Efficient Iterative Neural Architecture Search for Object Detection on IoT Devices

Engineering Data Science

Tony Tran

thtran37@cougarnet.uh.edu

Cullen College of Engineering, Research Computing

University of Houston, Houston, USA

04.17.2026

NAIL
Networked Autonomous Intelligent Learning

UNIVERSITY OF HOUSTON

Tony Tran

Publications

1) **Tony Tran**, Qin Lin, and *Bin Hu*. ELASTIC: Efficient Once For All Iterative Search for Object Detection on Microcontrollers. IEEE Transactions on Computers 2026.
2) **Tony Tran** and *Bin Hu*. TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops 2026.
3) Richie R. Suganda, **Tony Tran**, Miao Pan, Lei Fan, Qin Lin, and *Bin Hu*. Distributed Perception Aware Safe Leader Follower System via Control Barrier Methods. In 2025 IEEE International Conference on Robotics and Automation (ICRA).
4) **Tony Tran** and *Bin Hu*. FACETS: Efficient Once-For-All Object Detection via Constrained Iterative Search. In 2025 IEEE International Conference on Robotics and Automation (ICRA) Late Breaking Session.

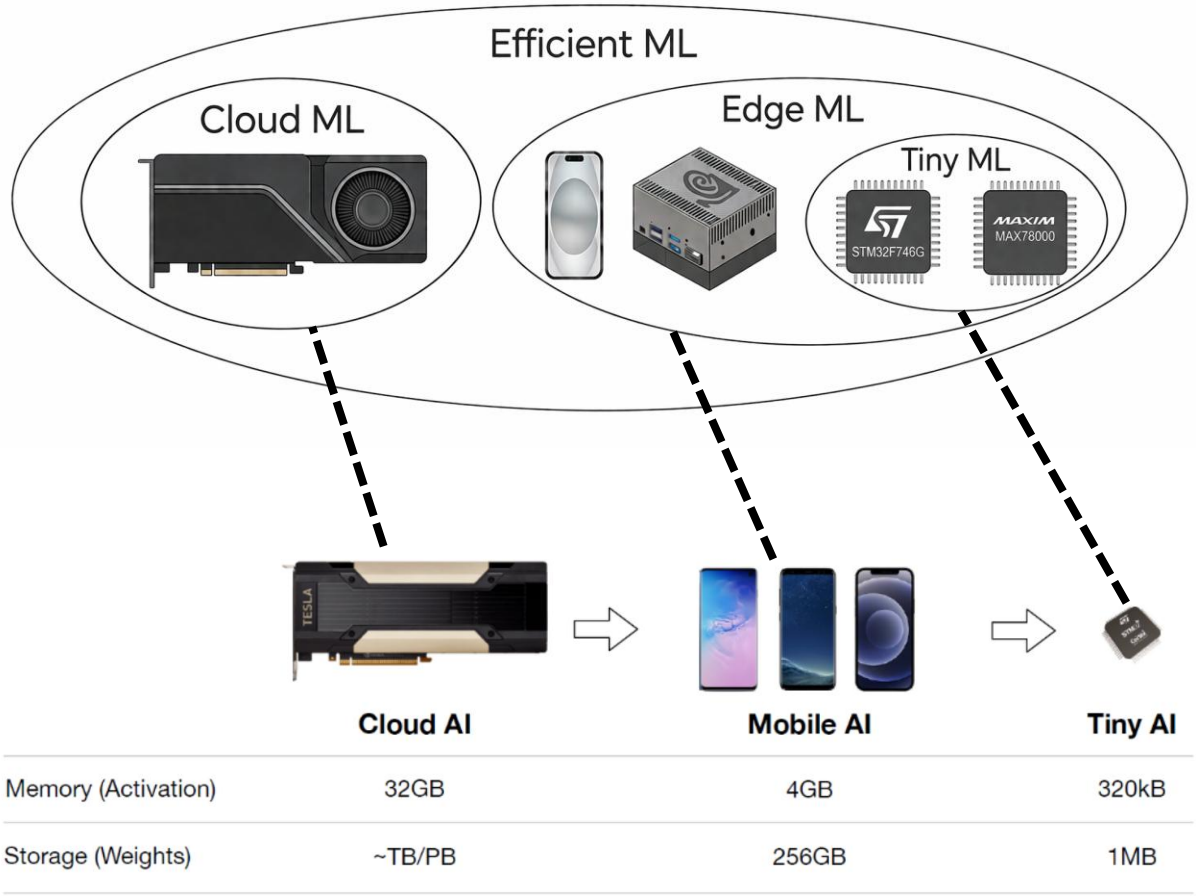
Achievements

1) *Bin Hu* and **Tony Tran**. NVIDIA Academic Grant Recipient (Robotics and Edge AI) (March 2026): Heterogeneous Multi-Robot Waste Detection with Conformal Runtime Monitoring.

Tiny Machine Learning

UNIVERSITY OF HOUSTON

Tiny Machine Learning (TinyML) is the Intersection of Machine Learning and Embedded IoT Devices, Allowing Machine Learning Algorithms to Run on Low-Power Microcontrollers



	Cloud AI	Mobile AI	Tiny AI
Memory (Activation)	32GB	4GB	320kB
Storage (Weights)	~TB/PB	256GB	1MB

TinyML and Efficient Deep Learning Computing, Introduction [Han, S., MIT 6.590 2024].

Why TinyML?

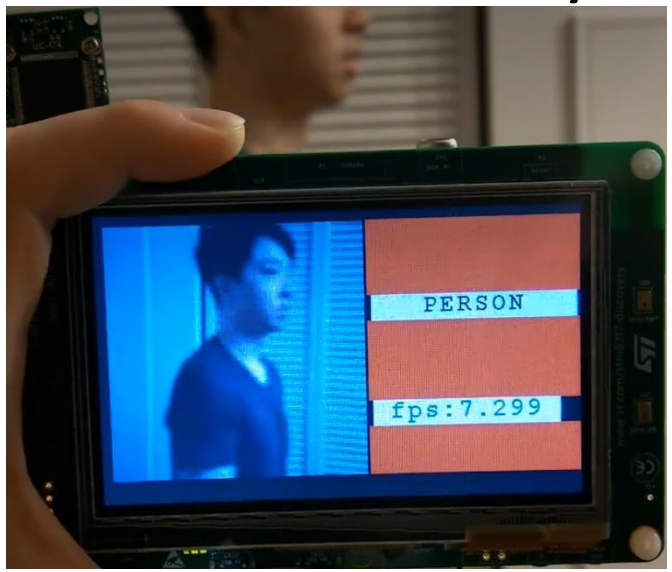
- **Privacy and Security**: Sensitive Data Processed Onboard and not Sent to Cloud
- **Low-Latency Applications**: Useful for Autonomous Systems, Industrial Automation, and Safety-Critical Applications
- **Energy Efficiency**: Sustainable for Battery-Powered Devices and Remote Applications

TinyML and Embedded AI

UNIVERSITY OF HOUSTON

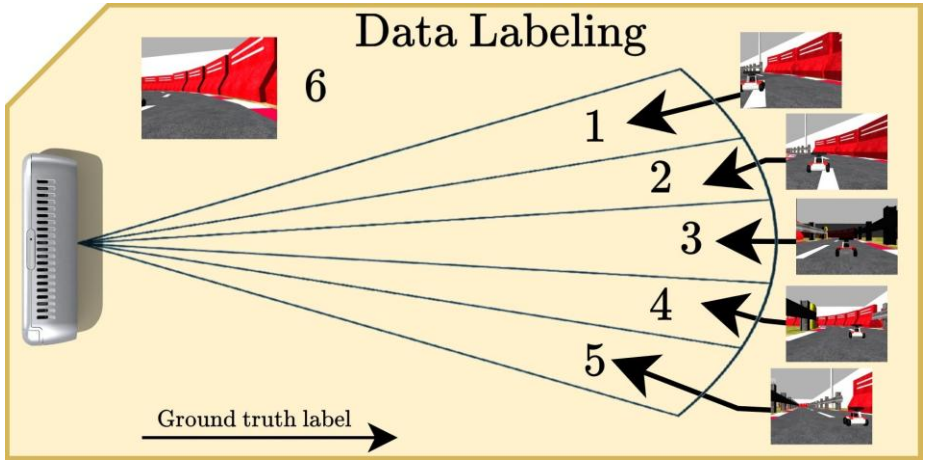
TinyML and Embedded AI Enable Intelligent, On-Device Decision-Making in Applications such as Wearables, Robotics, Monitoring Systems, etc.

Visual Wake Words on STM32F746 in Under 200 KB Memory



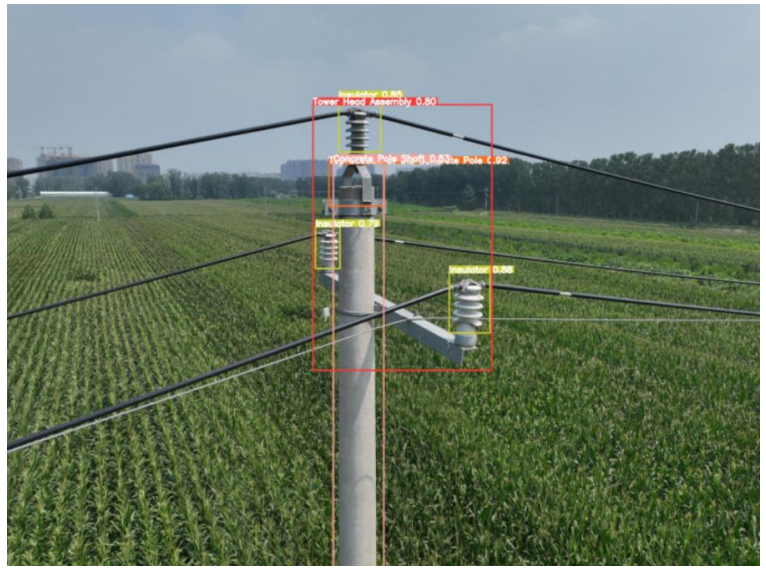
MCUNet: Tiny Deep Learning on IoT Devices [Lin et al., NIPS 2020].

Onboard Perception on Rosbot Pro 2.0 for State Estimation



Distributed Perception Aware Safe Leader Follower System via Control Barrier Methods [Suganda et al., ICRA 2025]

Infrastructure Inspection via Drone Vision Systems



A Lightweight Drone Vision System for Autonomous Inspection with Real-Time Processing [Zhou et al., Drones 2026].

The Embedded AI Problem

Energy

- Embedded Devices have **Limited Battery Life**
  - Minimize Energy Consumption to **Maximize Operation Time**

Latency

- Embedded Devices Need to Support **Real-Time Decision Making**
  - **Minimize Inference Latency** on Resource-Constrained Hardware

Storage

- Embedded Devices have **Limited On-Chip Flash** and **Memory**
  - Must Satisfy **Strict Flash Constraints** (Weights) and **Memory Constraints** (Activation)

Thesis Presentation Outline

UNIVERSITY OF HOUSTON

- ✓ Prior Work and Challenges
- ✓ Iterative Neural Architecture Search Framework
- ✓ Iterative Search for Waste Detection
- ✓ Conclusion and Future Work

Thesis Presentation Outline

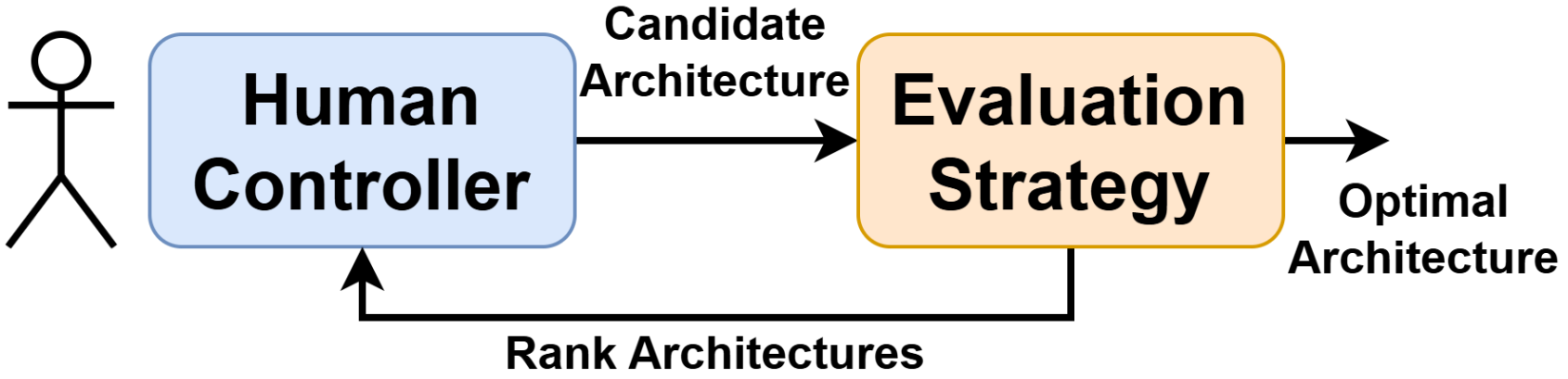
UNIVERSITY OF HOUSTON

- ✓ Prior Work and Challenges
- ✓ Iterative Neural Architecture Search Framework
- ✓ Iterative Search for Waste Detection
- ✓ Conclusion and Future Work

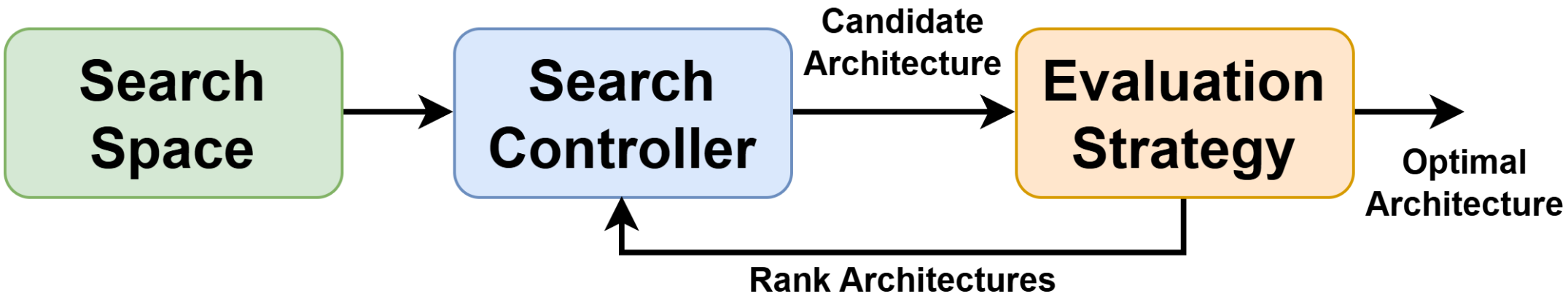
Neural Architecture Search

UNIVERSITY OF HOUSTON

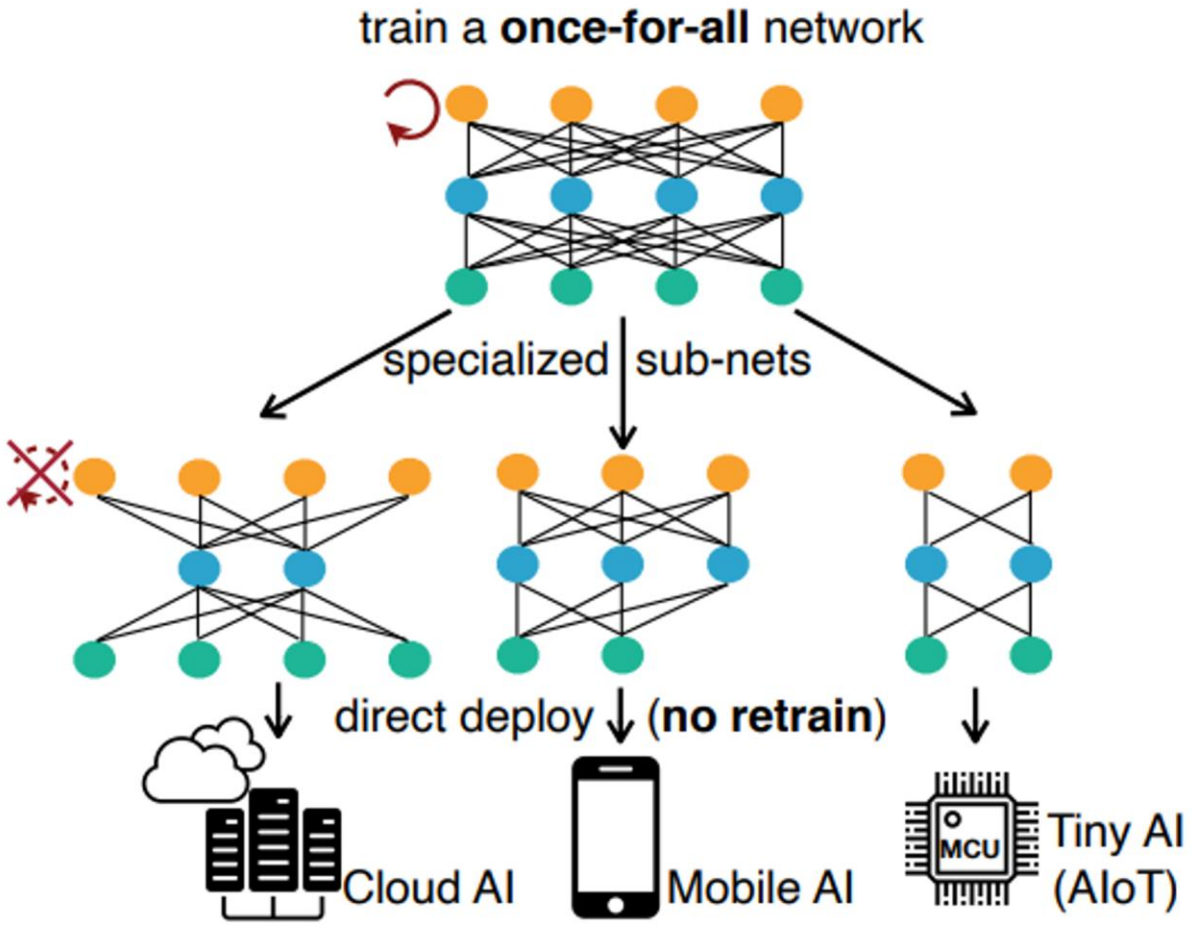
Traditional Neural Network Design



Neural Architecture Search (NAS) can Automate the Design of Neural Network Models



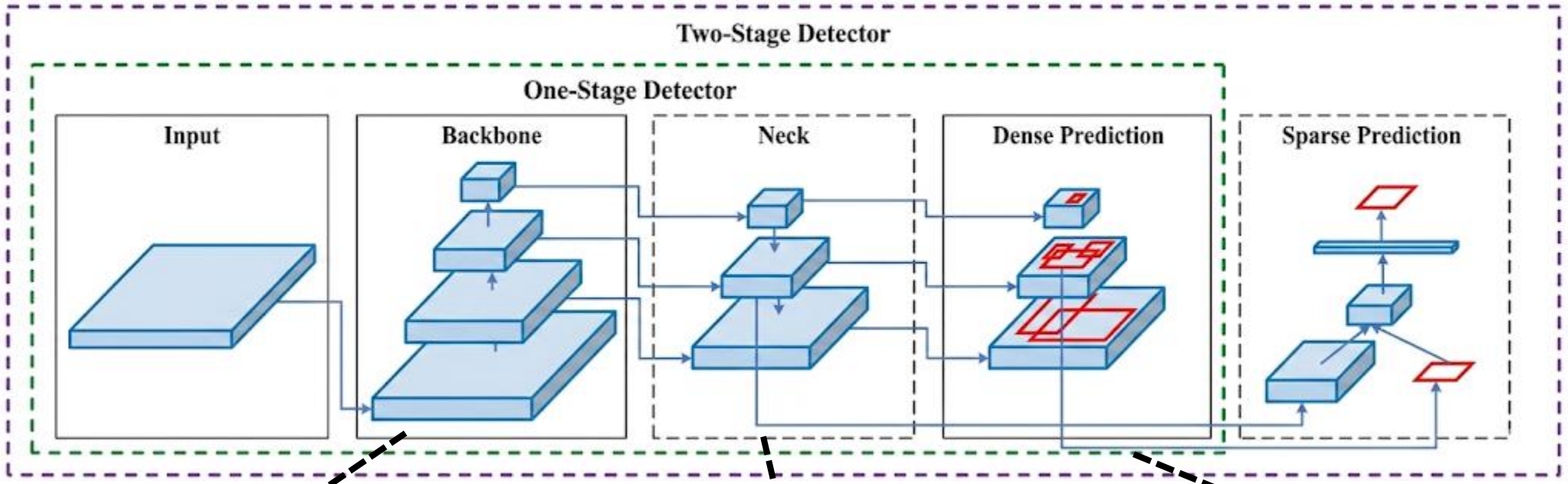
Once-For-All Neural Architecture Search



Once-for-All: Train One Network and Specialize it for Efficient Deployment [Cai et al., ICLR 2020].

1. Train a Once-For-All Network
   - Define Search Space
   - Train Supernet
   - Train Subnets Equally
2. Search for Specialized Subnets
   - Evolutionary Algorithm
   - Under Memory Constraints
3. Directly Deploy Subnetworks on Diverse Platforms
   - Yield Many Subnetworks
   - Directly Deploy on Platform

NAS for Object Detection



Backbone (Feature Extraction)

DetNAS [Chen et al., NIPS 2019]

DetOFA [Sakuma et al., ICCVW 2023]

HIO-NAS [Chen et al., IEEE Access 2025]

Neck (Feature Aggregation)

NAS-FPN [Ghiasi et al., CVPR 2019]

NAS-FCOS [Wang et al., CVPR 2020]

EASFP [Xue et al., SSRN 2025]

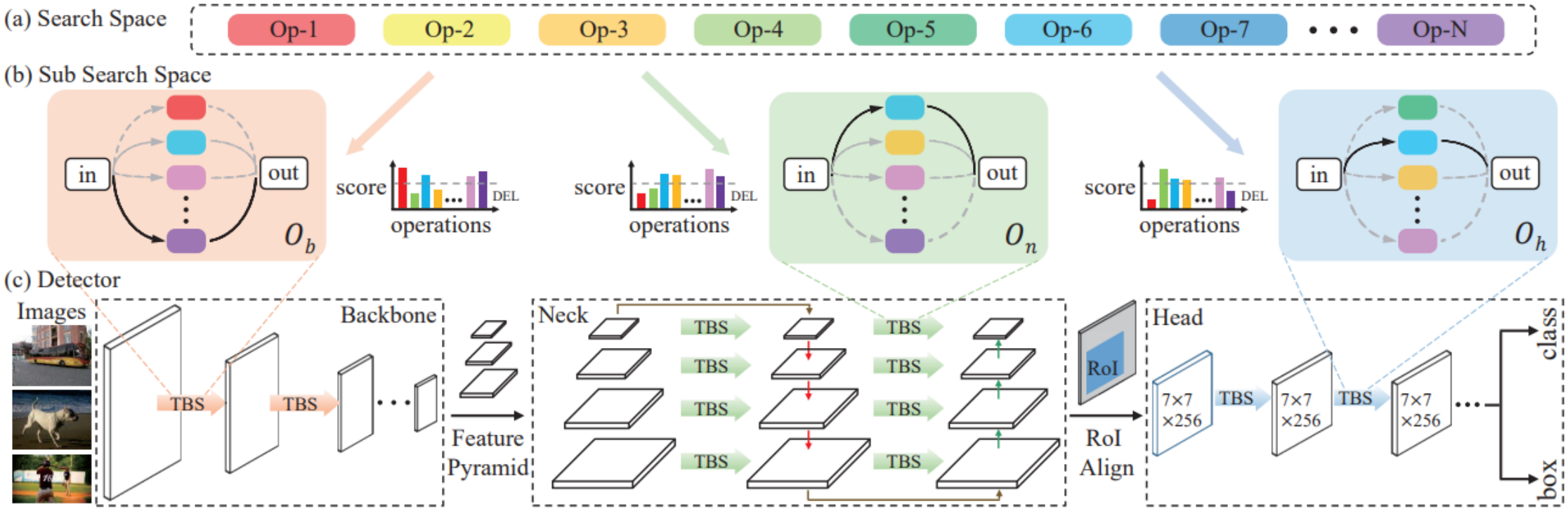
Head (Multi-Scale Prediction)

NAS-FCOS [Wang et al., CVPR 2020]

Hit-Detector [Guo et al., CVPR 2020]

NAS-OD [Rana et al., ICEIC 2024]

Hit-Detector [Guo et al., CVPR 2020]



Hit-Detector: Hierarchical Trinity Architecture Search for Object Detection [Guo et al., CVPR 2020]

Search Space Screening

- Screens Optimal Search Space for Each Component
- Jointly Optimizes the Object Detection Network

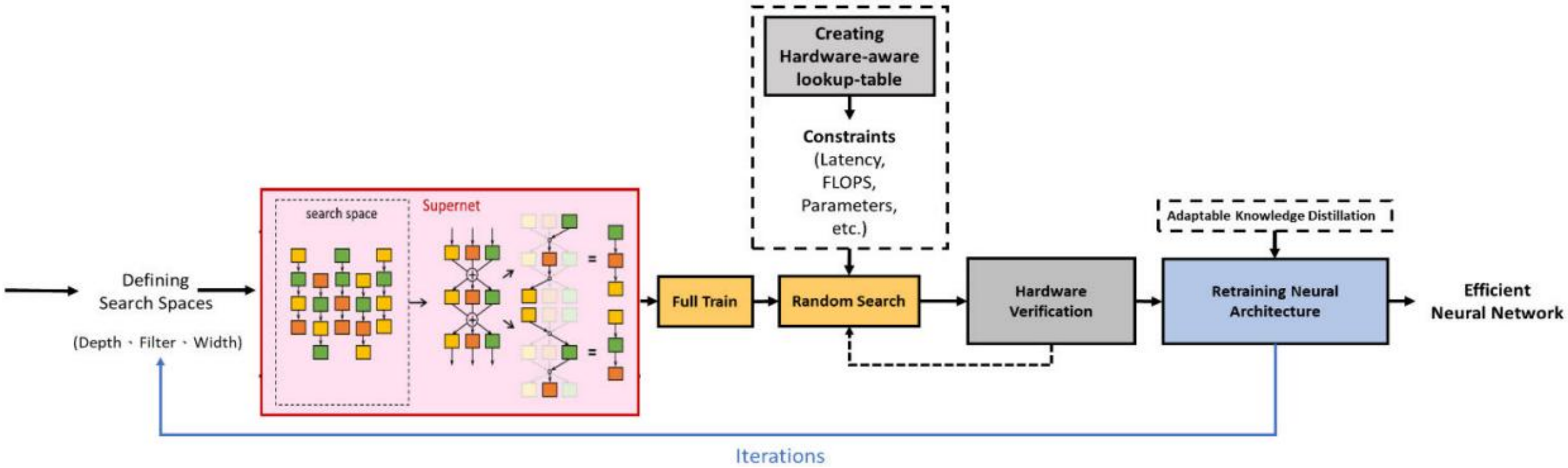
Jointly Optimize Object Detection Network

- Backbone, Neck, Head
- Ensures Compatible Components

Limitations

Joint Optimization is Computationally Expensive for the Large, Combinatorial Search Space

HIO-NAS [Chen et al., IEEE Access 2025]



Hardware-Aware Iterative One-Shot Neural Architecture Search With Adaptable Knowledge Distillation for Efficient Edge Computing [Chen et al., IEEE Access 2025]

Iterative Three Dimensional Search

- Depth, Filter Size, Width
- Avoids the Combinatorial Search Space

Backbone-Only Search for Object Detection

- Target Backbone of YOLOv7
- Backbone Responsible for Feature Extraction

Limitations

How to Ensure the Neck/Head Architecture is Optimal for the Efficient Backbone?

Gaps in Prior Work

UNIVERSITY OF HOUSTON

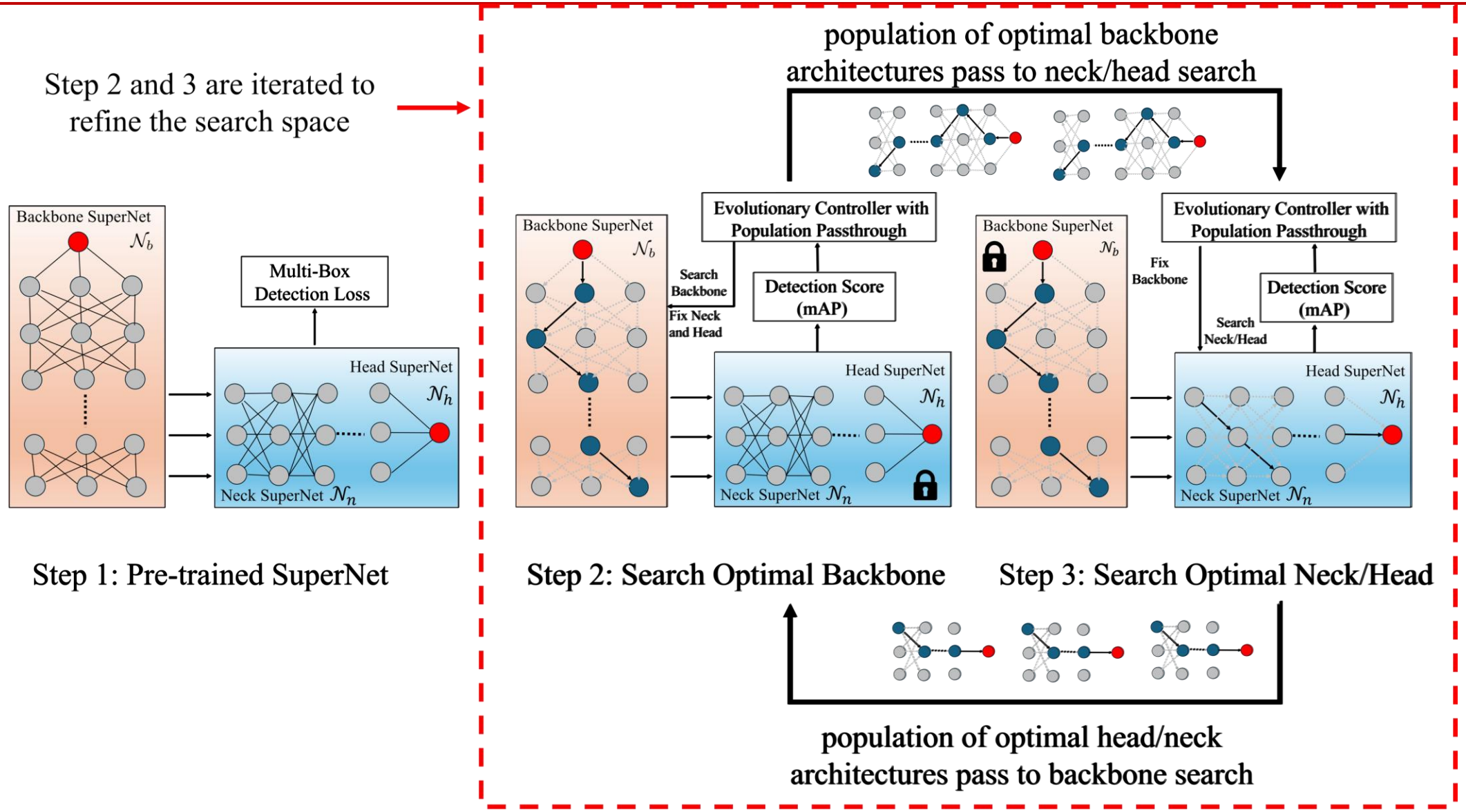
- ✓ Object Detection in TinyML is Difficult due to H**igh Memory Demand** for Networks and **Low Memory Budget** for Embedded Devices
- ✓ Prior Neural Architecture Search Methods for Object Detection Often **Target One Component** During Search, **Limiting Backbone-Neck-Head Co-Adaptation**
- ✓ Joint Neural Architecture Search is **Computationally Expensive** because **Search Space Grows Combinatorially**

Thesis Presentation Outline

UNIVERSITY OF HOUSTON

- ✓ Prior Work and Challenges
- ✓ Iterative Neural Architecture Search Framework
- ✓ Iterative Search for Waste Detection
- ✓ Conclusion and Future Work

Constrained Iterative Search Framework

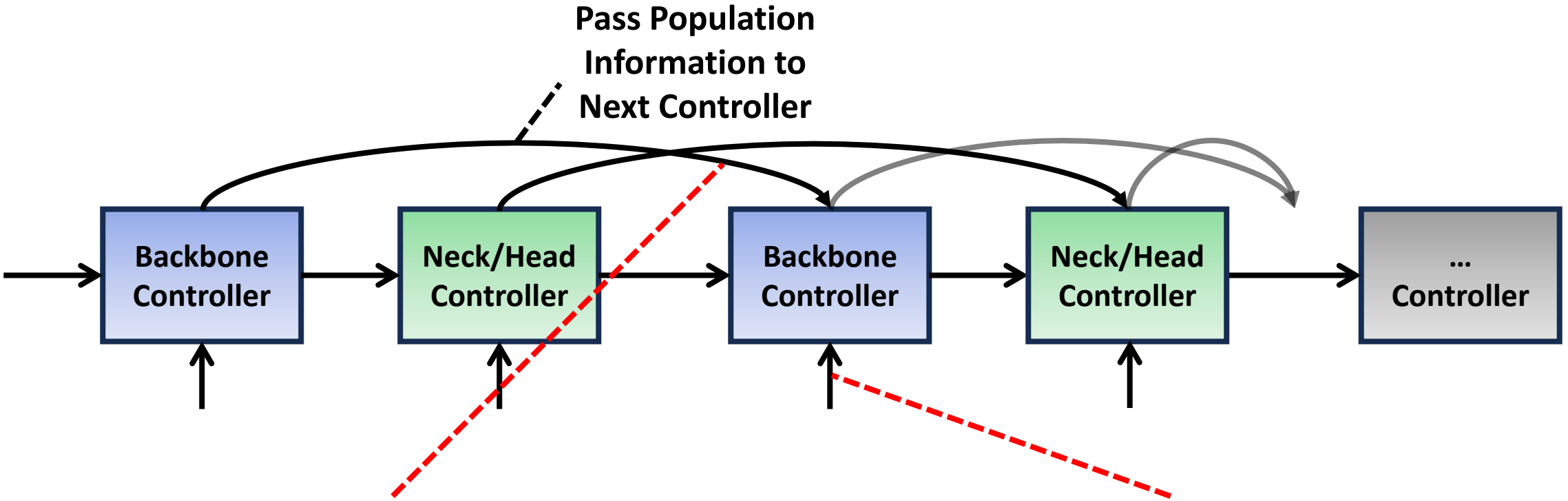


ELASTIC: Efficient Once For All Iterative Search for Object Detection on Microcontrollers [Tran et al., IEEE TC 2026].

Population Passthrough Across Stages

UNIVERSITY OF HOUSTON

We Adopt a Novel Population Passthrough Mechanism to Preserve High-Performing Candidates



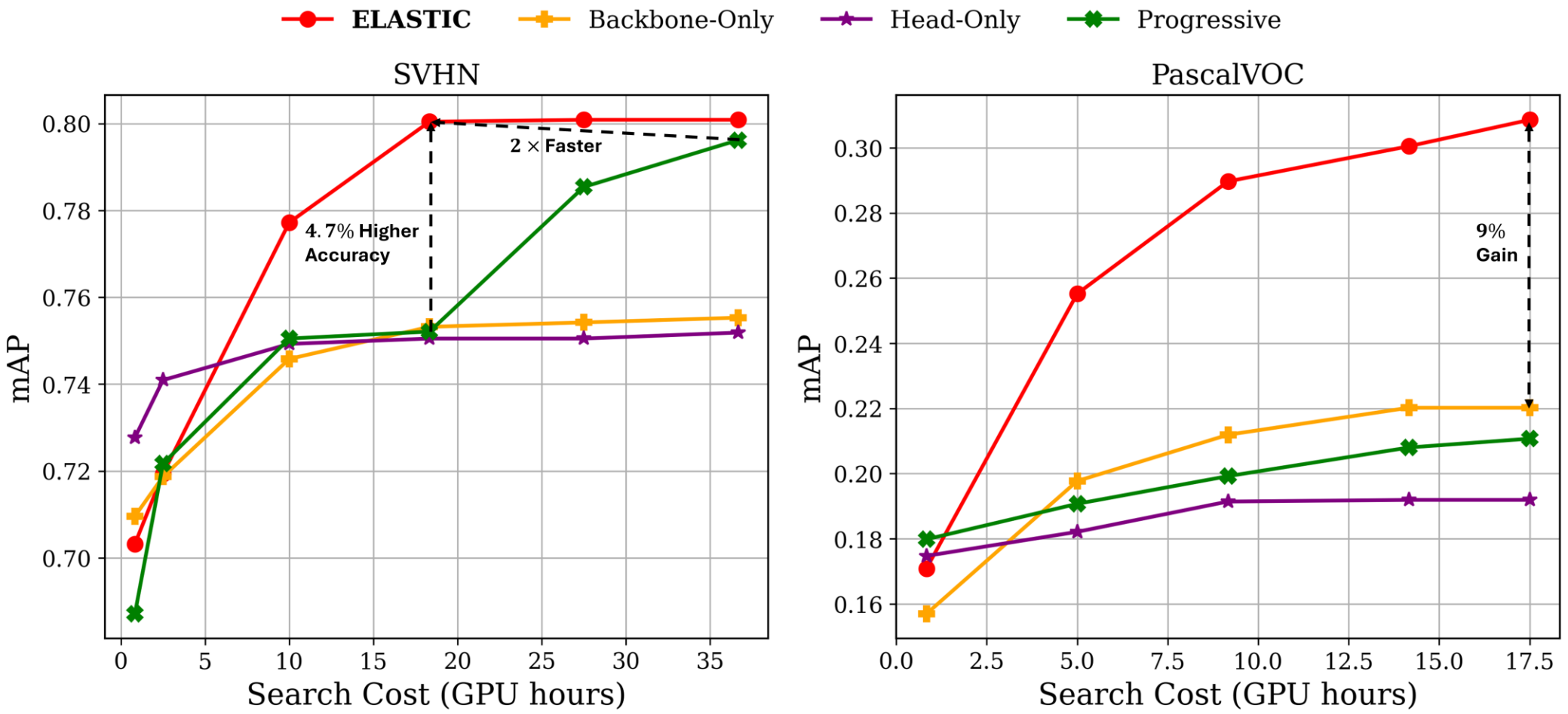
Elite Passthrough (Exploitation):

A Fraction of Best Performing Archs Passed to Seed Next Search

Diversity Augmentation (Exploration):

New Population is Randomly Seeded from Supernet

Comparative Search Results



ELASTIC: Efficient Once For All Iterative Search for Object Detection on Microcontrollers [Tran et al., IEEE TC 2026].

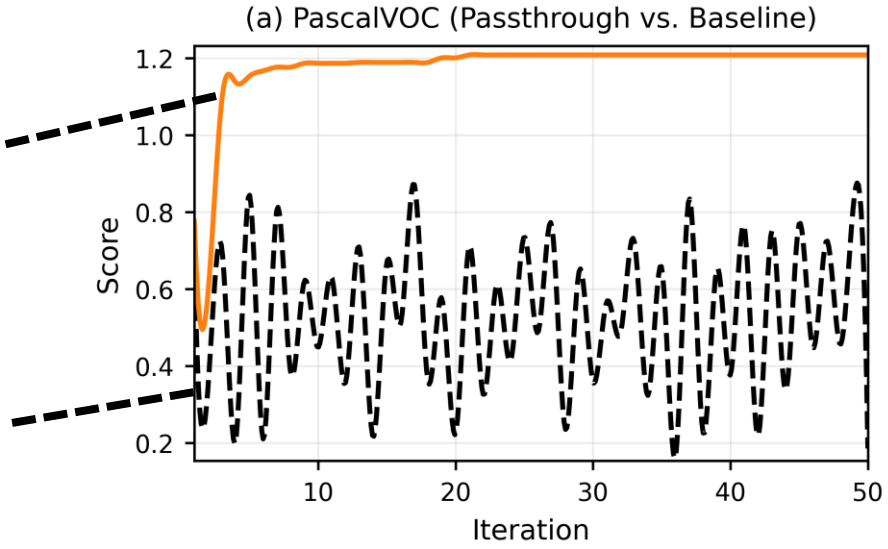
Iterative Search can Converge 2x Faster at 4.7% Higher Accuracy on SVHN Dataset and has 9% Gain on PascalVOC Subset

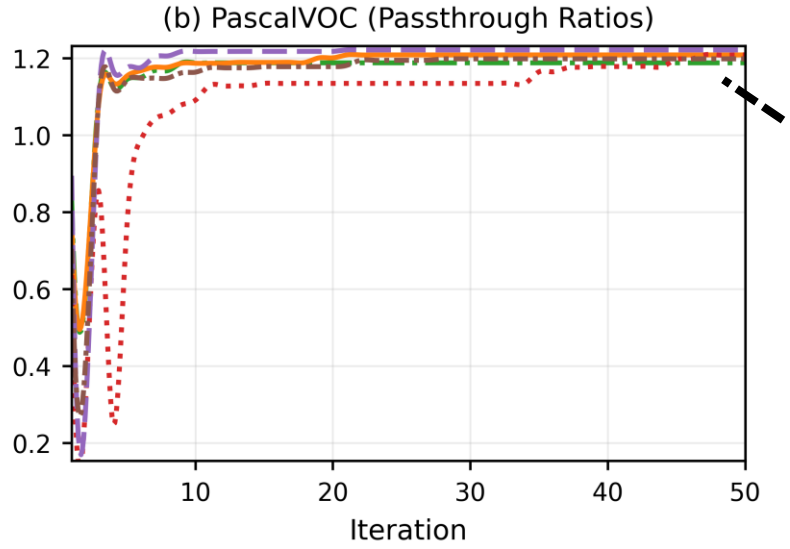
Search Stability via Population Passthrough

UNIVERSITY OF HOUSTON

With Population Passthrough

Without Population Passthrough



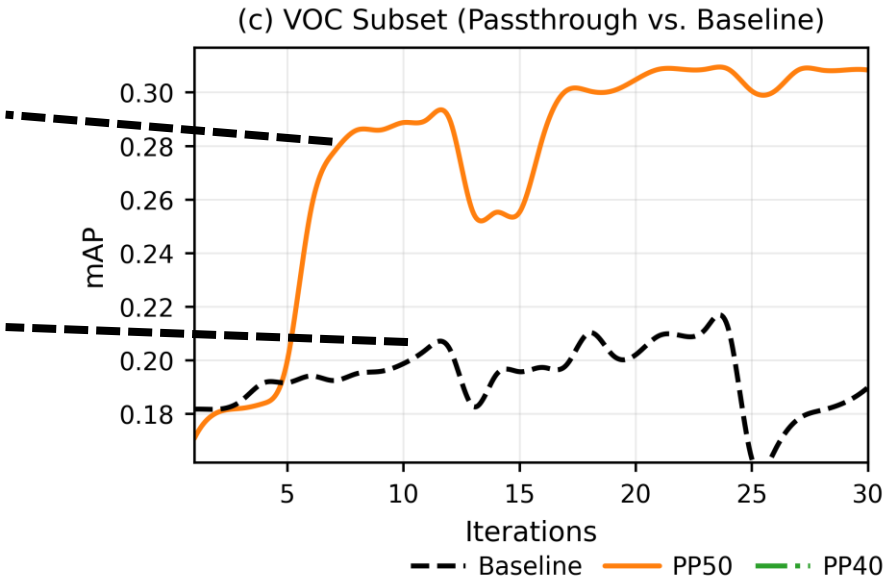


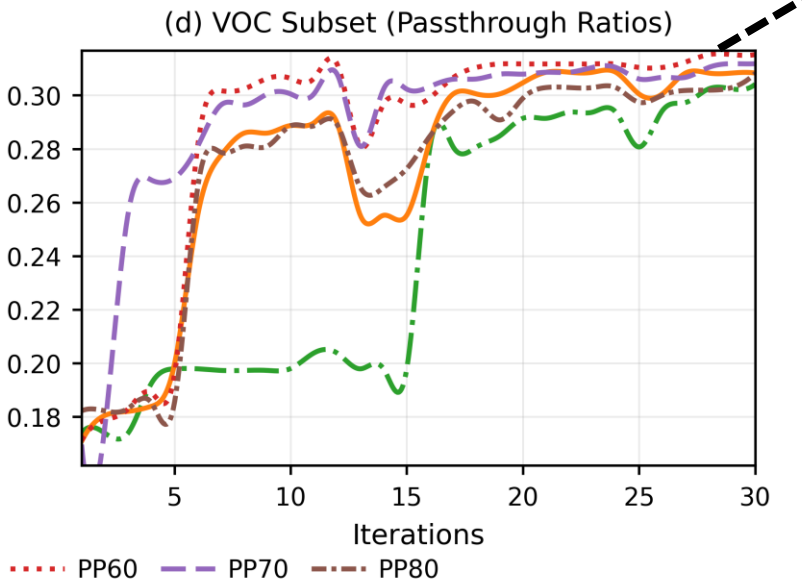
60% Performs Best

70% Performs Best

Can Converge

Fails to Converge

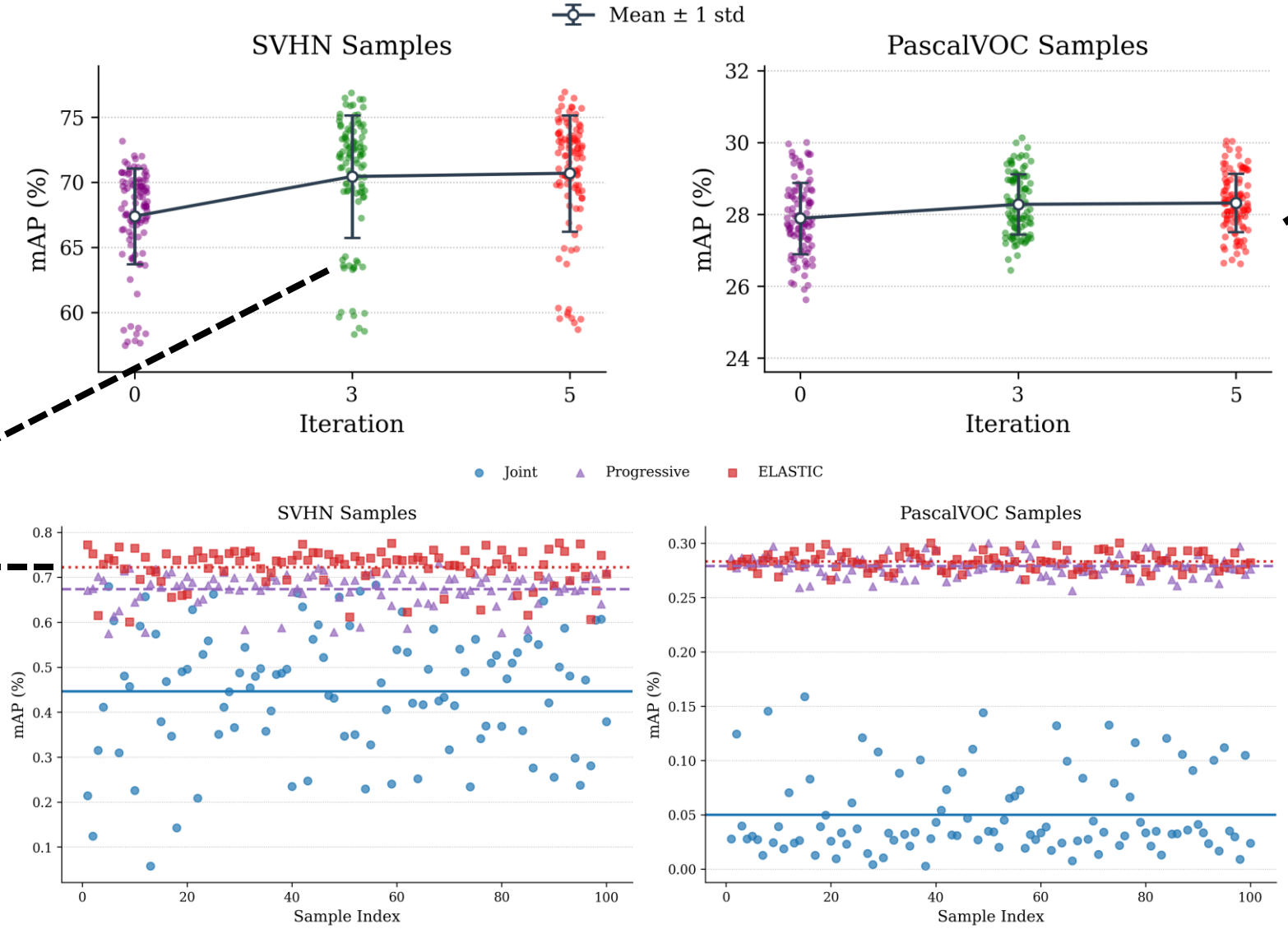




Highlights Tradeoff Between Exploration and Exploitation

ELASTIC: Efficient Once For All Iterative Search for Object Detection on Microcontrollers [Tran et al., IEEE TC 2026].

Search Space Refinement Analysis



Iterative Refinement of Search Space

Lower Standard Deviation and Higher Mean mAP

ELASTIC: Efficient Once For All Iterative Search for Object Detection on Microcontrollers [Tran et al., IEEE TC 2026].

MCU Deployment Results

UNIVERSITY OF HOUSTON

Model	Dataset	Device	Params	Energy (μJ)	Latency (ms)	Power (mW)	mAP (%)
ai87-fpndetector [1]	PascalVOC	MAX78002	2.18M	62001	122.6	445.76	50.66
ELASTIC (OURS)	PascalVOC	MAX78002	**1.32M**	**17581**	**51.1**	**285.02**	**57.65**
ai87-fpndetector [1]	TrashNet	MAX78002	2.18M	62001	122.6	445.76	83.1
ELASTIC (OURS)	TrashNet	MAX78002	**1.32M**	**17581**	**51.1**	**285.02**	**93.3**
ai85net-tinierssd-face [1]	VGGFace2	MAX78000	0.28M	1712	**43.4**	29.90	84.72
ELASTIC (OURS)	VGGFace2	MAX78000	**0.22M**	**1368**	45.6	**20.90**	**87.10**
ai85net-tinierssd-face [1]	VGGFace2	STM32746	0.277M	N/A	968.2	N/A	80.2
ELASTIC (OURS)	VGGFace2	STM32746	**0.168M**	N/A	**699.1**	N/A	82.1
ai85net-tinierssd [1]	SVHN	MAX78000	**0.19M**	573	14.0	29.20	83.60
ELASTIC (OURS)	SVHN	MAX78000	0.22M	**341**	**13.0**	**16.70**	**88.10**
ai85net-tinierssd [1]	SVHN	STM32746	0.229M	N/A	449.3	N/A	83.5
ELASTIC (OURS)	SVHN	STM32746	**0.235M**	N/A	523.7	N/A	**87.9**

ELASTIC: Efficient Once For All Iterative Search for Object Detection on Microcontrollers [Tran et al., IEEE TC 2026].

Iterative Search Reduces Energy Consumption, Latency, and Power Usage while Achieving Higher mAP

Digit Detection Demo on MAX78000

Face Detection Demo on MAX78000

UNIVERSITY OF HOUSTON

NAIL

Networked Autonomous Intelligent Learning

Thesis Presentation Outline

UNIVERSITY OF HOUSTON

- ✓ Prior Work and Challenges
- ✓ Iterative Neural Architecture Search Framework
- ✓ Iterative Search for Waste Detection
- ✓ Conclusion and Future Work

Computer Vision for Waste Monitoring

UNIVERSITY OF HOUSTON

Uncontrolled Waste Disposal and Littering Contribute Directly to Environmental Pollution

Waste in the Wild

TACO: Trash Annotations in Context for Litter Detection [Proenca et al., Arxiv 2020].

Underwater Waste

Trash-ICRA19: A Bounding Box Labeled Dataset of Underwater Trash [Fulton et al., DRUM 2020].

Waste from UAV-Perspective

Autonomous, Onboard Vision-Based Trash and Litter Detection in Low Altitude Aerial Images Collected by an Unmanned Aerial Vehicle [Kraft et al., Remote Sensing 2021].

Automatic Image-Based Waste Detection can Enable Large Scale Monitoring in Diverse Environments

Waste Monitoring on the Edge

UNIVERSITY OF HOUSTON

Suppose We Desire some Real-Time Solution for Waste Detection on UAVs

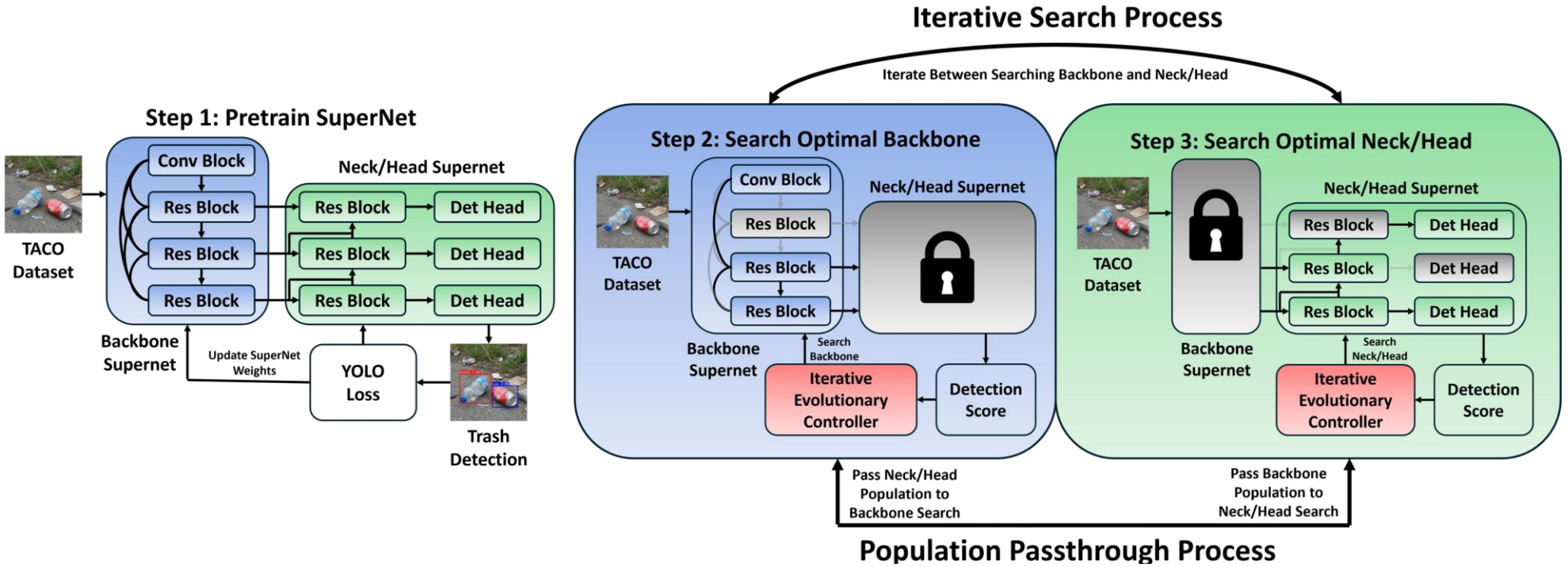
Waste Detection on UAVV

UAV Waste Detector

- Limited Edge Compute
- Limited Onboard Memory
- Limited Flash Capacity
- Power and Energy Constraints

How can we design an efficient real-time detection method on edge devices on systems with limited computational resources?

Iterative Search for Waste Vision



TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection [Tran et al., WACVW 2026].

Step 1

Pretrain Supernet on Waste Detection Dataset, Jointly Optimizing All Possible Subnets

Step 2

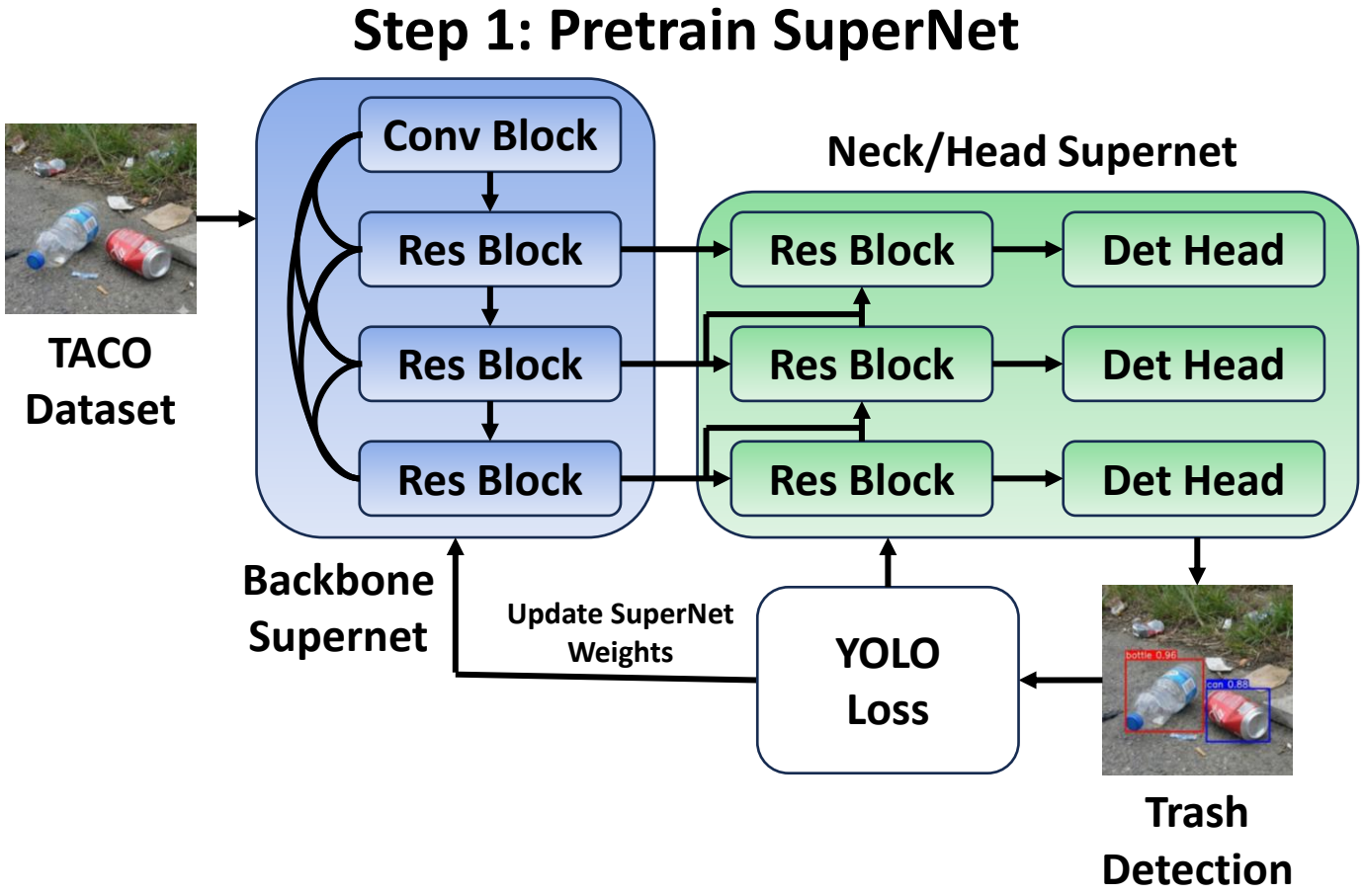
Freeze Neck/Head Module and Search Optimal Backbone Arch Subject to Hardware Constraints

Step 3

Freeze Backbone Module and Search Optimal Neck Arch Subject to Hardware Constraints

Iteratively Repeat Steps 2 and 3 Until Convergence

Pretrain Object Detection SuperNet



Objective 1:

Minimize Loss Function with Respect to the Supernet Detection Output

Objective 2:

Minimize Loss Function with Respect to Randomly Sampled Subnets

Search One Module, Freeze Other Module

UNIVERSITY OF HOUSTON

Step 2: Search Optimal Backbone

TACO Dataset

Conv Block

Res Block

Res Block

Res Block

Backbone Supernet

Neck/Head Supernet

Search Backbone

Iterative Evolutionary Controller

Detection Score

Target Search Module

Step 3: Search Optimal Neck/Head

TACO Dataset

Backbone Supernet

Neck/Head Supernet

Res Block

Res Block

Res Block

Det Head

Det Head

Det Head

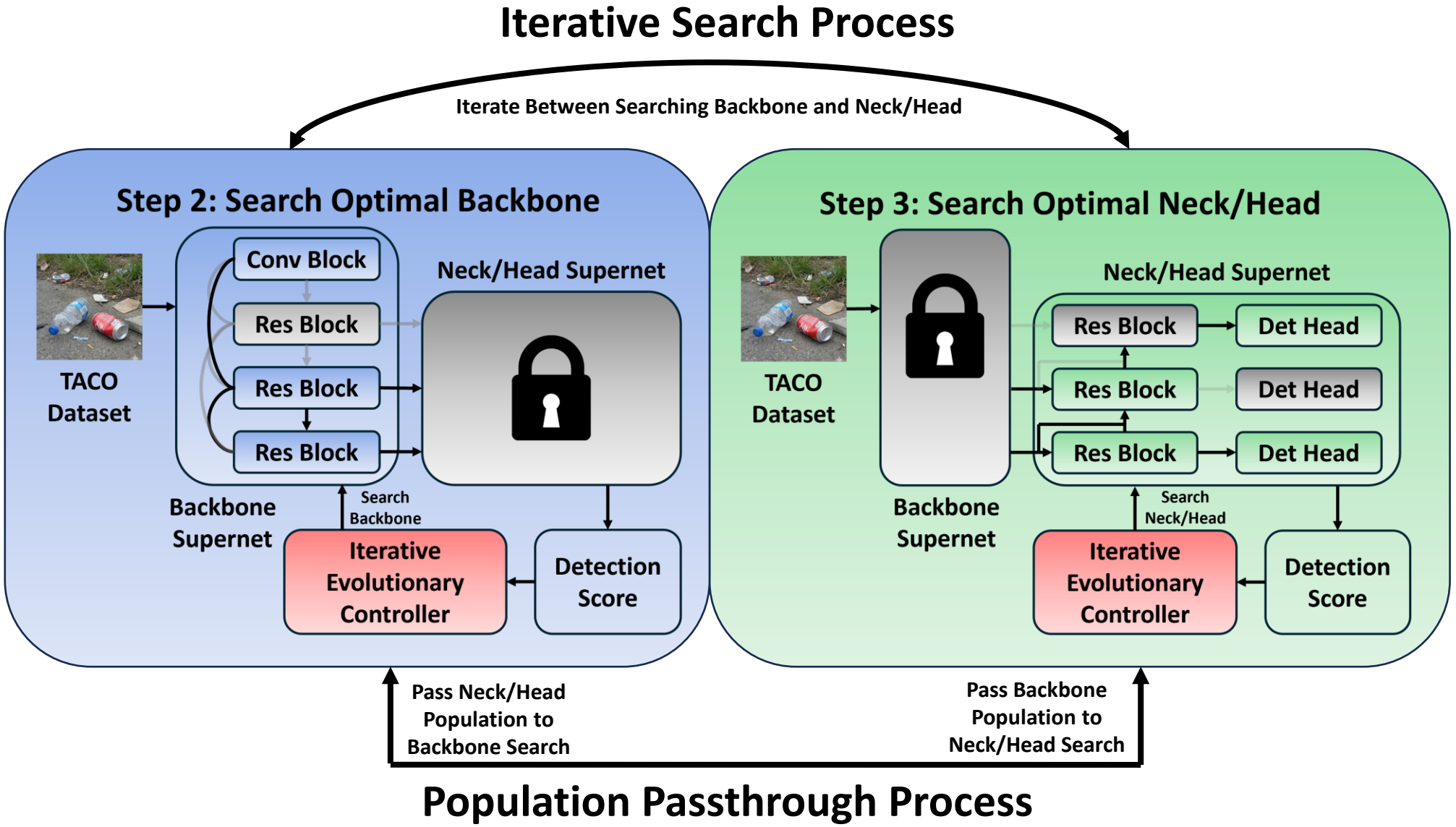
Search Neck/Head

Iterative Evolutionary Controller

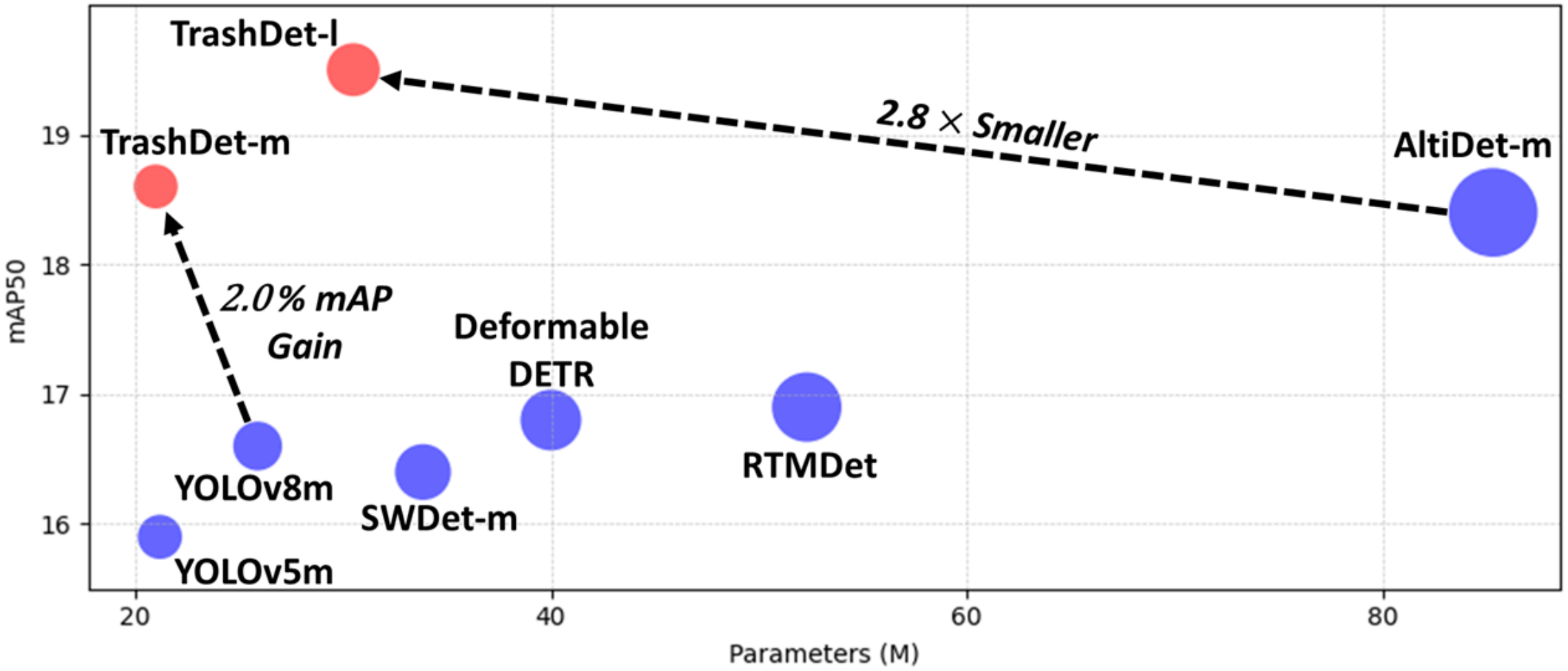
Detection Score

Frozen Companion Module

Iterative Search Process



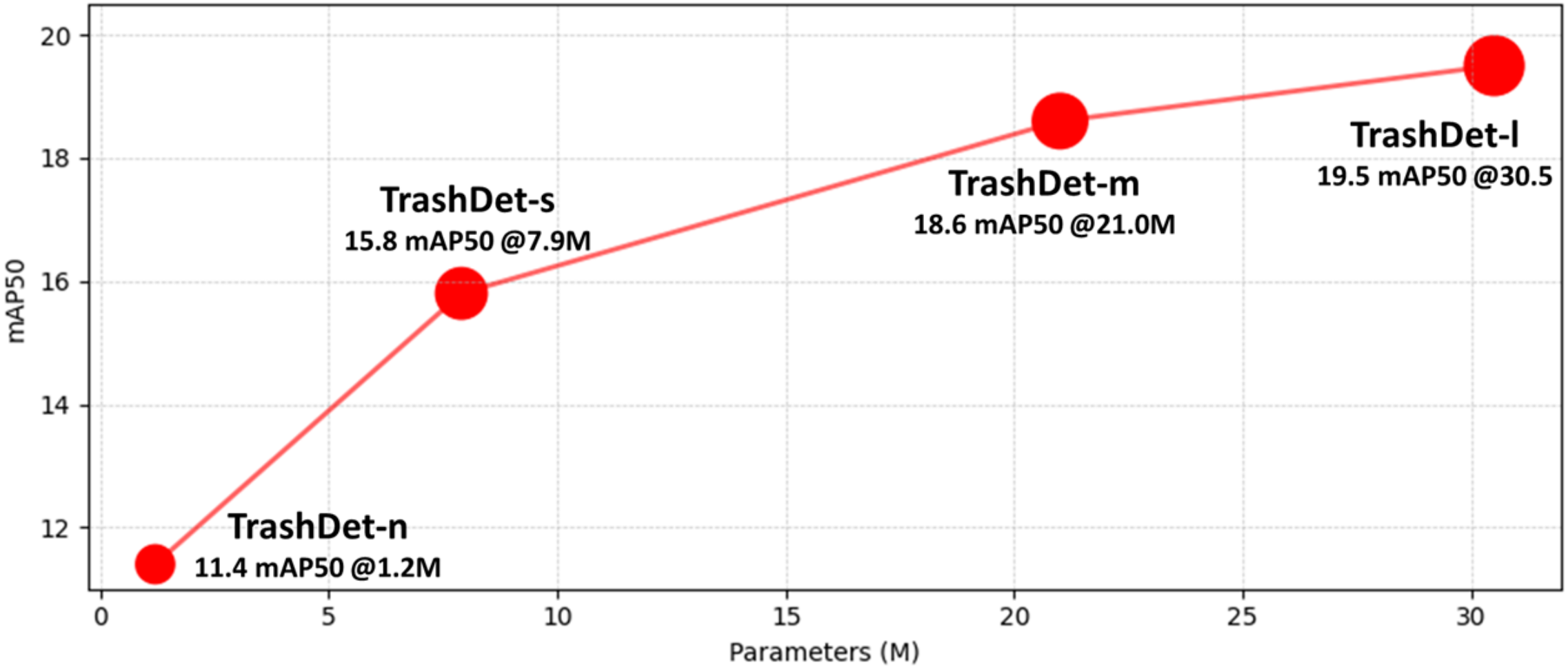
Waste Detection Baseline Comparison



TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection [Tran et al., WACVW 2026].

Our Framework Discovers Models Significantly More Compact and Accurate than Existing Detectors on TACO Dataset

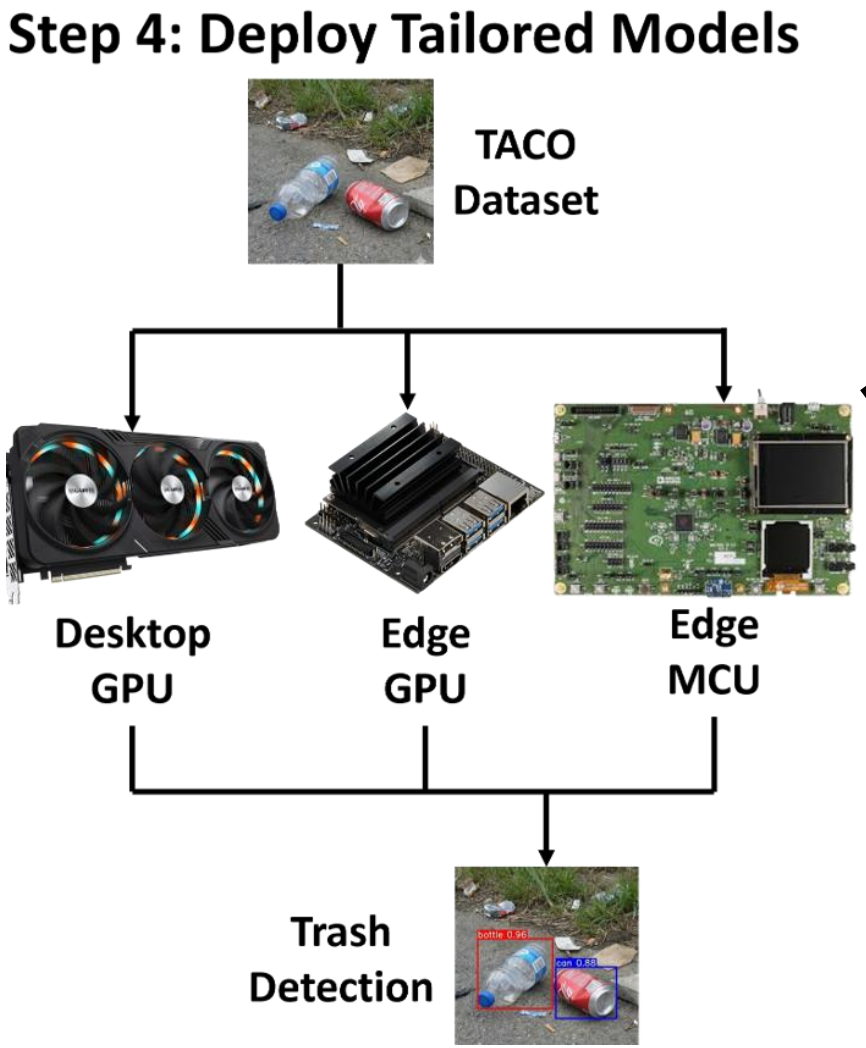
Scalable Waste Detection Variants



TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection [Tran et al., WACVW 2026].

Our NAS Framework can Target a Range of Parameter Budgets, Providing Practitioners with Scalable Options for Diverse Deployment Targets

Deployment on Resource Constrained MCUs



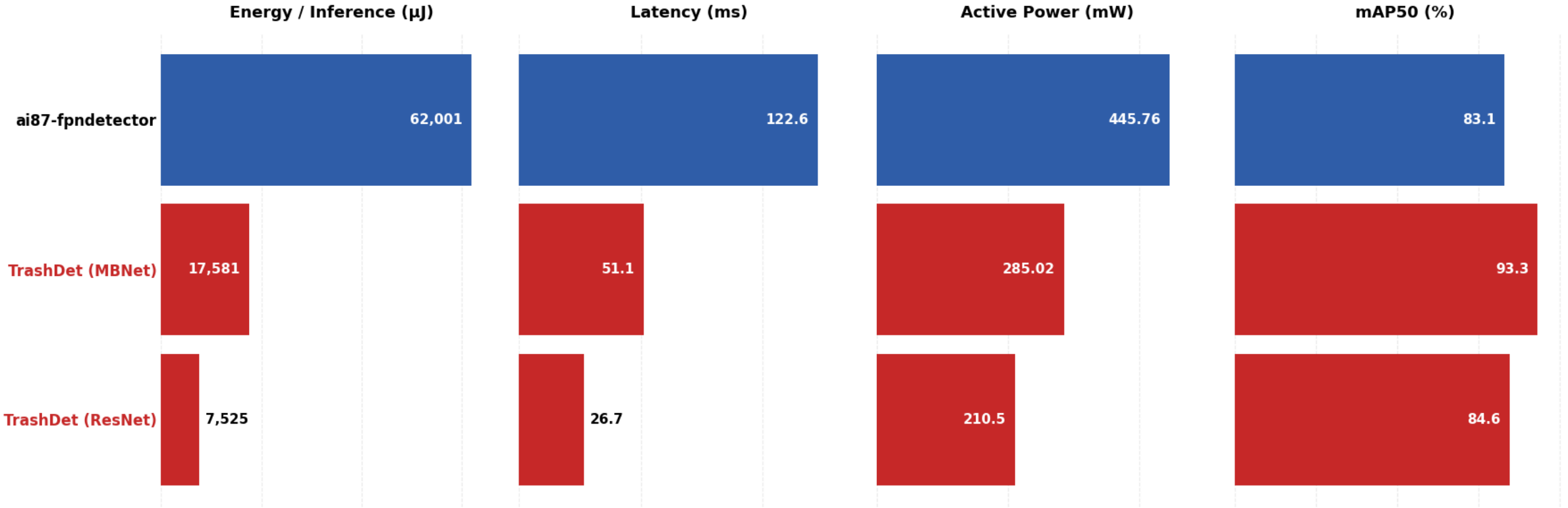
Target Device: MAX78002

Analog Devices' Low-Power AI Microcontroller with Low-Power CNN Accelerator

Hardware Constraints:

- ✓ Weight Memory: 2,340 KiB of Kernel Memory Reserved for Kernel Weights
- ✓ Data Memory: 80 KiB of Data Memory Reserved for Activation (No Streaming Mode)
- ✓ Limited Operators: Restrictions in Conv, Pooling, and Activation Function Parameters

Embedded Deployment for Waste Detection



TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection [Tran et al., WACVW 2026].

87.9%

Energy Reduction Compared Against TrashDet (ResNet)

78.2%

Latency Reduction Compared Against TrashDet (ResNet)

52.8%

Power Reduction Compared Against TrashDet (ResNet)

10.2%

mAP50 Increase Compared Against TrashDet (MBNet)

Comparison Measured on TrashNet Dataset for Object Detection on MAX78002

Trash Detection Demo on ModalAI Sentinel

UNIVERSITY OF HOUSTON

UNIVERSITY OF HOUSTON

TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection

Trash Detection Demo
on ModalAI Sentinel

Networked Autonomous Intelligent Learning

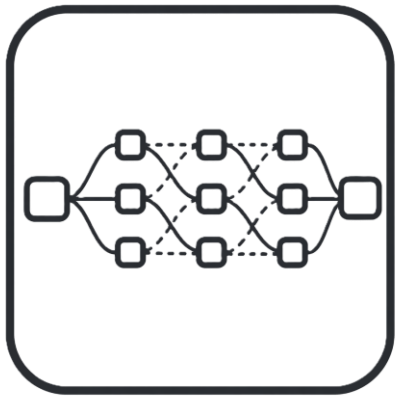
NAIL

Thesis Presentation Outline

UNIVERSITY OF HOUSTON

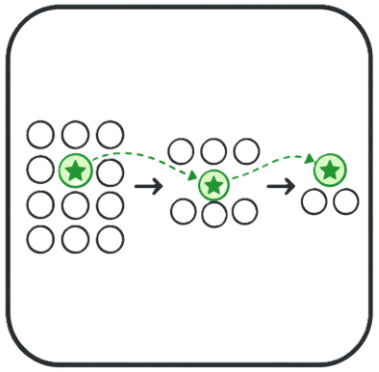
- ✓ Prior Work and Challenges
- ✓ Iterative Neural Architecture Search Framework
- ✓ Iterative Search for Waste Detection
- ✓ Conclusion and Future Work

Conclusion



Utilize a Once-For-All Supernet for Efficient Inherited Evaluation, Filtered by Hardware Constraints

Develop a Novel Iterative Evolutionary Search Method to Enable Efficient Modular Optimization



Introduce a New Population Passthrough Mechanism to Preserve Strong Candidates Across Stages

Future Work

UNIVERSITY OF HOUSTON

- ✓ Scale Iterative Search Beyond Current TinyML Setting to Diverse Robotic Platforms
- ✓ Strengthen the Hardware-Aware Objective, Incorporating Latency, Energy, Scheduling, etc.
- ✓ Refine the Granularity of Iterative Search, Finer or Coarser
- ✓ Broaden Detector Families and Applications Supported by Our Framework

Thesis Committee Members

UNIVERSITY OF HOUSTON

Dr. Bin Hu
Assistant Professor
Committee Chair
NAIL Lab Director

Dr. Xin Fu
Professor
Committee Member
ECOMS Lab Director

Dr. Qin Lin
Assistant Professor
Committee Member
AIR Lab Director

NAIL Lab Spring 2024

UNIVERSITY OF HOUSTON

NAIL/AIR Lab Spring 2025

UNIVERSITY OF HOUSTON

NAIL Lab Fall 2025

NAIL Lab 2024-2026

To my parents, **Anh Tuan Tran** and **Tho Thi Nguyen**, and my sister, **Tina Thaomy Tran**, thank you for being the foundation of everything I have been able to pursue.

Thank You, Questions?

Tony Tran

Master of Science in

Engineering Data Science

thtran37@cougarnet.uh.edu

Cullen College of Engineering, Research Computing

University of Houston, Houston, USA

Networked Autonomous
Intelligent Learning

National Science Foundation Award (IIS-2007386)

References

1) **Tony Tran**, Qin Lin, and Bin Hu. ELASTIC: Efficient Once For All Iterative Search for Object Detection on Microcontrollers. IEEE Transactions on Computers, (01):1–8, March 2026. ISSN 0018-9340. doi: 10.1109/TC.2026.3678184.
2) **Tony Tran** and Bin Hu. TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops, pages 585–593, 2026.
3) Richie R. Suganda, **Tony Tran**, Miao Pan, Lei Fan, Qin Lin, and Bin Hu. Distributed Perception Aware Safe Leader Follower System via Control Barrier Methods. In 2025 IEEE International Conference on Robotics and Automation (ICRA), pages7794–7800, May 2025. doi: 10.1109/ICRA55743.2025.11127883.
4) **Tony Tran** and Bin Hu. FACETS: Efficient Once-For-All Object Detection via Constrained Iterative Search. In 2025 IEEE International Conference on Robotics and Automation (ICRA) Late Breaking Session.
5) Song Han. Introduction [PowerPoint Slides]. MIT 6.5940 TinyML and Efficient Deep Learning Computing, 2024.
6) Ji Lin, Wei-Ming Chen, Yujun Lin, john cohn, Chuang Gan, and Song Han. MCUNet: Tiny Deep Learning on IoT Devices. In Advances in Neural In-formation Processing Systems, volume 33, pages 11711–11722. Curran Associates, Inc., 2020.
7) Zhengran Zhou, Wei Wang, Hao Wu, Tong Wang, and Satoshi Suzuki. A Lightweight Drone Vision System for Autonomous Inspection with Real-Time Processing. Drones, 10(2):126, February 2026.
8) Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. Once for All: Train One Network and Specialize it for Efficient Deployment. In 2020 International Conference on Learning Representations (ICLR), April 2020.
9) Yukang Chen, Tong Yang, Xiangyu Zhang, GAOFENG MENG, Xinyu Xiao, and Jian Sun. DetNAS: Backbone Search for Object Detection. In Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc., 2019.
10) Yuiko Sakuma, Masato Ishii, and Takuya Narihira. DetOFA: Efficient Training of Once-for-All Networks for Object Detection using Path Filter. In 2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), pages 1325–1334, October 2023. doi: 10.1109/ICCVW60793.2023.00143.

References

11) Oscal Tzyh-Chiang Chen, Yu-Xuan Chang, Chih-Yu Chung, Ya-Yun Cheng, and Manh-Hung HA. Hardware-Aware Iterative One-Shot Neural Architecture Search With Adaptable Knowledge Distillation for Efficient Edge Computing. IEEE Access, 13:54204–54222, 2025. ISSN 2169-3536. doi: 10.1109/ACCESS.2025.3554185.
12) Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V. Le. NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 7029–7038, June 2019. doi: 10.1109/CVPR.2019.00720.
13) Ning Wang, Yang Gao, Hao Chen, Peng Wang, Zhi Tian, Chunhua Shen, and Yanning Zhang. NAS-FCOS: Fast Neural Architecture Search for Object Detection. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 11940–11948, June 2020.
14) Yu Xue, Chenhang Yao, M. Wahib, and Moncef Gabbouj. Easfp: Efficient Auto-Searched Feature Pyramid for Object Detection, May 2025.
15) Jianyuan Guo, Kai Han, Yunhe Wang, Chao Zhang, Zhaohui Yang, Han Wu, XinghaoChen, and Chang Xu. Hit-Detector: Hierarchical Trinity Architecture Search forObject Detection. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 11402–11411, June 2020. doi: 10.1109/CVPR42600.2020.01142.
16) Amrita Rana and Kyung Ki Kim. NAS-OD: Neural Architecture Search for Object Detection. In 2024 International Conference on Electronics, Information, and Communication (ICEIC), pages 1–3, January 2024. doi: 10.1109/ICEIC61013.2024.10457265.
17) Pedro F. Proenc¸a and Pedro Sim˜oes. TACO: Trash Annotations in Context for Litter Detection, March 2020.
18) Michael Fulton, Jungseok Hong, Md Jahidul Islam, and Junaed Sattar. Trash-ICRA19: A Bounding Box Labeled Dataset of Underwater Trash. Data Repository for the University of Minnesota (DRUM), 2020.
19) Marek Kraft, Mateusz Piechocki, Bartosz Ptak, and Dawid Walas. Autonomous, Onboard Vision-Based Trash and Litter Detection in Low Altitude Aerial Images Collected by an Unmanned Aerial Vehicle. Remote Sensing, 13(5):965, 2021.

ELASTIC - Quantitative Comparison

UNIVERSITY OF HOUSTON

Dataset	Method	mAP	↑mAP	Cost	MACs	Params	Latency
SVHN	Backbone-Only [13], [20], [21]	75.53%	+0.18%	10.8 hrs	137.6 M	0.73 M	2.62 ms
	Neck/Head-Only [6], [22], [23]	75.35%	+0.00%	4.7 hrs	475.3 M	1.01 M	2.69 ms
	Progressive [14]	79.62%	+4.27%	30.8 hrs	78.4 M	0.52 M	2.36 ms
	ELASTIC (OURS)	**80.09%**	**+4.74%**	**12.5 hrs**	105.8 M	0.61 M	**2.34 ms**
PascalVOC	Backbone-Only [13]	22.02%	+2.80%	9.6 hrs	436.1 M	0.58 M	2.11 ms
	Neck/Head-Only [6], [22], [23]	19.22%	+0.00%	6.0 hrs	240.0 M	0.36 M	2.48 ms
	Progressive [14]	21.74%	+2.52%	14.8 hrs	540.0 M	0.41 M	2.66 ms
	ELASTIC (OURS)	**30.83%**	**+11.61%**	**14.65 hrs**	642.8 M	0.57 M	**2.00 ms**

Table 1. Quantitative comparison of search strategies on the SVHN and PascalVOC subset. ELASTIC consistently achieves the highest mAP on both datasets and reduces GPU hours by 59\% on SVHN.

ELASTIC – TinyissimoYOLO and MCUNet

UNIVERSITY OF HOUSTON

Method	MACs	↓MACs	Params	VOC mAP	↑mAP
TY: 20-3-88 [24]	32M	90.7%	0.58M	53%	+30%
TY: 20-7-88 [24]	44M	87.2%	0.58M	47%	+24%
TY: 20-3-112 [24]	54M	84.3%	0.89M	56%	+33%
TY: 20-7-112 [24]	70M	79.6%	0.91M	53%	+30%
TY: 20-3-224 [24]	218M	36.4%	3.34M	23%	+0%
MCUNet [1]	168M	51.0%	1.2M	51.4%	+28.4%
MCUNetV2-M4 [25]	172M	49.9%	0.47M	64.6%	+41.6%
MCUNetV2-H7 [25]	343M	0%	0.67M	68.3%	+45.3%
ELASTIC (OURS)	86M	74.9%	1.36M	**72.3%**	**+49.3%**

Table 2. Comparison of ELASTIC with TinyissimoYOLO and MCUNETo n the full PascalVOC dataset, considering all object classes and counts. ELASTIC achieves a 20.9% mAP boost over MCUNET, 4.0% over MCUNetV2, and a 16.3% over TinyissimoYOLO's best performing model, while discovering a model with significantly fewer MACs, enabling faster inference.

ELASTIC – Search Space

UNIVERSITY OF HOUSTON

Dataset	Search Space	Mean mAP	↑mAP	Std. Dev.	Variance
SVHN	Joint Search [15]	44.61%	0%	1.39	1.93
	Neck/Head Search [6], [22], [23]	67.38%	+22.77%	**0.37**	**0.136**
	ELASTIC$^{(3)}$	70.42%	+25.81%	0.47	0.221
	ELASTIC$^{(5)}$	**70.68%**	**+26.07%**	0.45	0.199
PascalVOC	Joint Search [15]	4.99%	0%	0.04	1.40×10^{-3}
	Neck/Head Search [6], [22], [23]	27.89%	+22.90%	**0.01**	9.86×10^{-5}
	ELASTIC$^{(3)}$	28.28%	+23.29%	**0.01**	7.04×10^{-5}
	ELASTIC$^{(5)}$	**28.32%**	**+23.33%**	**0.01**	$\mathbf{6.60 \times 10^{-5}}$

Table 3. Mean mAP and variance across search spaces. Superscripts denote iterative search iterations. Higher mean mAP is better; lower variance indicates more consistent architectures.

TrashDet – Quantitative Comparison

UNIVERSITY OF HOUSTON

Method	Backbone	Neck	Head	Params	AR	mAP50
YOLOv5m [11]	CSPDarknet [27]	SPPF [8] + PANet [14]	Yolov3 [22]	21.2M	**22.3**	15.9
YOLOv8m [10]	CSPDarknet [27]	SPPF [8] + PANet [14]	Yolov8 [10]	25.9M	16.6	16.6
TrashDet-m (Ours)	**OFA ResNet**	**OFA PANet**	**OFA Yolov3**	**21.0M**	19.1	**18.6**
SWDet-m [30]	ADA [29]	EAFPN	Yolov3 [22]	33.85M	21.0	16.4
Deformable DETR [31]	ResNet-101 [9]	DETR Encoder	DETR Decoder [4]	40M	**30.3**	16.8
RTMDet [17]	RTMDet-l	PANet [14]	RTMDet	52.3M	19.4	16.9
AltiDet-m [12]	ADA + HRFE [28]	A-IFPN	Yolov3 [22]	85.3M	22.4	18.4
TrashDet-l (Ours)	**OFA ResNet**	**OFA PANet**	**OFA Yolov3**	**30.5M**	18.6	**19.5**

Table 4. Comparison of detectors on the TACO Dataset. TrashDet-l achieves the highest mAP50 of 19.5 with 30.5M parameters, outperforming the strongest baseline AltiDet-m while using roughly one third of the parameters

TrashDet – Scalable Detectors

UNIVERSITY OF HOUSTON

Method	Resolution	Params	AR	mAP50	Latency (ms)	FPS
TrashDet-n	640	1.2M	21.2	11.4	2.21	452.79
TrashDet-s	640	7.9M	16.9	15.8	3.83	261.06
TrashDet-m	640	21.0M	19.1	18.6	4.39	227.70
TrashDet-l	640	30.5M	18.6	19.5	5.07	197.08

Table 5. Performance of TrashDet variants on TACO. Model capacity ranges from TrashDet-n with 1.2M parameters to TrashDet-l with 30.5 parameters. Latency measurements were performed on an Ubuntu 22.04.4 system on an Intel Core i9-13900KF CPU with an NVIDIA GeForce RTX 4090 GPU.

TrashDet – On TrashNet MAX78002

UNIVERSITY OF HOUSTON

Model	Resolution	Dataset	Params	Energy (μJ)	Latency (ms)	Power (mW)	FPS	mAP50
ai87-fpndetector [1]	256×320	TrashNet	2.18M	62001	122.6	445.76	8.16	83.1
TrashDet - MBNet	224×224	TrashNet	1.32M	17581	51.1	285.02	19.57	**93.3**
TrashDet - ResNet	224×224	TrashNet	**1.08M**	**7525**	**26.7**	**210.5**	**37.45**	84.6

Table 6. Energy, latency, and power comparison on the MAX78002 for detectors discovered within the TrashDet search space. We explore ResNet- and MobileNet-style (MBNet) backbones and obtain two deployment-ready models, **TrashDet–MBNet** and **TrashDet–ResNet**, evaluated on the TrashNet dataset for detection. Compared to ai87-fpndetector, TrashDet **reduces energy by up to 54,476uJ**, **latency by up to 95.9ms,** and **power by up to 235.3mW** while maintaining superior accuracy.