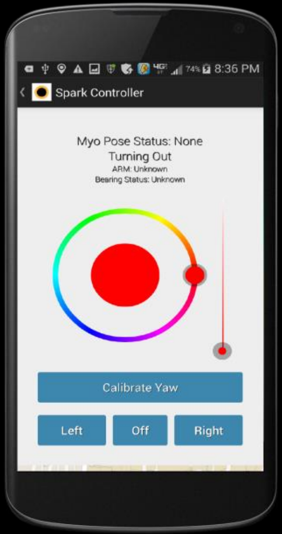
Makes use of additional controls embedded in clothing or attached to handlebars

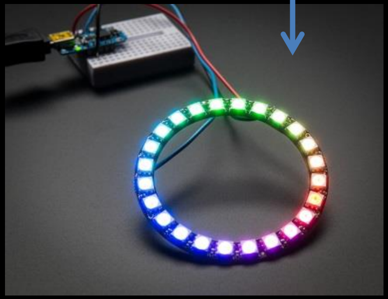
High price point! Gloves $80, Helmets $180+

The Liteturn Solution

Use commonplace smart devices and a small, wireless-enabled LED package to provide hands-free control and automation of turn lights.

The Liteturn Solution





The Liteturn Solution

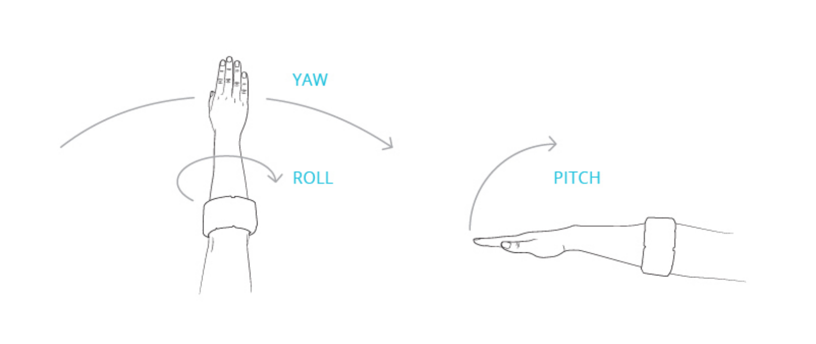
- Visibility: Central lights allow cyclists to signal their intentions at all angles, even in low-lit conditions.
- Mobility: Signal lights are sustained for the duration of the turn and can be enabled at any point using a brief, natural, and universally recognized gesture.
- Utility: Can be extended to replace head/tail lights, indicate braking and rolling, and provide customized lighting animations in neutral states.

System Overview

- Gesture Recognition
- End of Turn Detection
- Braking / Rolling Signals
- LED Communication
- Practical Implementation

Gesture Recognition

- Extract yaw-pitch-roll arm orientations using accelerometer and gyroscope data from arm/wrist wearables
- Reduce to relative coordinates using GPS location and temporal bearing calculations
- Discretize readings into segmentations of 360 degree space.



Gesture Recognition

YPR World Coordinates (0, 2pi)

Yaw Correction using GPS location bearing calculations

YPR Relative Coordinates (0, 2pi)

Discretization of degree space into k segments

YPR Discretized Coordinates [0, k]

Gesture Recognition

Detect valid YPR readings within window *w* of perfect gesture.
Force gesture hold delay of *d* ms to reduce false positives.

- Yaw: Arm rotation around body
  - Valid Turn Gesture: Arm is out to the side
- Roll: Device rotation around wrist
  - Valid Turn Gesture: Device screen facing back or up (for outer turn)
- Pitch: Arm rotation in vertical space
  - Valid Turn Gesture: Arm straight (outer turn) or bent 90° (inner turn)

End of Turn Detection

- Request GPS updates every second at a minimum 10 meter travel distance
- Keep a short running history of GPS bearings
- Ending Criteria: New bearing calculation roughly 90°+ from any recent bearing
- On turn end, clear bearing history.

End of Turn Detection

Good: location minimum prevents small position changes from triggering false positives.

- Safe to wait out a turn using localized movement.

Problems?

- Lane change detection
  - Use accelerometer for more minute actions
- Gentle turns, e.g. forks
  - Increase bearings tolerance
  - Add linearly increasing delay before turn end recognized
    - Sharp turns recognized immediately, while small turns will take longer to be detected to avoid canceling upcoming turns off of curved roads.

Braking / Rolling Signals

- Extract raw accelerometer data
- Grab vector component along traveling direction using last known GPS bearing
- Interpolate LED color and send to microcontroller over BLE or WiFi in intervals

Can use smaller intervals over BLE due to lower latency over direct paired connection.

Practical Implementation

HackTX 2014 Implementation

- Thalmic Myo Armband – Can use EMG sensors to detect hand poses (e.g. CLOSED_FIST) and reduce false positives.
- Android Companion App with naive tracking
- Spark Core (WiFi-enabled microcontroller)
- 24 RGB-LED NeoPixel Ring

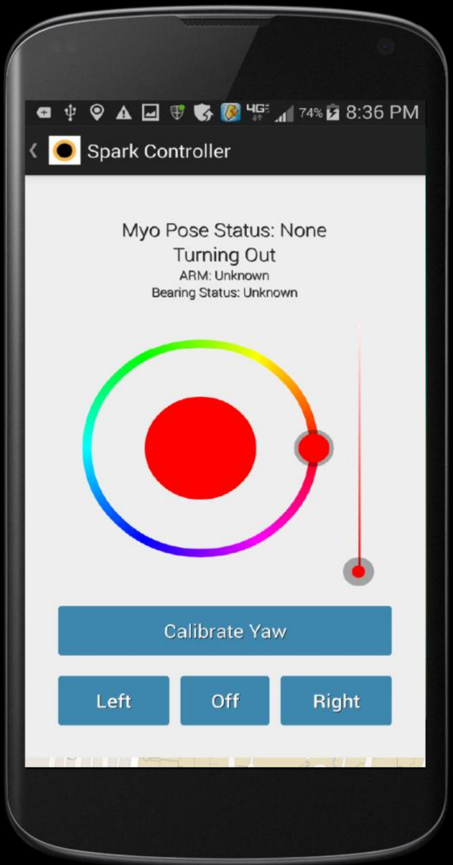
Practical Implementation

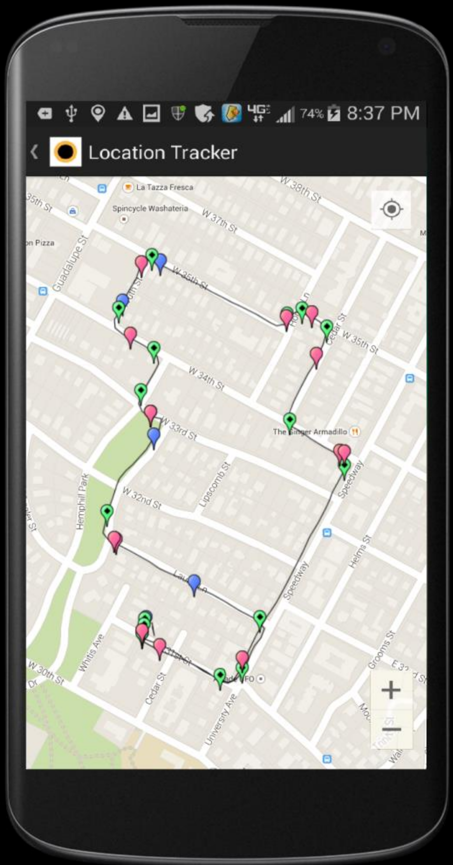
Current Implementation

- Samsung Galaxy S3 + Glove + Rubberband
  - “Galaxy Watch”
- Android Companion App with GPS tracking
- Spark Core
- 24 RGB-LED NeoPixel Ring

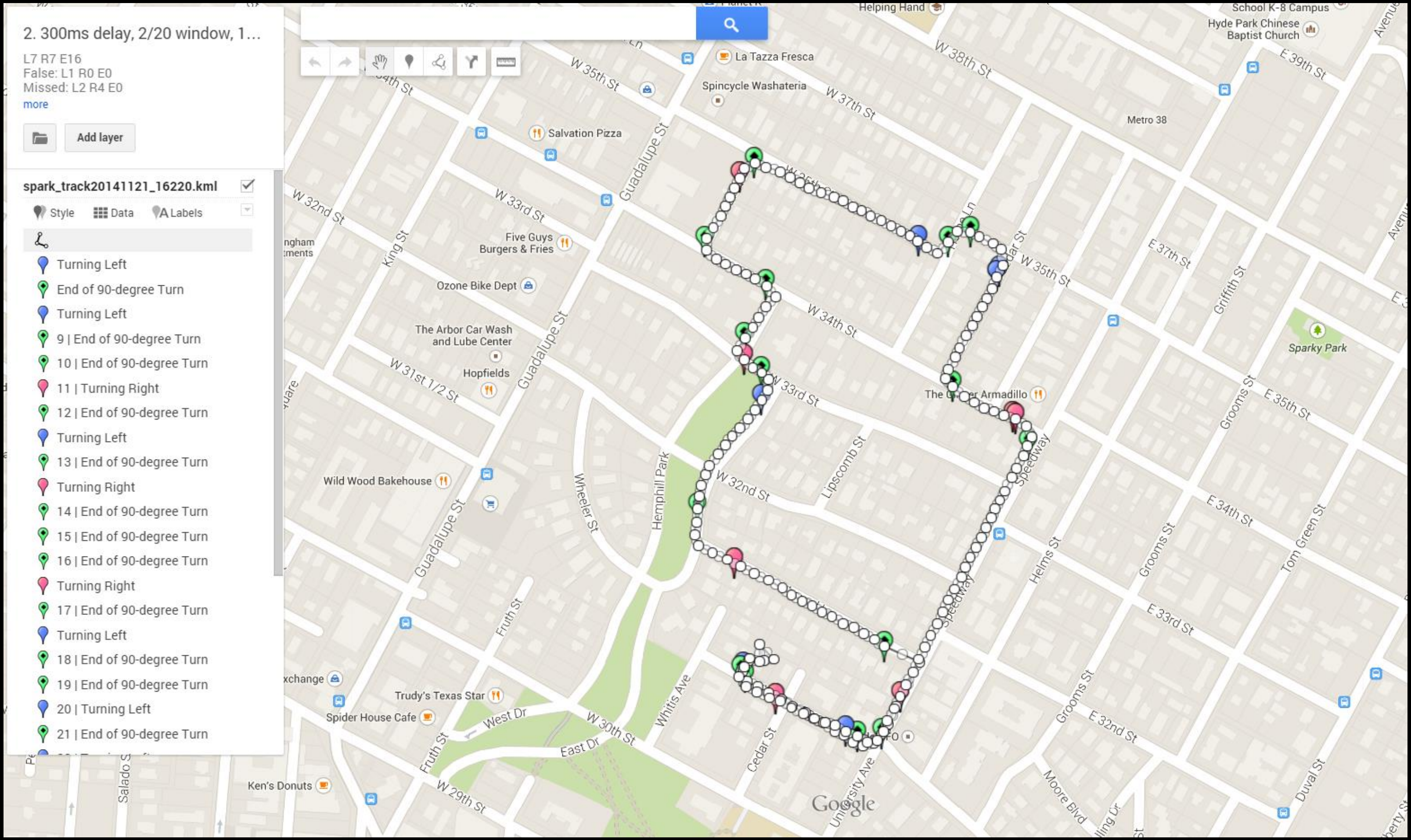
Practical Implementation

Companion App





Testbed

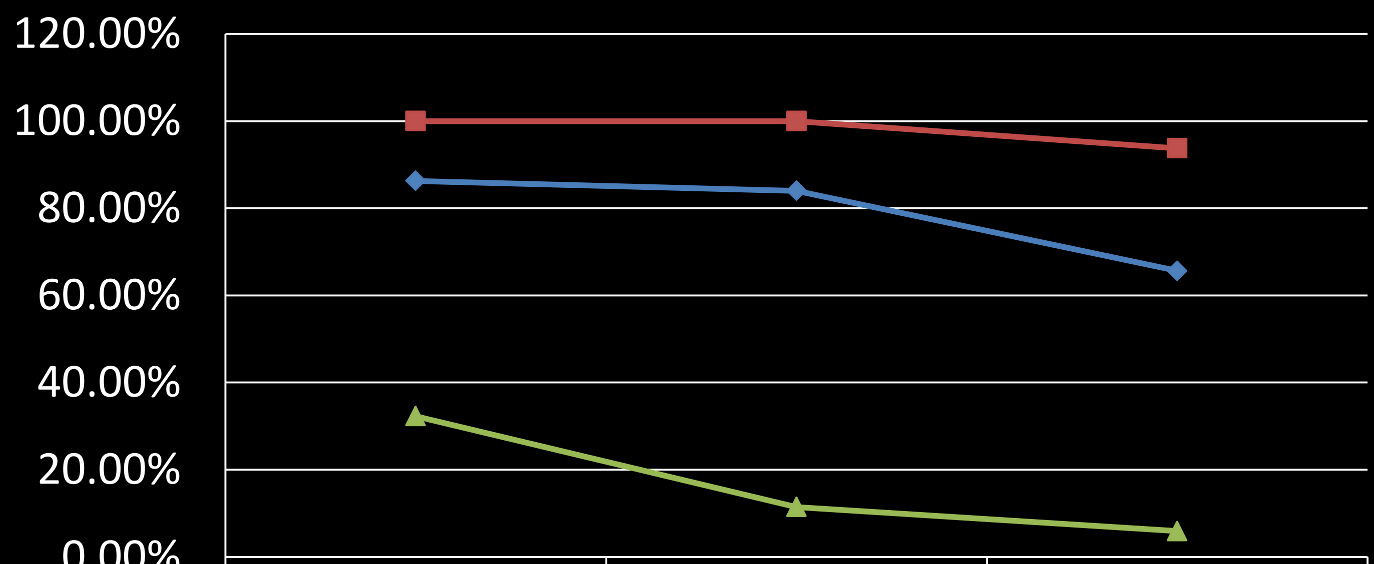

2. 300ms delay, 2/20 window, 1...
L7 R7 E16
False: L1 R0 E0
Missed: L2 R4 E0
more
Add layer
spark_track20141121_16220.kml
Style
Data
Labels
Turning Left
End of 90-degree Turn
Turning Left
9 | End of 90-degree Turn
10 | End of 90-degree Turn
11 | Turning Right
12 | End of 90-degree Turn
Turning Left
13 | End of 90-degree Turn
Turning Right
14 | End of 90-degree Turn
15 | End of 90-degree Turn
16 | End of 90-degree Turn
Turning Right
17 | End of 90-degree Turn
Turning Left
18 | End of 90-degree Turn
19 | End of 90-degree Turn
20 | Turning Left
21 | End of 90-degree Turn

Testbed

- 16 turns in North Campus area
  - Flats + Steep Uphill/Downhill
  - Potholes/Bumps
  - Short/Long Stretches
  - Rain/No rain
- Vary Hold Delay, Gesture Angle Windows, Location Polling Intervals
- 3 trials per test

Evaluation and Results

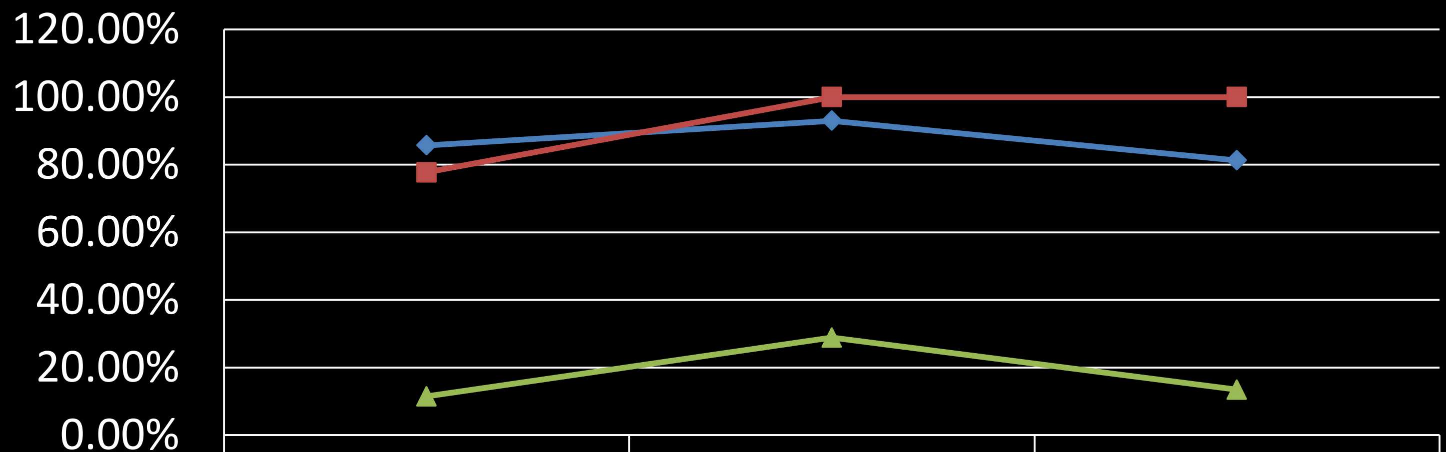
Gesture Hold Delay



	0ms	150ms	300ms
Turn Recognition Accuracy	86.25%	84%	65.63%
End Recognition Accuracy	100.00%	100%	93.75%
False Positive Rate	32.29%	11.45%	5.88%

Evaluation and Results

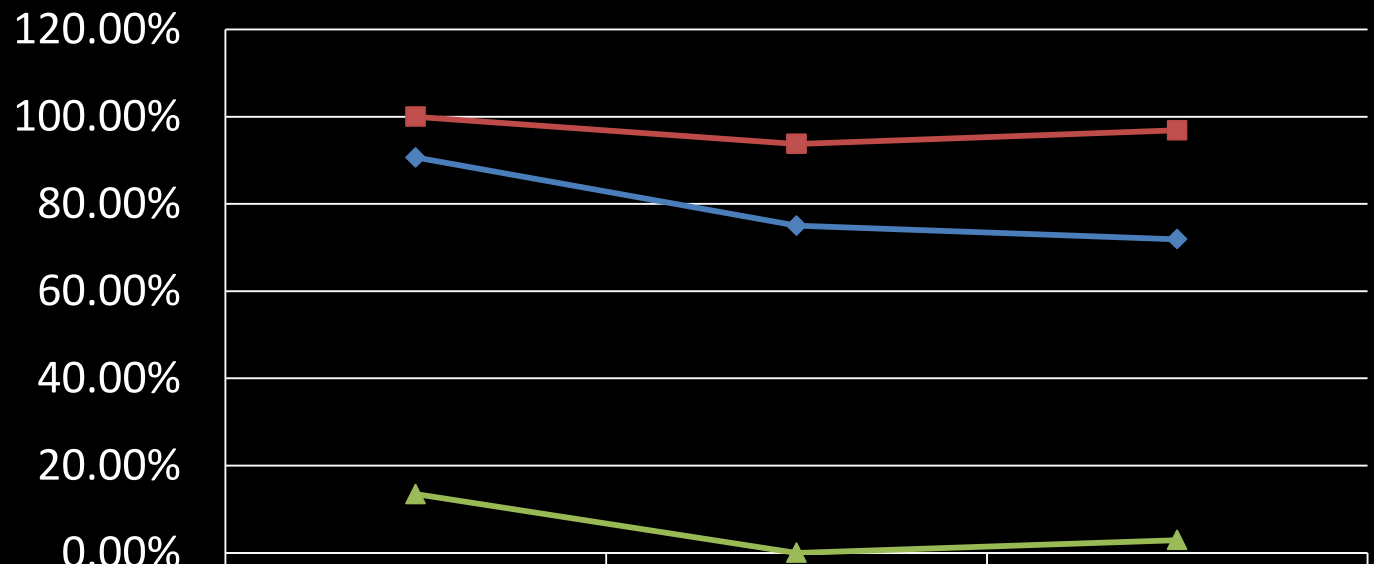
Gesture Recognition Window : YPR Divisions



	2:20	3:20	5:40
Left Turn Recognition Accuracy	85.70%	93%	81.25%
Right Turn Recognition Accuracy	78%	100%	100%
False Positive Rate	11.44%	28.85%	13.45%

Evaluation and Results

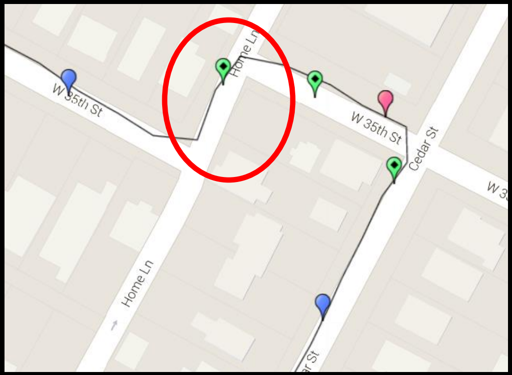
Location Polling Interval (Minimum time/dist)



	1s/10m	1.5s/15m	2s/20m
Turn Recognition Accuracy	90.63%	75%	71.88%
End Recognition Accuracy	100%	94%	96.88%
False Positive Rate	13.45%	0.00%	2.94%

Evaluation and Results

- False Positives
  - With some amount of hold delay, false positives limited to one per trial
    - Common irrelevant actions allowed by delay and YPR windows (e.g. hand repositioning, riding one-handed/hands-free, resting hand on top tube)
    - Steep hills alter pitch readings and can set off turn recognition
- Missed Turns
  - Turns in quick succession run ahead of location polling
  - Gesture recognition module adjusts using outdated bearings and misses turn



Implementation Revisited

Ideal Implementation

- Generic smartwatch or other wrist wearable
- Android or iOS Companion App with GPS
- “Dumb” $1 micro-controller with BLE module
- Generic, bright LEDs

➡ $10-20 System using GPS and BLE

12/1 – 12/5

- Extensive Testing
  - Larger sample sizes
  - Smaller parameter variations
  - Battery Drain Testing
  - Different test individuals
  - GPS location noise in denser urban areas
- Hardware loop timing organization for responsive swapping between neutral state, turn signals, and acceleration indicators.
- Full Paper

Future Work

- Pitch correction using elevation information from Google Maps or gyroscope
- Further work in detecting turn completion for lane changes and slight turns
  - Accelerometer, Google Maps road inference, Adaptive timeout
- Reduce costs, energy use, and form factor size to achieve ideal implementation
- Can be extended with distance sensors and video processing of lanes to further automate and improve road awareness

Conclusion

- Fully hands-free and automated. No need for additional clothing or control accessories.
- Replace your head and tail lights for an all-in-one solution that is robust and customizable.
- Cheap, lightweight, and energy-efficient.