tinyML® Talks

Enabling Ultra-low Power Machine Learning at the Edge

"Deploying AI to Embedded Systems"

Bernhard Suhm – MathWorks

April 13, 2021

TINY ML

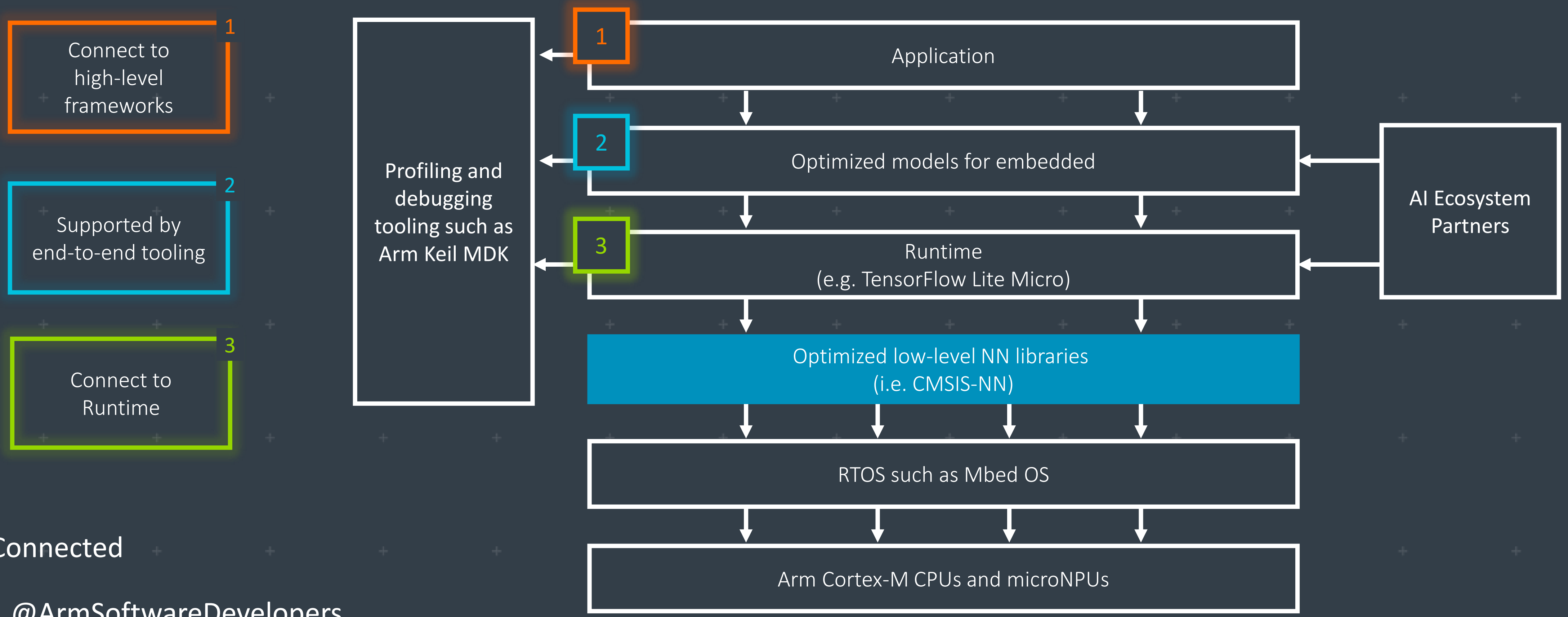
www.tinyML.org

tinyML Talks Sponsors

tinyML Strategic Partner

Additional Sponsorships available – contact Olga@tinyML.org for info

Arm: The Software and Hardware Foundation for tinyML



Stay Connected

@ArmSoftwareDevelopers

@ArmSoftwareDev

Resources: developer.arm.com/solutions/machine-learning-on-arm

arm

Deeplite

WE USE AI TO MAKE OTHER AI FASTER, SMALLER AND MORE POWER EFFICIENT

Automatically compress SOTA models like MobileNet to <200KB with **little to no drop in accuracy** for inference on resource-limited MCUs

Reduce model optimization trial & error from weeks to days using Deeplite's **design space exploration**

APPLY NOW

Deploy more models to your device without sacrificing performance or battery life with our **easy-to-use software**

BECOME BETA USER **bit.ly/testdeeplite**

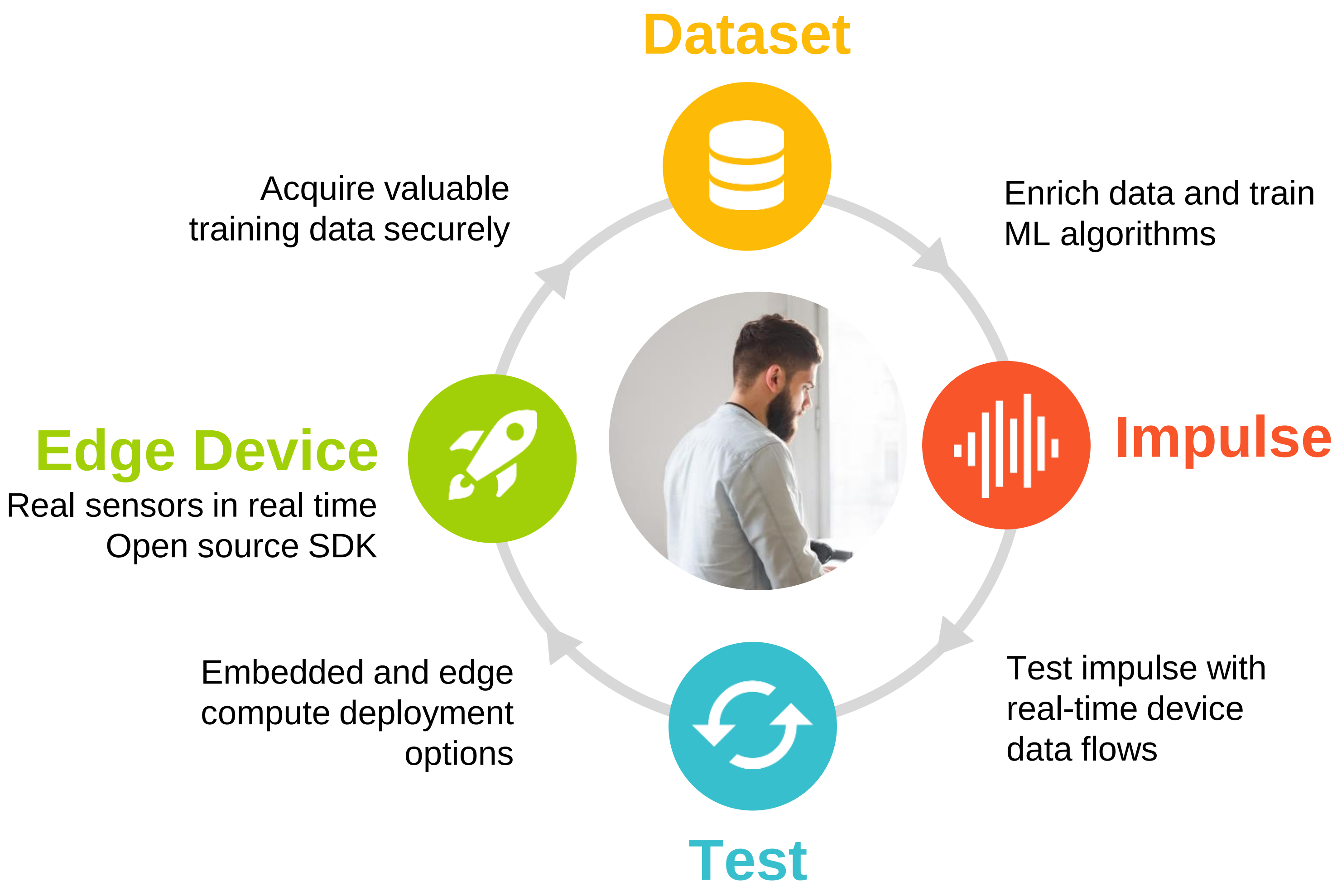
mobilityXlab arm AI 100 CBINSIGHTS

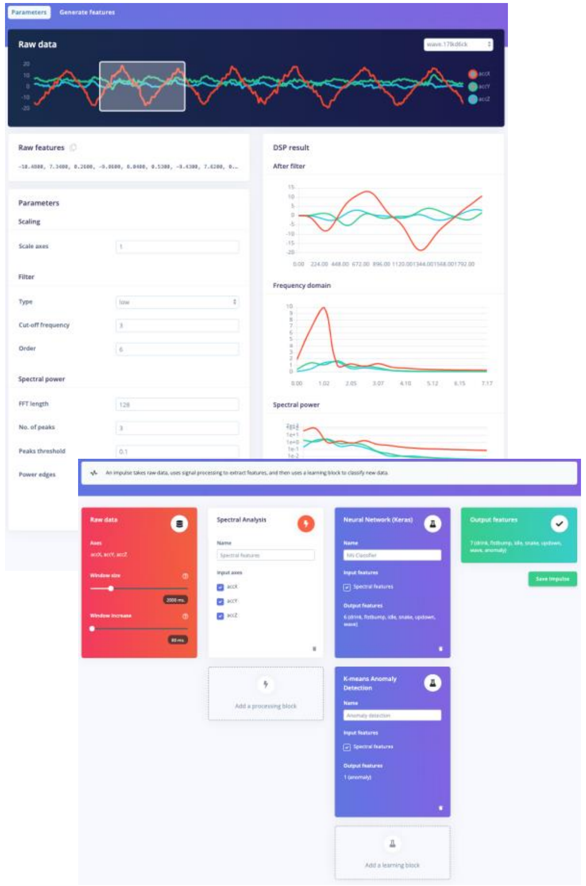
TinyML for all developers

C++ library

Arduino library

WebAssembly

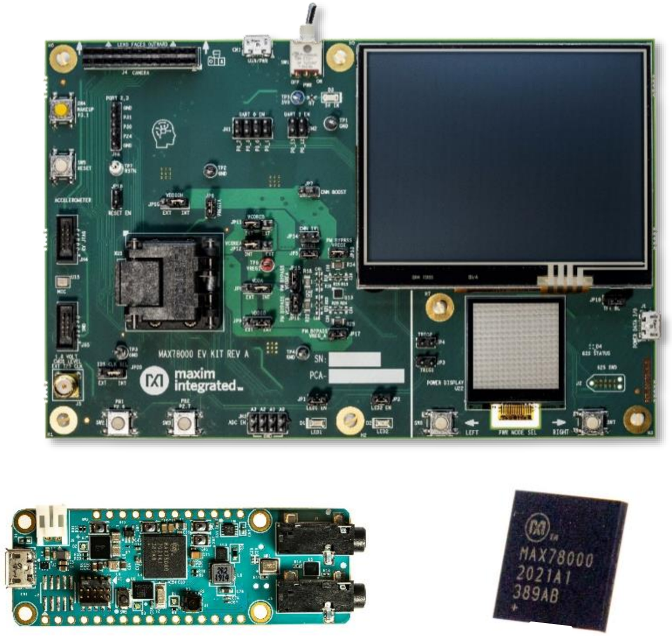




www.edgeimpulse.com

Maxim Integrated: Enabling Edge Intelligence

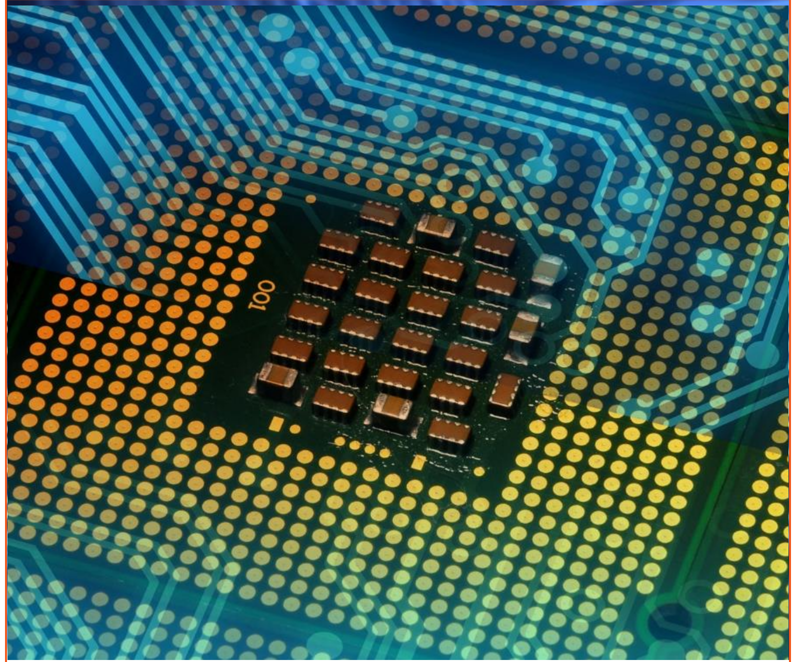
Advanced AI Acceleration IC



The new MAX78000 implements AI inferences at low energy levels, enabling complex audio and video inferencing to run on small batteries. Now the edge can see and hear like never before.

www.maximintegrated.com/MAX78000

Low Power Cortex M4 Micros



Large (3MB flash + 1MB SRAM) and small (256KB flash + 96KB SRAM, 1.6mm x 1.6mm) Cortex M4 microcontrollers enable algorithms and neural networks to run at wearable power levels.

www.maximintegrated.com/microcontrollers

Sensors and Signal Conditioning

Health sensors measure PPG and ECG signals critical to understanding vital signs. Signal chain products enable measuring even the most sensitive signals.

www.maximintegrated.com/sensors

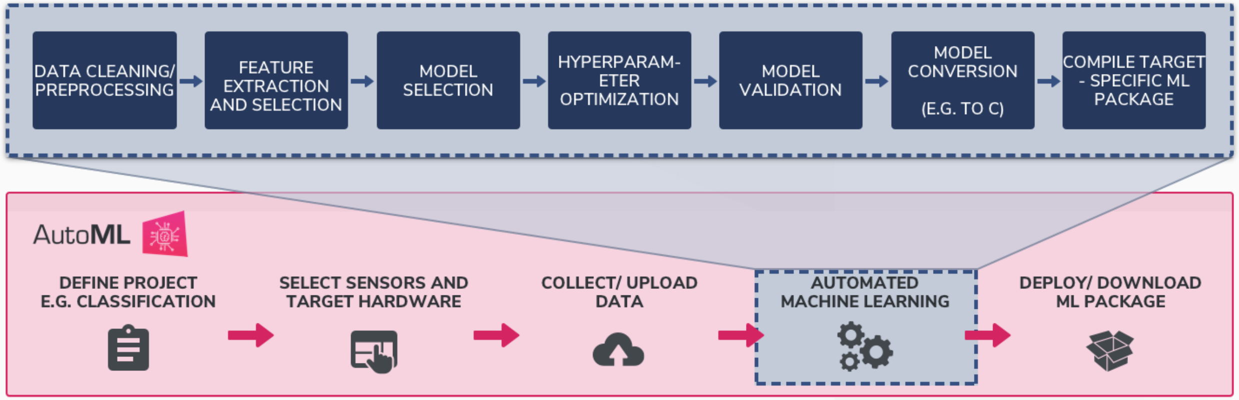
Qeexo AutoML

Automated Machine Learning Platform that builds tinyML solutions for the Edge using sensor data

Key Features

- Supports 17 ML methods:
  - Multi-class algorithms: GBM, XGBoost, Random Forest, Logistic Regression, Gaussian Naive Bayes, Decision Tree, Polynomial SVM, RBF SVM, SVM, CNN, RNN, CRNN, ANN
  - Single-class algorithms: Local Outlier Factor, One Class SVM, One Class Random Forest, Isolation Forest
- Labels, records, validates, and visualizes time-series sensor data
- On-device inference optimized for low latency, low power consumption, and small memory footprint applications
- Supports Arm® Cortex™- M0 to M4 class MCUs

End-to-End Machine Learning Platform



For more information, visit: www.qeexo.com

Target Markets/Applications

- Industrial Predictive Maintenance
- Smart Home
- Wearables
- Automotive
- Mobile
- IoT

Reality AI®

Add Advanced Sensing to your Product with Edge AI / TinyML

https://reality.ai info@reality.ai @SensorAI Reality AI

Pre-built Edge AI sensing modules, plus tools to build your own

Reality AI solutions

- Prebuilt sound recognition models for indoor and outdoor use cases
- Solution for industrial anomaly detection
- Pre-built automotive solution that lets cars “see with sound”

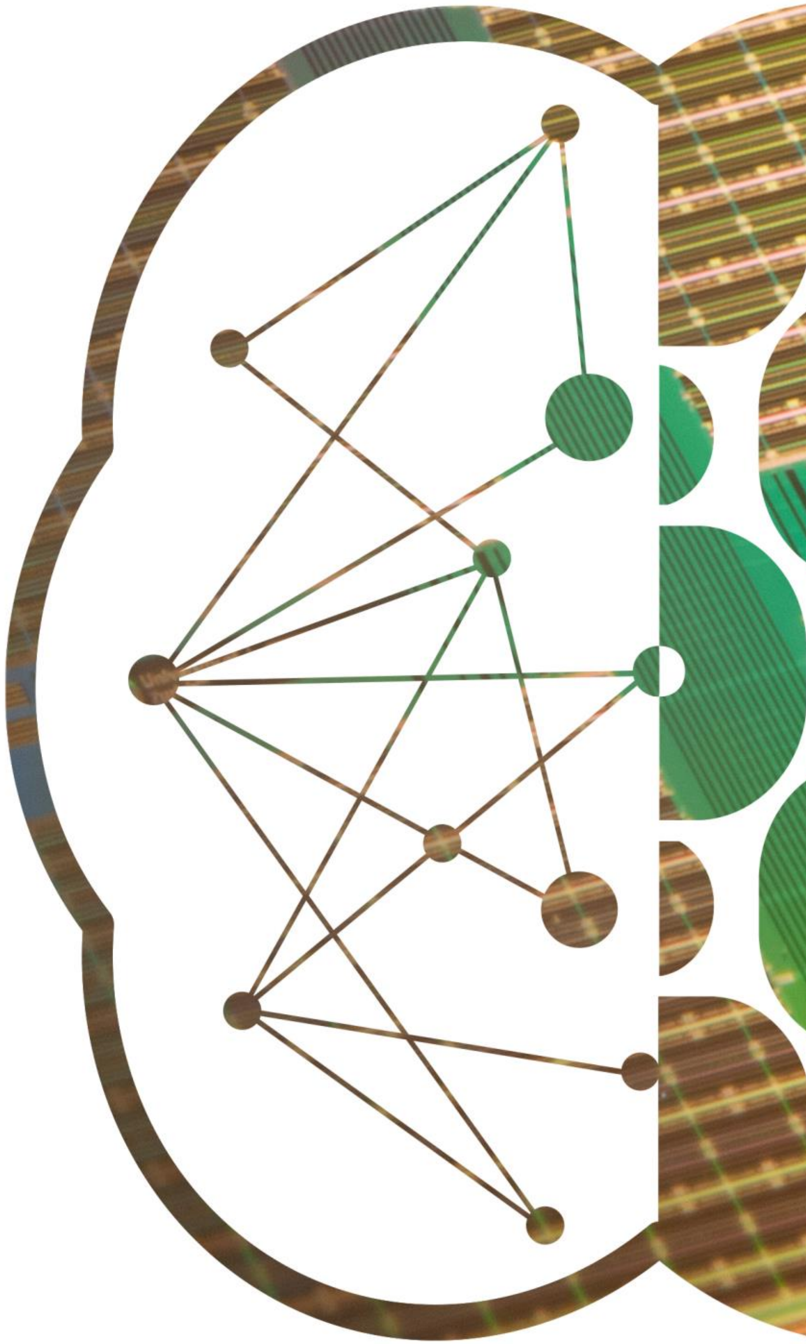
Reality AI Tools® software

- Build prototypes, then turn them into real products
- Explain ML models and relate the function to the physics
- Optimize the hardware, including sensor selection and placement

SynSense

SynSense builds **sensing and inference** hardware for **ultra-low-power** (sub-mW) **embedded, mobile and edge** devices. We design systems for **real-time always-on smart sensing**, for audio, vision, IMUs, bio-signals and more.

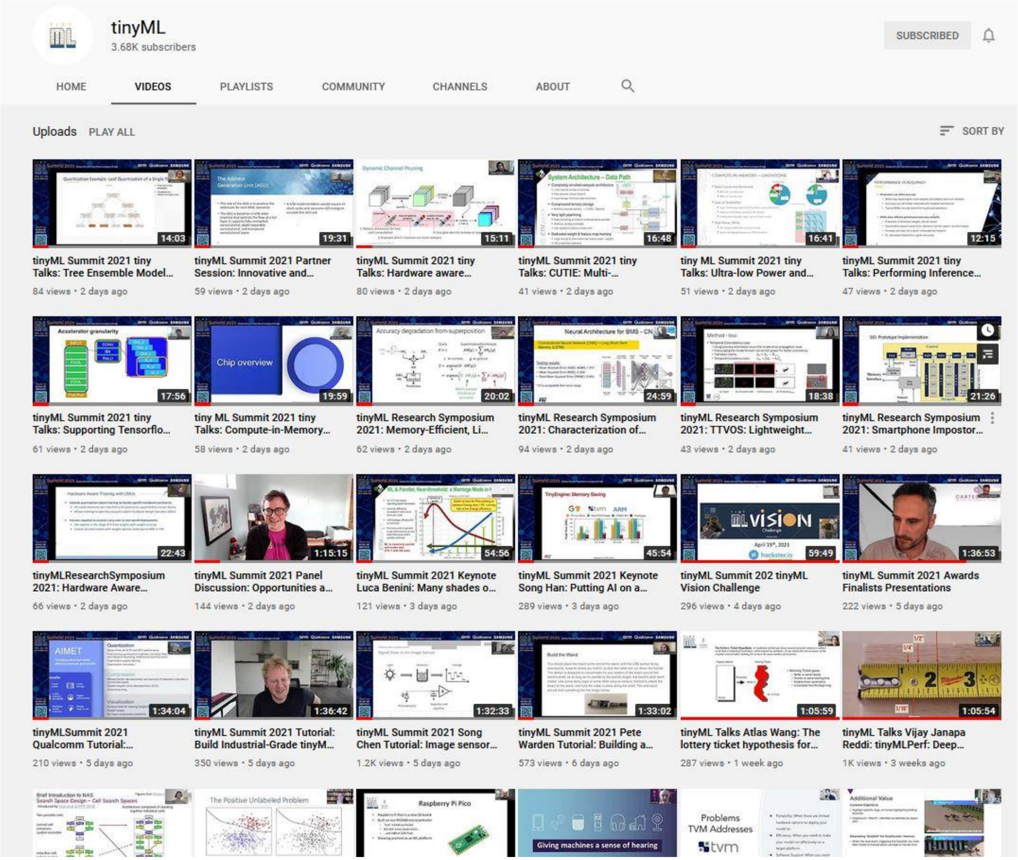
https://SynSense.ai



Successful tinyML Summit 2021:

- **5** days of tutorials, talks, panels, breakouts, symposium
  - **4** tutorials
  - **6** keynotes & **6** plenary tinyTalks (more in breakouts)
  - **2** panel discussions
  - **5** disruptive news presentations
  - **17** breakout/partner sessions
  - **6** Best Product and Innovation Award Finalists & Presentations
  - **89** Speakers
- **5006** registered attendees representing:
  - **104** countries, **1000+** companies and **400+** academic institutions
- **26** Sponsoring companies

www.youtube.com/tinyML with 150+ videos



tinyML Summit-2022, January 24-26, Silicon Valley, CA

EMEA

June 7-10, 2021 (virtual, but LIVE)

Deadline for abstracts: May 1

Sponsorships are being accepted: sponsorships@tinyML.org

April 15th, 2021 Launch

hackster.io
AN AVNET COMMUNITY

Next tinyML Talks

Date	Presenter	Topic / Title
Tuesday, April 27	**Michael Jo and Xingheng Lin** Rose-Hulman Institute of Technology	Train-by-weight (TBW): Accelerated Deep Learning by Data Dimensionality Reduction

Webcast start time is 8 am Pacific time

Please contact talks@tinyml.org if you are interested in presenting

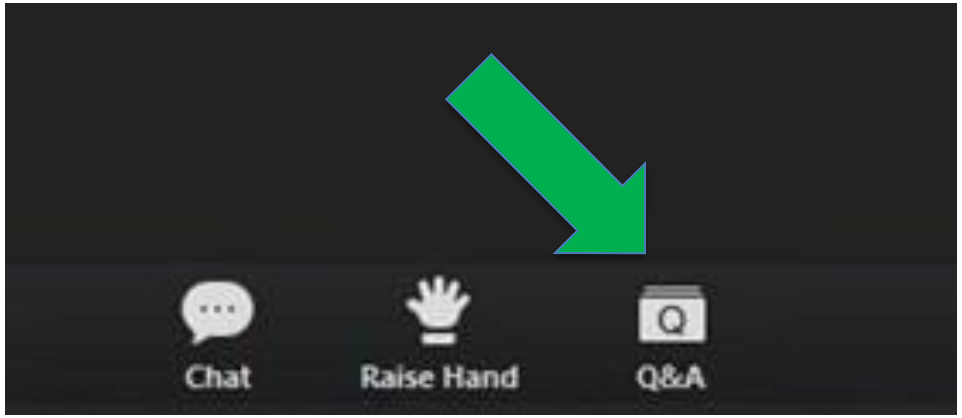
Reminders

Slides & Videos will be posted tomorrow

tinyml.org/forums

youtube.com/tinyml

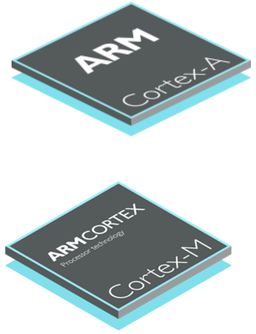
Please use the Q&A window for your questions



Bernhard Suhm

Bernhard Suhm is the product manager for Machine Learning at MathWorks. He works closely with customer facing and development teams to address customer needs and market trends in our machine learning related products, primarily the Statistics and Machine Learning toolbox. Prior to joining MathWorks Bernhard led a team of analysts consulting call centers on optimizing the delivery of customer service. He also held positions at a usability consulting company and Carnegie Mellon University. He received a PhD in Computer Science specializing in speech user interfaces from Karlsruhe University in Germany.

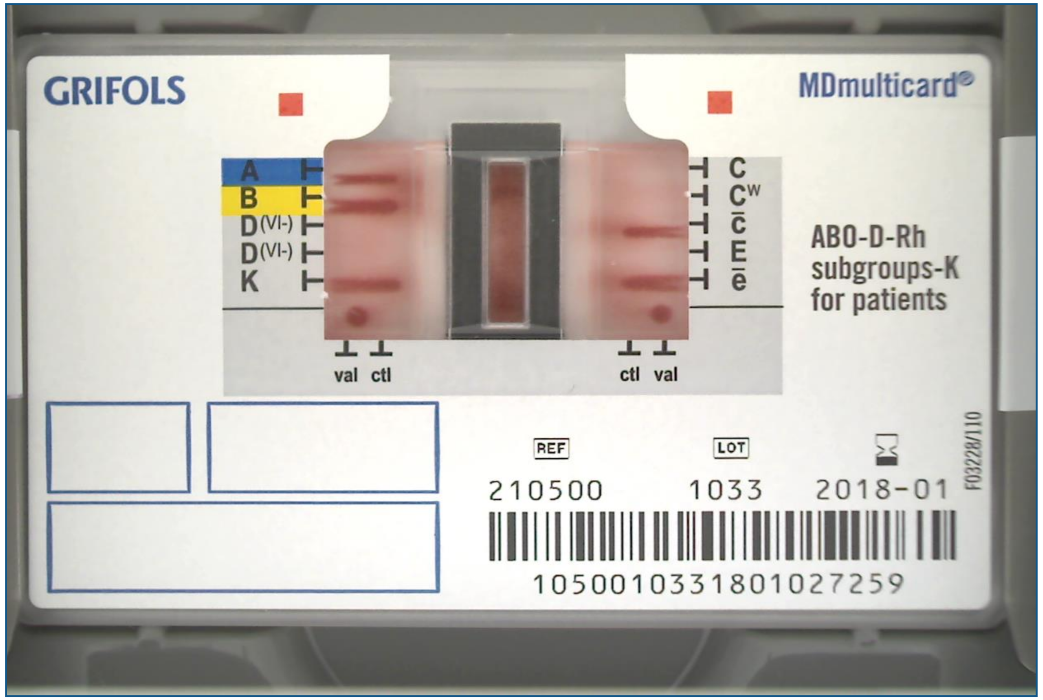
Deploying AI to Embedded Systems



Bernhard Suhm

Product Manager – Machine Learning

Examples for Embedded Deployment



Card to Classify Blood Type
IDNEO

Oversteering Detection

Agenda

Deploying AI is difficult

Four specific challenges:

1. Integrate AI model with an embedded (or industrial) system
2. Fit large AI models on limited hardware
3. Test & Verify before deployment to production
4. Ongoing changes in environment or system behavior

MathWorks®

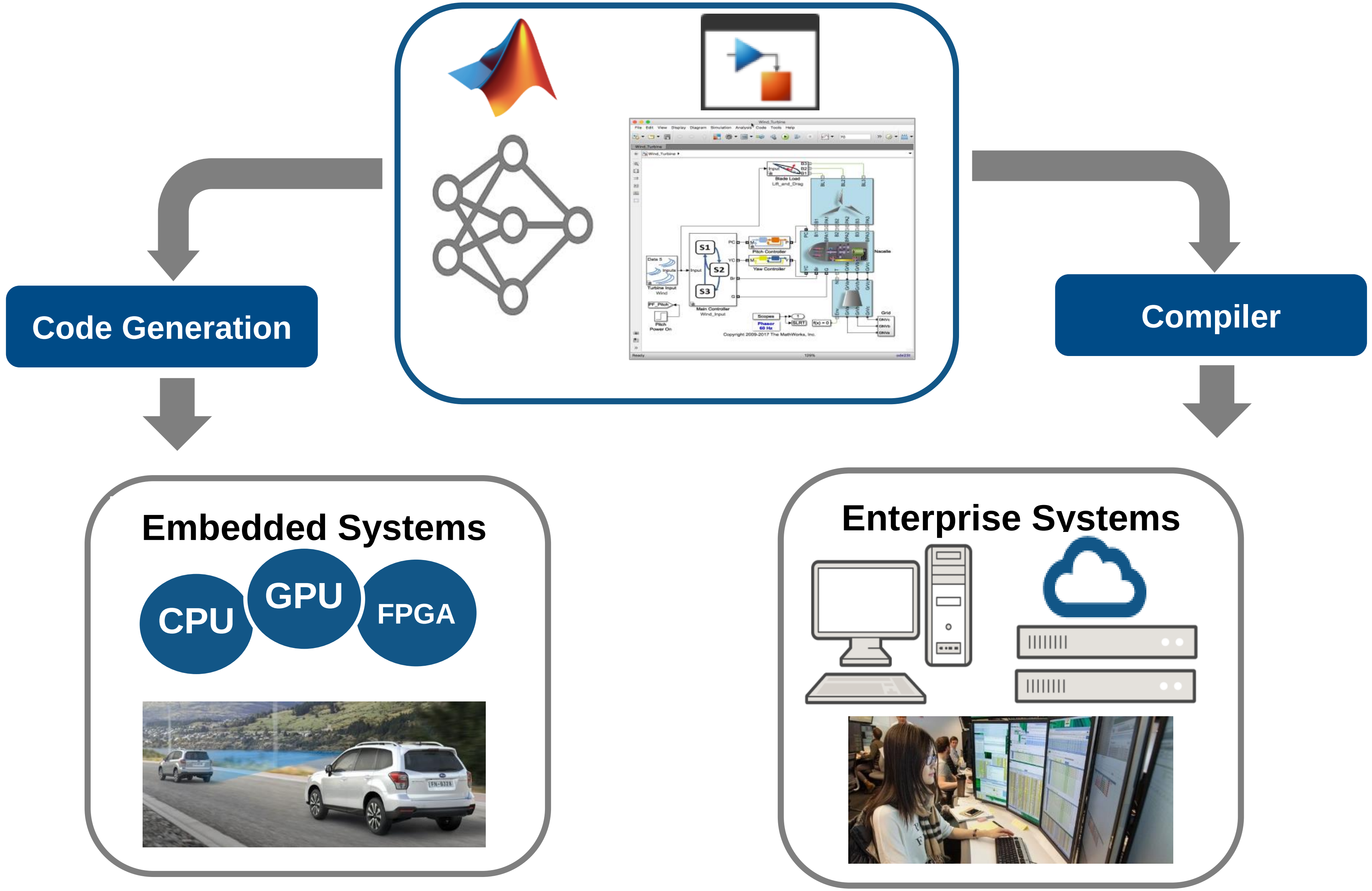
is a **Leader** in the Gartner Magic Quadrant for 2021 Data Science and Machine Learning Platforms for the Second Year in a Row

Figure 1: Magic Quadrant for Data Science and Machine Learning Platforms

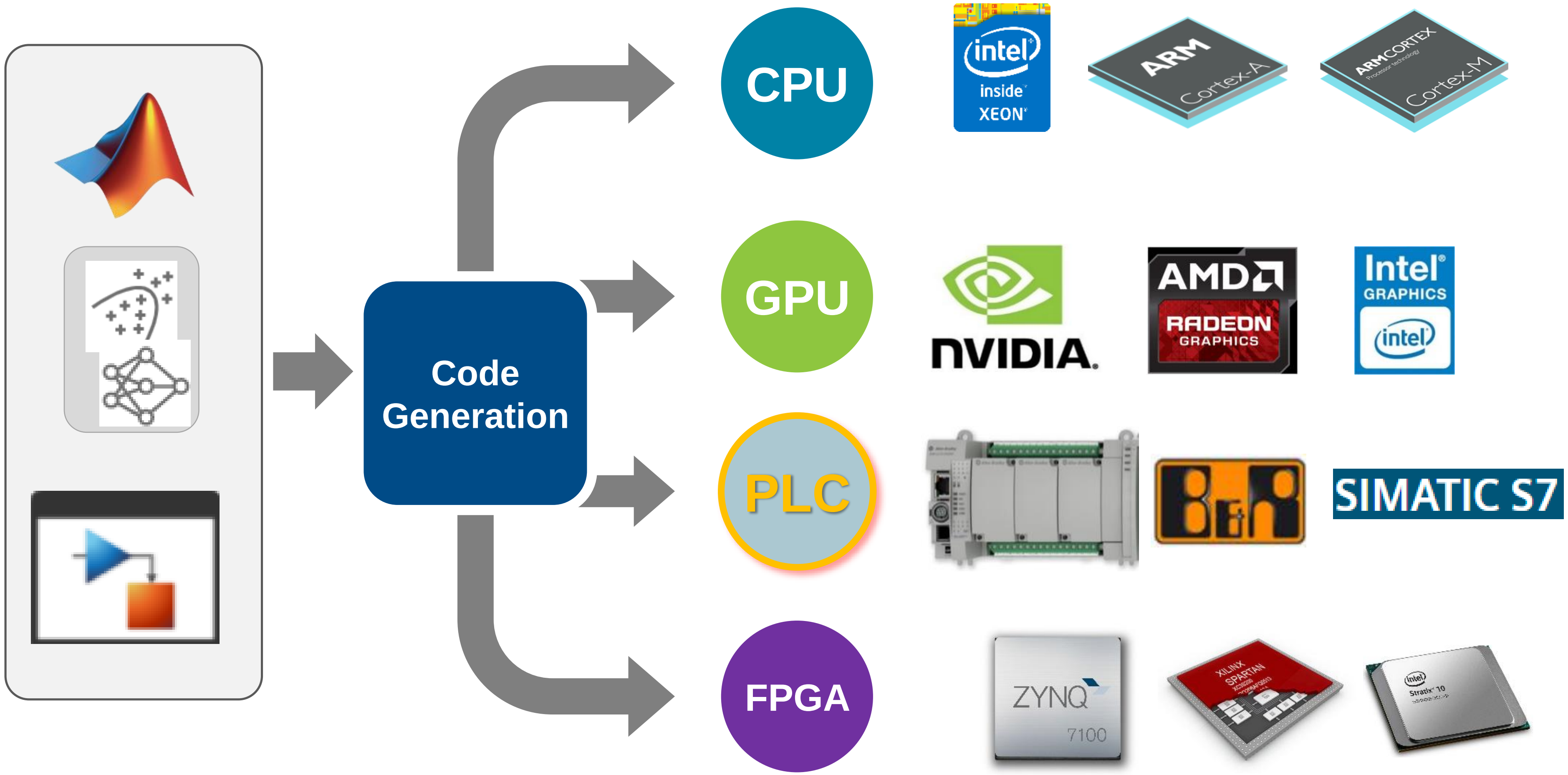
Source: Gartner (March 2021)

Gartner Magic Quadrant for Data Science and Machine Learning Platforms, Peter Krensky, Erick Brethenoux, Pieter denHamer, Farhan Choudhary, Afraz Jaffri, Subhangi Vashisth, 1st March 2021.

Two Approaches for integrating AI with Larger System



One Codebase – Many Deployment targets



NVIDIA, the NVIDIA logo, and TensorRT are registered trademarks of NVIDIA Corporation
Intel logo is a registrered trademark of Intel Coroporation

Human Activity Classification using Smartphones

Data:

Accelerometer from mobile

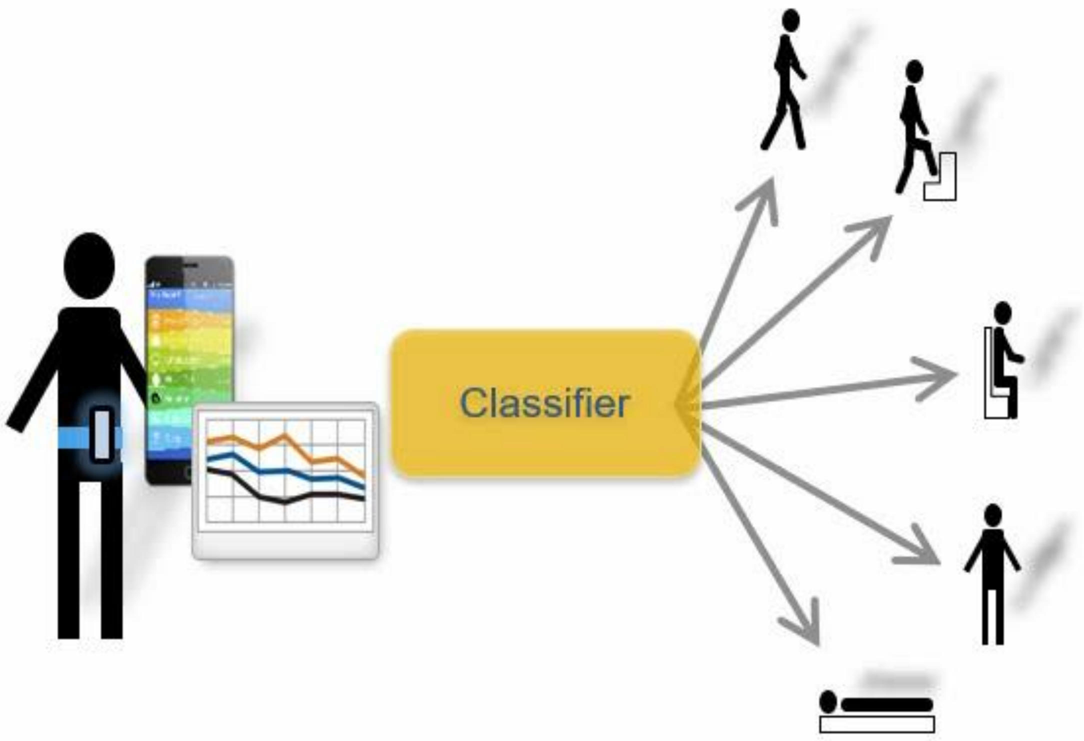
~7K observations from 30 subjects

Steps:

- Interactively build classifiers
- Integrate model with system
- Generate inference C-code
- Deploy to Android using Simulink

Dataset courtesy of:
Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra and Jorge L. Reyes-Ortiz. *Human Activity Recognition on Smartphones using a Multiclass Hardware-Friendly Support Vector Machine.* International Workshop of Ambient Assisted Living (IWAAL 2012). Vitoria-Gasteiz, Spain. Dec 2012 http://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones

Human Activity Learning using Smartphones

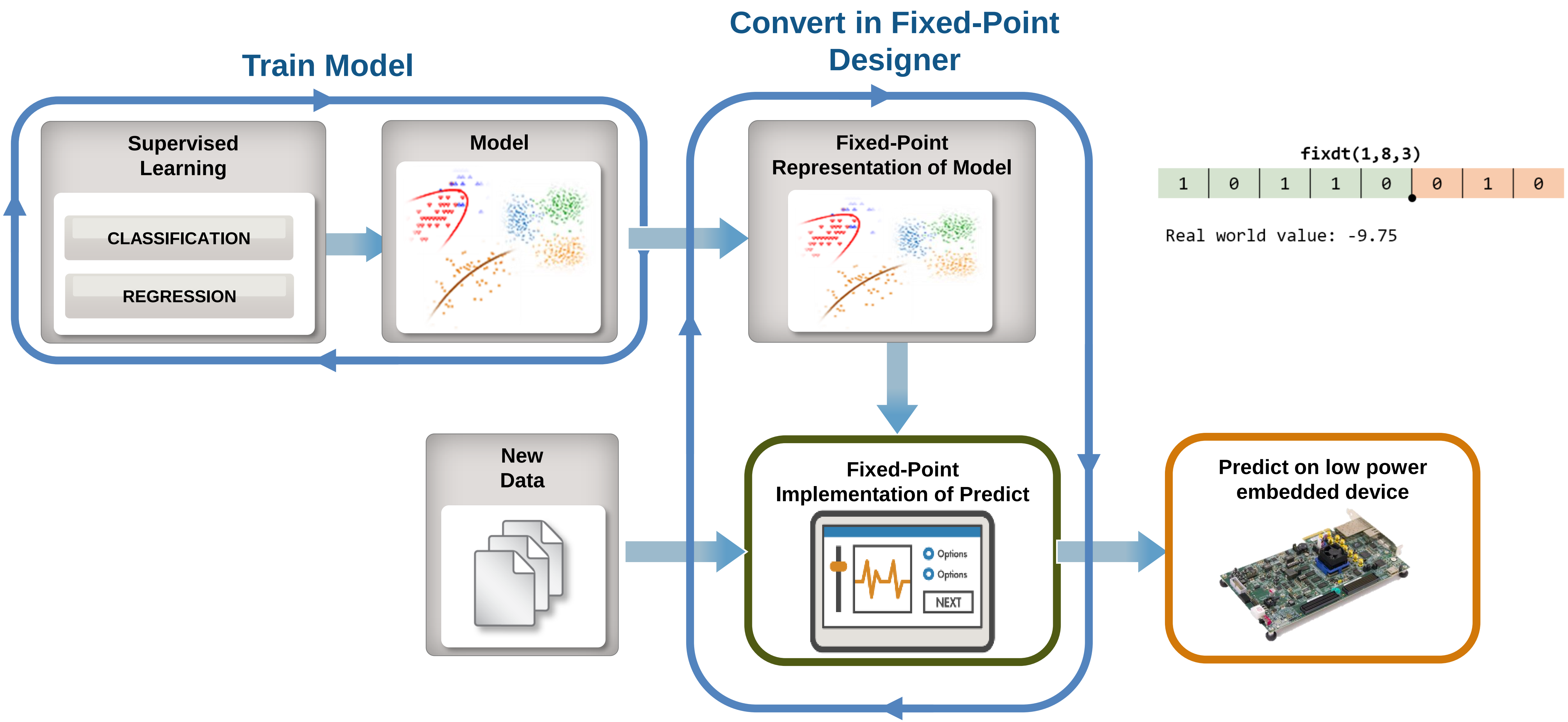


Human Activity Recognition Data

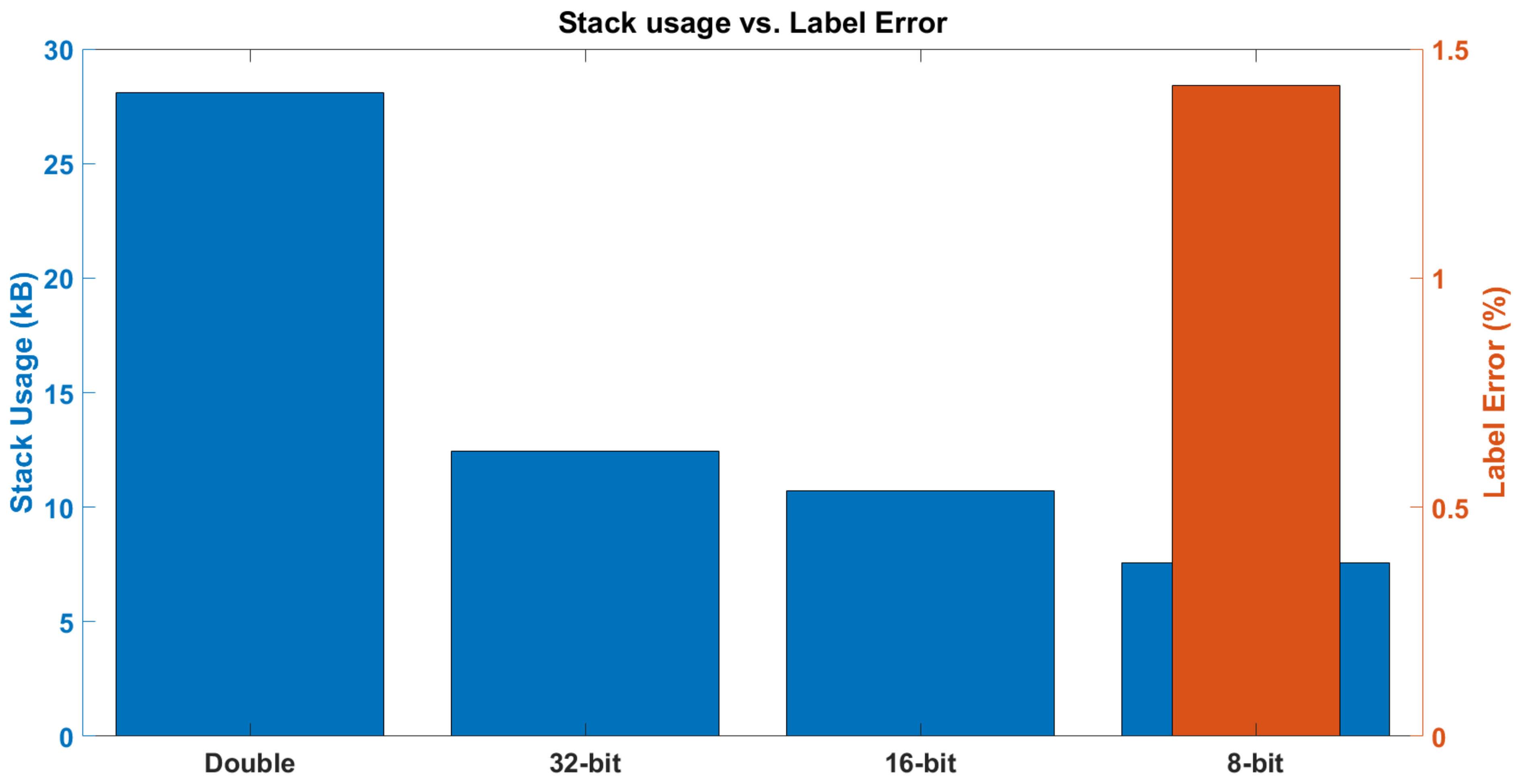
- Accelerometer from mobile
- 7K observations
- 66 manual features, reduced to 2
- 6 classes reduced to “Walking” vs “Idle”

Dataset courtesy of:
Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra and Jorge L. Reyes-Ortiz. *Human Activity Recognition on Smartphones using a Multiclass Hardware-Friendly Support Vector Machine.* International Workshop of Ambient Assisted Living (IWAAL 2012). Vitoria-Gasteiz, Spain. Dec 2012 http://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones

Convert machine learning to fixed-point to reduce memory footprint

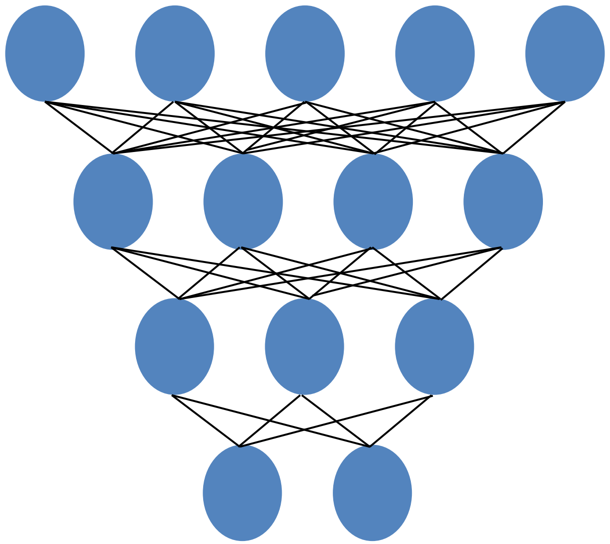


Fixed-point conversion is a trade-off between resource usage optimization and accuracy

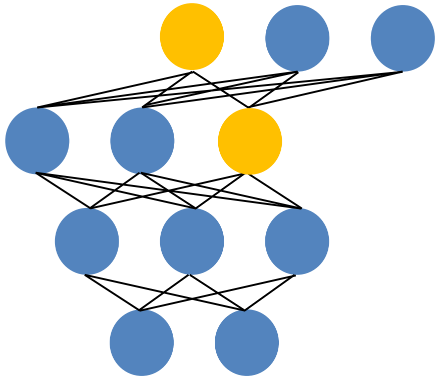


Three Approaches to reducing size of Deep Neural Nets

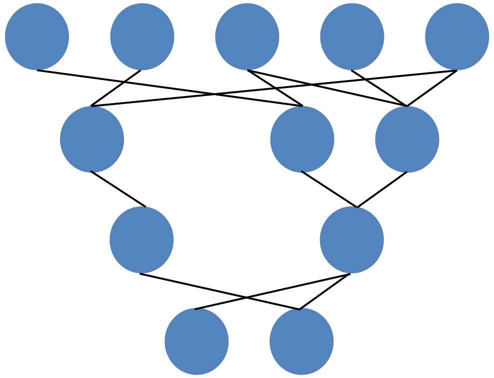
Original Deep Neural Net



1. Layer Fusion

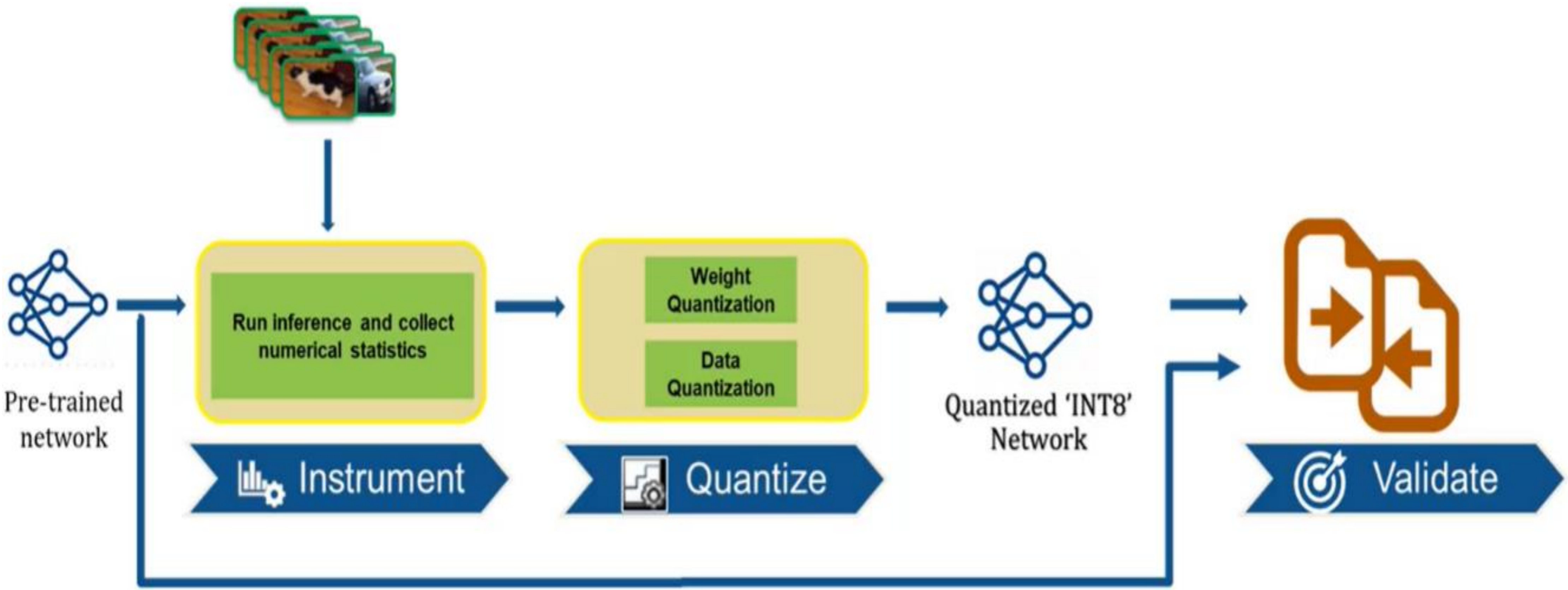


2. Pruning

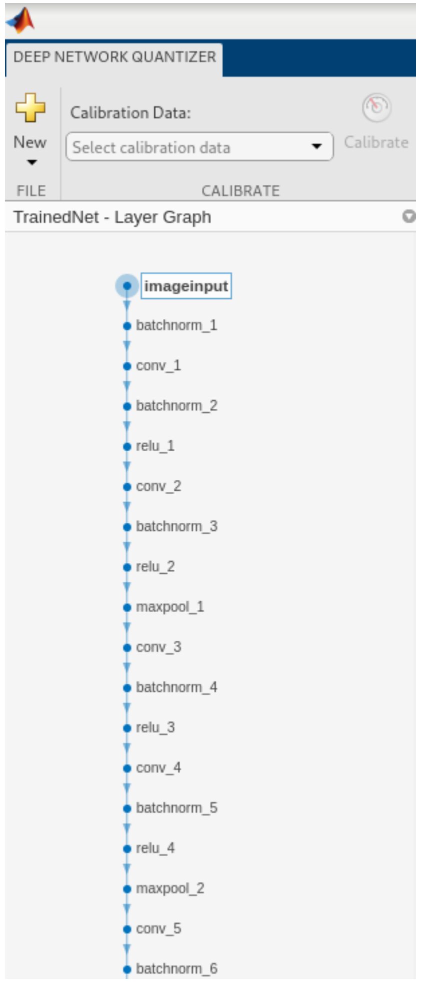


3. Quantization

Model Quantization Workflow



Deep Learning Quantization



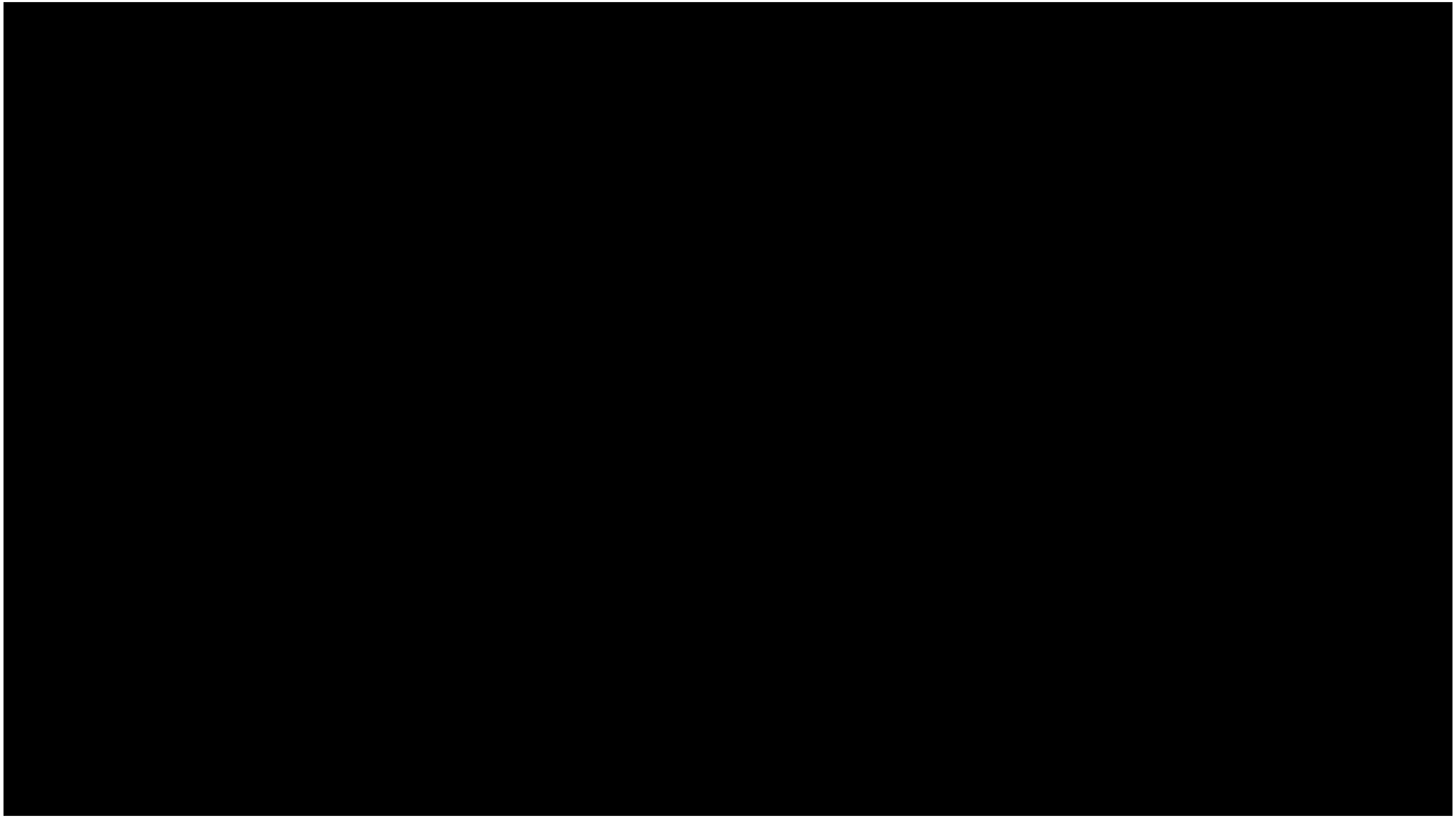
Use Deep Network Quantizer to Optimize the Inference Network

```
load('trainedNet');
analyzeNetwork(trainedNet);
numData = size(xTrain);
numData = numData(end);
augImds = augmentedImageDatastore(trainedNet.Layers(1).InputSize, xTrain, yTrain);
calDS = augImds.subset(1:floor(numData * 0.8));
valDS = augImds.subset(floor(numData * 0.8)+1:numData);
dq = dlquantizer(trainedNet, 'ExecutionEnvironment', 'GPU');
dq.calibrate(calDS)
```

- Load trained network
- Split data: calibration – 80%, validation – 20%
- Launch Deep Network Quantizer App

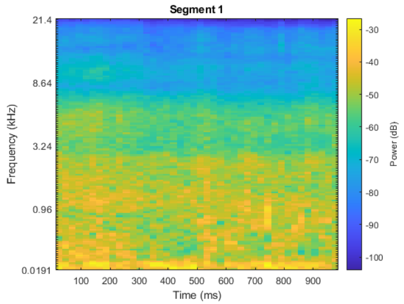
Video *(start ~1:10)*

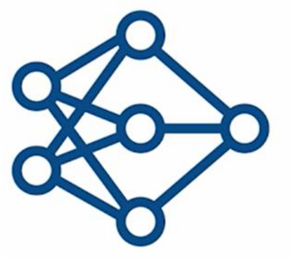
Quiz: Which Sounds do you hear?



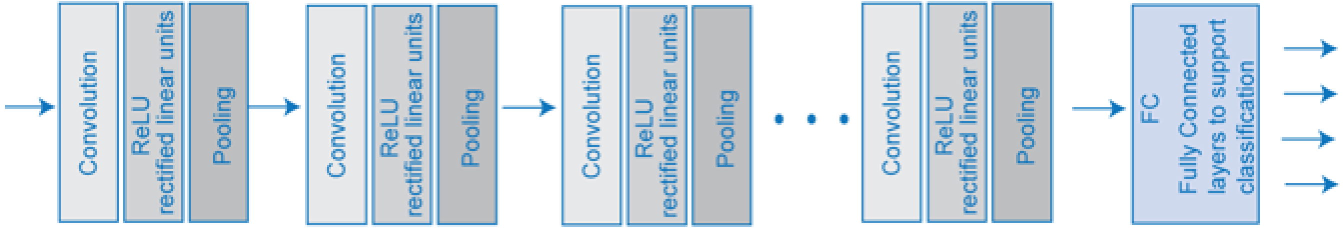
Embedded Deployment of Acoustic Scene Recognition

Reformat the data





Convolutional Neural Networks (**CNN**)



SqueezeNet ~5MB
ResNet-50 ~100MB

Limited resources

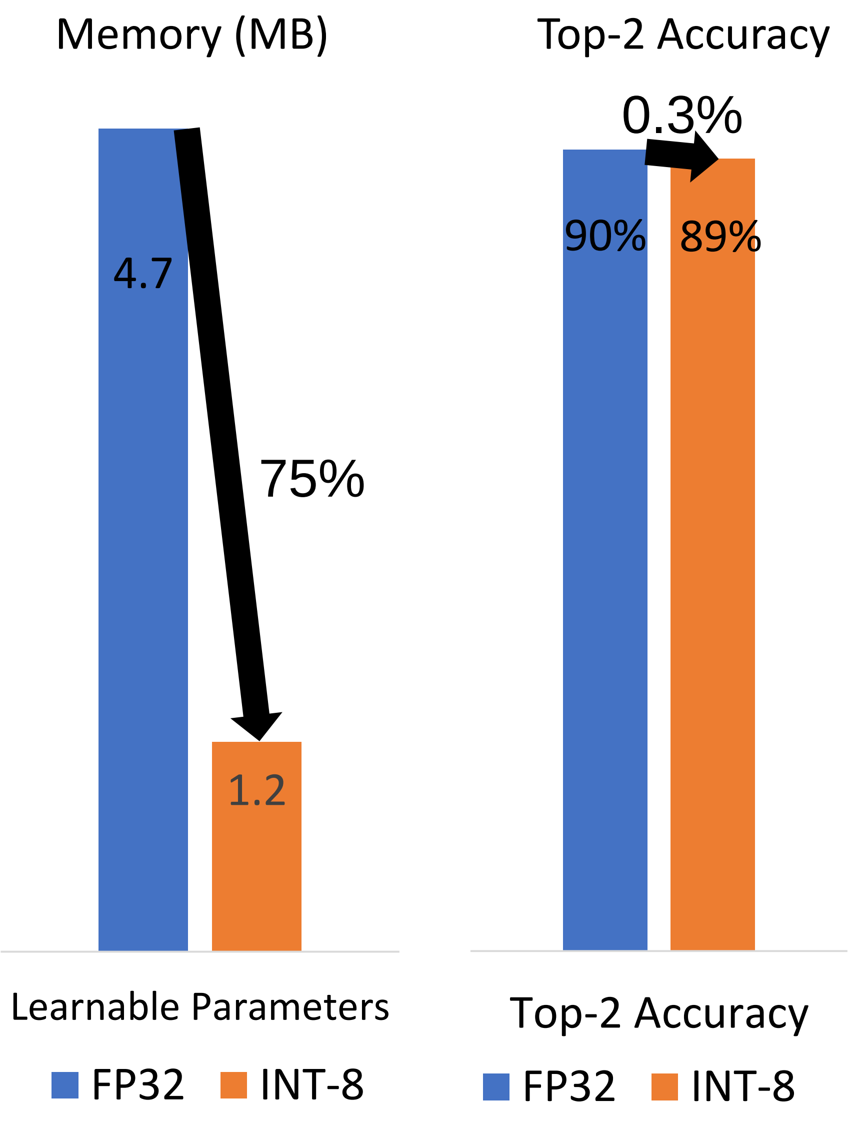
4x reduction in Memory – less than 5% increase in error rate

Validation Summary

Validation Results

Number of samples: 2,280

Metric	Floating-Point Network Results	Quantized Network Results	Perc
Learnable parameter memory (MB)	4.7033	1.1903	74.6
hComputeAccuracy	0.9022	0.8991	0.34



Agenda

Deploying AI is difficult

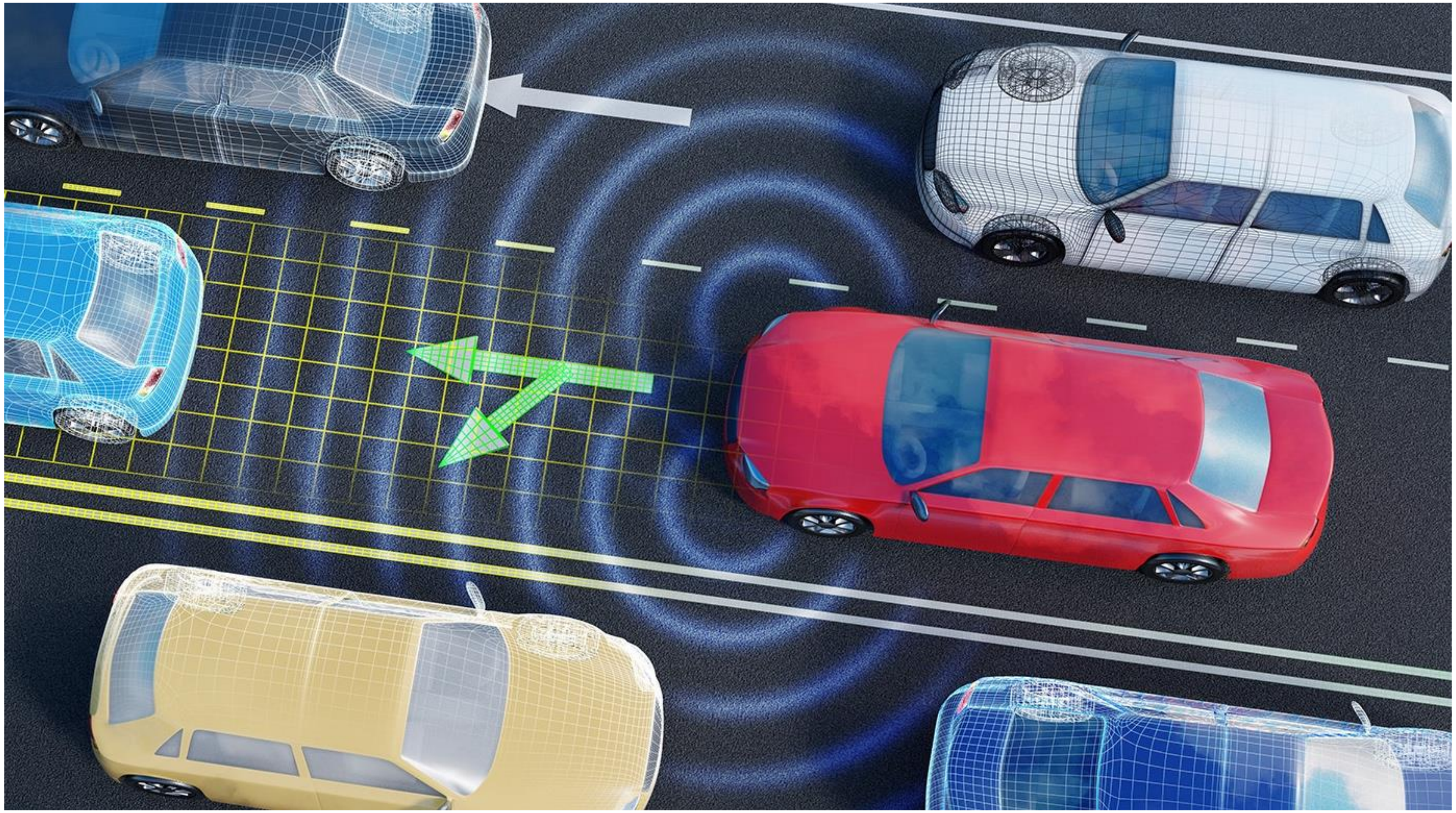
Four specific challenges:

✓ Integrate AI model with an industrial / embedded system

✓ Fit large AI models on limited hardware

3. Test & Verify before deployment on hardware
4. Ongoing changes in environment or system behavior

Integrate AI into system-wide context, simulate before moving to hardware, and verify effectiveness



Facilitate Verification & Validation of your AI Application

MATLAB

SIMULINK

GPU Coder	MATLAB Coder	Simulink Coder

AI Model
Additional Logic

Software In The Loop

Processor In The Loop

Hardware In The Loop

Embedded Systems

Agenda

Deploying AI is difficult

Three specific challenges:

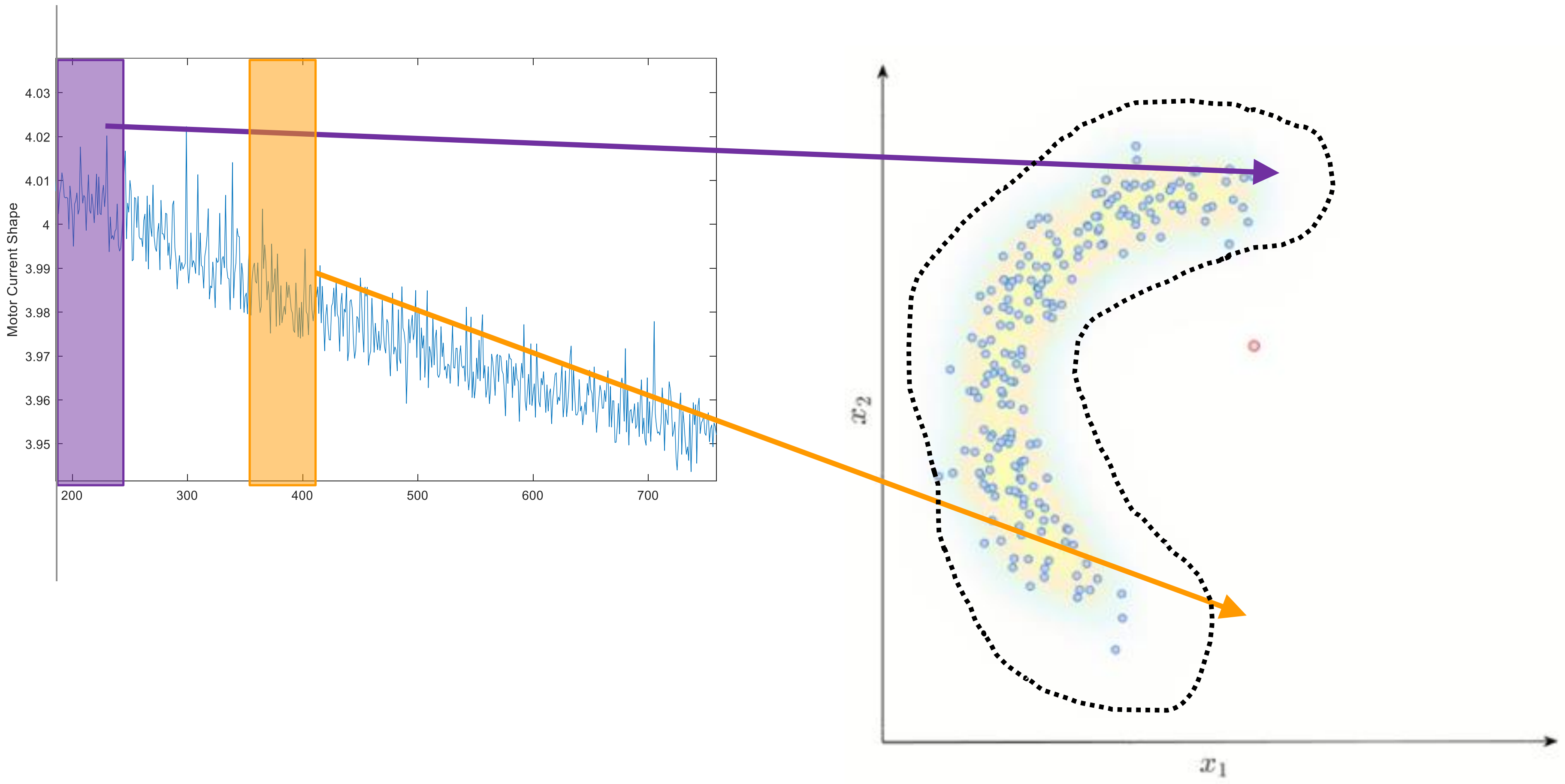
✓ Integrate AI model with an industrial / embedded system

✓ Fit large AI models on limited hardware

✓ Test & Verify before deployment on hardware

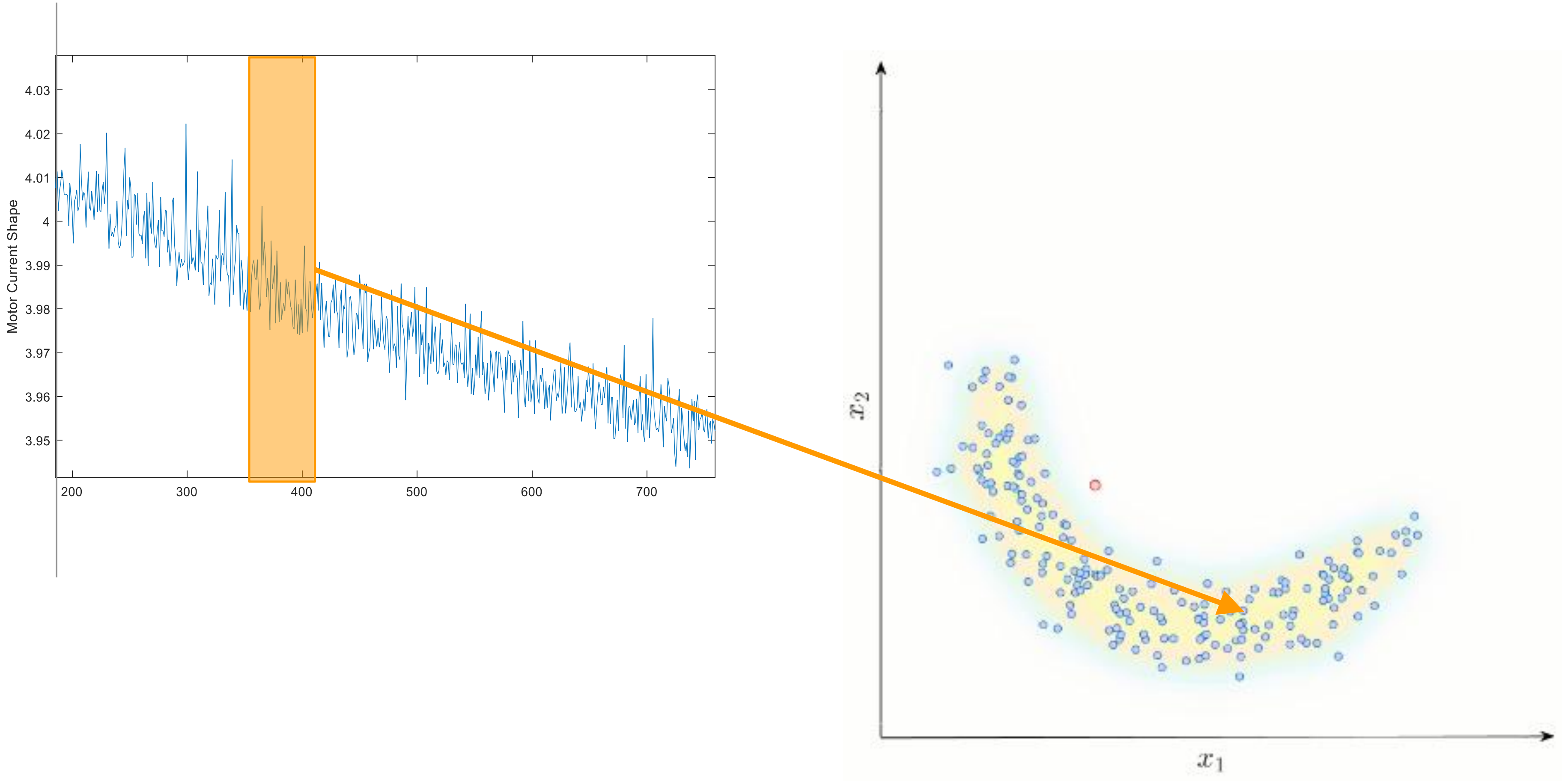
4. Ongoing changes in environment or system behavior

AI models reflect System behaviors and Environment

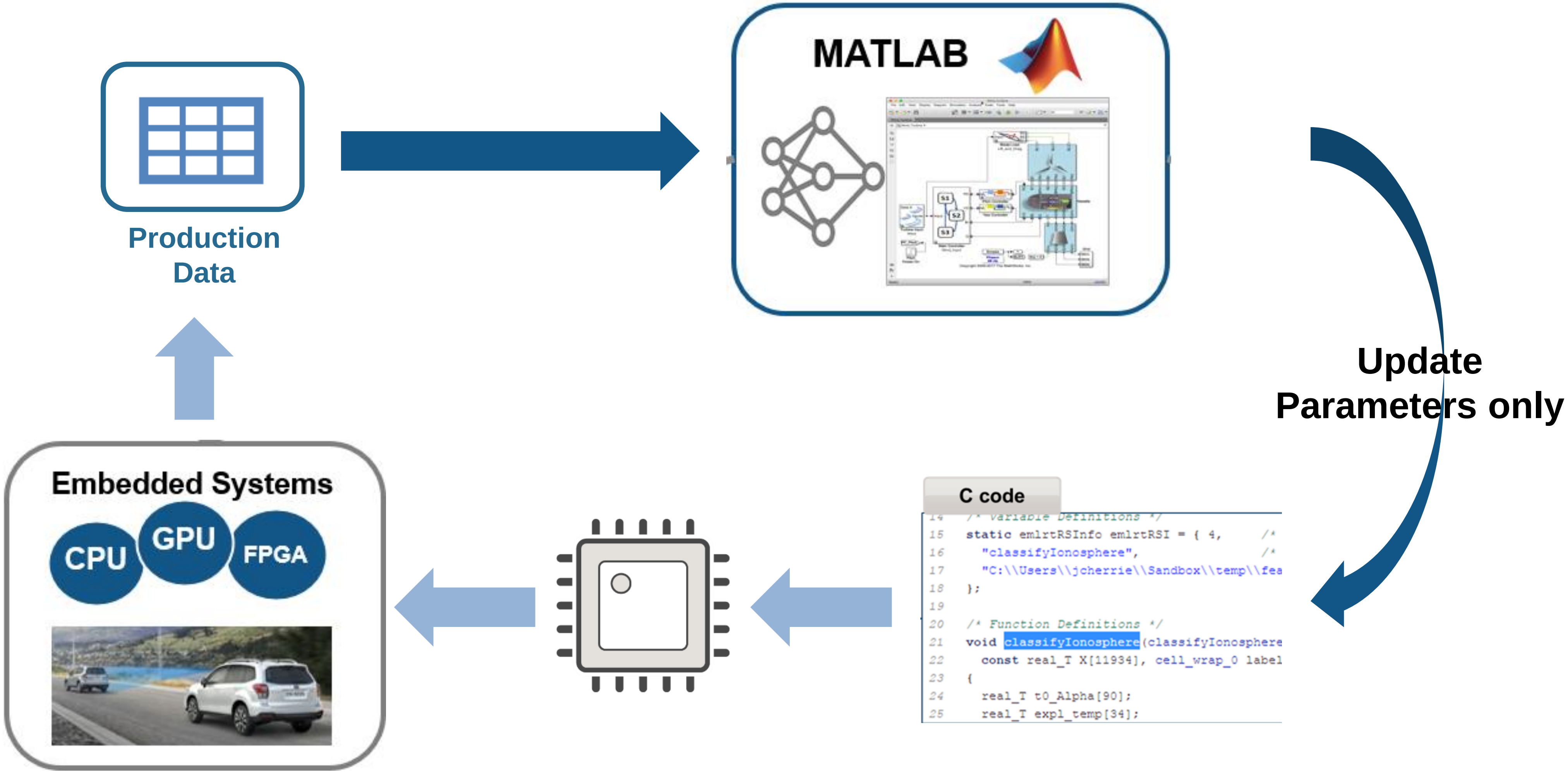


(illustration only; not based on actual data)

Deployed Models Need to Adapt.



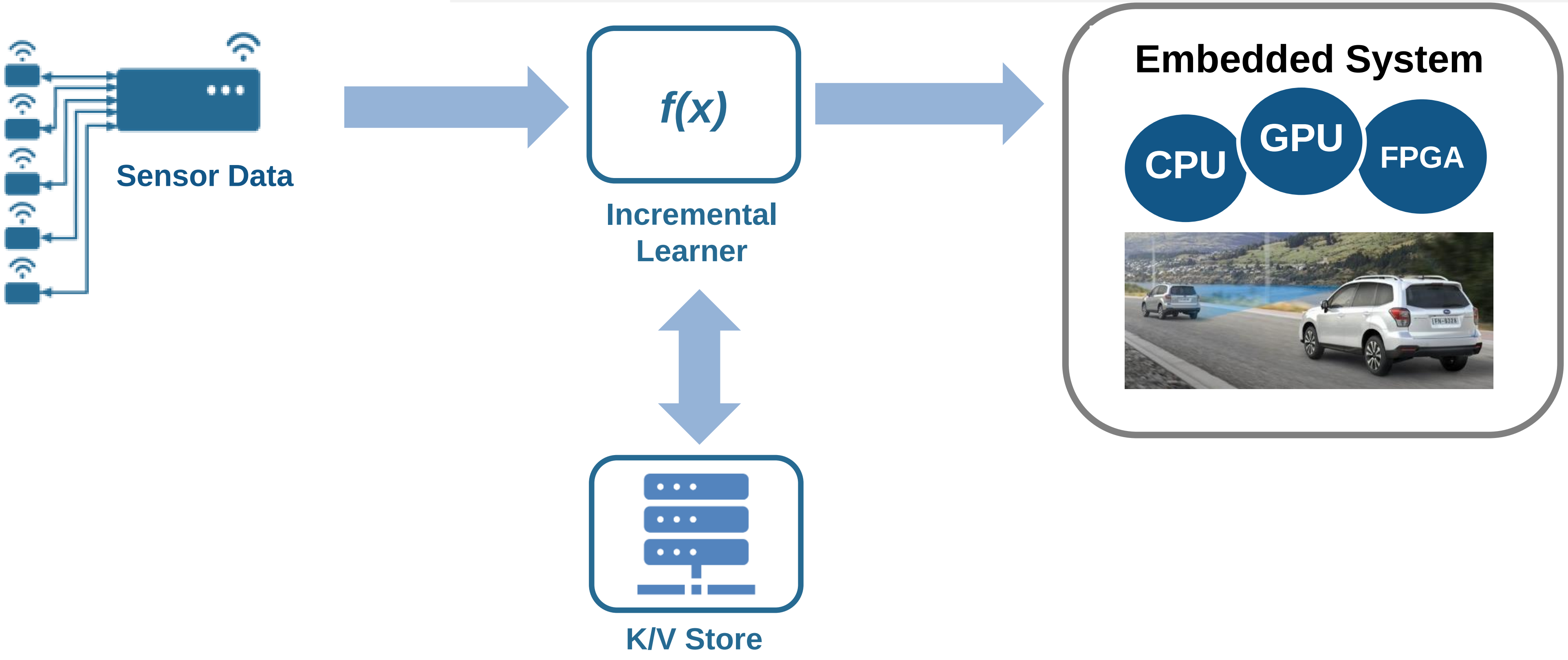
“In-place” Model Updates



Incremental Learning – on device training

```
incMdl = incrementalLearner(mdl);

while dataStreaming
   featureChunk = extractFeatures(streamdata);
   inclMdl = updateMetricsAndFit(incMdl,featureChunk,labels);
End
```



Embedded Deployment is about finding the right trade-off

✓ Memory footprint

Energy Cost

– Computation
– Data Access

Computation speed

Rough energy costs for various operations in 45nm 0.9V

Operation	Energy (pJ)
8b Add	0.03
32b FP Add	0.9
8b Multiply	0.2
32b FP Mult	3.7

Quantization greatly reduces energy consumption

	%Reduction
32b FP to 8b Add	97%
32b FP to 8b Multiply	95%

Source: Mark Horowitz "Computing's Energy Problem (and what we can do about it)", ISSCC 2014

Conclusions

Deploy to many targets from one codebase

Tools for handling these challenges in production deployments:

- Fit models to embedded hardware with Quantization / Fixed-Point conversion
- Simulink facilitates system integration, verification and testing
- Incrementally adapt deployed models

Learn More

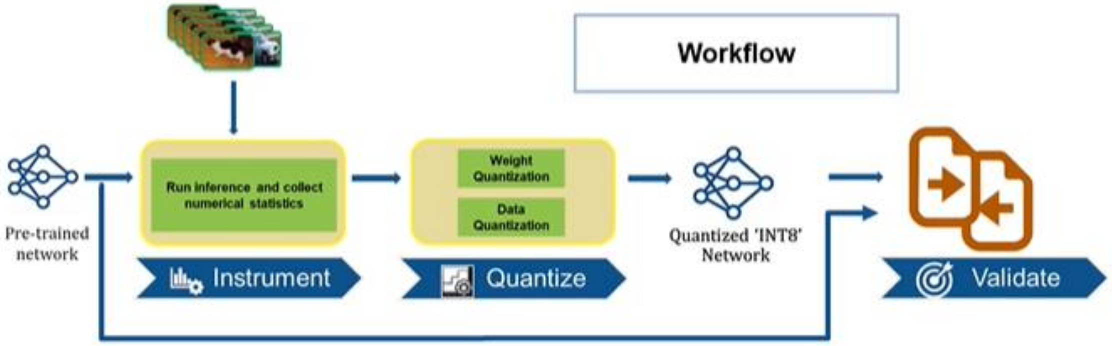
Machine Learning

- Fixed-Point Prediction with SVM
- Instrument MATLAB Code
- Update Model Parameters for Code Generation
- Incremental Learning with Logistic Regression
- Hand Gesture recognition on Arduino Nano

Deep Learning

- Quantize Residual Network Trained for Image Classification
- Video walk through Deep Network Quantizer

Deep Learning Model Quantization in MATLAB



Copyright Notice