RICE ece redefining limits

Edge Computing for Low-Latency Applications

Jenny Penaloza, Cynthia Chen, Peter Wang, Samatar Dalmar,
Nadya Mohamed, Chance Tarver, and Joseph R. Cavallaro

Parallel Hardware Applications in
Science and Technology (PHAST)
VIP Group

Smart Cities and the Challenges of Dense Connectivity in the Future

Background:

- Future applications in scenarios such as Smart Cities collect a huge amount of data from various entities such as personal devices, cameras, vehicles
- Big data processing problem for this large sensor network

Issue:

Cloud-based processing cannot meet the latency demands for many future applications that only require low-latency

- Vehicle-to-vehicle communication
- Industry 4.0 Smart Factories

Hypothesis:

We may reduce the latency by running smaller, localized applications at the edge.

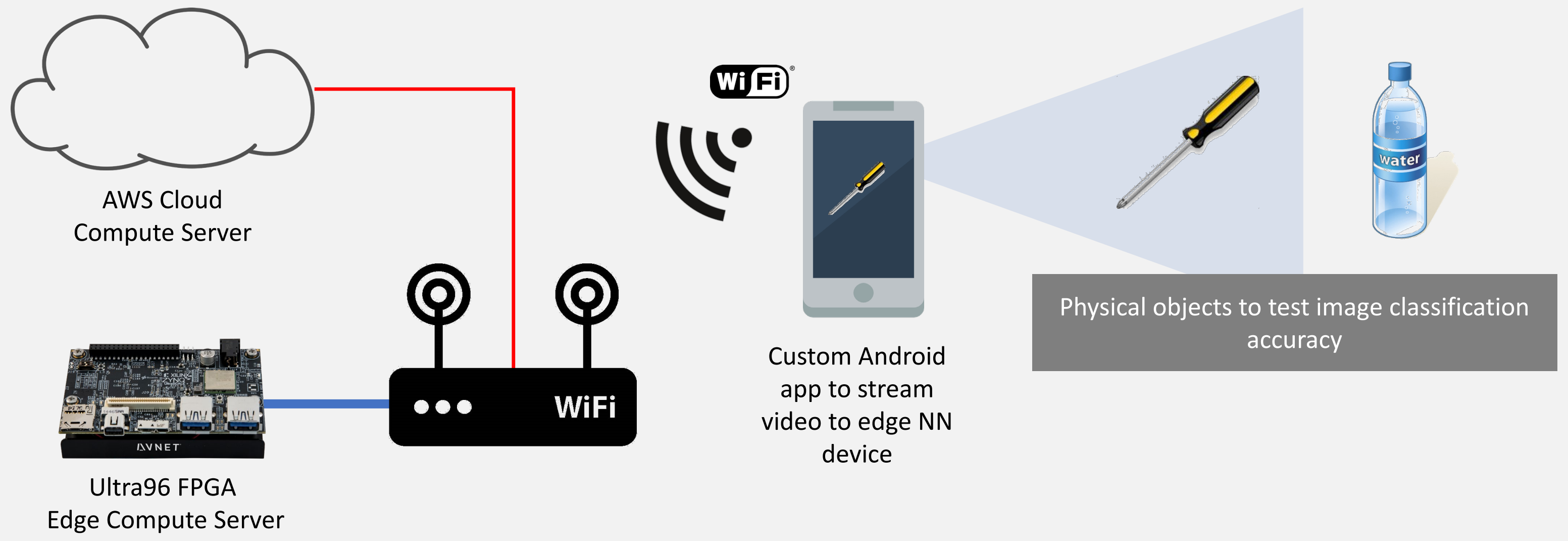
Goal:

To design high performance algorithms such as neural network inference on lower cost hardware to improve latency.

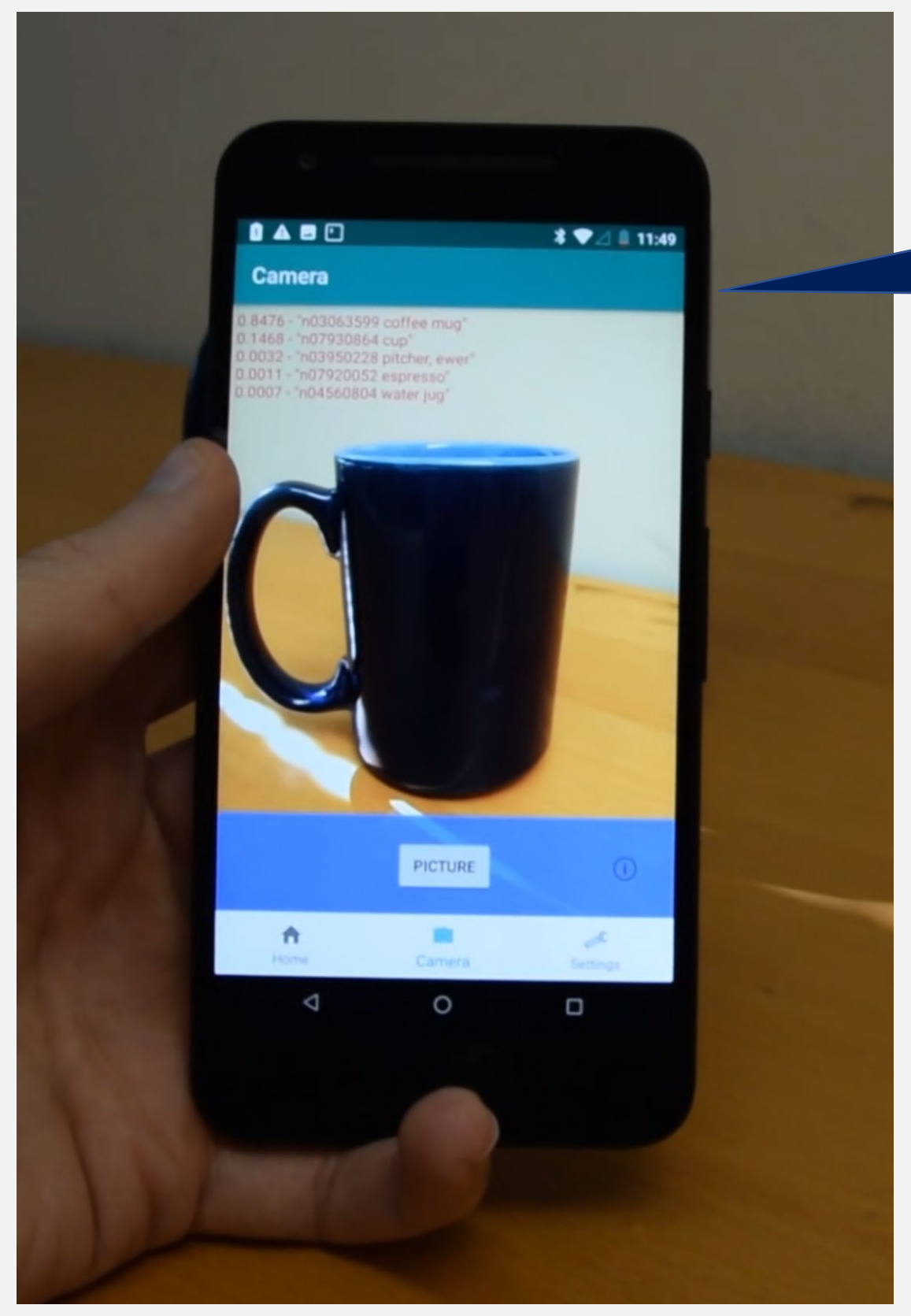
Neural Networks on the Edge

Goal: *Investigate tradeoffs between cloud and edge computing for low-latency applications*

FPGA-based Edge Compute Testbed:


AWS Cloud Compute Server
Ultra96 FPGA Edge Compute Server
WiFi
Wi Fi
Custom Android app to stream video to edge NN device
Physical objects to test image classification accuracy

Example result on custom Android app:

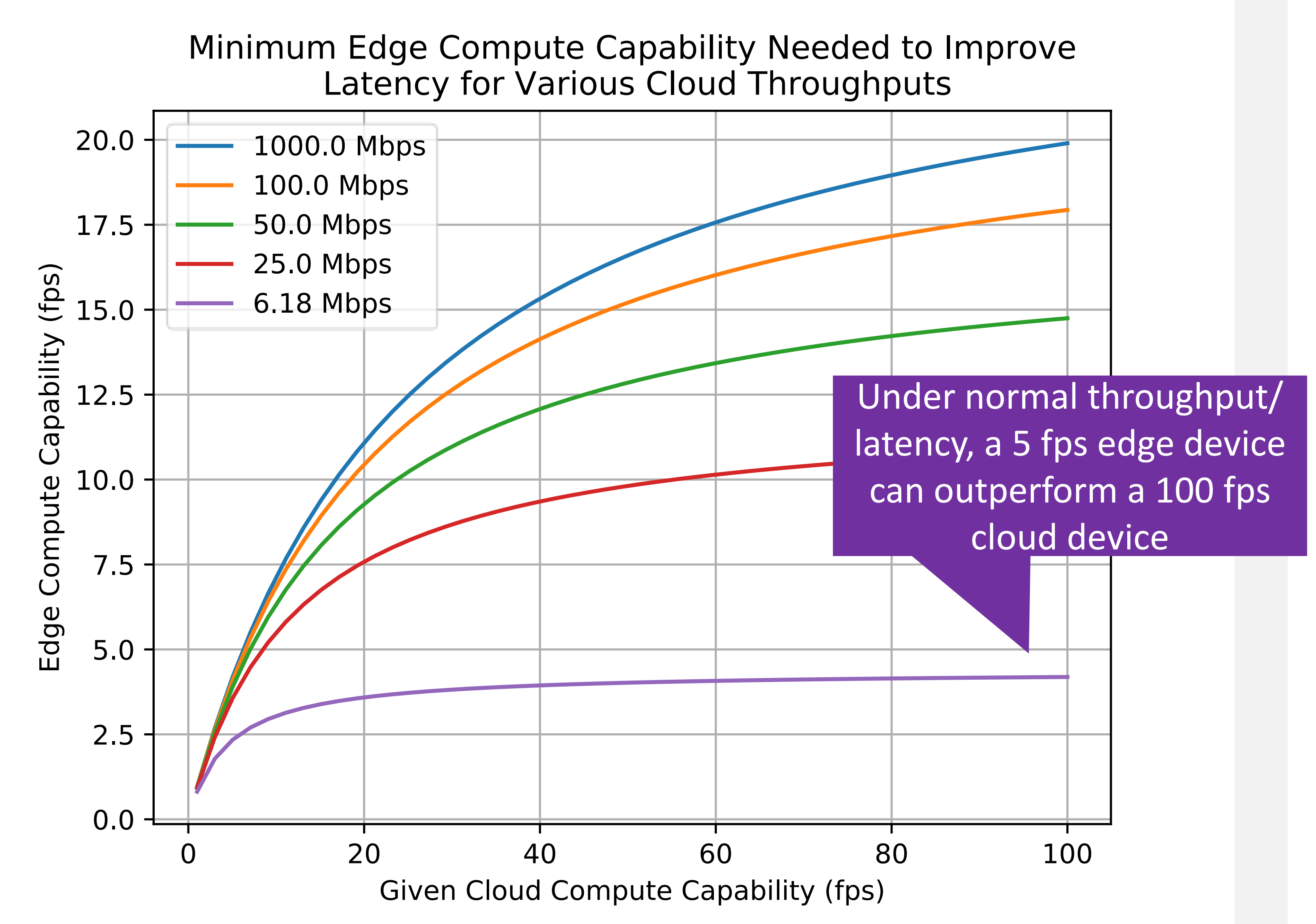


Edge FPGA sends Resnet50 classification results to the phone for display at a rate of ~10 fps.

Edge vs Cloud:

- 10x reduction in latency
- 30x increase in throughput

Theoretical result showing edge/cloud tradeoff



Under normal throughput/latency, a 5 fps edge device can outperform a 100 fps cloud device

Preprocessing for Sensor Networks

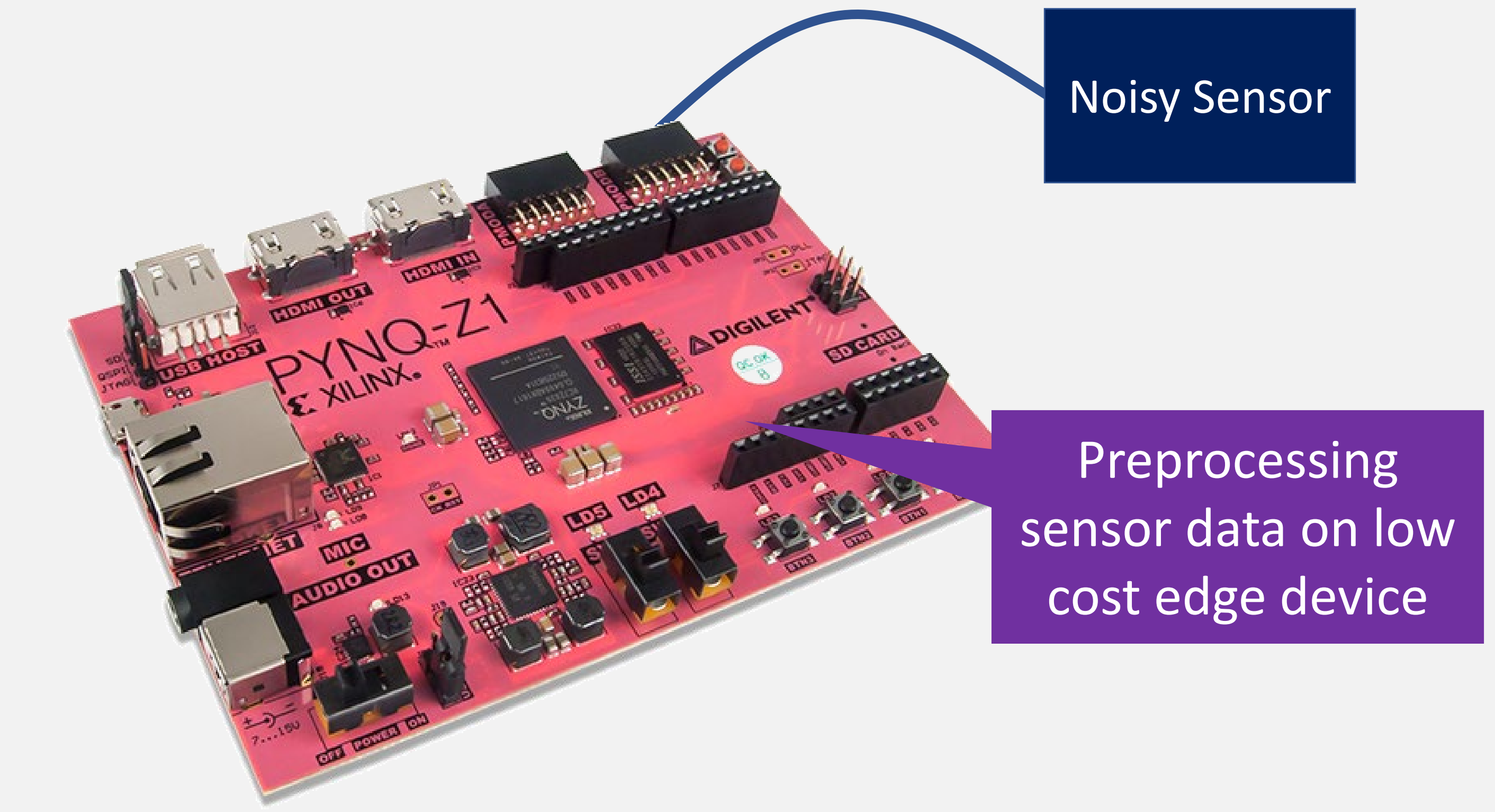
Goal: *Use low-cost edge devices to perform computations on raw data before forwarding to cloud.*

Example Application: *Filtering noise sensor input.*

- Low-cost sensors can be noisy
- Implement an FIR filter in FPGA to attach to sensor node to preprocess and smooth the data.

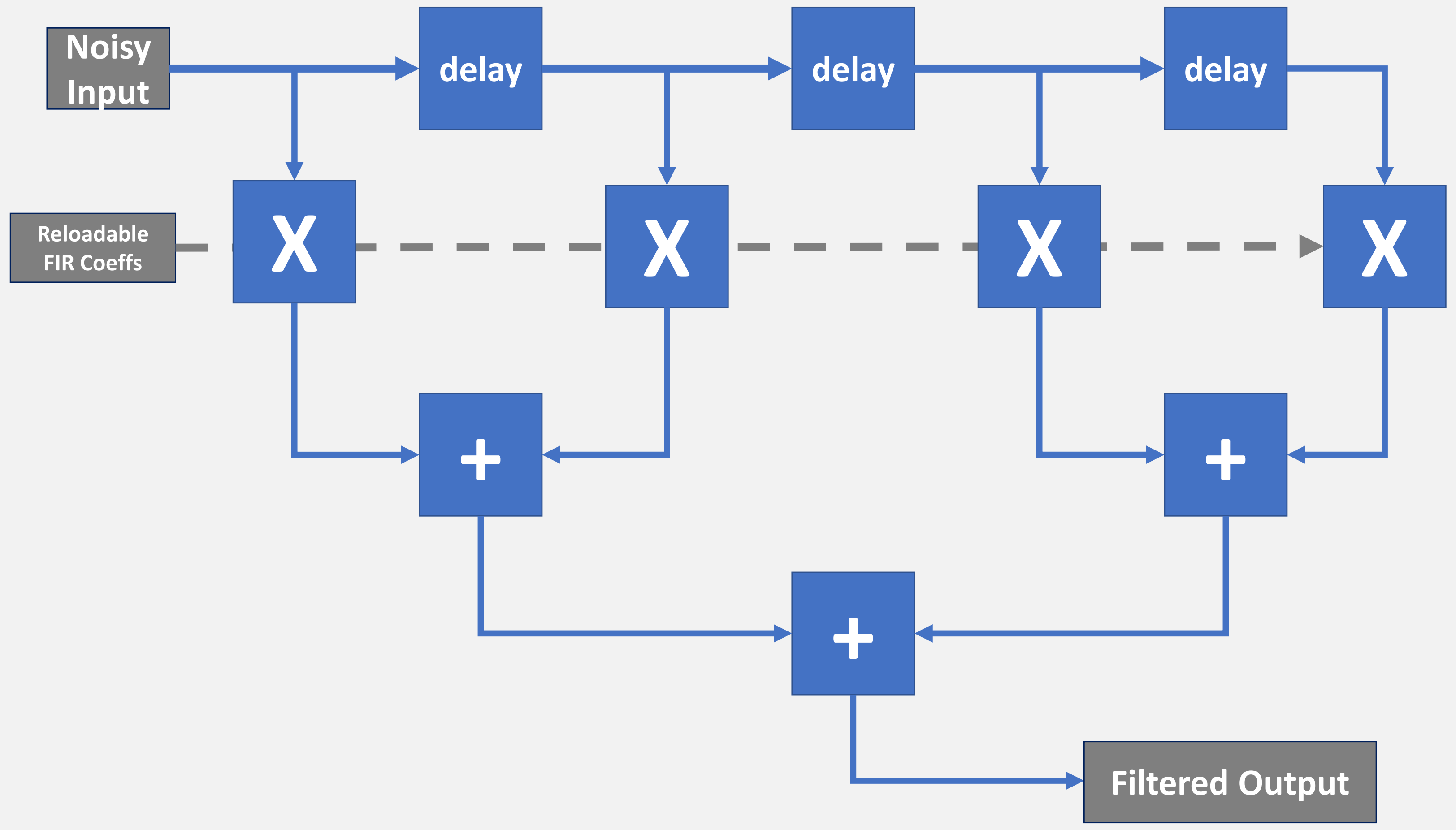
FIR Filter FPGA Design:

- Streamable, fully pipelined design
- Computing each of the multiples in parallel.
- Can support high throughputs
- Targeting new PYNQ platform where FPGA design can be used in a python environment



Noisy Sensor

Preprocessing sensor data on low cost edge device


Noisy Input
delay
delay
delay
Reloadable FIR Coeffs
X
X
X
X
+
+
+
Filtered Output