TinyML and Efficient Deep Learning

make AI efficient for edge devices

Song Han
Assistant Professor, MIT

tinyml.mit.edu
AI is Quickly Coming to the Edge
Privacy, Latency, Cost

% of Enterprise Data from Edge

Number of Edge Devices 2021
- 15 Billion Mobile phones
- 1.4 Billion Cars
- 770 Million Security cameras
- 15 Million Robots

Source: The Silent Intelligence: The Internet of Things
Today’s AI is too Big
We need new algorithms and hardware for TinyML and Green AI
Low Energy, Low Latency, Low Cost, Better Privacy

GPT-3: 175 billion parameters, 355 GPU years to train, cost $4.6M
AlphaGo: 1920 CPUs and 280 GPUs, $3000 per game for electric bill
Edge AI Demands New Algorithms for Efficient Computing

Extreme memory constraint: 4-5 orders of magnitude smaller memory than a GPU
Low power budget.

<table>
<thead>
<tr>
<th></th>
<th>Cloud AI</th>
<th>Mobile AI</th>
<th>Tiny AI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>32GB</td>
<td>4GB</td>
<td>256kB</td>
</tr>
</tbody>
</table>

16,000x smaller
100,000x smaller

tinyml.mit.edu
We Need Algorithm and Hardware Co-Design

Algorithm

Benchmark

Hardware

CPU/GPU...

Efficient Algorithm

co-design

Domain-Specific Hardware

?PU

design across the full stack

"There is plenty of room at the top by optimizing the algorithm. We found that DNN models can be significantly compressed and simplified."

TinyML and Efficient Deep Learning

Make AI smaller, faster, more accessible and more sustainable

Model Compression / Neural Architecture Search (NAS):
[NeurIPS'21] [MLSys'21] [NeurIPS'20, spotlight] [NeurIPS'20] [ICLR'20] [CVPR'20] [CVPR'20] [ICLR'19] [CVPR'19, oral] [ECCV'18] [ICLR'16, BP] [NIPS'15] [ICRA'21] [CVPR'21] [NeurIPS'20] [ACL'20] [CVPR'20] [ECCV'20] [ICLR'20] [NeurIPS'19, spotlight] [ICCV'19] [MICRO'21] [HPCA'21] [HPCA'20] [DAC'21] [DAC'20] [FPGA'17, BP] [ISCA'18]
Deep Compression
compress an existing model by pruning & quantization

Deep Compression, ICLR’16, best paper award
Once for All Network: Train $10^{19}$ networks at the same time

Design new efficient models rather than compressing existing models
Train once, get many, reduce the design cost

Once-for-All Network

Get many ($10^{19}$) child nets for free

OFA network contains many child networks that are sparsely activated

Child networks share the weights with the once-for-All network, trained jointly, amortize the training cost.

OFA can design specialized NN models for diverse hardware platforms

MIT News

Reducing the carbon footprint of artificial intelligence

MIT system cuts the energy required for training and running neural networks.

Rob Matheson | MIT News Office
April 23, 2020

Once-for-All, ICLR’20
MCUNet for Tiny Deep Learning

- TinyML: design light-weighted neural networks and deploy on cheap edge devices that has low power, computing, and memory.
- Billions of IoT devices around the world based on microcontrollers, much cheaper ($1-2), much smaller, everywhere in our lives, but very memory-constraint.
- MCUNet proposed Tiny Neural Architecture Search (TinyNAS) and Tiny Inference Engine (TinyEngine) that can design and deploy neural networks on micro controllers with only 256KB memory.
- MCUNet-v2 proposed a patch-based inference method and reduces the memory by 4x, further reduced memory usage to 30KB, paving the way for tiny machine learning on IoT devices.
TinyML for Automotive Applications and Sensor Fusion

Large number of sensors increased the demand for efficient deep learning computing

Sensor suite of self-driving cars (from Waymo)

Diverse sensors are necessary

Camera in low-light condition

LiDAR in rainy weather
TinyML for LiDAR and Automotive Applications

- 3D point cloud models: 10x more computationally expensive than 2D CNNs
- Challenge: highly sparse & irregular, large memory footprint
- Random memory access is unfriendly for CPU/GPU/TPU => SW & HW co-design

3D neural architecture search

New design space, new primitive for point cloud

Point-Based Feature Transformation (Fine-Grained)

Normalization

Voxelize

Voxel Conv

Devoxelize

Voxel-Based Feature Aggregation (Coarse-Grained)

Algorithm

Software

Hardware

Maps (in, out, wgt)

Gather By Weight

Add

W_{1,1}

W_{1,2}

W_{1,3}

W_{2,1}

W_{2,2}

W_{2,3}

W_{3,1}

W_{3,2}

W_{3,3}

Point-Voxel CNN, NeurIPS'19 spotlight]

Software

Hardware accelerator for point cloud

GPU library for 3D sparse convolution

Hardware accelerator for point cloud

3D neural architecture search

New design space, new primitive for point cloud

Point-Based Feature Transformation (Fine-Grained)

Normalization

Voxelize

Voxel Conv

Devoxelize

Voxel-Based Feature Aggregation (Coarse-Grained)

Algorithm

Software

Hardware

Maps (in, out, wgt)

Gather By Weight

Add

W_{1,1}

W_{1,2}

W_{1,3}

W_{2,1}

W_{2,2}

W_{2,3}

W_{3,1}

W_{3,2}

W_{3,3}

Point-Voxel CNN, NeurIPS'19 spotlight]

Software

Hardware accelerator for point cloud

GPU library for 3D sparse convolution

Hardware accelerator for point cloud

3D neural architecture search

New design space, new primitive for point cloud

Point-Based Feature Transformation (Fine-Grained)

Normalization

Voxelize

Voxel Conv

Devoxelize

Voxel-Based Feature Aggregation (Coarse-Grained)

Algorithm

Software

Hardware

Maps (in, out, wgt)

Gather By Weight

Add

W_{1,1}

W_{1,2}

W_{1,3}

W_{2,1}

W_{2,2}

W_{2,3}

W_{3,1}

W_{3,2}

W_{3,3}

Point-Voxel CNN, NeurIPS'19 spotlight]

Software

Hardware accelerator for point cloud

GPU library for 3D sparse convolution

Hardware accelerator for point cloud

3D neural architecture search

New design space, new primitive for point cloud

Point-Based Feature Transformation (Fine-Grained)

Normalization

Voxelize

Voxel Conv

Devoxelize

Voxel-Based Feature Aggregation (Coarse-Grained)

Algorithm

Software

Hardware

Maps (in, out, wgt)

Gather By Weight

Add

W_{1,1}

W_{1,2}

W_{1,3}

W_{2,1}

W_{2,2}

W_{2,3}

W_{3,1}

W_{3,2}

W_{3,3}

Point-Voxel CNN, NeurIPS'19 spotlight]

Software

Hardware accelerator for point cloud

GPU library for 3D sparse convolution

Hardware accelerator for point cloud

3D neural architecture search

New design space, new primitive for point cloud

Point-Based Feature Transformation (Fine-Grained)

Normalization

Voxelize

Voxel Conv

Devoxelize

Voxel-Based Feature Aggregation (Coarse-Grained)

Algorithm

Software

Hardware
Tiny Transfer Learning

- AI systems need to continually adapt to new data collected from the sensors
- Not only inference, but also re-training the model on edge devices

- On-device: data cannot be sent to the cloud for privacy reason
- But edge devices have limited memory
- TinyTL reduces the training memory from 300MB to 16MB
Video Demos
TinyML Brings AI to Mobile Devices
Running on Qualcomm Snapdragon 855 SoC
HW-aware NAS technique achieves real-time inference on top of existing inference engine (SNPE)
Unlock many mobile AI applications: healthcare, smart home, automotive…

- on-device car/person detection
- on-device pose estimation
- on-device segmentation
Joint Detection and Segmentation for Automotive Applications

runs 102 FPS on Jetson Xavier AGX, TensorRT 8.0 with FP16
3D Object Detection with Multi-Sensor Fusion

Camera View (6-camera fusion)  Bird Eye View

Ours: 10x fewer GMACs, 4.8x faster on NVIDIA Jetson Xavier AGX