tinyML® Summit

Miniature dreams can come true…

March 28-30, 2022 | San Francisco Bay Area

TINY ML

www.tinyML.org

TinyML for All: Full-stack Optimization for Diverse Edge AI Platforms

Di Wu

Co-founder and CEO, OmniML

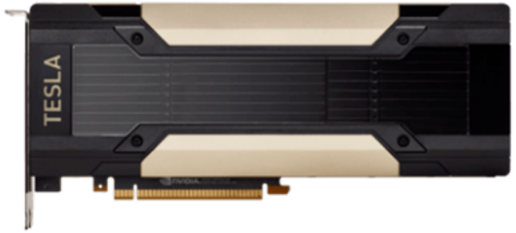
Song Han

Assistant Professor, MIT EECS

Co-founder and Chief Scientist, OmniML

TinyML is about Constraints

Mismatch: AI has been evolving unconstrained for many years



	Cloud AI	**Mobile AI**	**Tiny AI**
Computation	10 TFLOPS	GFLOPS	MFLOPS
Memory	32GB	4GB	256KB

100,000x smaller

Everything Together: Real-world AI on Tiny MCUs

Two Generations of Innovations: MCUNet-v1 (2020), MCUNet-v2 (2021)

[MCUNet, NeurIPS'20]
[MCUNet-v2, NeurIPS'21]

Facemask Detection

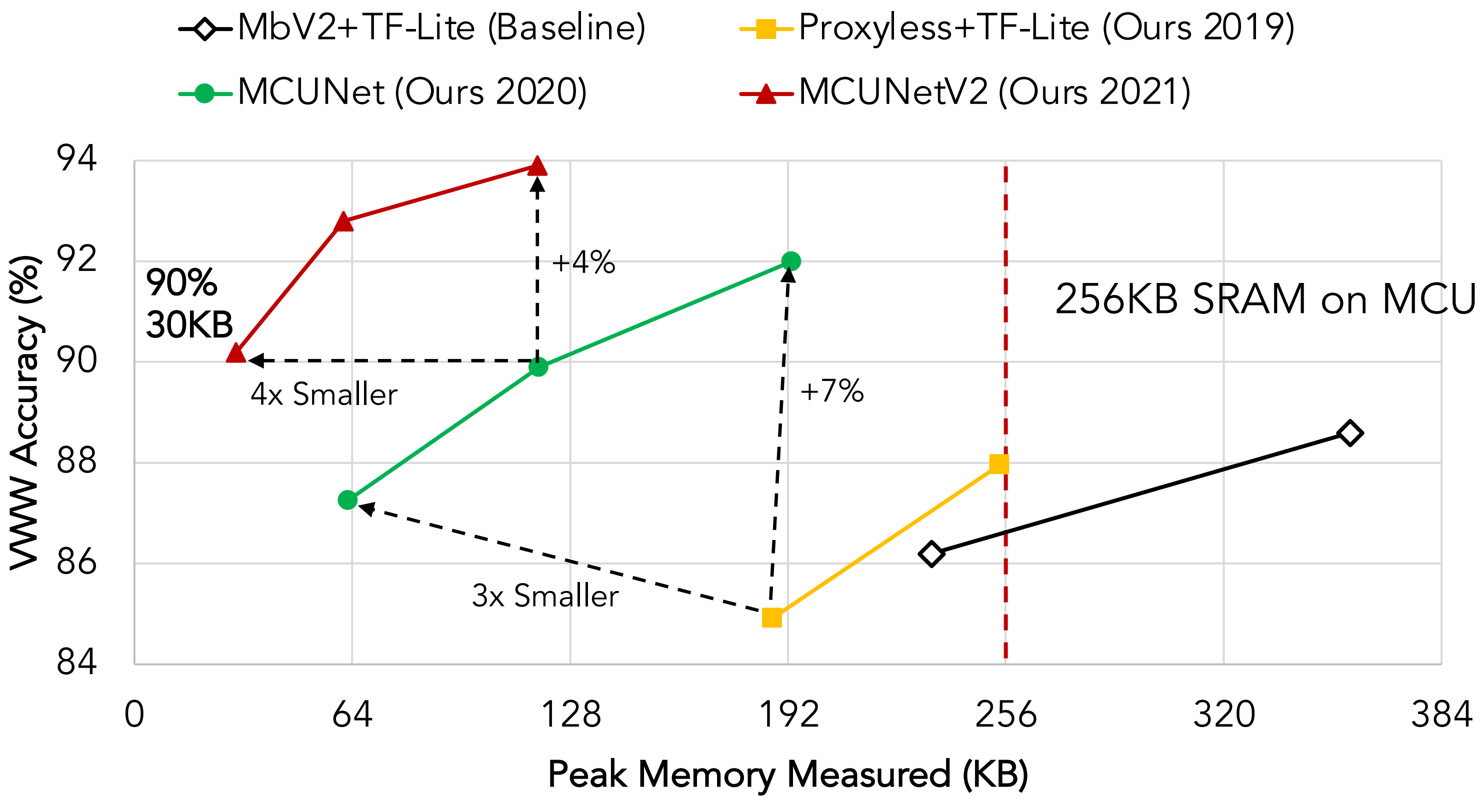
Person Detection

Works on Cortex M7 MCU

Brief History of MCUNets

Reducing the model sizes with increasing accuracy

[MCUNet, NeurIPS'20]
[MCUNet-v2, NeurIPS'21]



Opportunity in Fundamental ML Algorithms

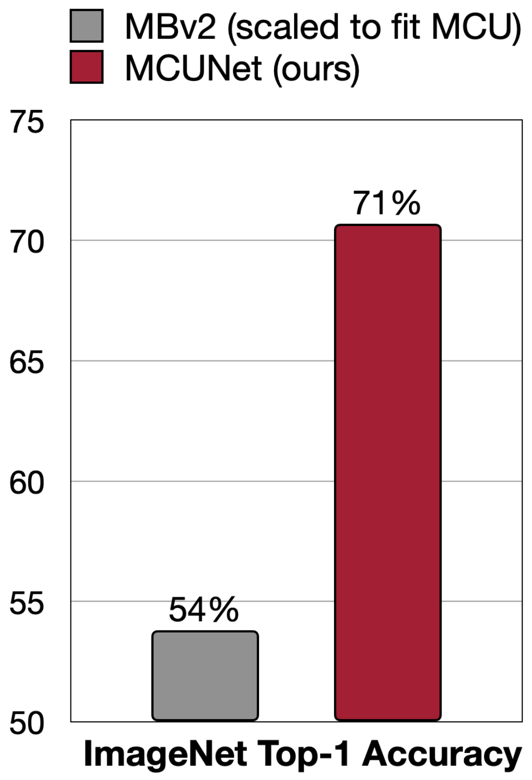
[MCUNet, NeurIPS'20]
[MCUNet-v2, NeurIPS'21]

Making algorithm more efficient under existing constraints

Faster than Moore's Law:
3.5x model size reduction every 12 months

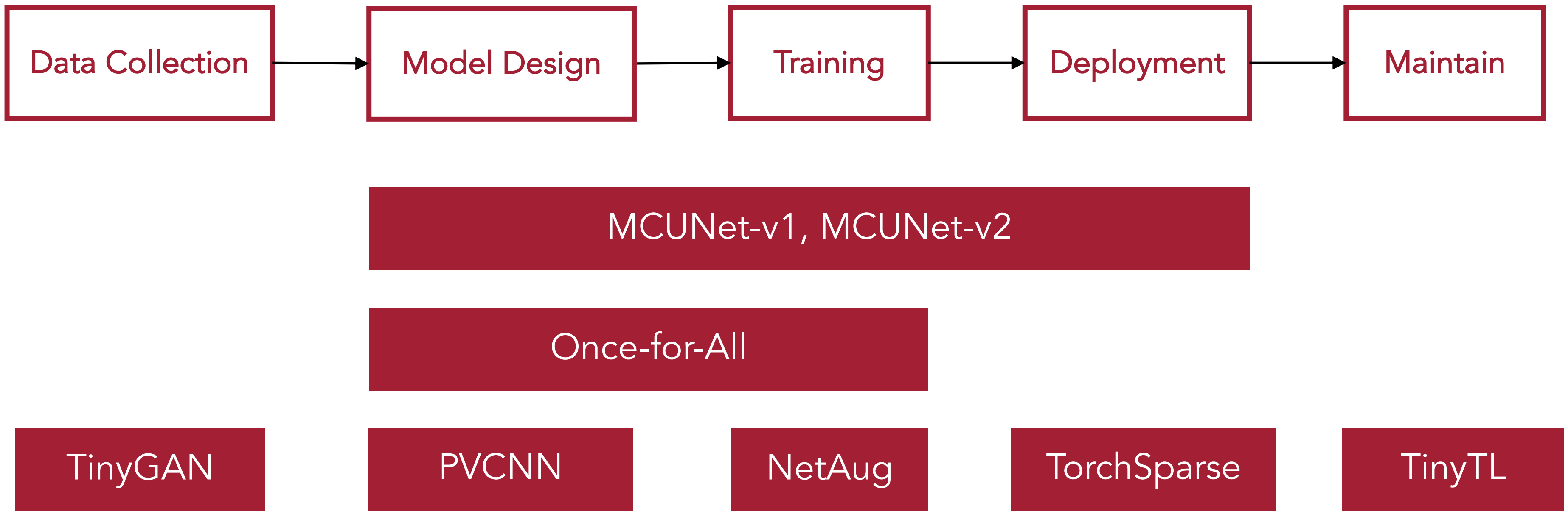
Improving efficiency means *more accurate* models, too

TinyML is about improving the entire stack: from design to deployment, from computation to data



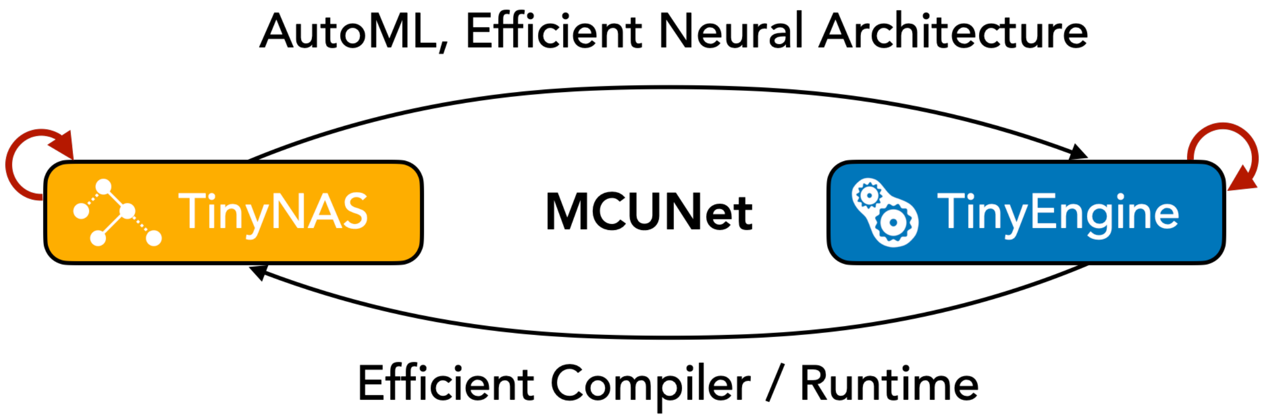
Agenda

Focus on Constraints on the Entire Stack



MCUNet-v1: TinyNAS+TinyEngine Co-design

[MCUNet, NeurIPS'20]



TinyNAS:

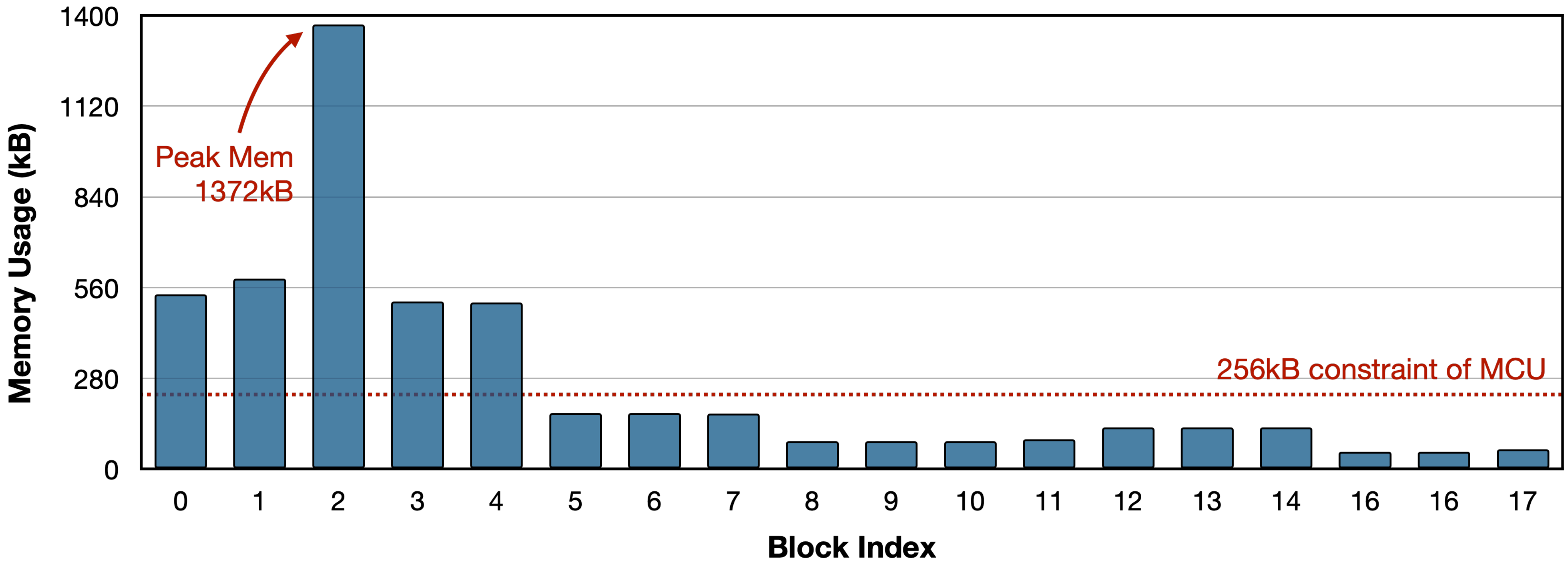
- Re-design the design space
- Latency-aware
- Energy-aware
- Once-for-all Network

TinyEngine:

- Co-design, specialization
- Offload run-time to compile-time
- Graph optimizations
- Memory-aware scheduling
- Low-precision
- Assembly-level optimizations

New Problem: Imbalanced Memory Distribution of CNNs [MCUNet-v2, NeurIPS'21]

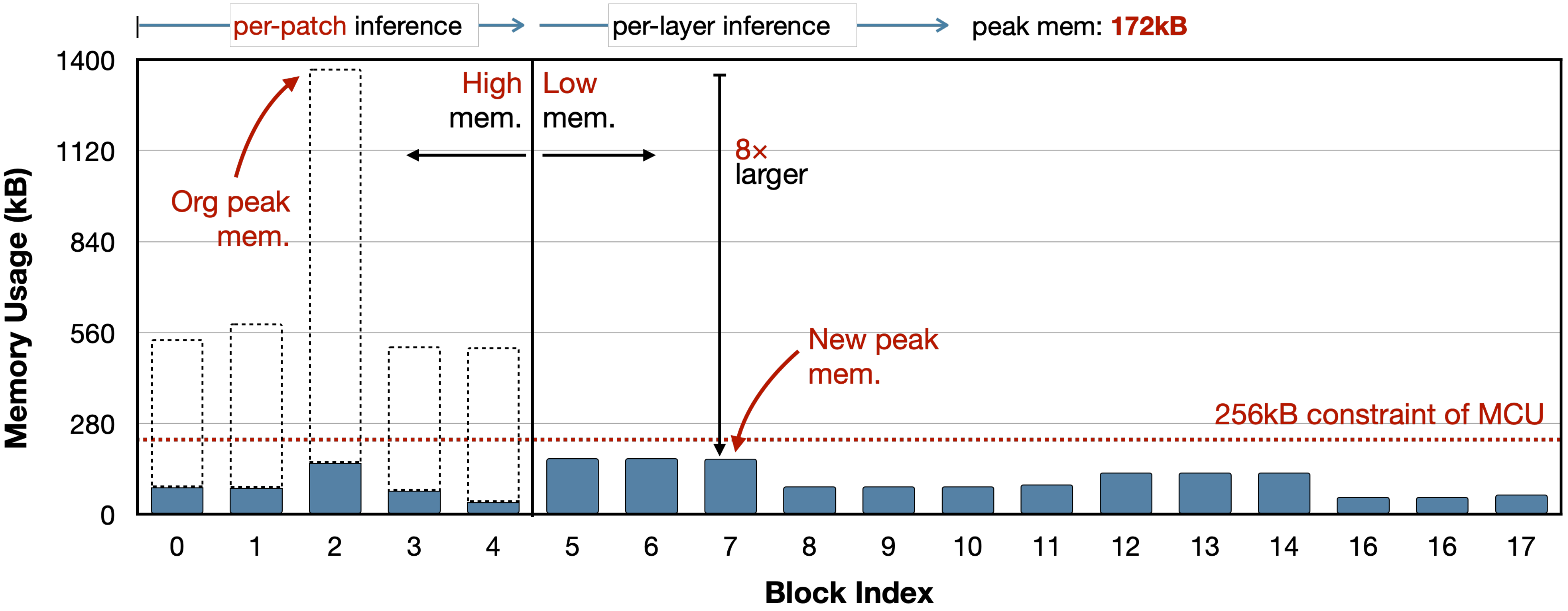
Per-block memory usage of MobileNetV2



Solving the Imbalance with Patch-based Inference

[MCUNet-v2, NeurIPS'21]

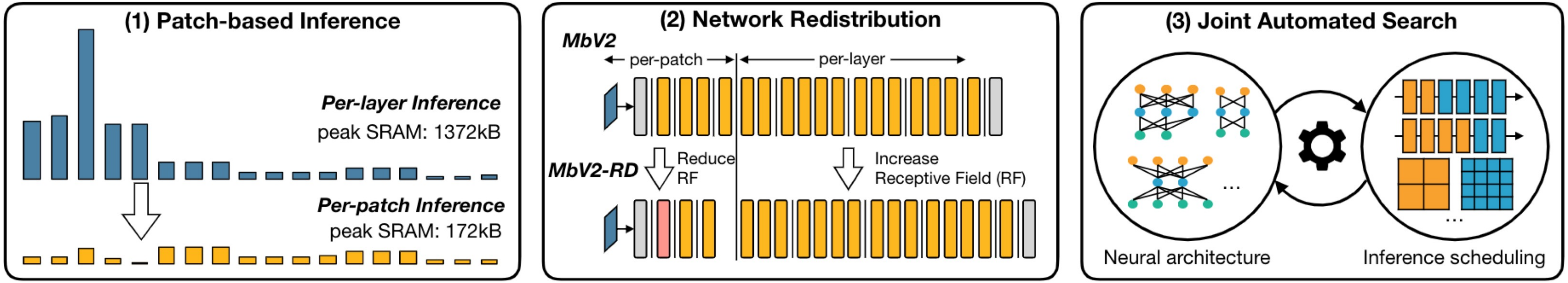
After applying Patch-based Inference



MCUNet-v2 Takeaways

[MCUNet-v2, NeurIPS'21]

Solving inference bottleneck (peak memory) results in smaller and better models



Agenda

Focus on Constraints on the Entire Stack

Data Collection → Model Design → Training → Deployment → Maintain

MCUNet-v1, MCUNet-v2

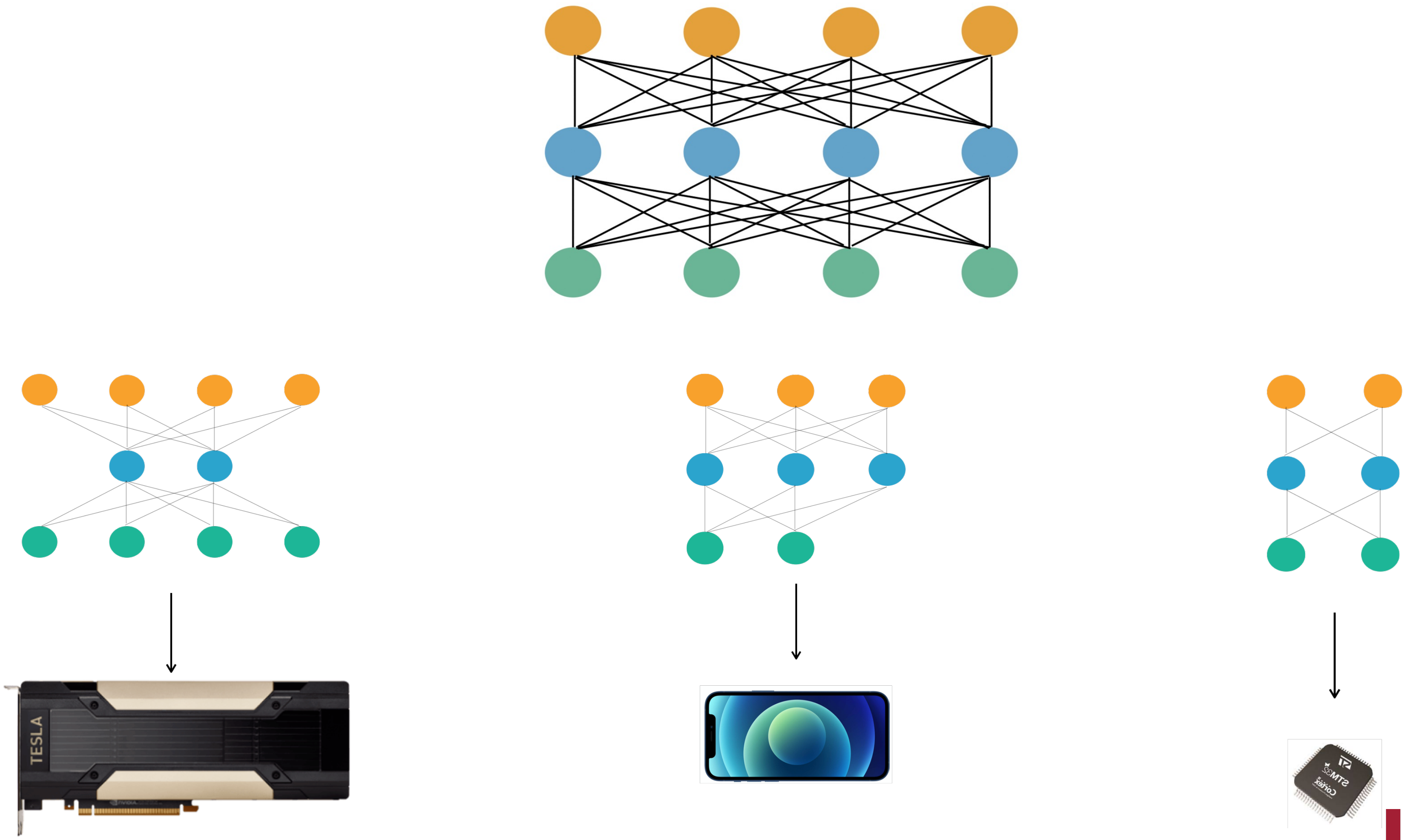
Once-for-All

TinyGAN | PVCNN | NetAug | TorchSparse | TinyTL

Once-for-All Network

[OFA, ICLR'20]

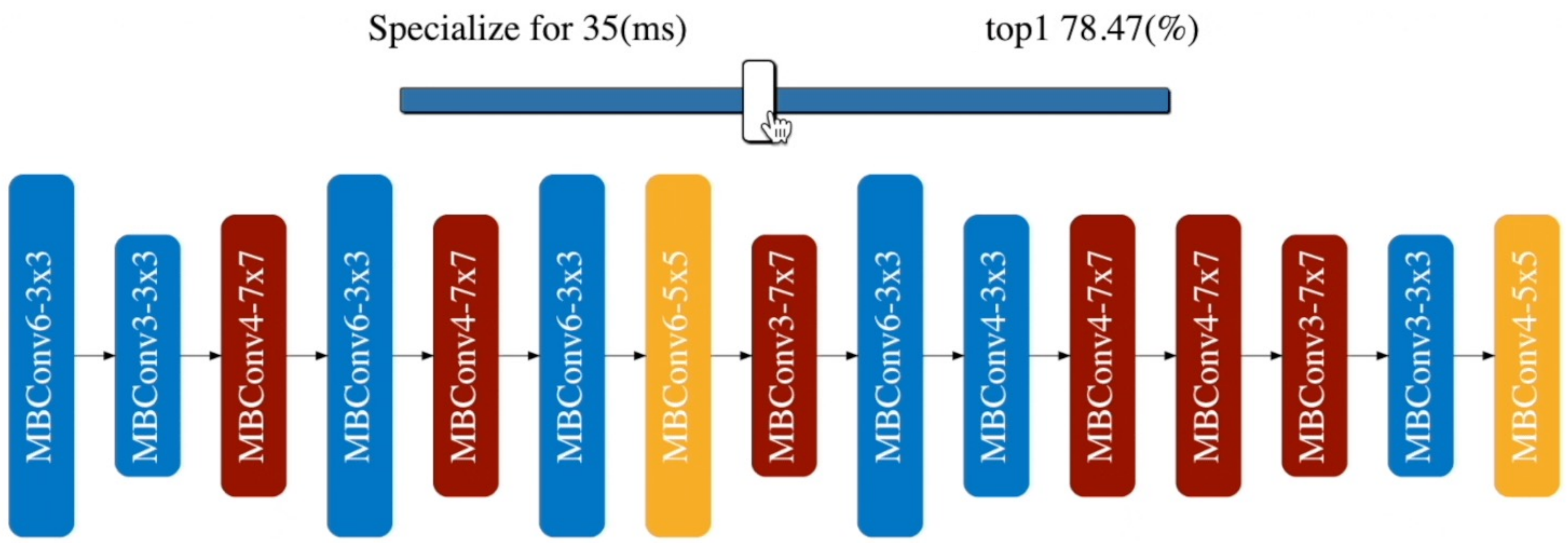
Train once, get many; Fit diverse hardware constraints



Better Results with Much Smaller Training Cost

[OFA, ICLR'20]

Reduce the search cost from 42,000 GPU hours (Google) to 200 GPU hours



Existing: Lots of hand tuning for different devices and latency.

OFA: Auto design the NN architecture at low cost

Agenda

Focus on Constraints on the Entire Stack

Data Collection → Model Design → Training → Deployment → Maintain

MCUNet-v1, MCUNet-v2

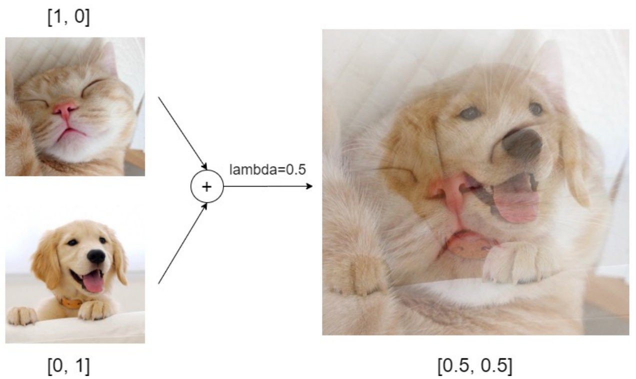
Once-for-All

DiffAugment | PVCNN | **NetAug** | TorchSparse | TinyTL

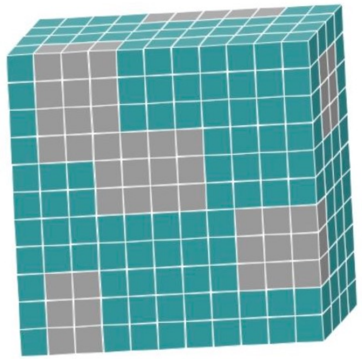
Problem in Training for Tiny Models

[NetAug, ICLR'22]

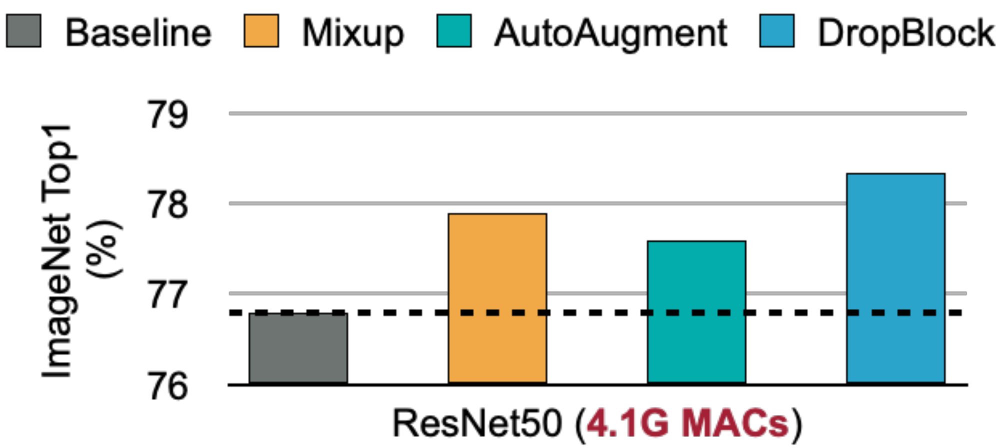
Existing Training Techniques don't Apply to TinyML



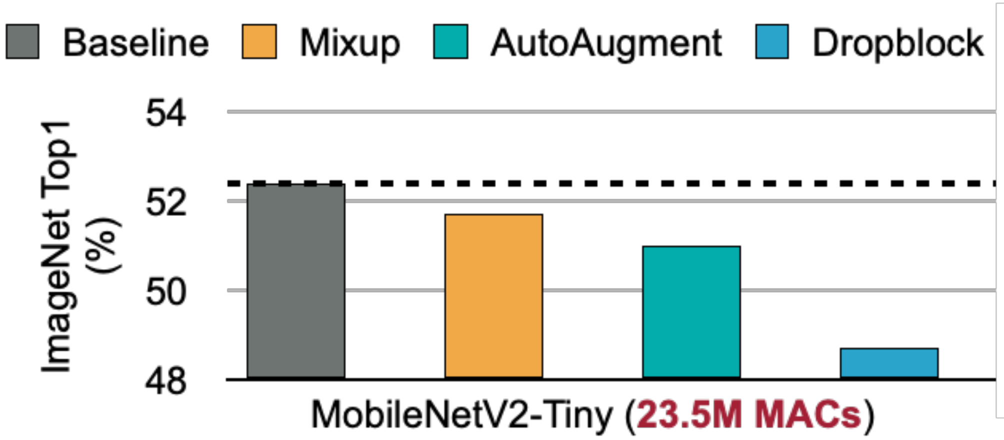
Mixup



DropBlock



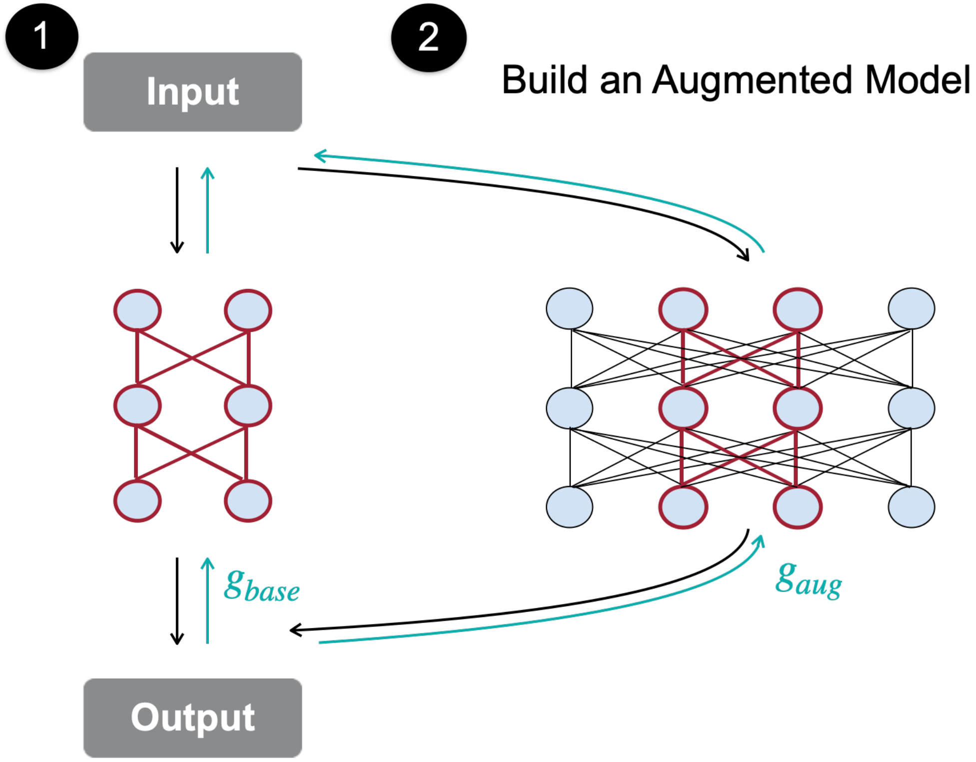
AutoAugment



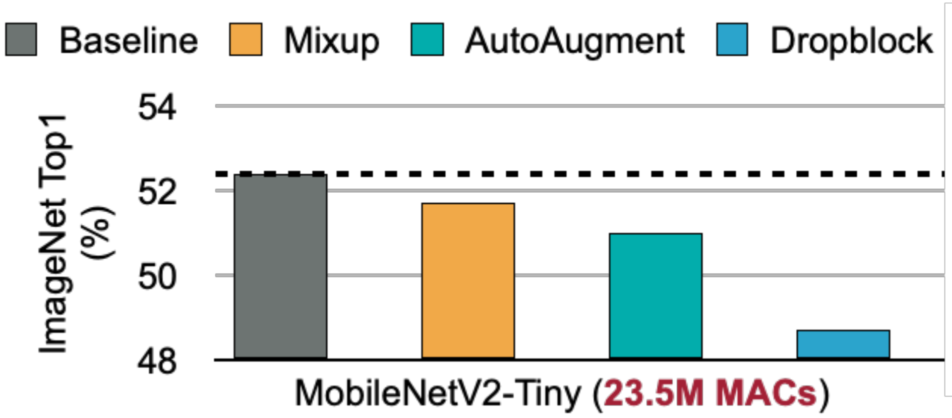
NetAug for TinyML

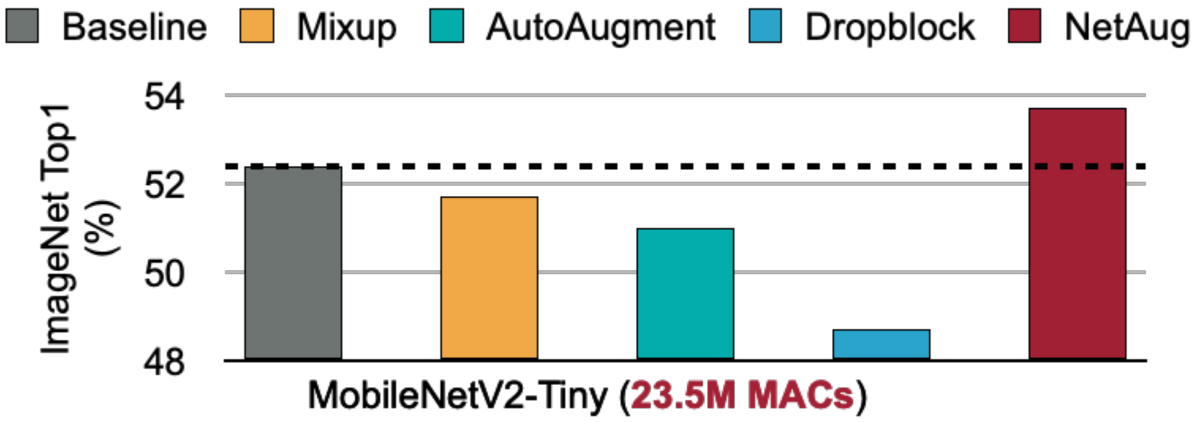
[NetAug, ICLR'22]

Augment Model Rather than Data



Zero Inference Overhead





Agenda

[TinyTL, NeurIPS'20]

Focus on Constraints on the Entire Stack

Data Collection → Model Design → Training → Deployment → Maintain

MCUNet-v1, MCUNet-v2

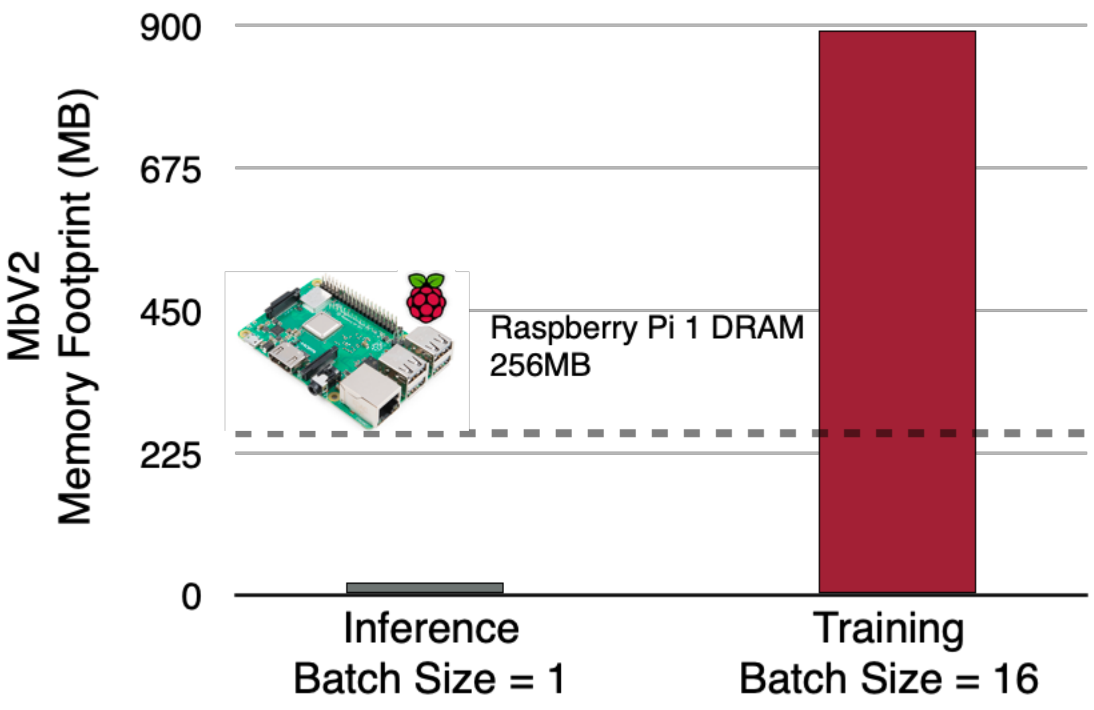
Once-for-All

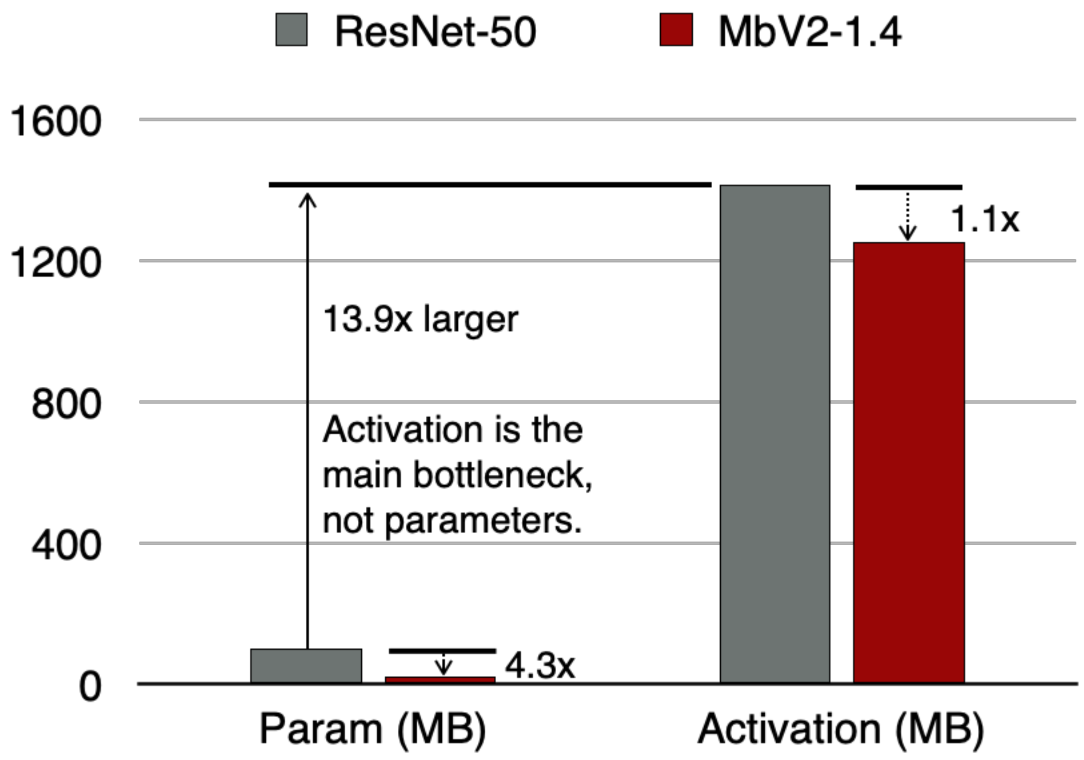
TinyGan | PVCNN | NetAug | TorchSparse | **TinyTL**

Problem: Training Memory is much Larger

[TinyTL, NeurIPS'20]

Bottleneck is Activation rather than Parameters

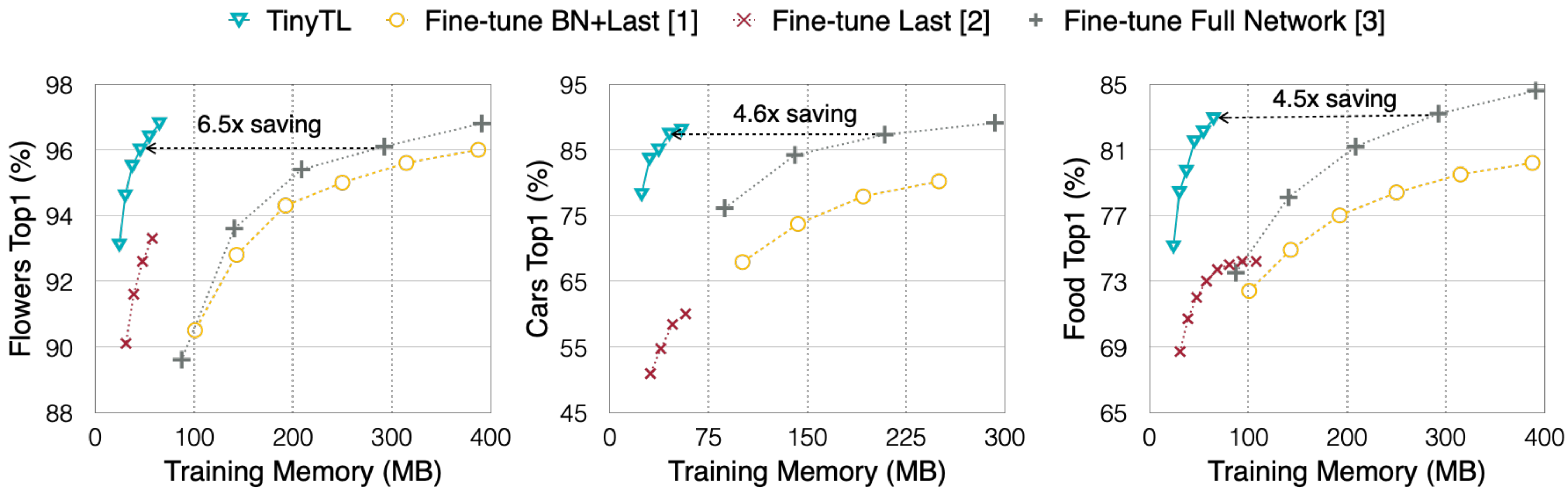




TinyTL: Up to 6.5x Memory Saving without Accuracy Loss

Use Fine-Tune Bias Only and Lite Residual Learning

[TinyTL, NeurIPS'20]



Agenda

[DiffAugment, NeurIPS'20]

Focus on Constraints on the Entire Stack

Data Collection → Model Design → Training → Deployment → Maintain

MCUNet-v1, MCUNet-v2

Once-for-All

TinyGAN

PVCNN

NetAug

TorchSparse

TinyTL

Data is Also Constrained

[DiffAugment, NeurIPS'20]

Many TinyML Applications Have Limited Access to Data

Rare Defects

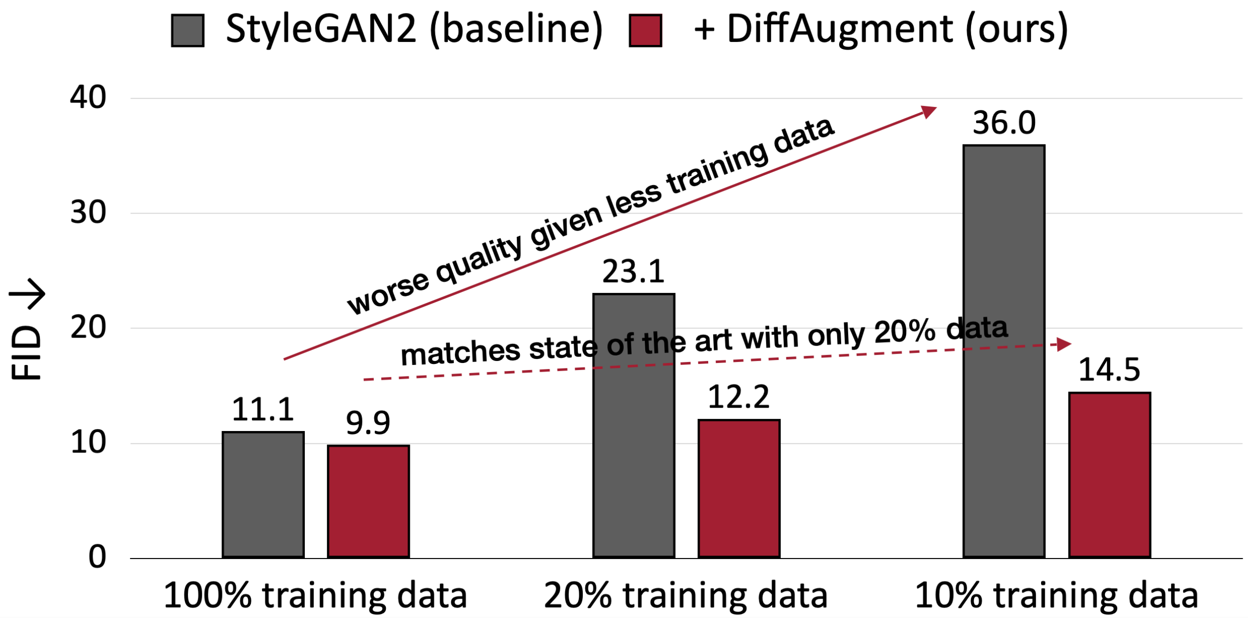
Specific Tasks

Privacy Concerns

Differentiable Augmentation

[DiffAugment, NeurIPS'20]

Photo-realistic and Smooth Generation with 100 Training Images



Agenda

Focus on Constraints on the Entire Stack

Data Collection → Model Design → Training → Deployment → Maintain

MCUNet-v1, MCUNet-v2

Once-for-All

DiffAugment

PVCNN

NetAug

TorchSparse

TinyTL

TinyML for LIDAR & Point Cloud

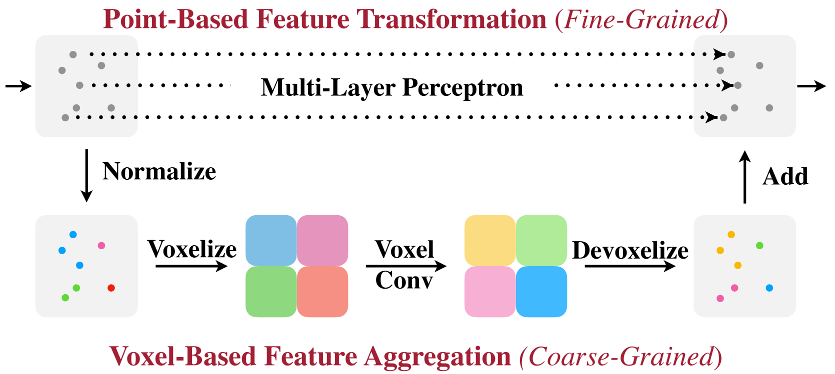
Challenge: High Algorithm Complexity vs. Limited Computational Resource

[PCVNN, NeurIPS'19]
[SPVNAS, ECCV'20]
[PointAcc, Micro'21]
[TorchSparse, MLSys'22]

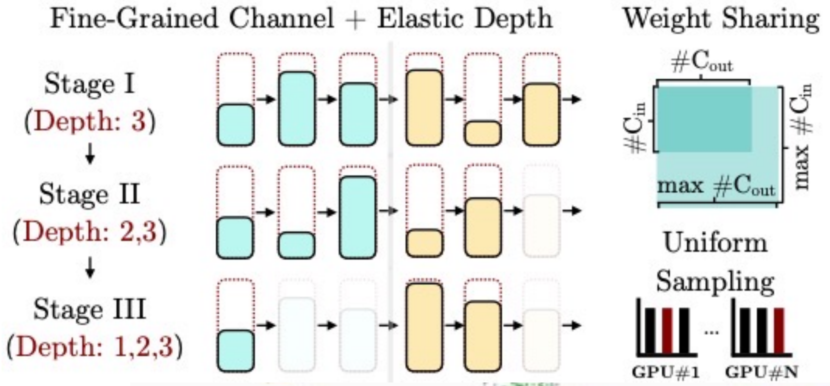
Full Stack LIDAR & Point Cloud Processing

New Design Space + NAS + Inference Library

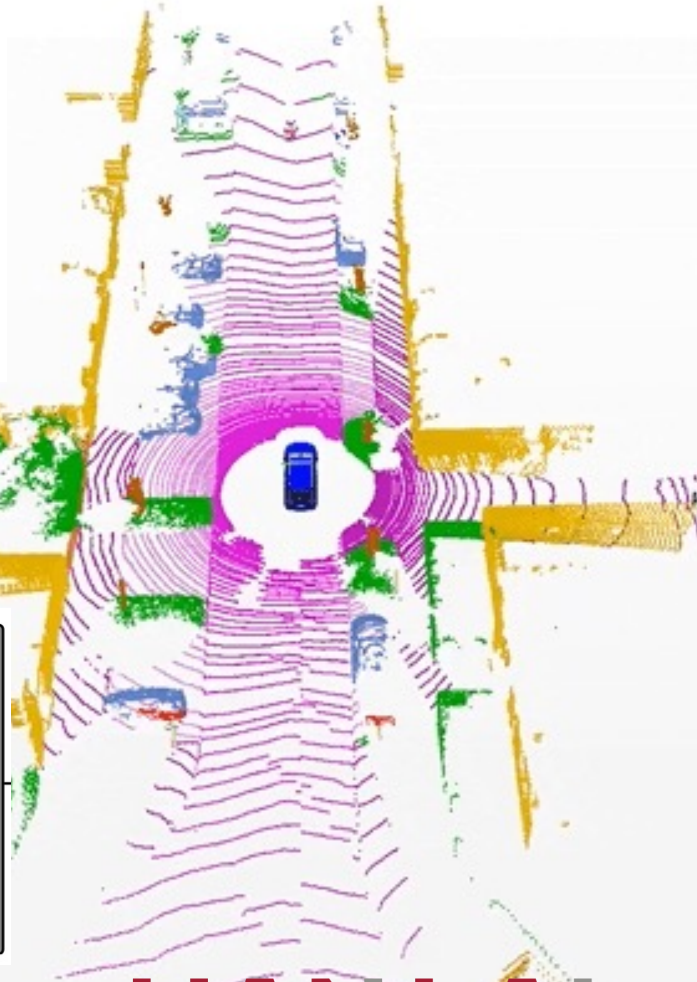
[PCVNN, NeurIPS'19]
[SPVNAS, ECCV'20]
[PointAcc, Micro'21]
[TorchSparse, MLSys'22]

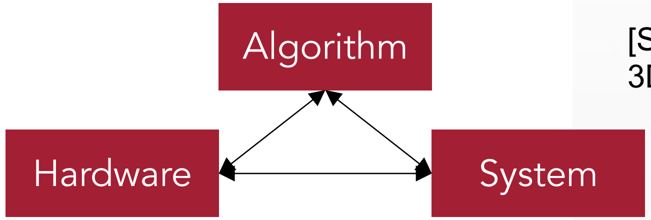


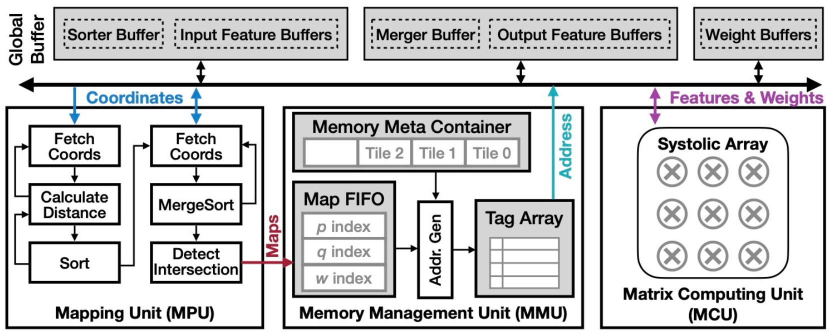
[Point-Voxel CNN, NeurIPS'19]
New design space for Point Cloud

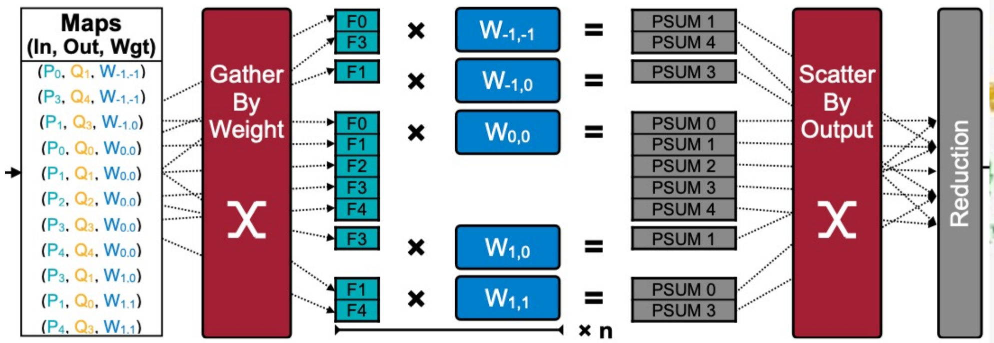


[SPVNAS, ECCV'20]
3D neural architecture search





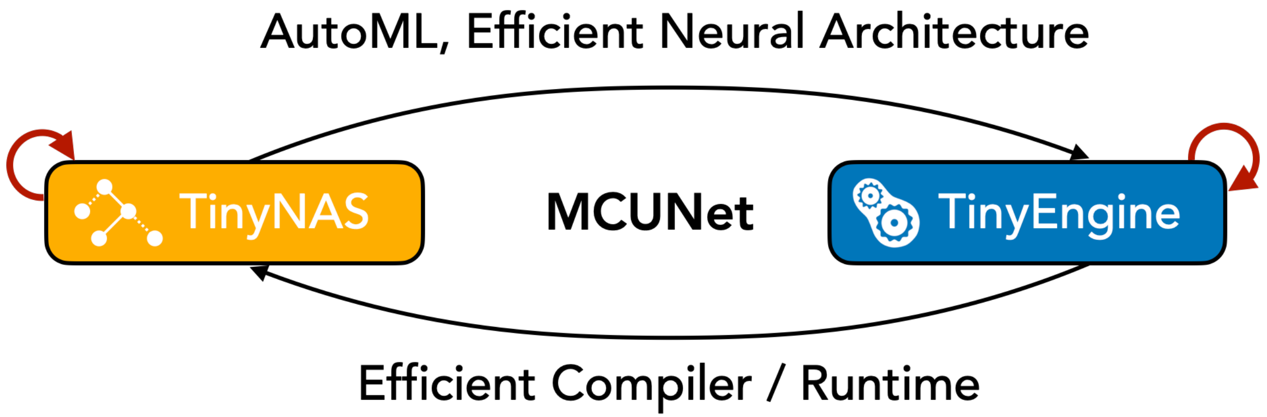




[PointAcc, MICRO'21]
Hardware accelerator for point cloud

[TorchSparse, MLSys'22]
GPU library for 3D sparse convolution

Takeaways: Coming Back to MCUNets



Co-optimization on the entire stack is the key to unlock the most potential for TinyML

OMNI ML

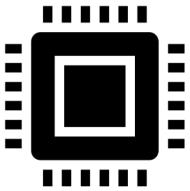
Effortlessly Empower Edge AI Everywhere

Fundamental Problems in TinyML

ML under new HW constraints is very hard

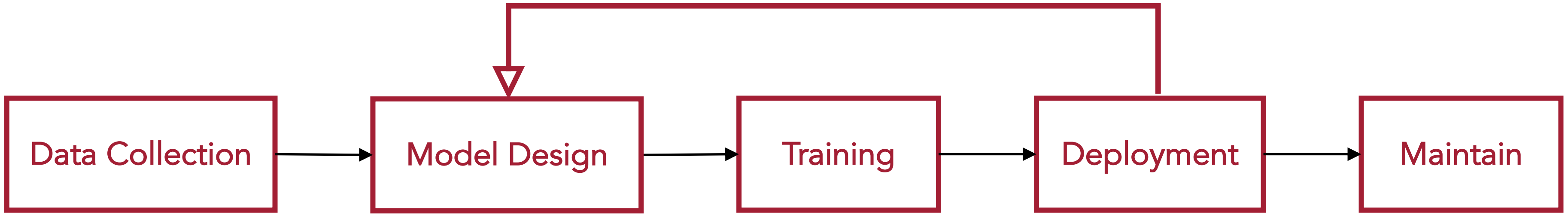
Mismatch

X



Slow Adoption ➔ **Less Revenue/Volume**

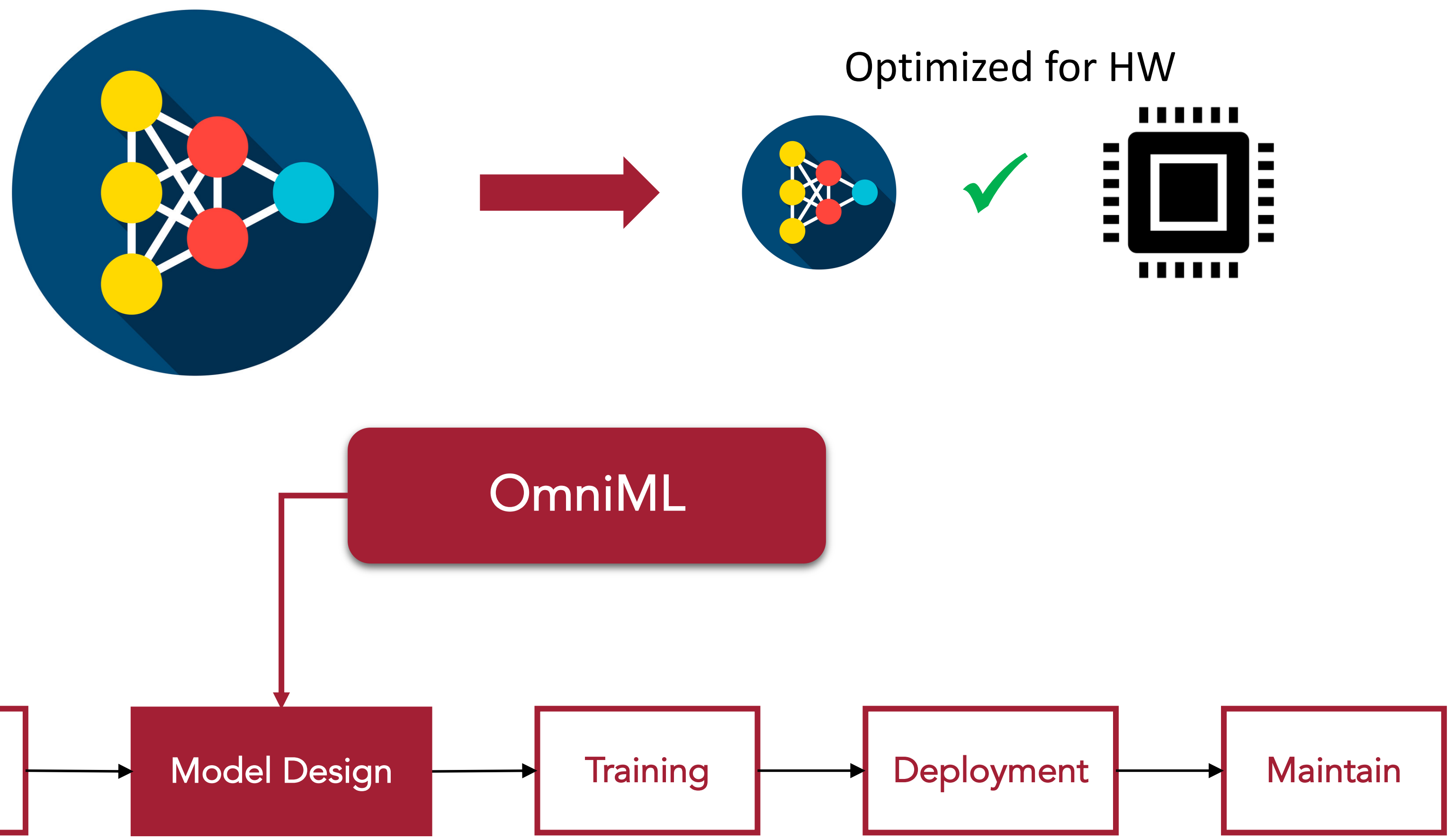
Typical AI/ML Development



Designing new models that works on different HW is still a manual and iterative approach

OmniML "Compress" the Model Before Training

Bring HW deployment constraints into model design and training



OmniML: Enable TinyML for All Vision Tasks

Create the Best Models on Different Platforms Effortlessly

CV on Mobile Devices

- Pose estimation
- Scene Segmentation
- Image denoise, super resolution
- AR/VR

Sensor Fusion 3D Detection

Multi-sensor 3D object detection for automotive applications.

Smarter Cameras

Turn “dumb” cameras into AI-powered cameras with advanced CV features on low-power, low-cost chip.

Computer Vision on MCUs

Not only classification but also object detection on microcontrollers with only 256~512KB of memory.

40+ Customers Conversations **10+** POCs **100K** Installed devices

Founding Team

Leading Experts in Efficient Deep Learning

Song Han

- Assistant professor at MIT, PhD from Stanford
- Co-founder of DeePhi Tech (acquired by Xilinx)
- "35 Innovators Under 35" by MIT Technology Review
- NSF CAREER Award, IEEE "AIs 10 to Watch"
- Inventor of "Deep Compression"
- 29K Google Scholar citations

Di Wu

- Previous tech lead at Facebook AI, PyTorch accelerator enablement
- Product and engineering leader at Falcon Computing Solutions (acquired by Xilinx)
- PhD from UCLA, years of experience in customized hardware systems at Intel Lab, MSRA.

Huizi Mao

- PhD from Stanford. Co-Inventor of "Deep Compression"
- Early member of DeePhi and Megvii.
- Worked at Google Research, Facebook AML and NVIDIA.
- NVIDIA Fellowship Recipient.

Thank you

Come talk to us to learn more

Follow us:

https://www.linkedin.com/company/omniml
https://twitter.com/OmniML_AI

We are hiring:

https://omniml.ai/career/
contact@omniml.ai

TINY ML SUMMIT

tinyML Summit 2022 Sponsors

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™

AONdevices

arm

ASPINITY

brainchip
The Neuromorphic Computing Company

CEVA®

Deeplite

EDGE IMPULSE

emza
visual sense

FotaHub

GREENWAVES
TECHNOLOGIES

Grovety Inc.

Himax

HOTG

imagimob

infineon

itemis

KLIKA TECH
GLOBAL IOT SOLUTIONS

LatentAI

LATTICE
SEMICONDUCTOR

Micro.ai

OmniML

NXP

PX!

Plumerai

PROPHESEE

Qeexo

Qualcomm

Rackner

RealityAI®
Engineering Solutions for the Edge

REEXEN
technology

RENESAS

SAP

seeed studio
The IoT Hardware Enabler

SensiML™

Sony
Semiconductor
Solutions
Corporation

ST
life.augmented

SA STREAM ANALYZE

synaptics®

SynSense

SYNTIANT

Tensil.ai

TensorFlow

XMOS

Copyright Notice

www.tinyml.org