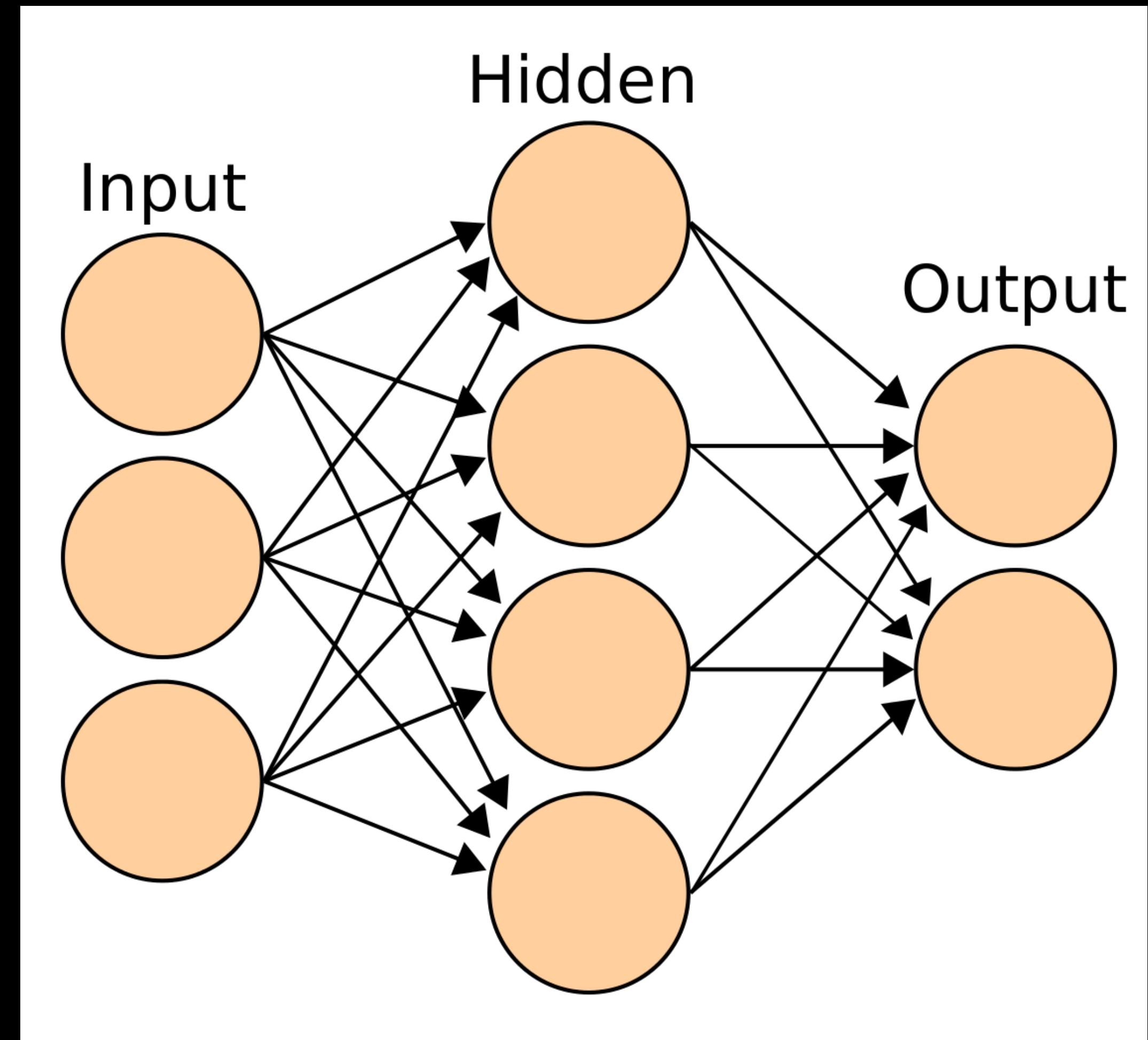


# Extending on both ends: Neural Nets and PCB manufacturing

David Goedicke & Nik Martelaro

# What is a neural net?

- A network of connected “neurons”
- Each neuron is activated or deactivated based on an input
- Kinda like the brain, but not really
- Baked in representation



# Running on single-board systems?

## Pros

- Increased privacy - not sending data over the network
- No need to pay for API usage or cloud compute
- No need to debug networking issues
- (sorta) Real-time analysis

# Running on single-board systems?

## Pros

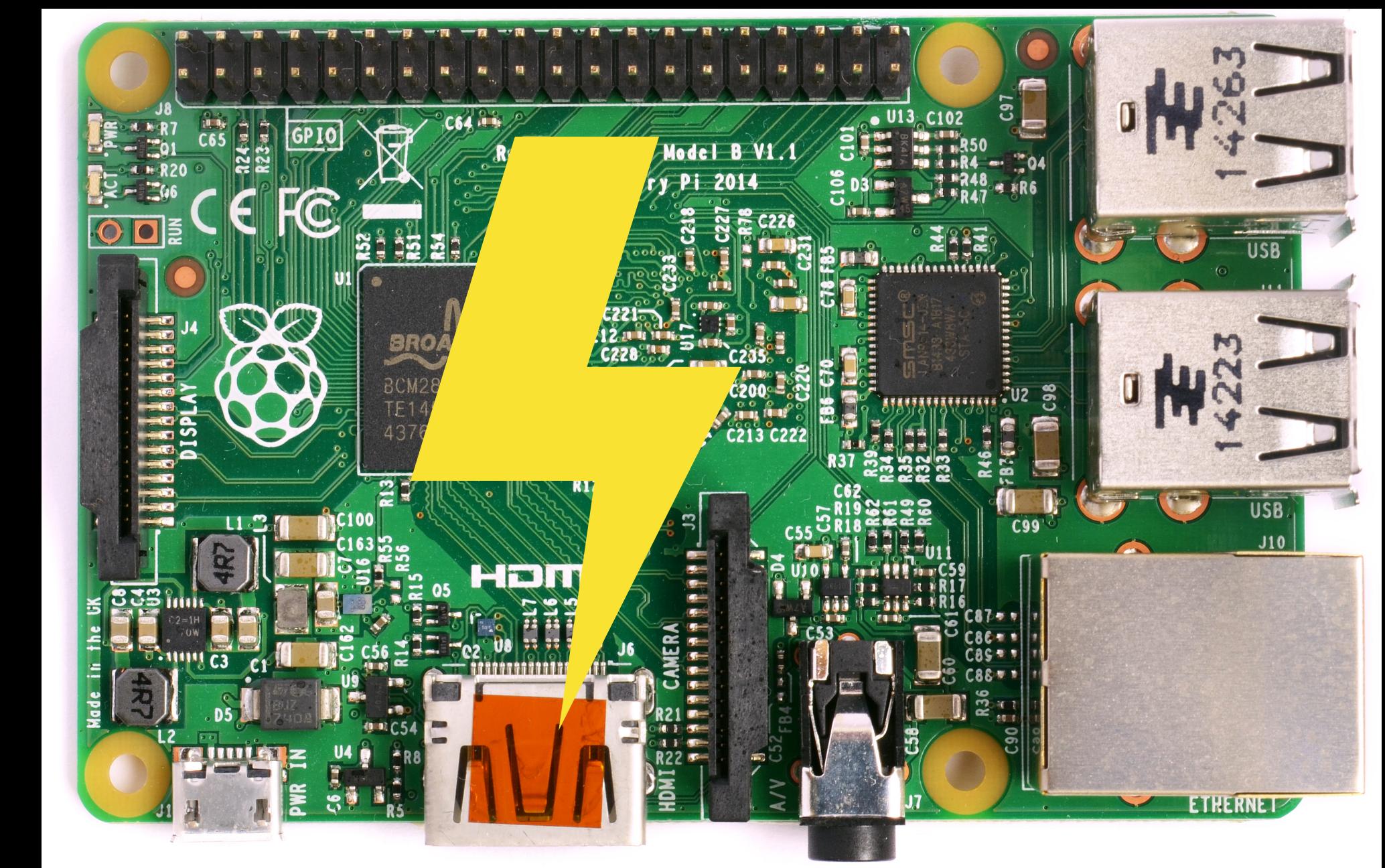
- Increased privacy - not sending data over the network
- No need to pay for API usage or cloud compute
- No need to debug networking issues
- (sorta) Real-time analysis

## Cons

- Slow (0.5 FPS on RPi 3B)
- More complicated application code
- More tooling and plumbing setup

# But is it really that slow?

- Depends on your needs
- CPU optimizations help
  - NEON (ARM)
  - VFPv3 (ARM)
- RaspberryPi 4
- Google Coral (TPU)



<https://www.pyimagesearch.com/2017/10/09/optimizing-opencv-on-the-raspberry-pi/>

Test Case:  
*Single Shot Multibox Detection (SSD)*  
*Using MobileNET-SSD*

# Test Systems



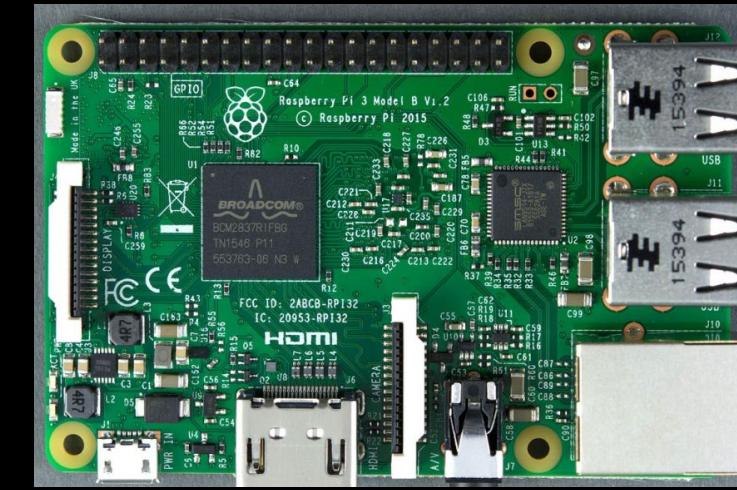
**Mac Mini**

2.6 GHz Dual-Core i5

8 GB RAM

MacOS

\$699



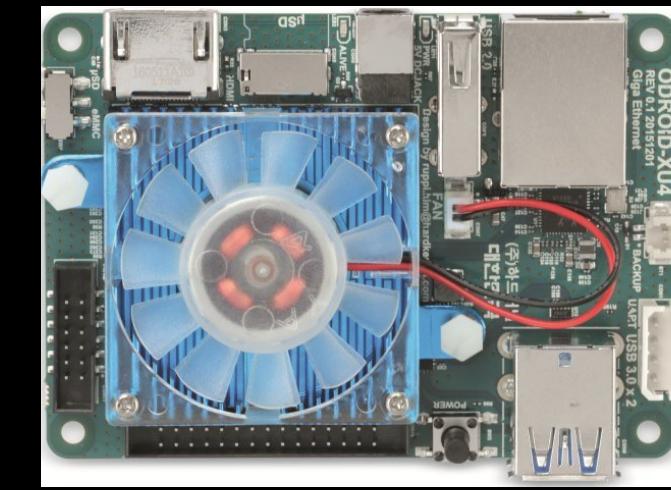
**Raspberry Pi 3**

1.2 GHz Quad Core ARM

1 GB RAM

Ubuntu Mate

\$35



**ODROID-XU4**

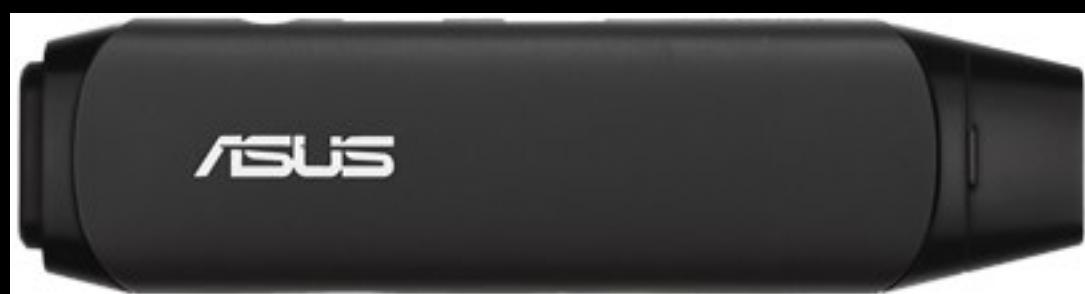
2.0 GHz Quad Core ARM (A15)

1.4 GHz Quad Core ARM (A7)

2 GB RAM

Ubuntu Mate

\$62



**Asus VivoStick**

1.4 GHz Quad Core Intel Z8350

2 GB RAM

Windows 10

\$116



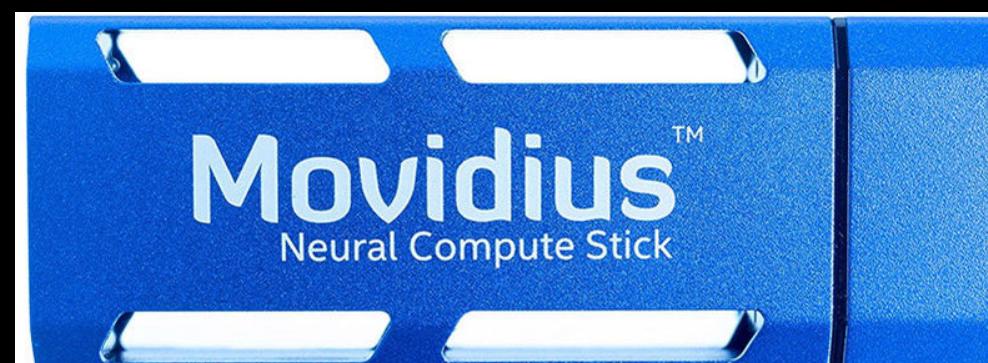
**LattePanda**

1.4 GHz Quad Core Intel Z8350

2 GB RAM

Ubuntu

\$89



**Intel Movidius Neural Compute Stick**

Movidius Vision Processing Unit (VPU)

Co-Processor tested on RPi 3

w/Optimized OpenCV

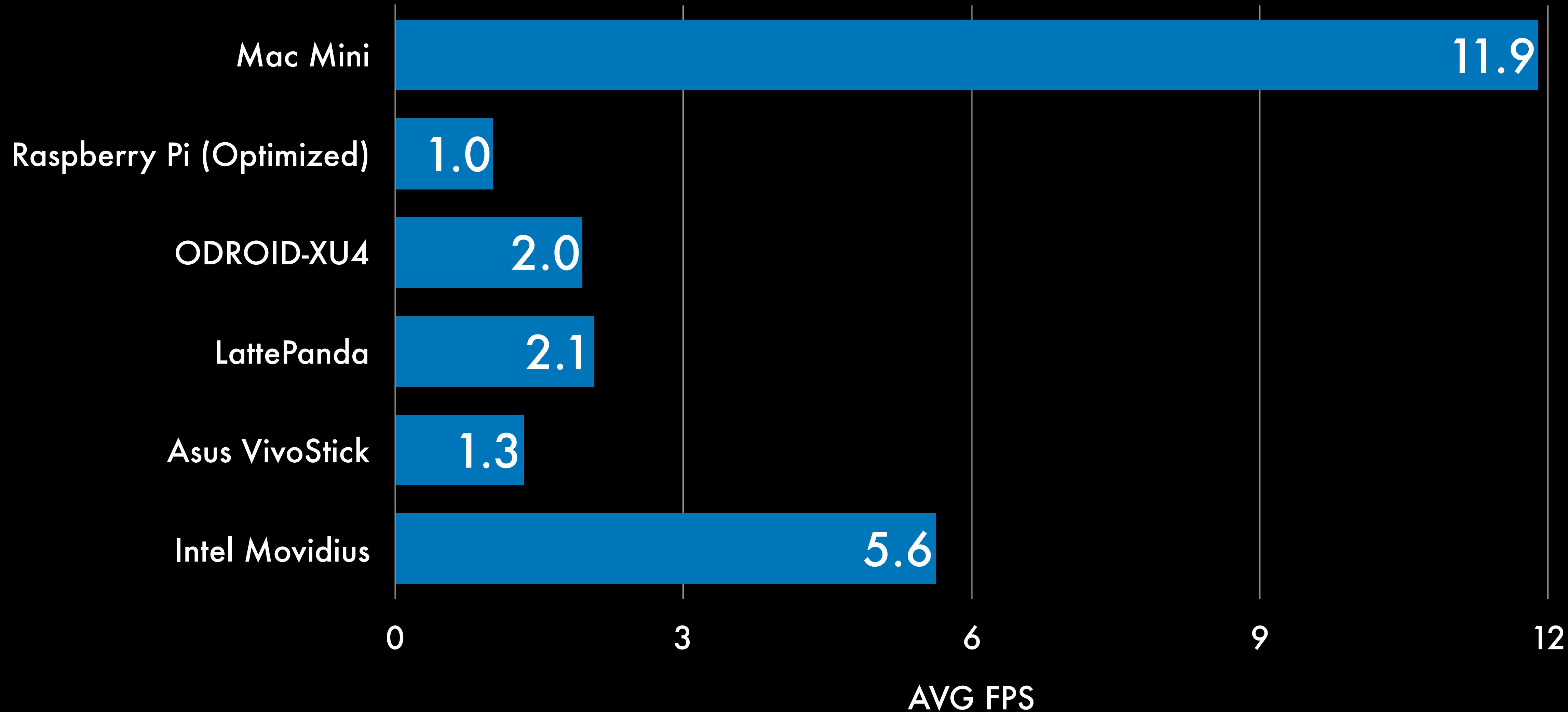
\$79 + \$35 RPi3

# Procedure

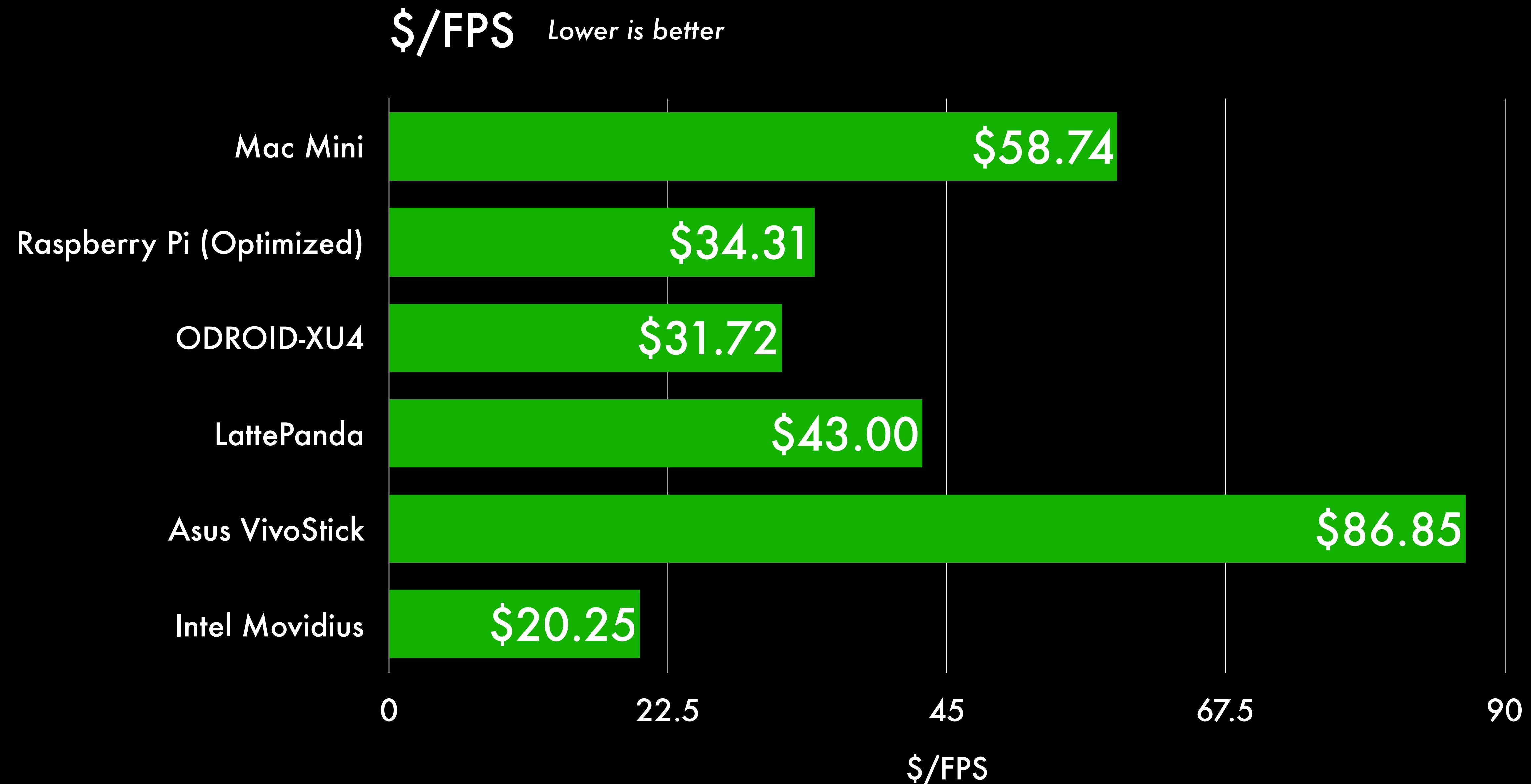
- Install optimized OpenCV
- Run real-time MobileNET-SSD sample for 60 seconds, 10 times
- Measure FPS, compute average

# Results

MobileNET-SSD Real-time FPS    *Higher is better*

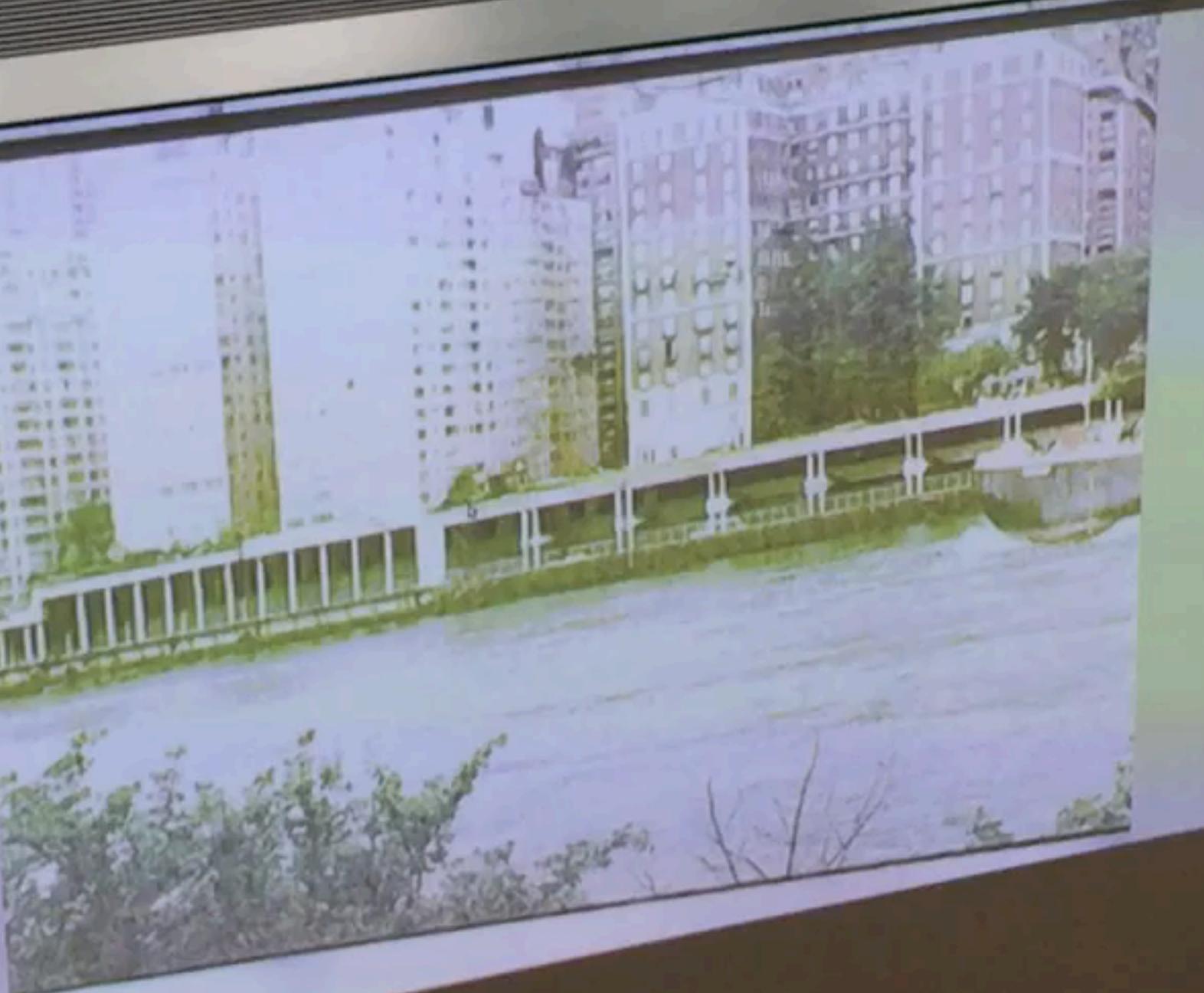


# Results



# Boat Detector





# Sample Code: PyImageSearch

- Amazing free tutorials on computer vision with Python by Dr. Adrian Rosebrock
- Lots of examples with the Raspberry Pi

The screenshot shows the PyImageSearch website with a dark theme. The header features the logo 'pyimagesearch' with the tagline 'be awesome at building image search engines'. The navigation bar includes links for 'Start Here', 'Practical Python and OpenCV', 'PyImageSearch Gurus', 'PyImageConf', 'OpenCV 3 Tutorials', 'FREE OpenCV Course', 'About', and 'Contact'. Below the header, there's a search bar and a 'Resource Guide (it's totally free)' section with a PDF download offer. Three tutorial articles are listed:

- Simple object tracking with OpenCV** by Adrian Rosebrock on July 23, 2018 in Object Tracking, Tutorials. The snippet: Today's tutorial kicks off a new series of blog posts on object tracking, arguably one of the most requested topics here on PyImageSearch. Object tracking is the process of: Taking an initial set of object detections (such as an input set of bounding box coordinates) Creating a unique ID for each of the initial detections [...].
- OpenCV Tutorial: A Guide to Learn OpenCV** by Adrian Rosebrock on July 19, 2018 in Tutorials. The snippet: Whether you're interested in learning how to apply facial recognition to video streams, building a complete deep learning pipeline for image classification, or simply want to tinker with your Raspberry Pi and add image recognition to a hobby project, you'll need to learn OpenCV somewhere along the way. The truth is that learning OpenCV used [...].
- OpenCV Saliency Detection** by Adrian Rosebrock on July 16, 2018 in Tutorials. The snippet: Today's tutorial is on saliency detection, the process of applying image processing and computer vision algorithms to automatically locate the most "salient" regions of an image. In essence, saliency is what "stands out" in a photo or scene, enabling your eye-brain connection to quickly

<https://www.pyimagesearch.com>

# People Counter

- 3 lines of code changed

# PCB Manufacturing

# PCB Manufacturing

- Electronics components sourcing/ Data sheets. (e.g. [mouse.com](#), <https://lcsc.com> [google.com](#))
- Schematics and Foot prints
- Drawing a schematic and PCB layout ([easyeda.com](#) Eagle, Kicad

# Electronics components sourcing and Data Sheets

**opamp Operational Amplifiers - Op Amps**

# Schematic and Footprint

SYMBOLS | FOOTPRINTS | 3D MODELS Back X

**SamacSys** **MOUSER ELECTRONICS**

Texas Instruments TLV9061SIDBVR

Symbol	Footprint	3D

Normal View ▾ Device on Footprint ▾

[DOWNLOAD ECAD MODELS](#)

Download the free [Library Loader](#) to convert this file for your ECAD Tool. [Learn More.](#)

# Thank you!

I will post links to the mentioned projects into the discord.