tinyML® Talks

Enabling Ultra-low Power Machine Learning at the Edge

"Single Lead ECG Classification On Wearable and Implantable Devices"

Arijit Ukil – TCS Research

Gitesh Kulkarni - TCS Research

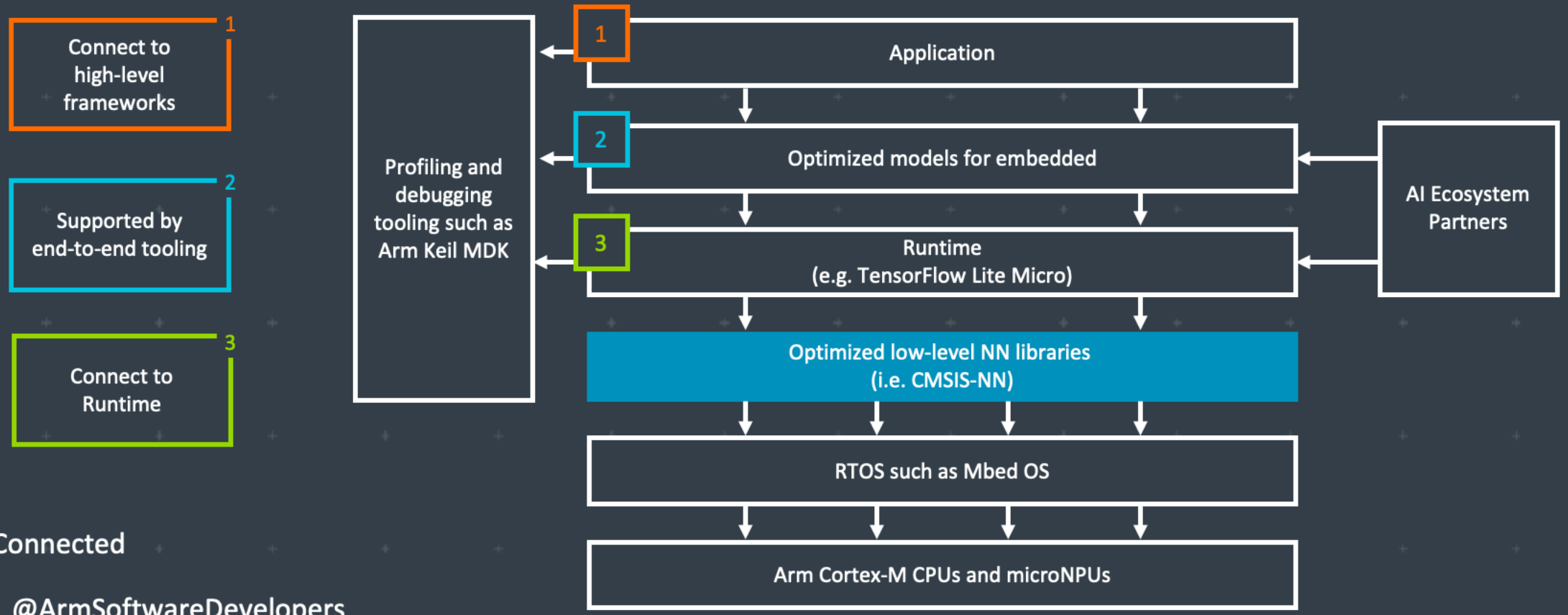
December 15, 2021

www.tinyML.org

tinyML Talks Strategic Partners

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**

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

Dataset

Acquire valuable training data securely

Enrich data and train ML algorithms

Edge Device

Real sensors in real time

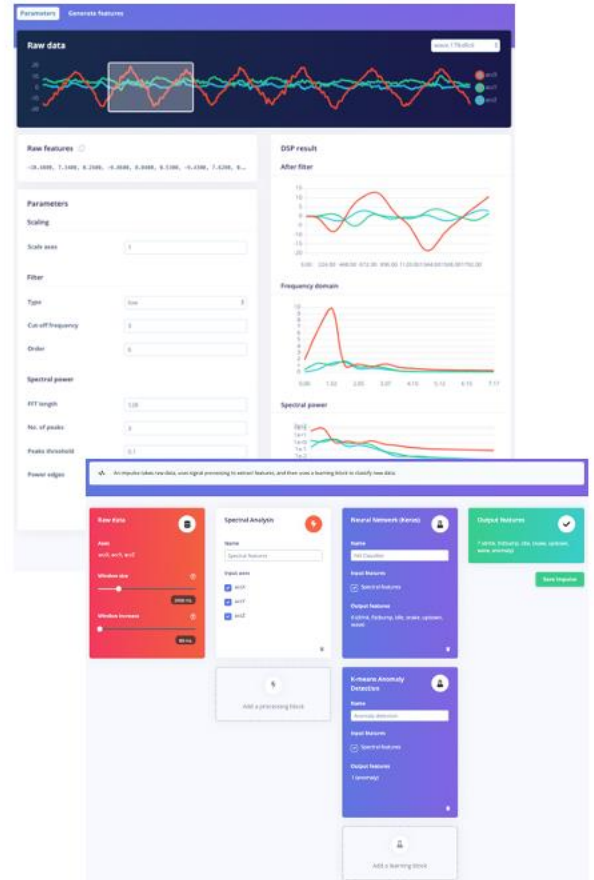
Open source SDK

Impulse

Embedded and edge compute deployment options

Test impulse with real-time device data flows

Test

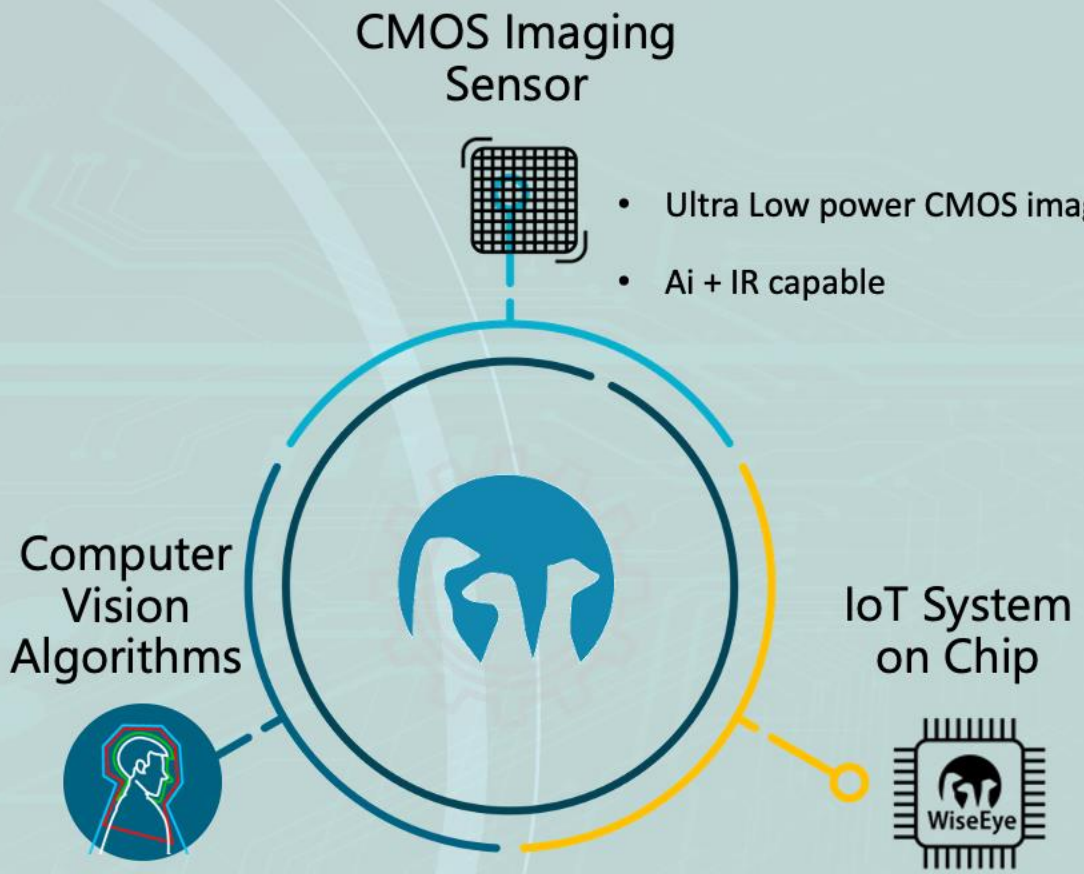


www.edgeimpulse.com

emza
visual sense

The Eye in IoT

Edge AI Visual Sensors



- Ultra Low power CMOS imager
- Ai + IR capable

- Machine Learning algorithm
- <1MB memory footprint
- Microcontrollers computing power
- Trained algorithm
- Processing of low-res images
- Human detection and other classifiers

- Machine Learning edge computing silicon
- <1mW always-on power consumption
- Computer Vision hardware accelerators

info@emza-vs.com

Enabling the next generation of Sensor and Hearable products to process rich data with energy efficiency

Visible Image

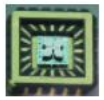
Sound

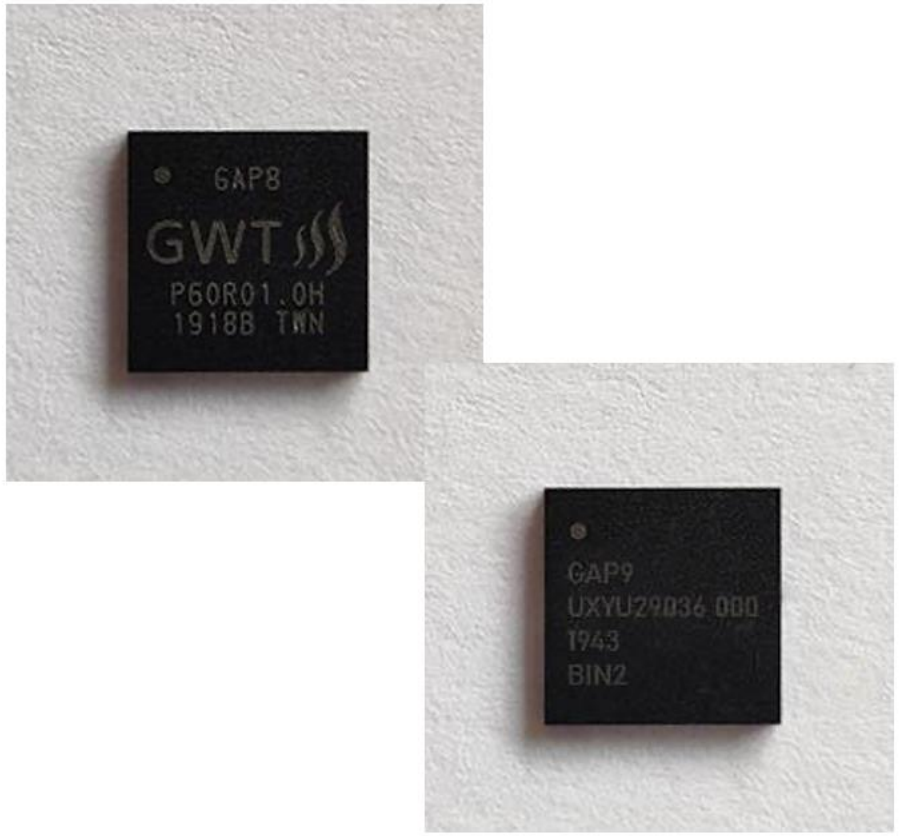
IR Image

Radar

Bio-sensor

Gyro/Accel

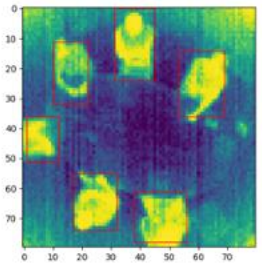




Wearables / Hearables

Battery-powered consumer electronics

IoT Sensors



GREENWAVES TECHNOLOGIES

Grovety Inc.

SOFTWARE DEVELOPMENT SERVICES FOR TINYML SOLUTIONS

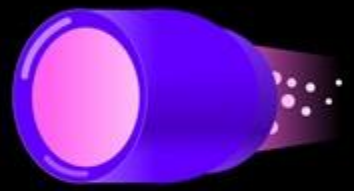
1 **Development tools**
SDK, IDE, compilers, leveraging on TVM, uTVM & LLVM

2 **Firmware**
Drivers, BSP, protocols, etc.

arm AI PARTNER

Distributed infrastructure for TinyML apps

HOTG
Decoupling intelligence



Develop at warp speed

Automate deployments

Device orchestration

HOTG is building the **distributed infrastructure** to pave the way for **AI** enabled **edge applications**

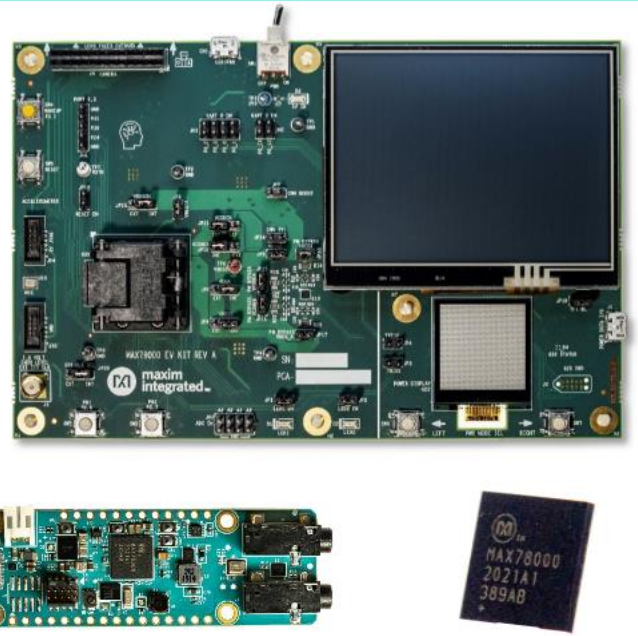
LatentAI

Adaptive AI for the Intelligent Edge

Latentai.com

Maxim Integrated: Enabling Edge Intelligence

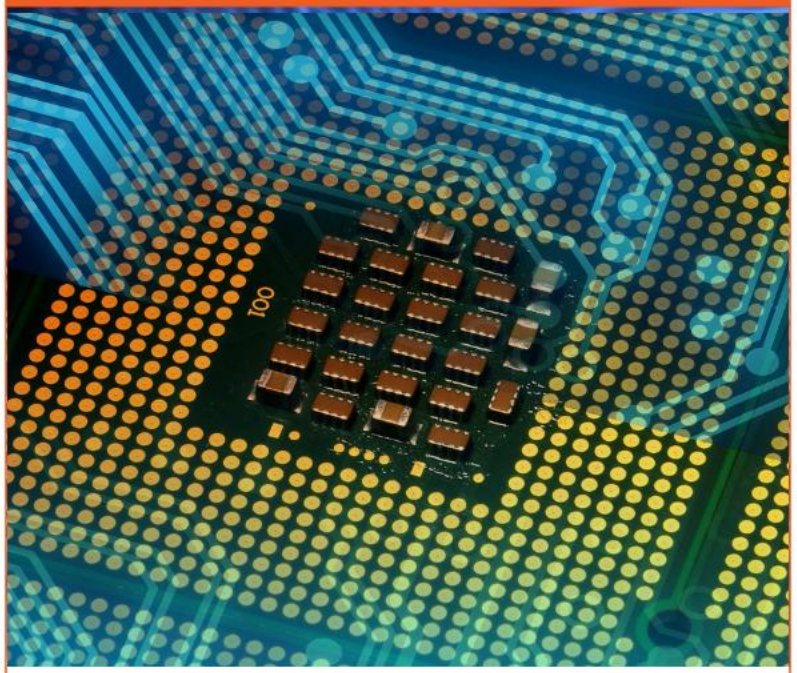
Advanced AI Acceleration IC



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

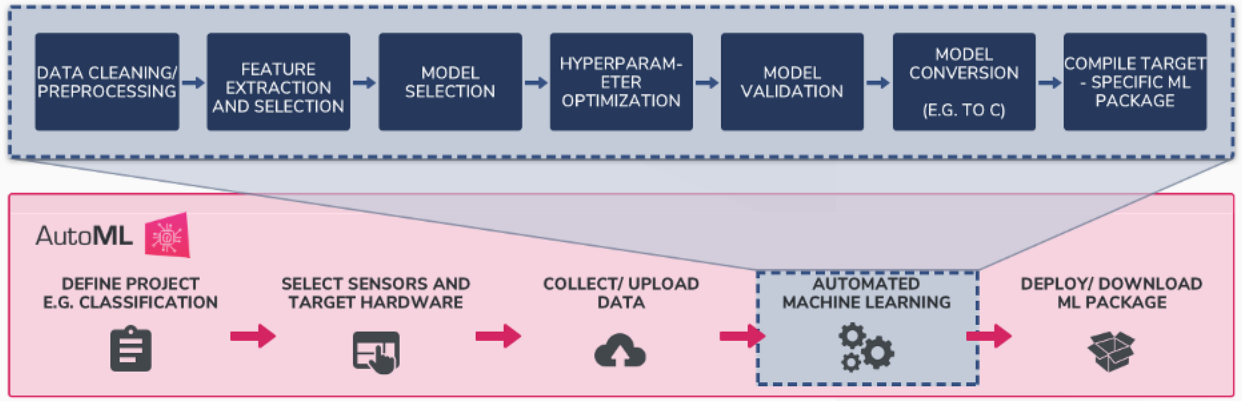
Qeexo AutoML

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

Qualcomm
AI research

Advancing AI research to make efficient AI ubiquitous

Power efficiency

Model design, compression, quantization, algorithms, efficient hardware, software tool

Personalization

Continuous learning, contextual, always-on, privacy-preserved, distributed learning

Efficient learning

Robust learning through minimal data, unsupervised learning, on-device learning

A platform to scale AI across the industry

Perception

Object detection, speech recognition, contextual fusion

Reasoning

Scene understanding, language understanding, behavior prediction

Action

Reinforcement learning for decision making

Edge cloud

Cloud

IoT/IIoT

Automotive

Mobile

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

BROAD AND SCALABLE EDGE COMPUTING PORTFOLIO

Microcontrollers & Microprocessors

Arm® Core

RA — Arm® Cortex®-M 32-bit MCUs
Arm ecosystem, Advanced security, Intelligent IoT

RZ — Arm®-based High-end 32 & 64-bit MPUs
High-resolution HMI, Industrial network & real-time control

RE — Arm® Cortex®-M0+ Ultra-low Power 32-bit MCUs
Innovative process tech (SOTB), Energy harvesting

Renesas Synergy™ — Arm®-based 32-bit MCUs for Qualified Platform
Qualified software and tools

Renesas Core

RL78 — Ultra-low Energy 8 & 16-bit MCUs
Bluetooth® Low Energy, SubGHz, LoRa®-based Solutions

RX — High Power Efficiently 32-bit MCUs
Motor control, Capacitive touch, Functional safety, GUI

RH850 — 40nm/28nm process Automotive 32-bit MCUs
Rich functional safety and embedded security features

Core technologies

AI

A broad set of high-power and energy-efficient embedded processors

Security & Safety

Comprehensive technology and support that meet the industry's stringent standards

Digital & Analog & Power Solution

Winning Combinations that combine our complementary product portfolios

Cloud Native

Cross-platforms working with partners in different verticals and organizations

BIG IDEAS FOR EVERY SPACE RENESAS

seeed studio

The IoT Hardware Enabler

SensiML™

Build Smart IoT Sensor Devices From Data

SensiML pioneered TinyML software tools that auto generate AI code for the intelligent edge.

- End-to-end AI workflow
- Multi-user auto-labeling of time-series data
- Code transparency and customization at each step in the pipeline

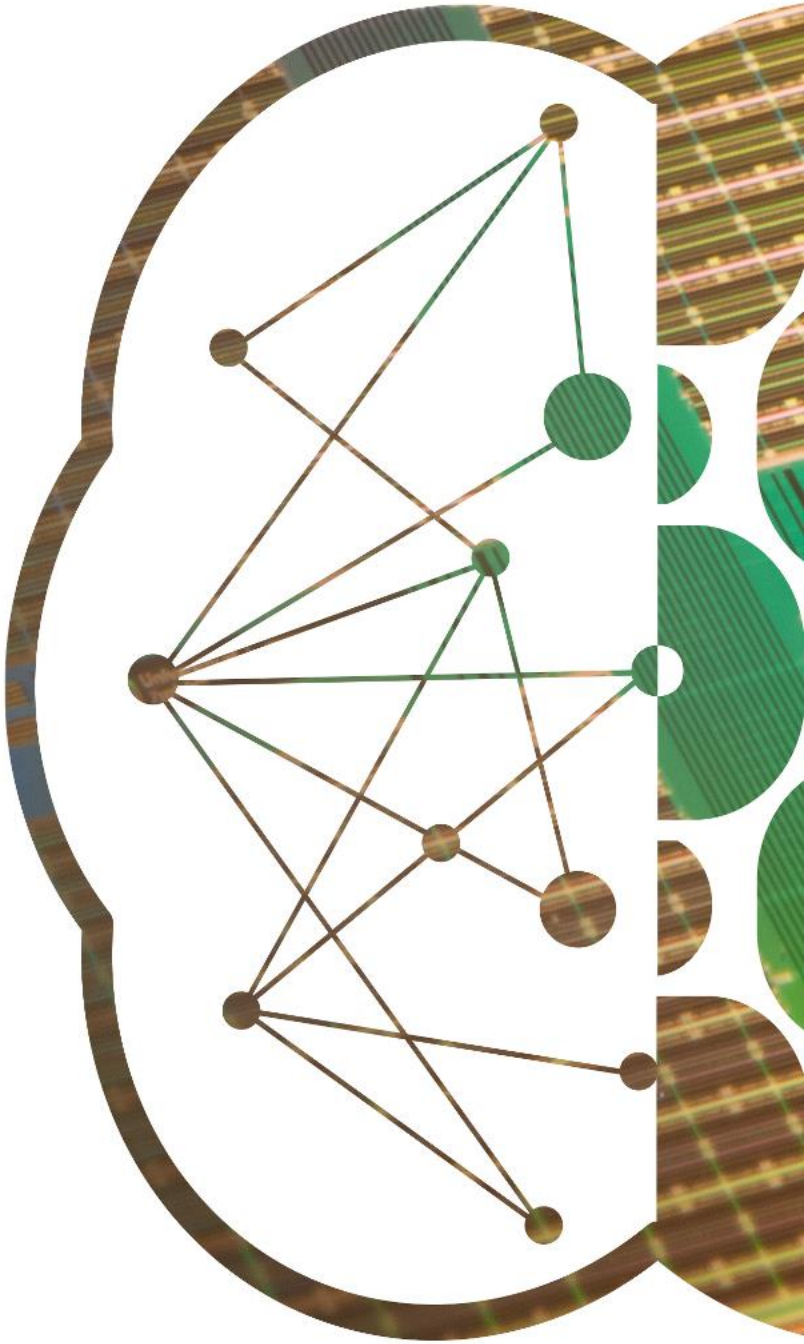
We enable the creation of production-grade smart sensor devices.

sensiml.com

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

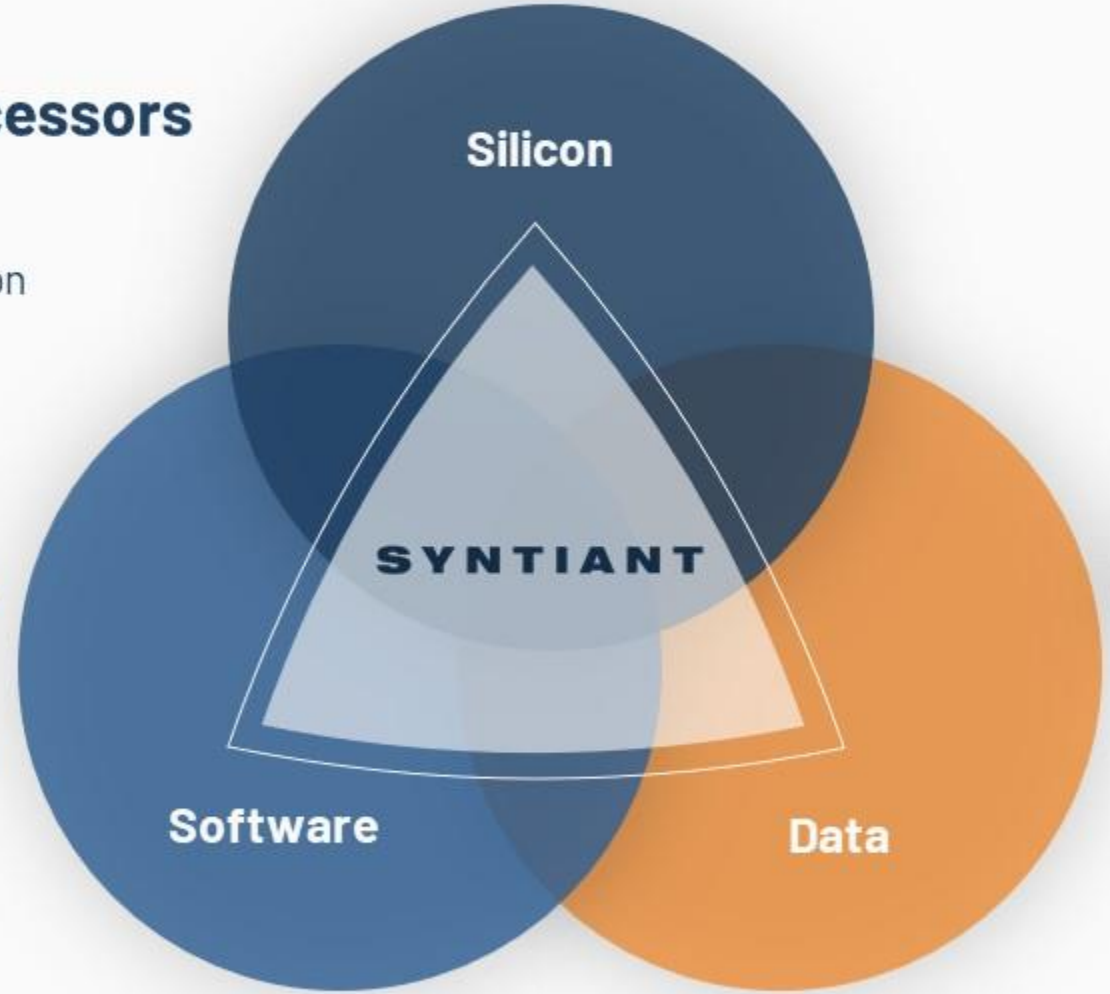


SYNTIANT

End-to-End Deep Learning Solutions for TinyML & Edge AI

Neural Decision Processors

- At-Memory Compute
- Sustained High MAC Utilization
- Native Neural Network Processing



ML Training Pipeline

- Enables Production Quality Deep Learning Deployments

Data Platform

- Reduces Data Collection Time and Cost
- Increases Model Performance

partners@syntiant.com

www.syntiant.com

tinyML Summit 2022

Miniature dreams can come true…

March 28-30, 2022

Hyatt Regency San Francisco Airport

https://www.tinyml.org/event/summit-2022/

Registration will be open on **December 15**, 2021.

Deadline for poster submission is **December 17**.

*The Best Product of the Year and the Best Innovation of the Year awards are open for nominations between **November 15 and February 28.***

tinyML Research Symposium 2022

March 28, 2022

https://www.tinyml.org/event/research-symposium-2022

Call for papers – Submission deadline is **December 17**, 2021.

More sponsorships are available: sponsorships@tinyML.org

Next tinyML Talks

Date	Presenter	Topic / Title
Thursday, December 16	Alasdair Allan, Raspberry Pi	Standing at the Edge, Looking into the Future

Webcast start time is 8:00 am Pacific time

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

TINY ML TALKS webcast

Reminders

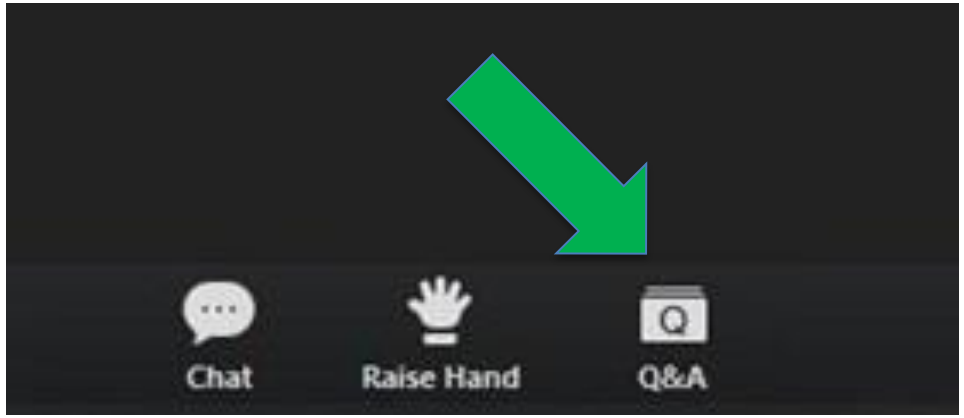
Slides & Videos will be posted tomorrow

PDF

tinyml.org/forums

youtube.com/tinyml

Please use the Q&A window for your questions



Local Committee in India

Chetan Singh Thakur, PhD
Assistant Professor at the Indian Institute of Science (IISc), Bangalore. He is a Ph.D. in neuromorphic engineering. Dr. Thakur's research interest spans VLSI Design, Edge Computing, Neuromorphic Engineering.

Sandipan Chatterjee
Sandipan is Lead Data scientist at DXC Technology where he develops and implements vision-based automation in manufacturing, automotive and healthcare. He has a background in image and statistical analysis.

Arijit Das
Arijit is a 15-year-old high-schooler. He is the youngest Ambassador for Edge Impulse and has been in the AIoT field since 2017. His interests include Edge Computing, EdgeAI, and Low-Power Wide Area Networks.

Anup Rajput
Co-founder at Envir AI, trying to bring ML into the real world. Anup has a background in semiconductor design and applied ML from edge to cloud.

Abhishek Nair
Abhishek is a PhD student at IISc Neuronics lab. His research area includes exploring low power ML algorithms for digital hardware implementation.

Follow us for more updates at:
https://www.linkedin.com/company/tinyml-india

TINY ML TALKS webcast

Arijit Ukil

Arijit Ukil is having more than 18 years of industrial research experience in different capacities. He is working as Senior Scientist in TCS Research, Tata Consultancy Services, India. He has published more than 50 research papers in distinguished conferences and journals. He has authored 4 book chapters. He has filed more than 40 patents with more than 30 grants in different geographies including Europe, China, USA, Japan, Australia, India. He holds Master's in Engineering from Jadavpur University, Kolkata, India. He is a Senior Member, IEEE. He is steering committee member of HealthyIoT, 2016, 2017 and the General Chair in KDAH-CIKM-2018, 2019, 2020, 2021.

TINY ML TALKS webcast

Gitesh Kulkarni

Gitesh Kulkarni is working as a Scientist in TCS research. He is an accomplished designer of embedded systems and a Maker at heart. In one of his pioneering works, he was the lead designer of the world's first industrial safety watch for the TATA group. He is a Master of Science in Electrical engineering from Colorado State University, Fort Collins, Colorado, USA. His research interests are at the cusp of edge computing, computer architectures, and sensing. He has several filed and granted patents in edge computing and wearable devices. Gitesh is a member of IEEE and ACM. Before joining TCS, Gitesh created embedded systems and systems products for leading technology companies with total experience of more than 19 years. He is a licensed Ham radio operator as well.

tcs Research

Inventing for impact

Single Lead ECG Classification On Wearable and Implantable Devices

Arijit Ukil and Gitesh Kulkarni
TCS Research, India

Single Lead ECG Classification On Wearable and Implantable Devices - Agenda

- The global challenge of Cardio-Vascular Disease (CVD)
- Current Solution - Techniques and Challenges
- Issues in using AI/ML in embedded systems due to Large models
- Large models in embedded systems -solutions
- "ECG TinyML" - a Piecewise Linear Approximation (PLA) approach
- Dataset and Deployment setup
- Experimental process and results
- Result Summary

Single Lead ECG Classification On Wearable and Implantable Devices – TCS Publication and Patents

This talk is based on the following TCS publications and patents

- Publications

  Ishan Sahu, Arpan Pal, Arijit Ukil, and Angshul Majumdar “Compressing Deep Neural Network: A Black-Box System Identification Approach” International Joint Conference on Neural Networks (IJCNN), 2021.

  Arijit Ukil, Ishan Sahu, Angshul Majumdar, Sai Chander Racha, Gitesh Kulkarni, Anirban Dutta Choudhury, Sundeep Khandelwal, Avik Ghose, Arpan Pal, "Resource Constrained CVD Classification Using Single Lead ECG On Wearable and Implantable Devices," IEEE EMBC, 2021.

- Patent

  India Patent Filing Details -

  CAP Application No: 202021053450

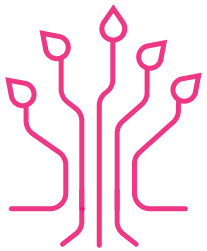
  Date of Filing: 30/03/2021

  Title: METHOD AND SYSTEM FOR DYNAMIC COMPRESSION OF DEEP NEURAL NETWORK (DNN)

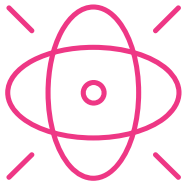
Embedded Devices & Intelligent Systems

We conduct research in Intelligent Sensing Systems that gives rise to perceptive machines in different industry verticals.

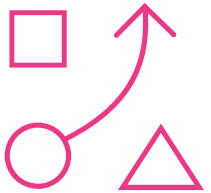
Research themes we work on



Machine Sensing



People Sensing



Device Edge Computing

Device Edge: WHY and WHAT

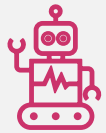
AI at Edge -

- Consumer Edge – AI cores as part of SoC
- Enterprise Edge – AI cores, standalone ASIC on robots, drones, autonomous cars

TinyML: energy cost < 1mW (Warden & Situnayake)
Heterogenous embedded devices, microcontrollers which are constrained in terms of CPU, memory, bandwidth, cache, and battery

Objectives: reduce latency, preserve privacy, improve reliability, reduce cost

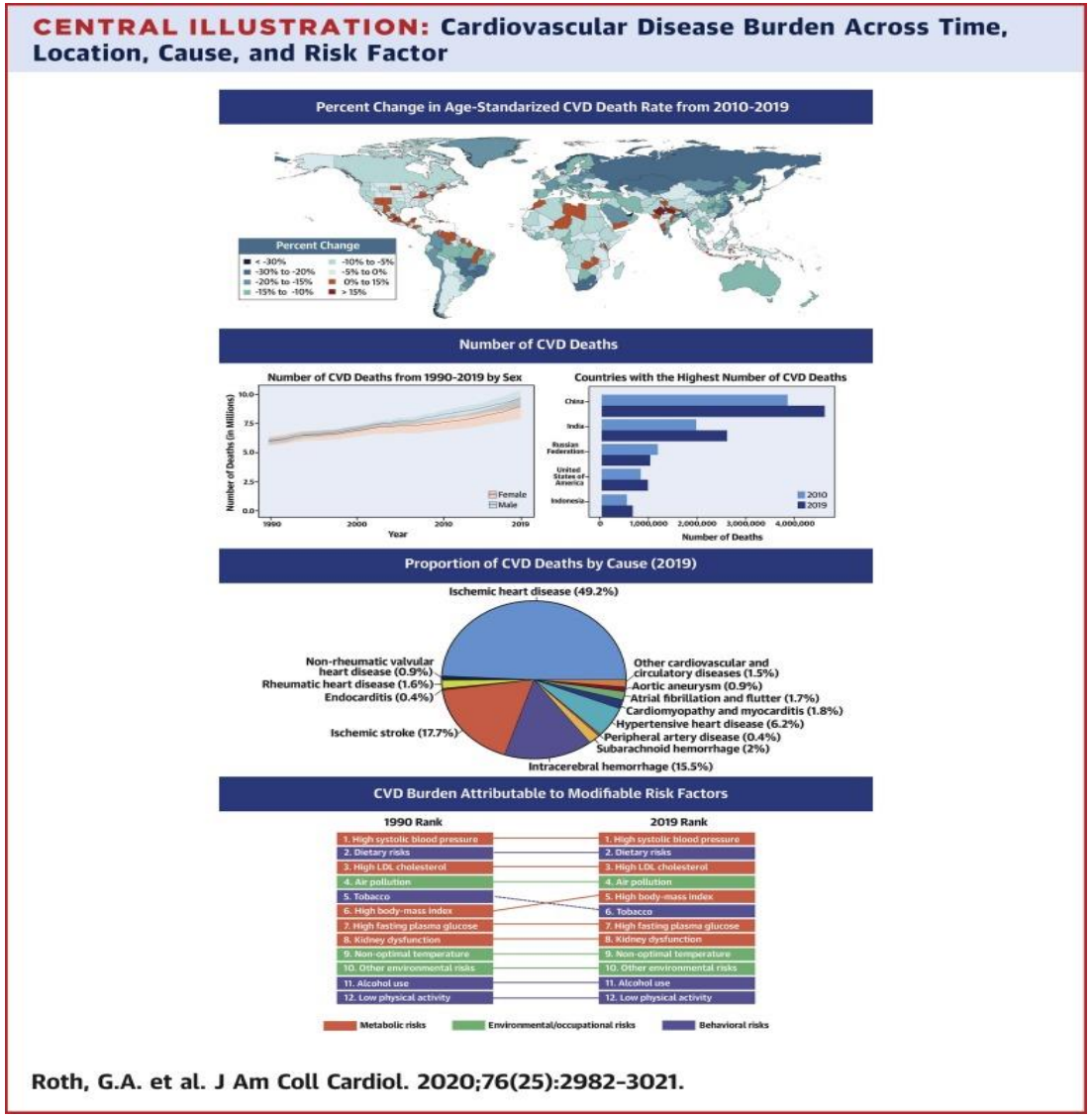
Challenges: constrained in terms of CPU, memory, bandwidth, battery, size



Applications: Real-time Sensing, Control and Rendering, Privacy-preserving Local Analytics – in IoT, robotics, space, AR/VR/XR etc

The global challenge of Cardio-Vascular Disease (CVD)

- Cardio-Vascular disease – a leading cause of deaths in developed and developing countries[1]
- Increasing number of deaths in all age groups[1]
- As per Center for Disease prevention and Control, USA -
  - One person dies **every 36 seconds** in the United States from cardiovascular disease[2]
  - **1 in 5** heart attacks is silent[2]



1. Roth, G. A et al. J Am Coll Cardiol. 2020; 76(25):2982-3021
2. https://www.cdc.gov/heartdisease/facts.htm, Accessed 01 Oct 2021

Current Solution - Techniques and Challenges

Prevention of CVD - Needs regular screenings at a sophisticated facility

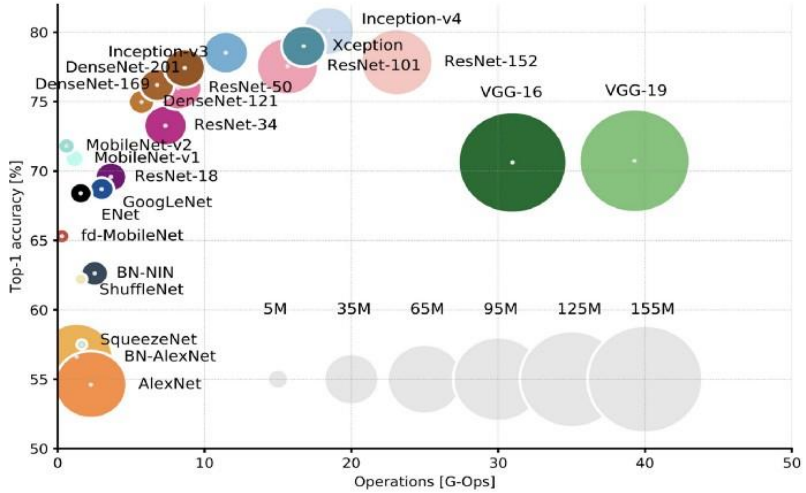
Such continuous and even remote monitoring of heart condition now possible due to wide availability of ECG sensors

- Using a wearable ECG analytics devices
- Implantable loop recorder (ILR)

https://www.fitbit.com/pl/shop/surge

But wearables (and implantables) need the inferencing model size to be as small as possible and still maintain accuracy of original model to be useful clinically

Typical MCUs in wearables are ARM Cortex M0 to Cortex M7 with limited flash memory from 32KB to 1MB and RAM sizes ranging from 10KB to 256KB



An Analysis of Deep Neural Network Models for Practical Applications by Canziani et. Al

https://culurciello.medium.com/analysis-of-deep-neural-networks-dcf398e71aae, Accessed 01 Oct 2021

Typical Wearable devices Hardware

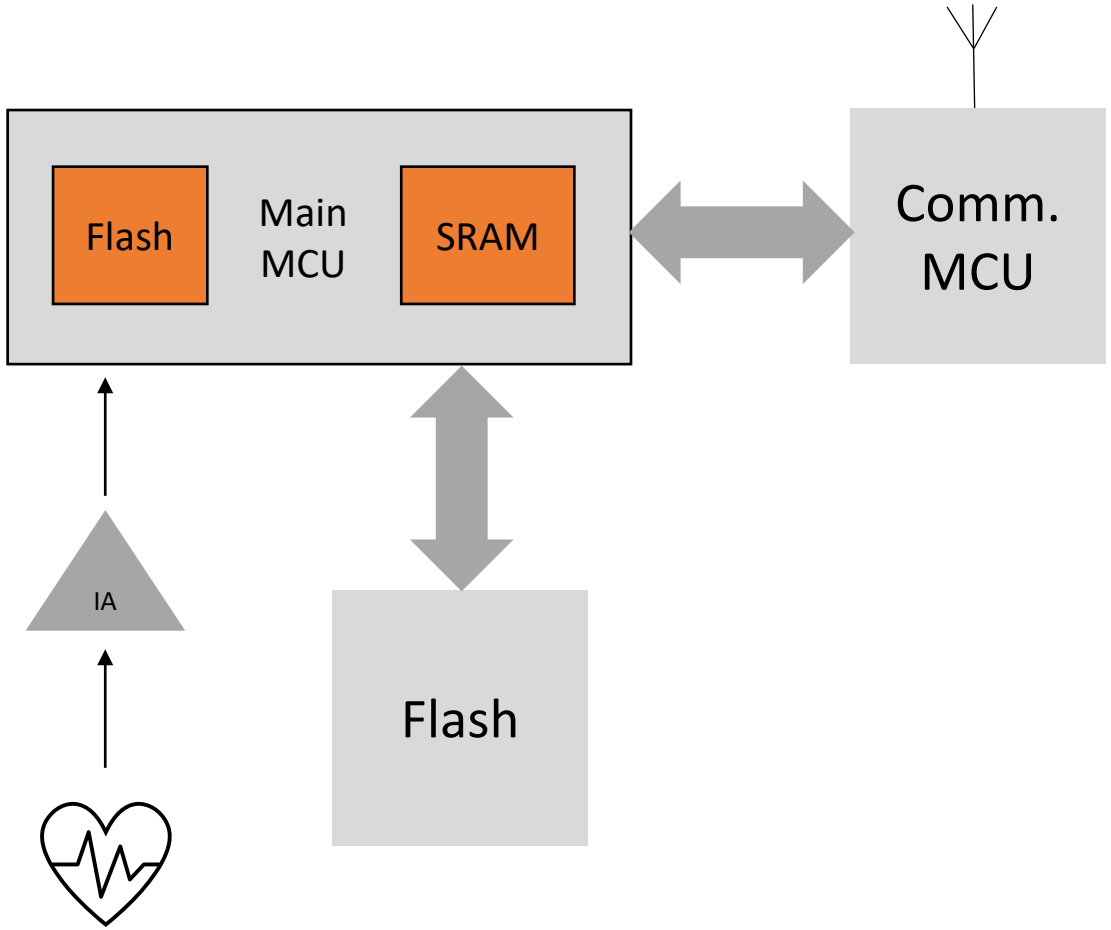
Make, Model	CPU	Speed	Flash	RAM	External Flash	Battery
Fitbit Surge *	ARM Cortex-M3	48 MHz	1024 KB	128 KB	64Mb	100mAh
MI band 3 ~	ARM Cortex-M0	Upto 96 MHz	• 64 kB (OTP) • 128 kB ROM	• 128 kB Data SRAM • 16 kB Cache SRAM	32 Mb	110mAh
Sony Smart Band 2 #	ARM Cortex-M0	--	256KB	32KB	--	100mAh

* https://www.ifixit.com/Teardown/FitBit+Surge+Teardown/42344

~ https://www.ifixit.com/Teardown/Mi+Band+3+Teardown/139902

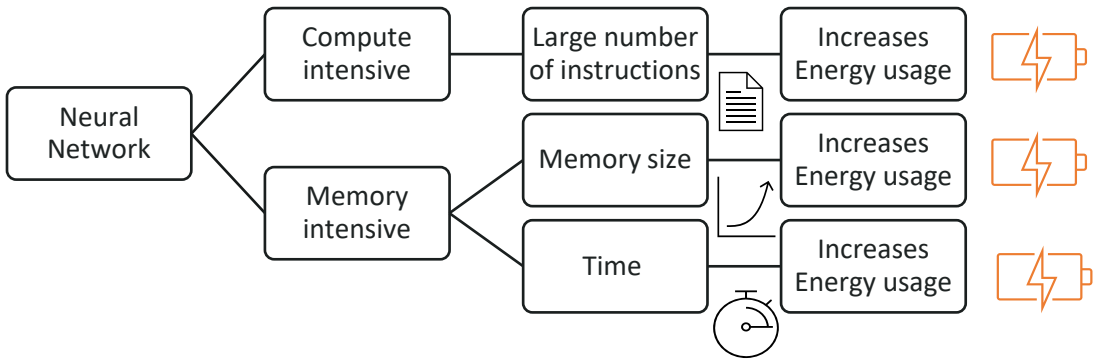
A survey of Wearable devices and Challenges by Seneviratne et. al

Typical Design of a Wearable



- Heart rate sensor
- Main MCU to run the algorithm and data capture
- Internal Flash and RAM
- (Optional) External Flash
- (Optional) Communication MCU for external Bluetooth Low Energy communication

Issues in using AI/ML in embedded systems due to large models

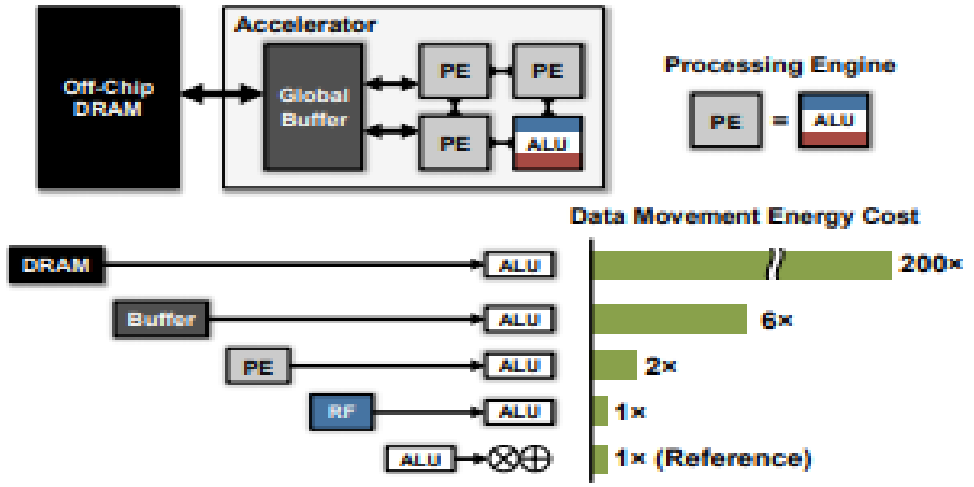


All options lead to more energy consumption and execution time -

- Large number of instruction due to large models
- Large networks do not fit in on-chip storage and hence require the external memory

Leading to more energy consumption and to increased execution time

Not useful for a good consumer experience



Hardware for Machine Learning: Challenges and Opportunities by Sze et. al

Approximate timing for various operations on a typical PC:

execute typical instruction	1/1,000,000,000 sec = 1 nanosec
fetch from L1 cache memory	0.5 nanosec
branch misprediction	5 nanosec
fetch from L2 cache memory	7 nanosec
Mutex lock/unlock	25 nanosec
fetch from main memory	100 nanosec
send 2K bytes over 1Gbps network	20,000 nanosec
read 1MB sequentially from memory	250,000 nanosec
fetch from new disk location (seek)	8,000,000 nanosec
read 1MB sequentially from disk	20,000,000 nanosec
send packet US to Europe and back	150 milliseconds = 150,000,000 nanosec

http://norvig.com/21-days.html#answers, Accessed 01 Oct 2021

Practical Challenges and Motivation

Motivation and use case

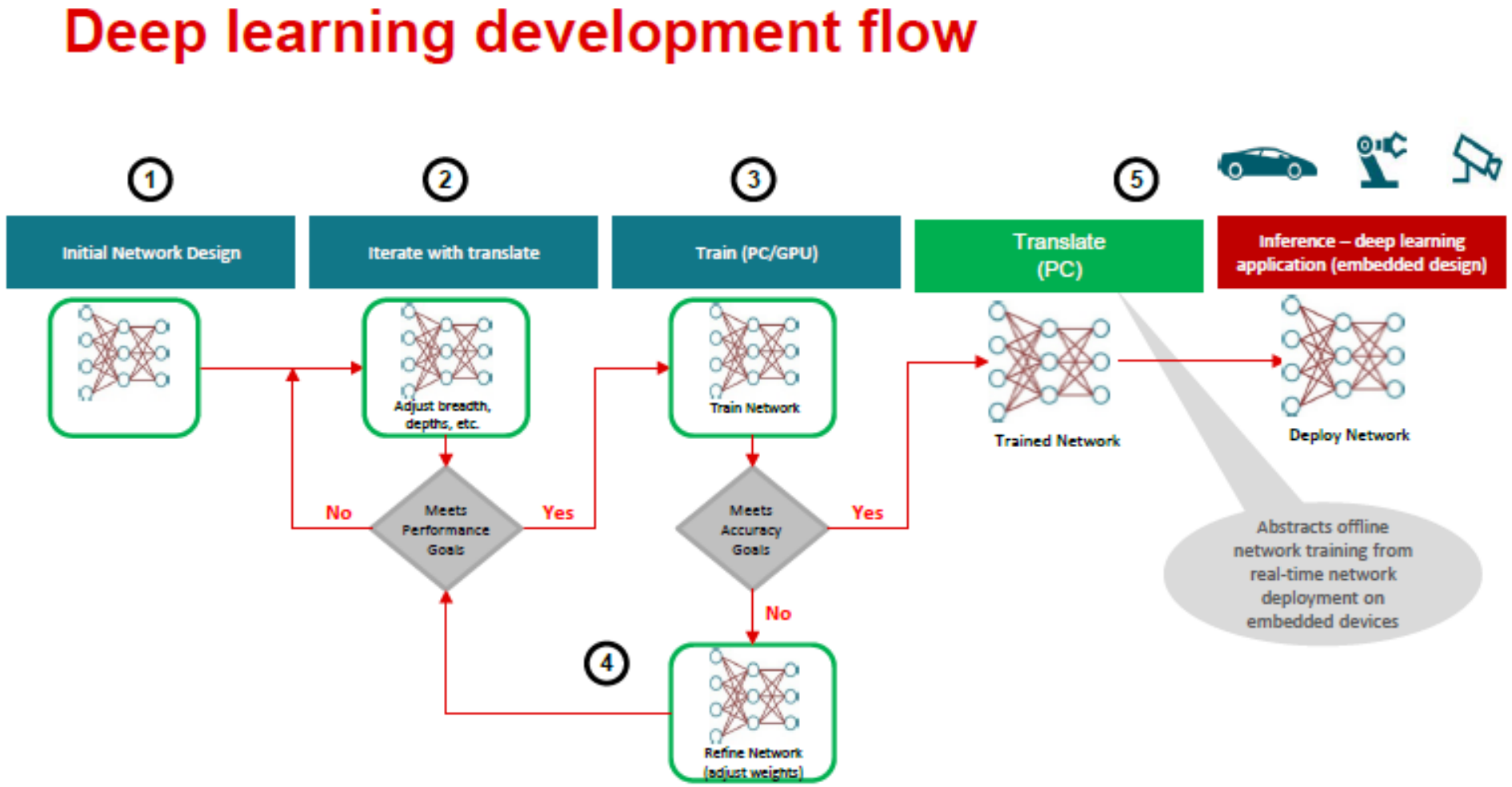
- Wearable devices with single lead ECG sensor can be used to detect heart arrhythmias and life-threatening Atrial Fibrillation condition
- Recent Deep Learning models can demonstrate high quality performance for the ECG classification task*
- Such models have high memory requirement which is not guaranteed in small devices like microcontroller unit (MCU)

Challenges

- Typical DL model for 1D signal classification (ResNet) ~ 10 MB, Stanford ECG model > 100MB
- Typical MCU RAM < 500 KB
- The classification performance (in terms of accuracy, F1-score,…) of the base DL model and compressed DL model in MCU should not differ much (within a budget of 2 – 3%)

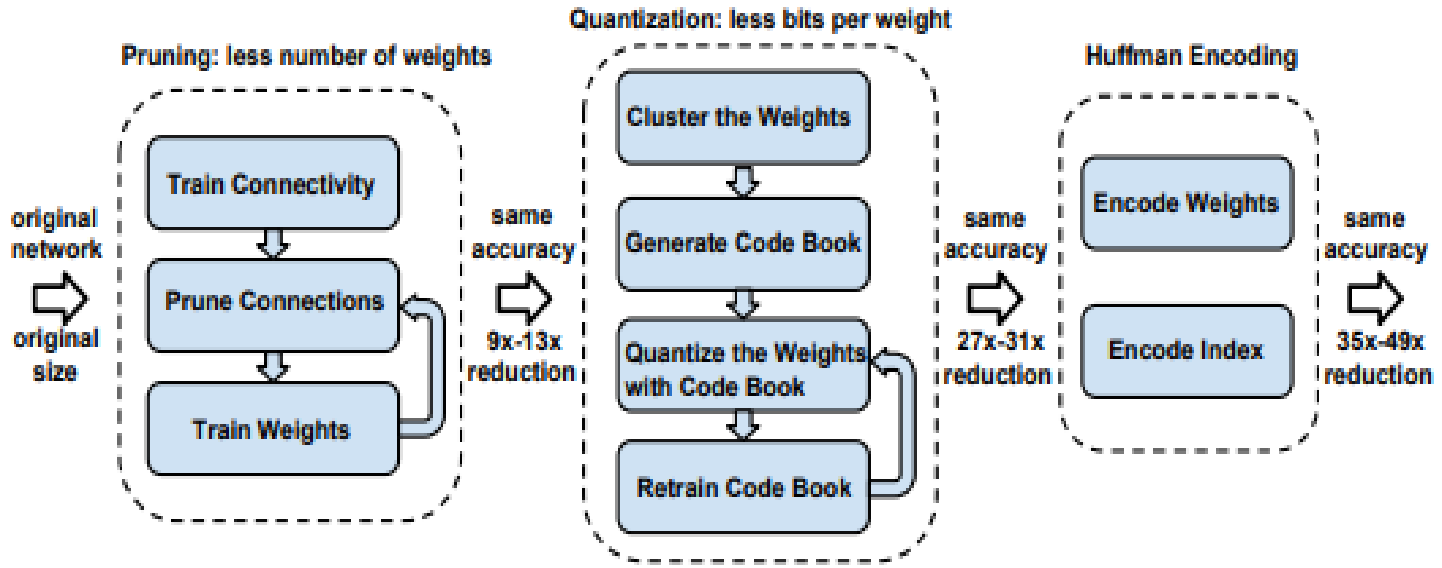
* Andrew Y. Ng et al. , "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," Nature, 2019.

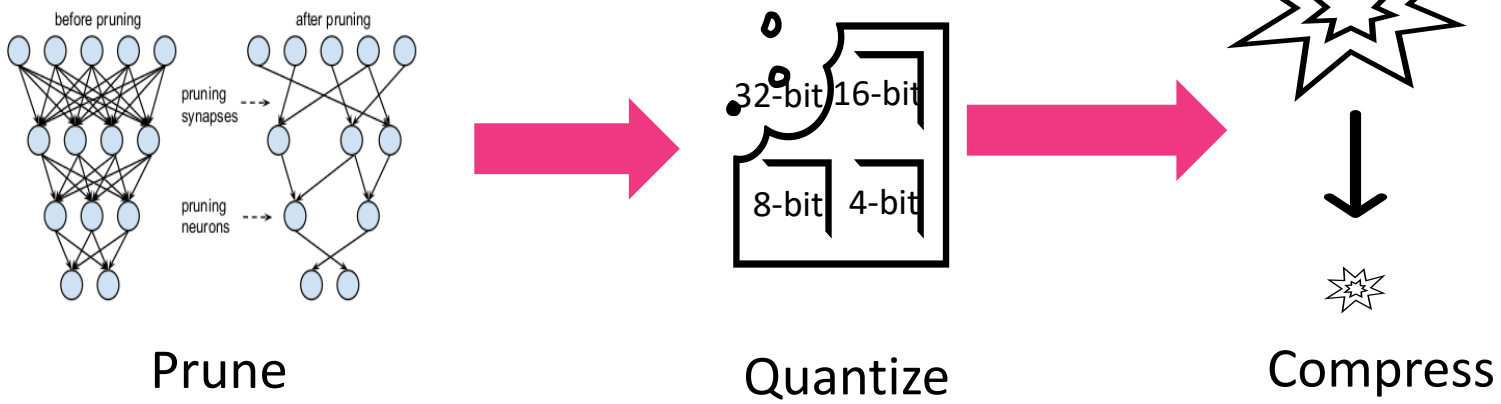
Deep learning model- development to deployment



Source- Texas Instrument

Large models in embedded systems - solutions





Prune

Quantize

Compress

- Model Pruning
- Quantization
- Network Compression

These methods are not sufficient to reduce the baseline DNN for MCU-based ECG analytics -

- Knowledge Distillation

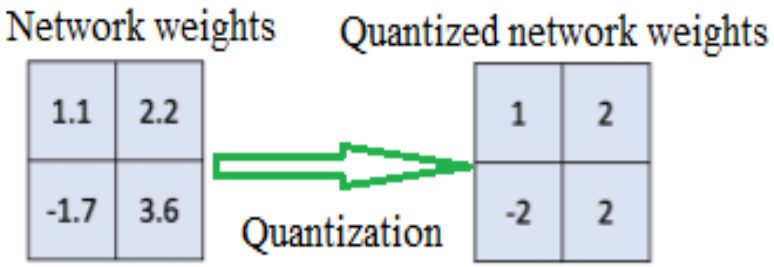
A Deep Neural Network Compression Pipeline: Pruning, Quantization, Huffman Encoding by Song Han et. Al

Learning both Weights and Connections for Efficient Neural Networks - Song Han et. al

Different approaches

Quantization:

Quantization in general is the process of mapping values from a large set to values in a smaller set, meaning that the output consists of a smaller range of possible values than the input, ideally without losing too much information in the process.



Assumption: Network weights are over-precisioned

Different approaches

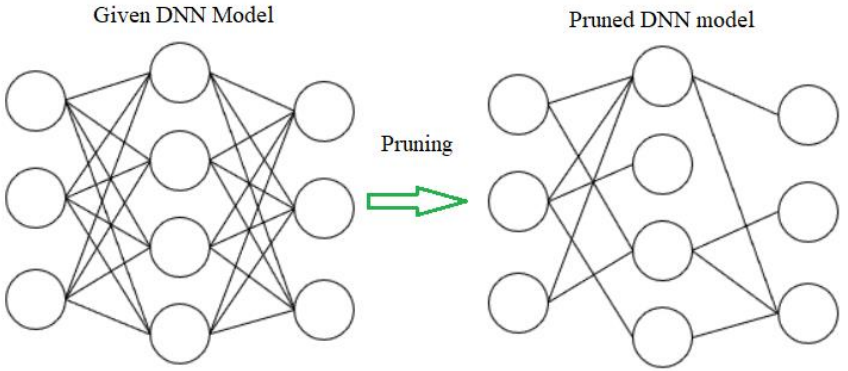
Low rank factorization:

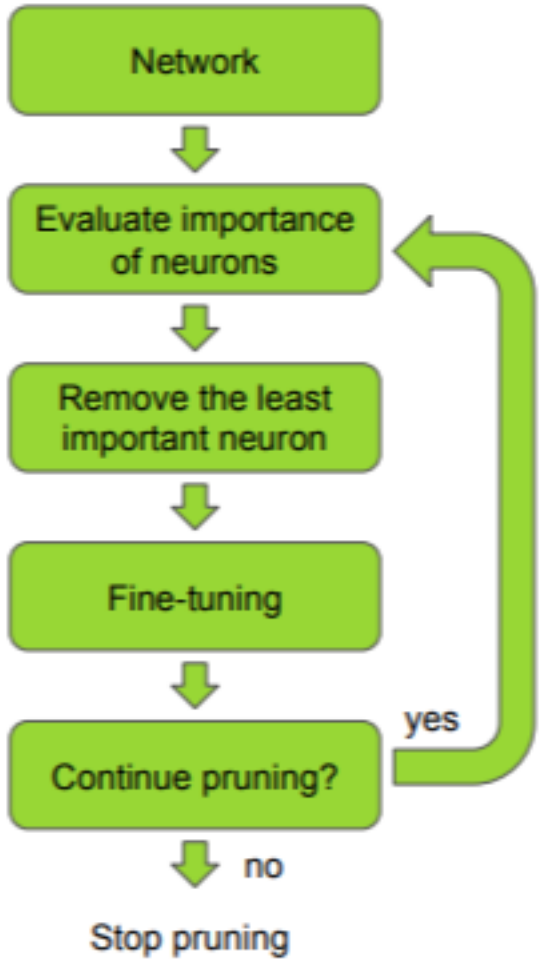
The goal of low-rank approximation is to approximate the numerous redundant filters of a layer using a linear combination of fewer filters. This uses matrix/tensor decomposition to estimate the informative parameters.

Assumption: Redundancy in the network parameters

Different approaches

Pruning:





Pruning involves removing connections between neurons or entire neurons, channels, or filters from a trained network, which is done by zeroing out values in its weights matrix or removing groups of weights entirely; for example, to prune a single connection from a network, one weight is set to zero in a weight's matrix,….

Assumption: Network is over-parametrized

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, Jan Kautz, “Pruning Convolutional Neural Networks For Resource Efficient Inference ,” ICLR, 2017.

Different approaches

Lottery Ticket Hypothesis*:

A randomly-initialized, dense neural network contains a subnetwork that is initialized such that—when trained in isolation—it can match the test accuracy of the original network after training for at most the same number of iterations.

Assumption: Equivalent sub-network exists

Identifying winning tickets. We identify a winning ticket by training a network and pruning its smallest-magnitude weights. The remaining, unpruned connections constitute the architecture of the winning ticket. Unique to our work, each unpruned connection's value is then reset to its initialization from original network *before* it was trained. This forms our central experiment:

1. Randomly initialize a neural network $f(x; \theta_0)$ (where $\theta_0 \sim \mathcal{D}_\theta$).
2. Train the network for j iterations, arriving at parameters θ_j.
3. Prune $p\%$ of the parameters in θ_j, creating a mask m.
4. Reset the remaining parameters to their values in θ_0, creating the winning ticket $f(x; m \odot \theta_0)$.

As described, this pruning approach is *one-shot*: the network is trained once, $p\%$ of weights are pruned, and the surviving weights are reset. However, in this paper, we focus on *iterative pruning*, which repeatedly trains, prunes, and resets the network over n rounds; each round prunes $p^{\frac{1}{n}}\%$ of the weights that survive the previous round. Our results show that iterative pruning finds winning tickets that match the accuracy of the original network at smaller sizes than does one-shot pruning.

*Jonathan Frankle, Michael Carbin, "The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks," ICLR, 2019.

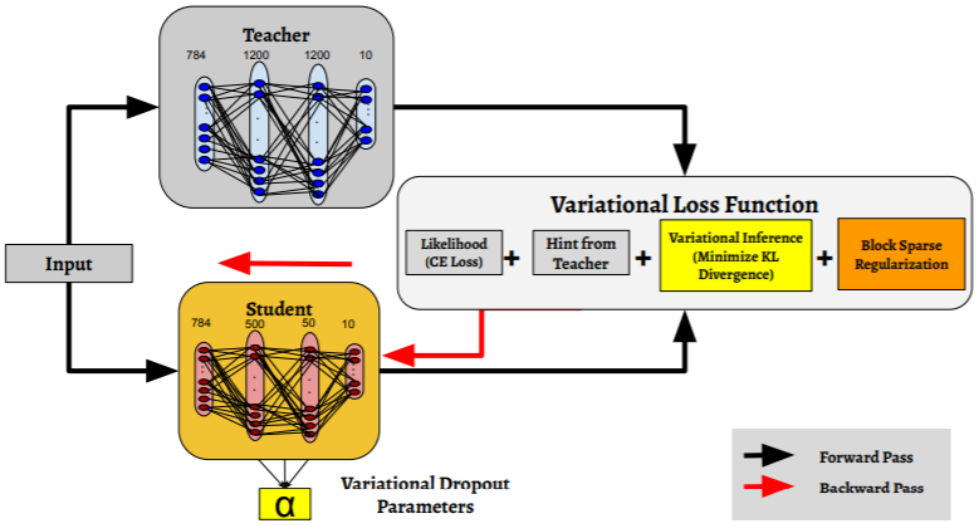
Different approaches

Knowledge distillation (KD):

It is transferring the knowledge from a large trained model to a smaller model for deployment by training it to mimic the larger model's output.

Assumption: Network is over-parametrized (overly complex)

Variational Student Approach



Knowledge Distillation (KD) with variational student approach for 2D model compression on CIFAR-10 dataset

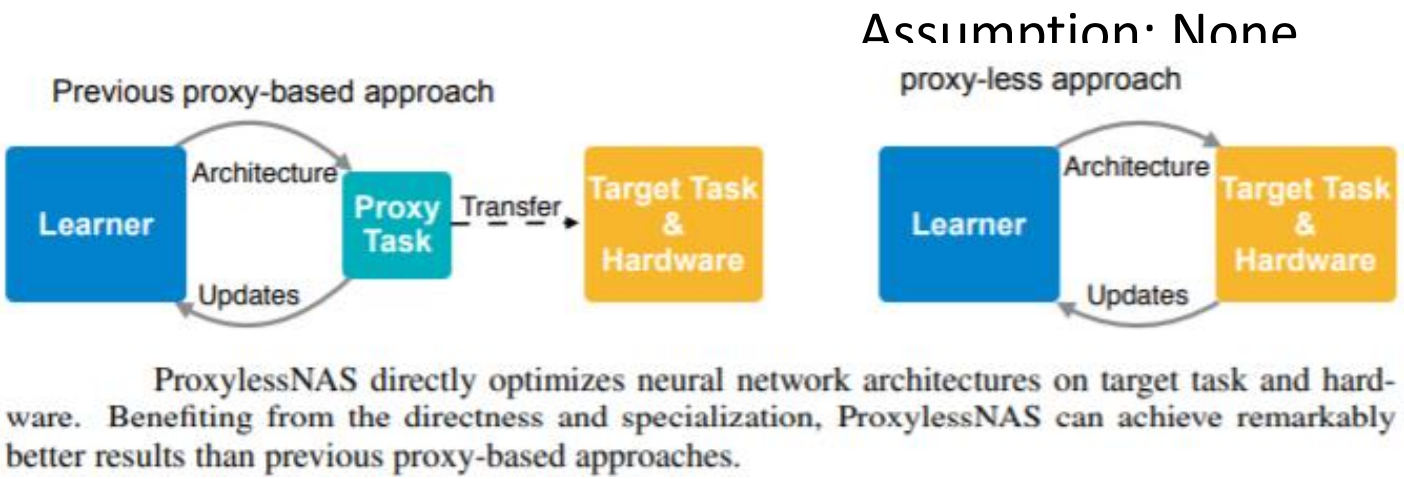
- Memory footprint reduction- 9.3MB to 147KB (LeNet-5) and 532MB to 2.5MB (VGGNet-19)
- Accuracy drop 0.08% (LeNet-5), 8.7% (VGGNet-19)
- Inference time for student model varies 0.257 – 0.470 ms.

S. Hegde, R. Prasad, R. Hebbalaguppe, V. Kumar, "Variational Student: Learning Compact and Sparser Networks in Knowledge Distillation Framework," ICASSP 2020.

Different approaches

Network architecture search (NAS):

NAS in the most general sense is a search over a set of decisions that define the different components of a neural network—it is a systematic, automized way of learning optimal model architecture. The idea is to remove human bias from the process to arrive at novel architectures that perform better than human-designed ones.

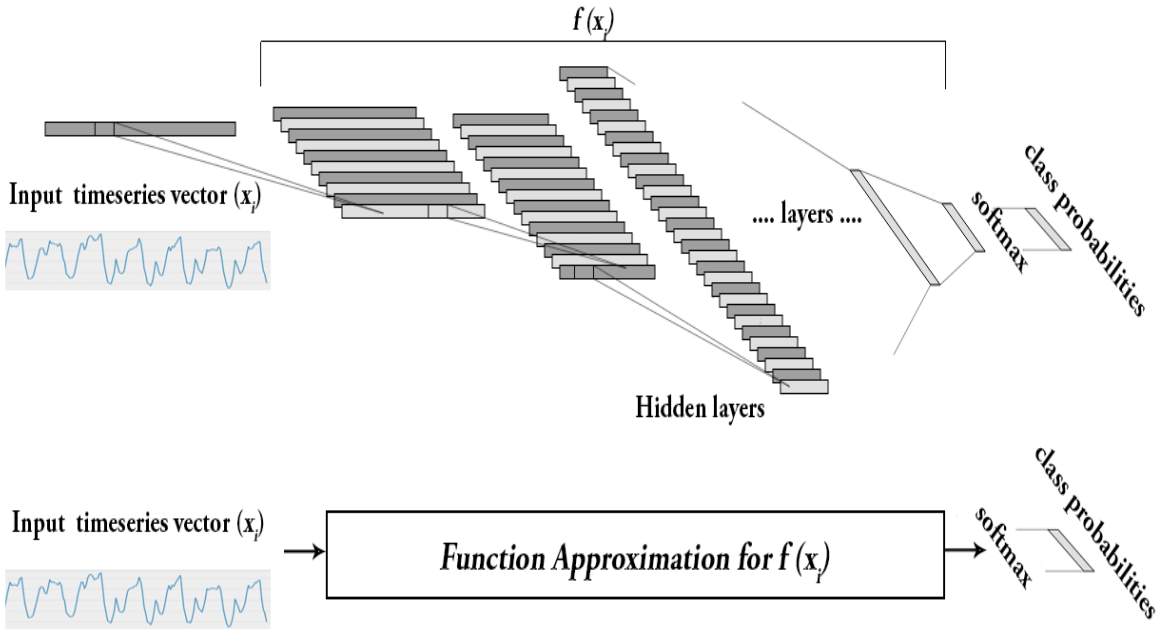


ProxylessNAS directly optimizes neural network architectures on target task and hardware. Benefiting from the directness and specialization, ProxylessNAS can achieve remarkably better results than previous proxy-based approaches.

Han Cai, Ligeng Zhu, Song Han, "Proxylessnas: Direct Neural Architecture Search On Target Task And Hardware," ICLR, 2019.

Our Method

- A DNN model can be considered as a complete black box. Given an input, we get an output
- End-to-end DNN takes in a real vector as an input and outputs a binary class label as output
- We aim to approximate the mapping from the space of input real vectors to the real valued output of the pre-final layer by system identification method (as a linear regression solution)



"ECG TinyML" - a Piecewise Linear Approximation (PLA) approach

We propose a piecewise linear approximation (PLA) of a ResNet based ECG diagnostic inferencing model

Main idea : The relationship between input data and output of the penultimate layer of DNN model is approximated using a piecewise approach.

Knowledge-Distillation with heterogeneity in KD process

Divided the input space into smaller pieces and then approximate the input – output relationships for each of these small spaces using separate linear models

Functional approximation of DNN representation before final activation (e.g., softmax)

- DNN: $f(x_i) \leftarrow$ output for input x_i
- Parameterized function: $g(\theta, x_i)$
- Least square estimate: $\underset{\theta}{\operatorname{argmin}} \sum_i \|f(x_i) - g(\theta, x_i)\|_2^2$
- Use $g(\theta, x)$ instead the DNN

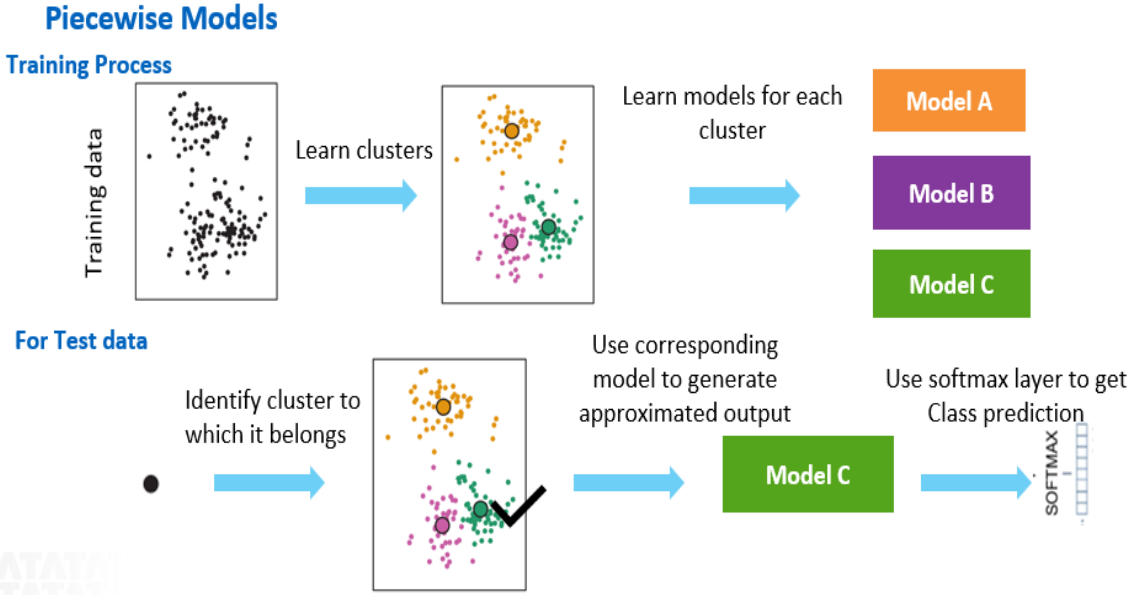
"ECG TinyML" - a Piecewise Linear Approximation (PLA) approach

Inferencing using the compressed model

ECG signals cluster membership is identified by locating the nearest cluster centroid. Then, intermediate output is computed using the cluster's associated linear regression model. This intermediate output is multiplied by last layer weights to generate the class predictions

ECG signal classification using $\Pi^{ECG\ TinyML}$:

Input: Test ECG signal: $\mathcal{S}_{test}$
Output: Label denoting health condition
Procedure:
Step 1: First the cluster membership of $\mathcal{S}_{test}$ is ascertained by determining the nearest cluster centroid present in $\Pi^{ECG\ TinyML}$.
Step 2: Using the linear regression model associated with the identified cluster, ρ'_{out} is computed.
Step 3: Finally, after multiplication of ρ'_{out} with the last layer weights, we get the predicted health condition.



Ishan Sahu, Arpan Pal, Arijit Ukil, and Angshul Majumdar "Compressing Deep Neural Network: A Black-Box System Identification Approach" International Joint Conference on Neural Networks (IJCNN), 2021.

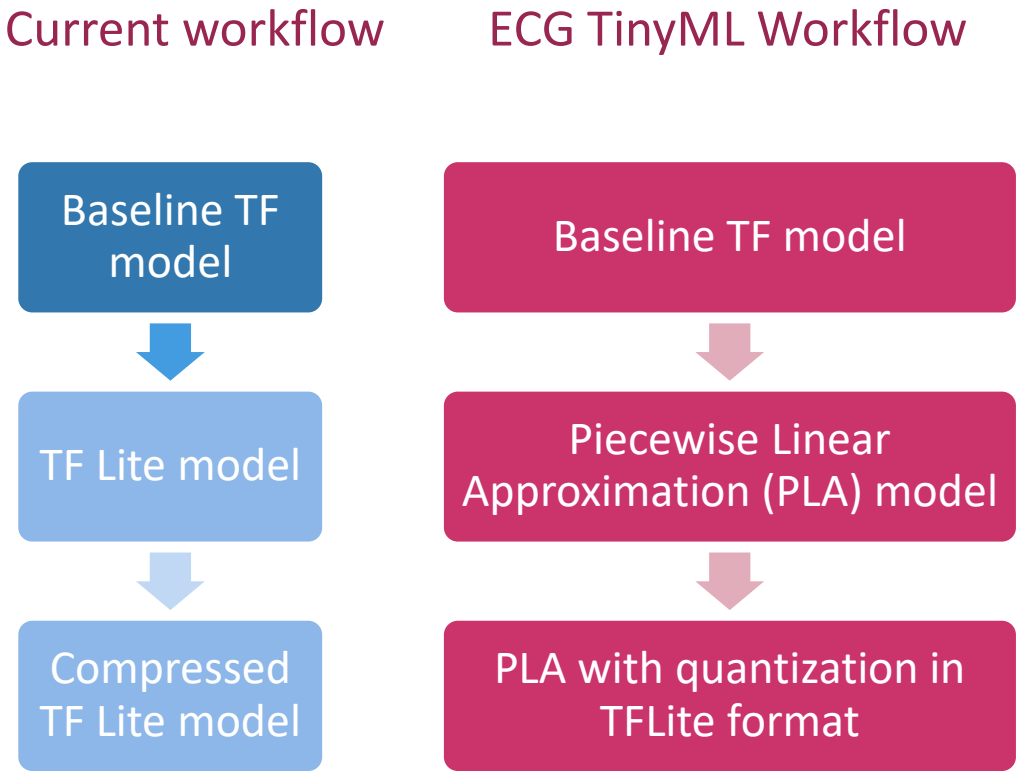
Dataset and Deployment setup

ECG dataset - BIDMC Congestive Heart Failure Database *

The original data is sampled at 250 Hz -
Pre-processed in two steps:
(1) extract each heartbeat
(2) make each heartbeat equal length using interpolation with time series length equals to 140-time steps

Classes in dataset -

1. Normal
2. R-on-T Premature
3. Ventricular Contraction
4. Supraventricular Premature
5. Premature Ventricular Contraction
6. Unclassifiable Beat



*Physiobank, Physiotoolkit, and Physionet: Components of A New Research Resource for Complex Physiologic Signals by Goldberger et. al

Experimental process and results

The baseline model size : 9.71 MB
Target MCU memory : < 100 KB
Target Model compression gain : 97x
Allowed performance penalty δ : 1%
Latency improvement: 10x

Two control parameters -
k, the number of pieces or clusters
λ, regularization parameter

We have used 5-fold cross-validation on training data to tune the parameters -

- k varied from 1 to 5 (in steps of 1)
- λ, search was done over the values ranging from 0.01 to 1000 (in multiples of 10)

Our publication on these results -

Ishan Sahu, Arpan Pal, Arijit Ukil, and Angshul Majumdar “Compressing Deep Neural Network: A Black-Box System Identification Approach” International Joint Conference on Neural Networks (IJCNN), 2021.

Arijit Ukil, Ishan Sahu, Angshul Majumdar, Sai Chander Racha, Gitesh Kulkarni, Anirban Dutta Choudhury, Sundeep Khandelwal, Avik Ghose, Arpan Pal, "Resource Constrained CVD Classification Using Single Lead ECG On Wearable and Implantable Devices," IEEE EMBC, 2021.

Method	Method Performance				
	Test accuracy	Test sensitivity	Test precision	Test F1-score	Model size
Baseline DNN (ResNet) [28]	0.931	0.931	0.924	0.926	9.71 MB
Baseline model TF Lite format	0.931	0.931	0.924	0.926	3.13 MB
Baseline model pruned and quantized TF Lite	0.929	0.931	0.927	0.928	0.801 MB

Method	Method Performance				
	Test accuracy	Test sensitivity	Test precision	Test F1-score	Model size
Baseline DNN (ResNet) [28]	0.931	0.931	0.924	0.926	9.71 MB
Piecewise Linear Approximation (PLA)	0.938	0.938	0.926	0.929	0.443 MB
PLA - quantized TF Lite (ECG TinyML)	0.937	0.937	0.925	0.928	0.062 MB

Method	FLOPs
Baseline DNN (ResNet) [28]	222908190
Baseline model pruned and quantized TF Lite	222908190
PLA - quantized TF Lite (ECG TinyML)	38554

Result Summary

The proposed "**ECG TinyML"** approach demonstrates the feasibility of transforms from larger models to Tiny models -

- High model compression gain ($\alpha = 156$)
- Insignificant performance penalty ($\delta \cong 0$)
- Substantially less computational load for effective run time execution ($\theta = 1.72 \times 10 - 4$)
- Tiny model foot-print of 62.3 KB obtained
- Low FLOPs of 38554 (5782x less computationally intensive)

In conclusion, with less than 1% accuracy loss, model size reduced by 97X and latency is decreased by 10x

Number of parameters -

- Baseline ResNet model = 804169
- Compressed PLA model = 55710

This type of ML model can run on Wearables and Implantable Loop Recorder

Would help in building early warning systems for CVD and reduce the burden of CVD on healthcare system

Future work

- Model size may be much higher for other types of disease detection tasks like Atrial Fibrillation detection from single-lead ECGs: > 100 MB model size*
- More sophistication in the model size reduction is required given that hardware or MCU remains the same
  - LTH, NAS are the near-future research approaches

*P. Rajpurkar, A. Hannun, M. Haghpanahi, C. Bourn, and A. Ng, “Cardiologist-Level Arrhythmia Detection With Convolutional Neural Networks,” Nature 2019.

tcs Research

Inventing for impact

Thank you

Copyright Notice

www.tinyml.org

Copyright Notice

www.tinyML.org