TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection

WASTEVISION: INTERNATIONAL WORKSHOP ON SMART WASTE MONITORING

Tony Tran[1], Bin Hu[2]

thtran37@cougarnet.uh.edu, bhu11@central.uh.edu

Cullen College of Engineering, Research Computing[1]

Department of Engineering Technology[2]

University of Houston, Houston, USA

NAIL
Networked Autonomous Intelligent Learning

WACV 2026
TUCSON, AZ
3/6 – 3/10

UNIVERSITY OF HOUSTON

Computer Vision for Waste Monitoring

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 Edge Devices

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

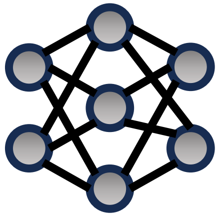
Waste Detection on UAVV

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?

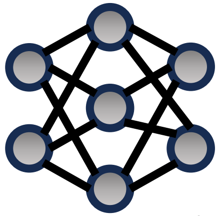
Existing SOTA Waste Detectors



YOLOv8m

16.6 mAP50 @21M

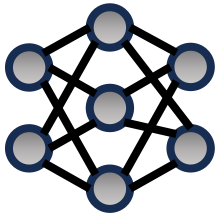
Ultralytics YOLOv8 [Jocher et al., GitHub 2023]



RTMDet-l

16.9 mAP50 @52M

RTMDet: An Empirical Study of Designing Real-Time Object Detectors [Lyu et al., Arxiv 2022].



AltiDet-m

18.4 mAP50 @85M

Altitude-informed fusion pyramid network for multi-scale waste detection in unmanned aerial vehicle images [Liew et al., Engineering Applications of Artificial Intelligence 2025].

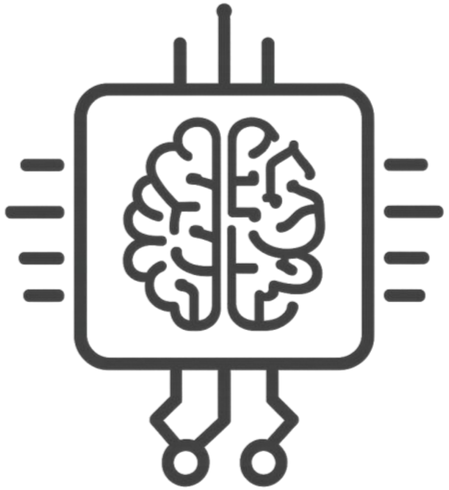
Flash Overflow

Out of Memory

Resource Constrained Devices

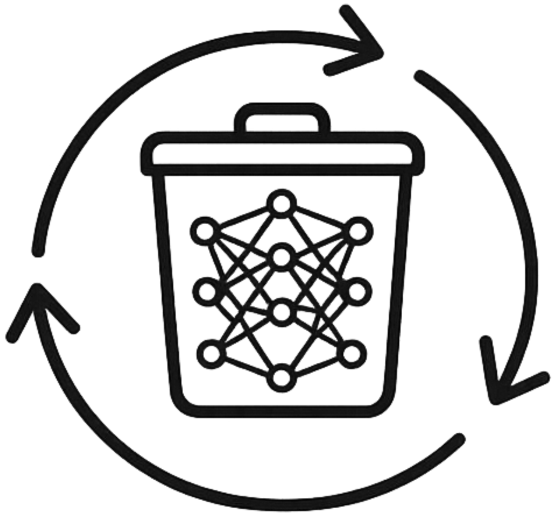
These Limitations Motivate the Need for More Efficient Detection Methods that Preserve Competitive Accuracy while Reducing Computational Costs

Hardware-Aware Iterative NAS Framework



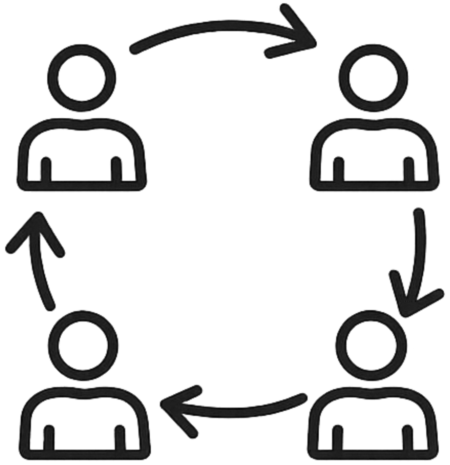
Hardware-Constrained Supernet

A Flexible, Overparameterized Network that Defines a Search Space of Many Subnetworks, Filtered by Hardware Constraints



Iterative Evolutionary Search

An Iterative Search Method that Efficiently Explores All Waste Detection Modules (i.e. Backbone, Neck, Head) Iteratively via Evolutionary Algorithm



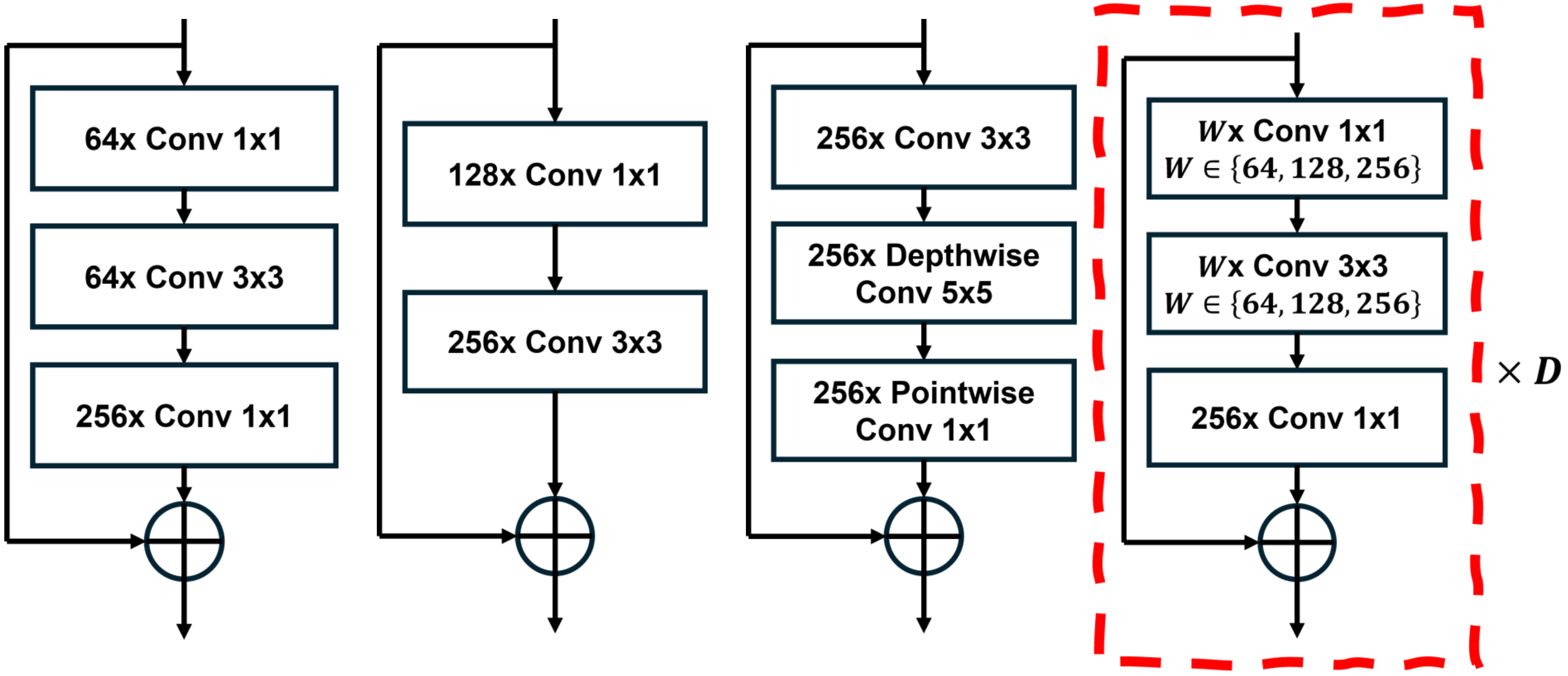
Population Passthrough

A Unique Search Memory Mechanism to Stabilize Search Process and Balance *Exploration* and *Exploitation*

*Our Method Yields a Family of Highly Efficient, Deployment-Ready Waste Detectors called **TrashDets***

Defining the Supernet Search Space

We Design a Supernet based on the OFA-Res Block Applied to All Detection Network Components (i.e. Backbone, Neck, Head)

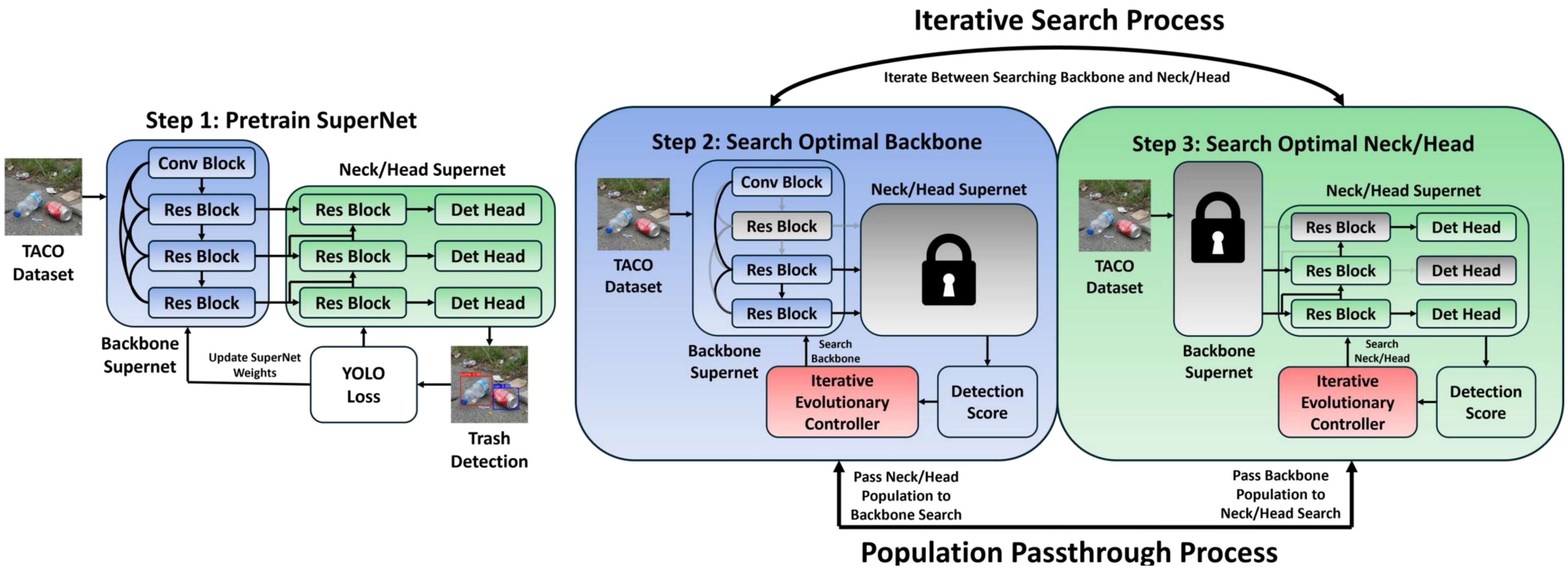


Bottleneck Block	DarkNet Block	RTMDet Block	OFA-Res Block
Used in ResNet	Used in CSPDarkNets	Used in RTMDet	Used in OFAResNet

Search Dimensions:

- **Depth:** Number of Active Blocks per Stage (2 to 8)
- **Width:** Channel Scaling Multipliers (e.g. 0.8x, 1.0x, 1.25x)
- **Expansion Ratio:** Controls Intermediate Channels within Blocks

Iterative Evolutionary Search Algorithm



Step 1

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

Step 2

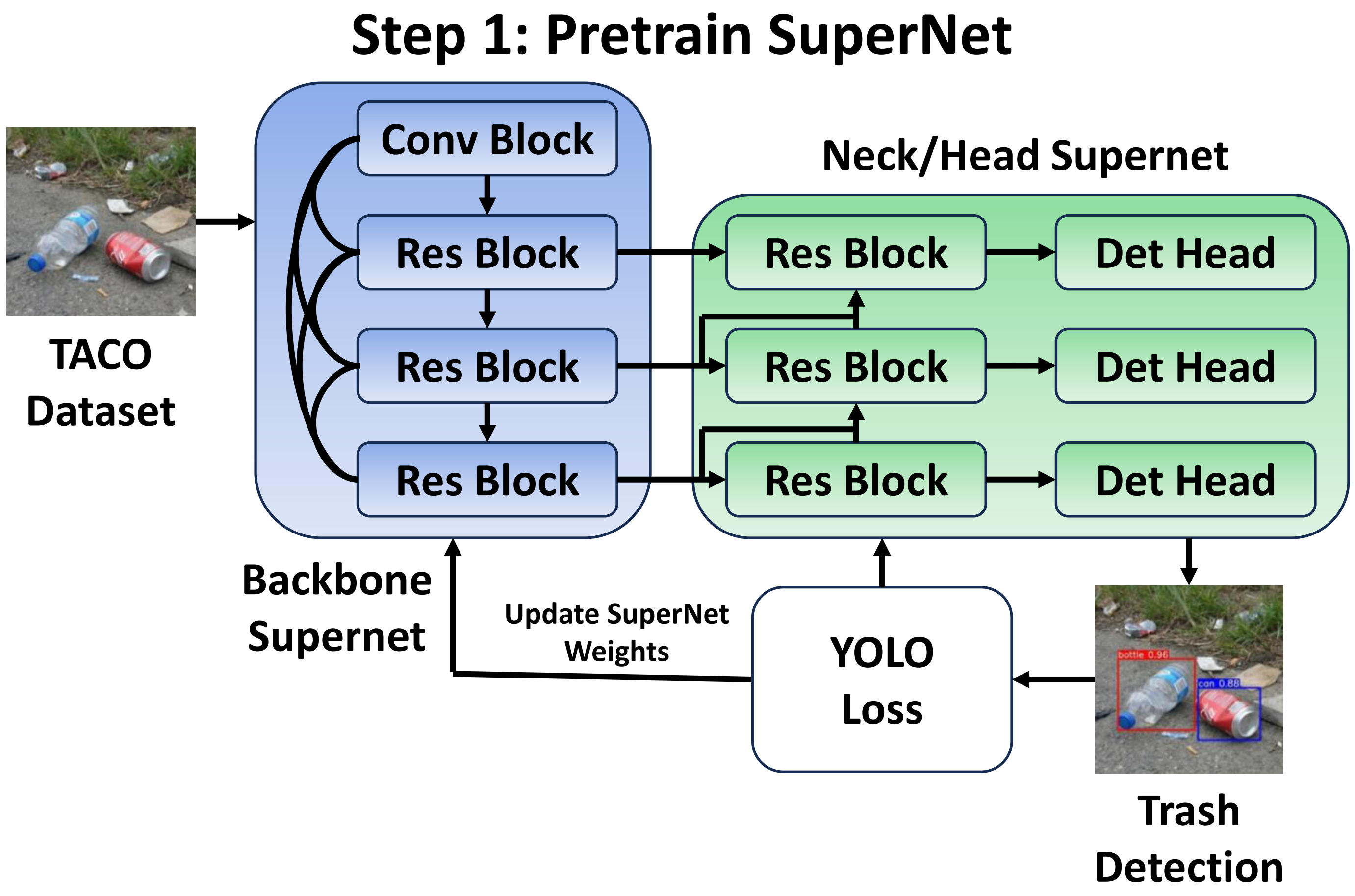
Freeze Neck/Head Module and Search Optimal Backbone Arch Subject to Hardware Constraints

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



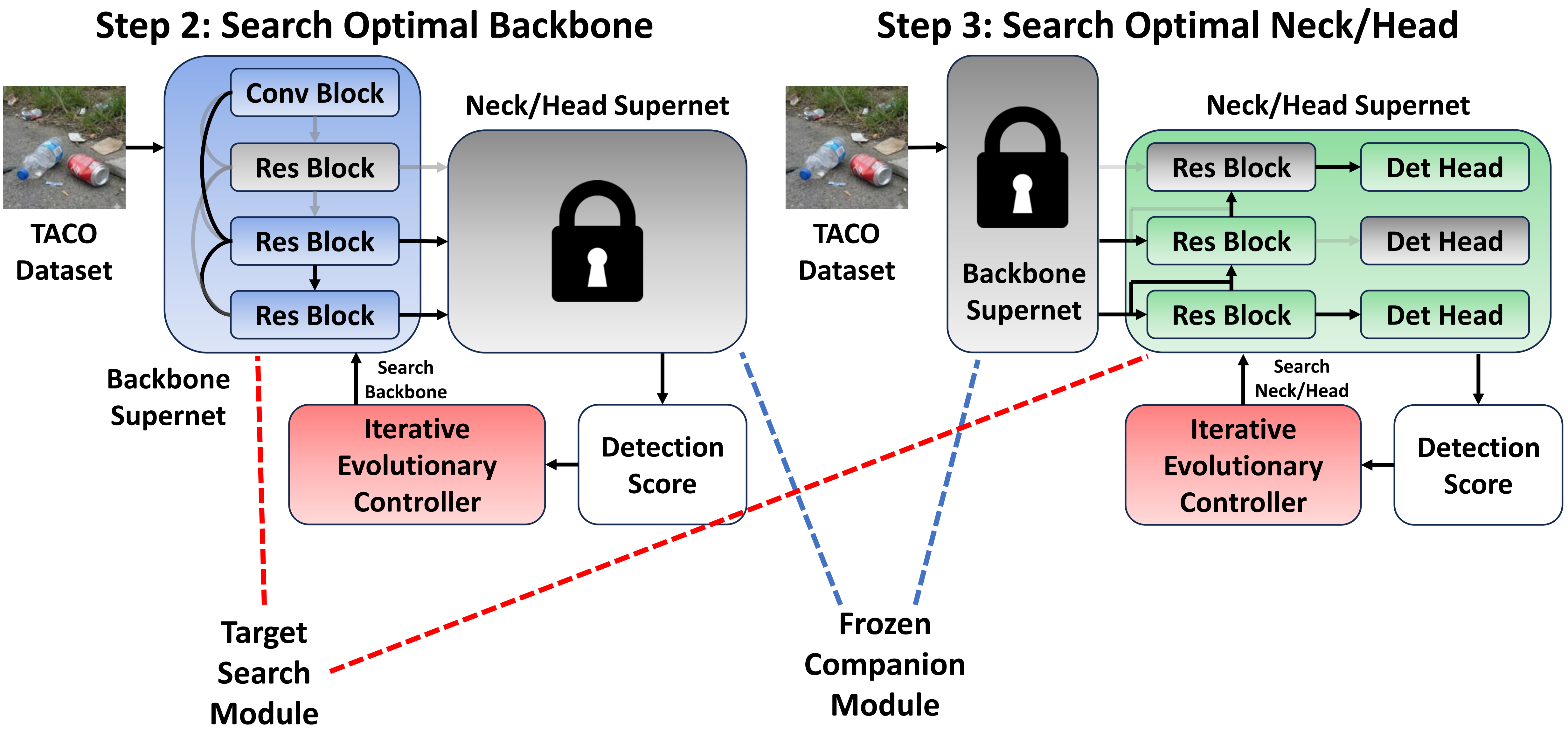
Objective 1:

Minimize Loss Function with Respect to the Supernet Detection Output

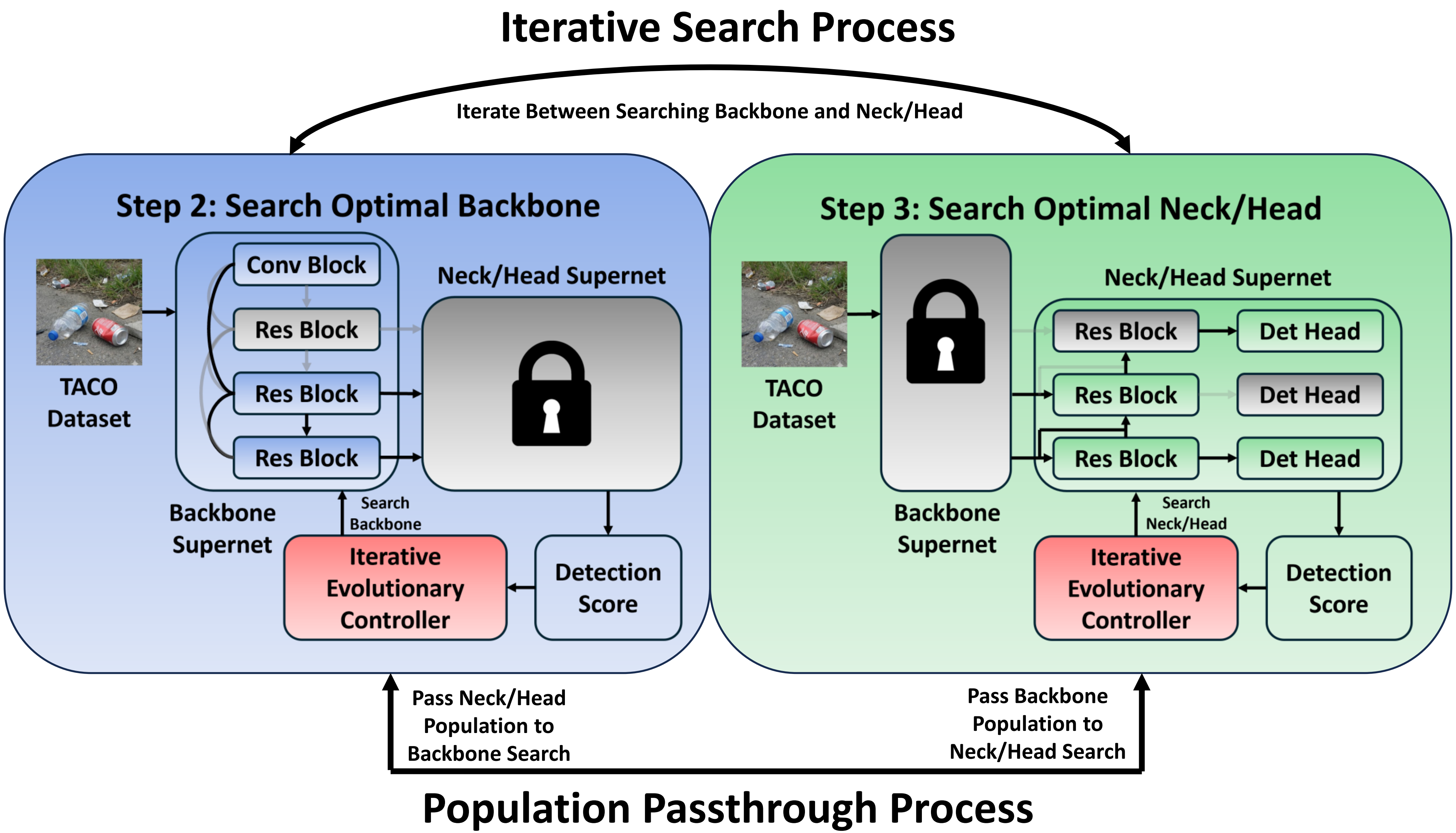
Objective 2:

Minimize Loss Function with Respect to Randomly Sampled Subnets

Search One Module, Freeze Other Module



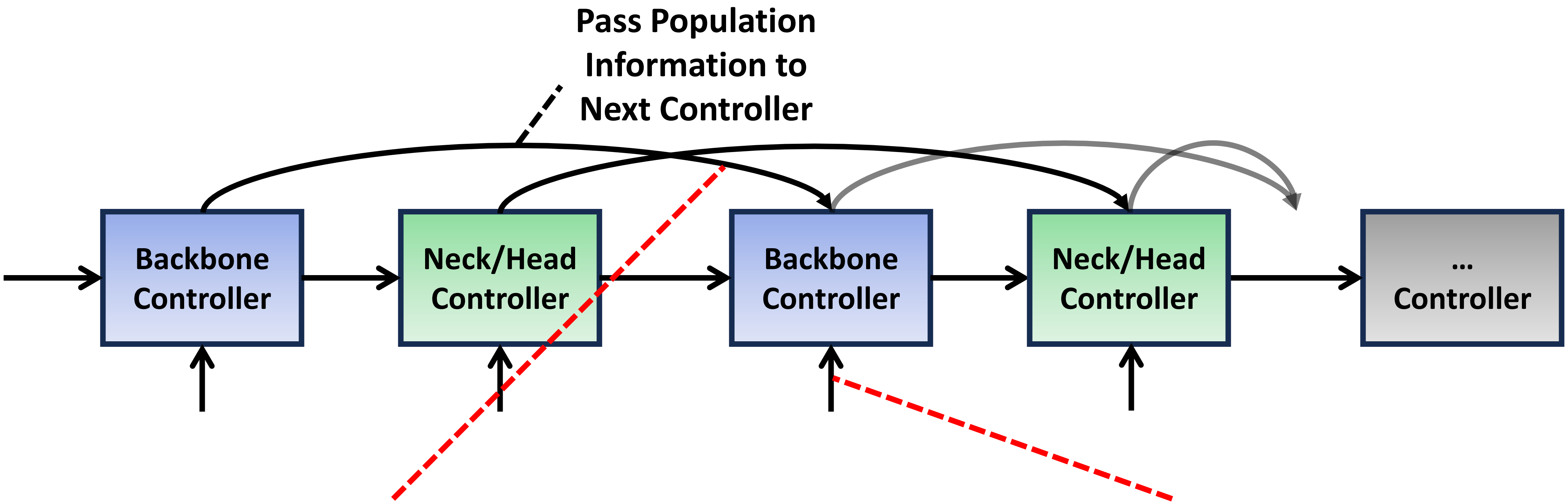
Iterative Search Process



Population Passthrough Process

UNIVERSITY OF HOUSTON

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



Elite Passthrough (Exploitation):

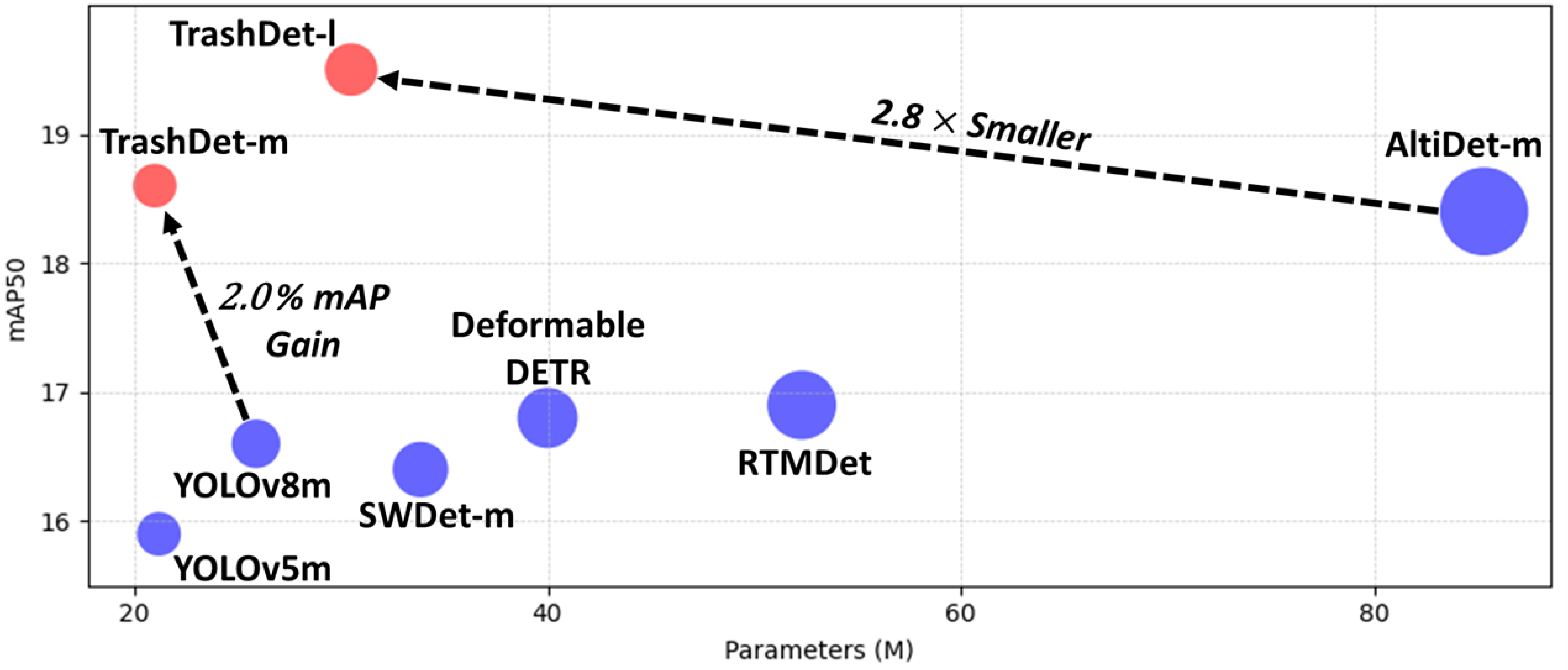
A Fraction of Best Performing Archs Passed to Seed Next Search

Diversity Augmentation (Exploration):

The Rest of New Population is Randomly Seeded from Supernet

TrashDet for Efficient Waste Detection

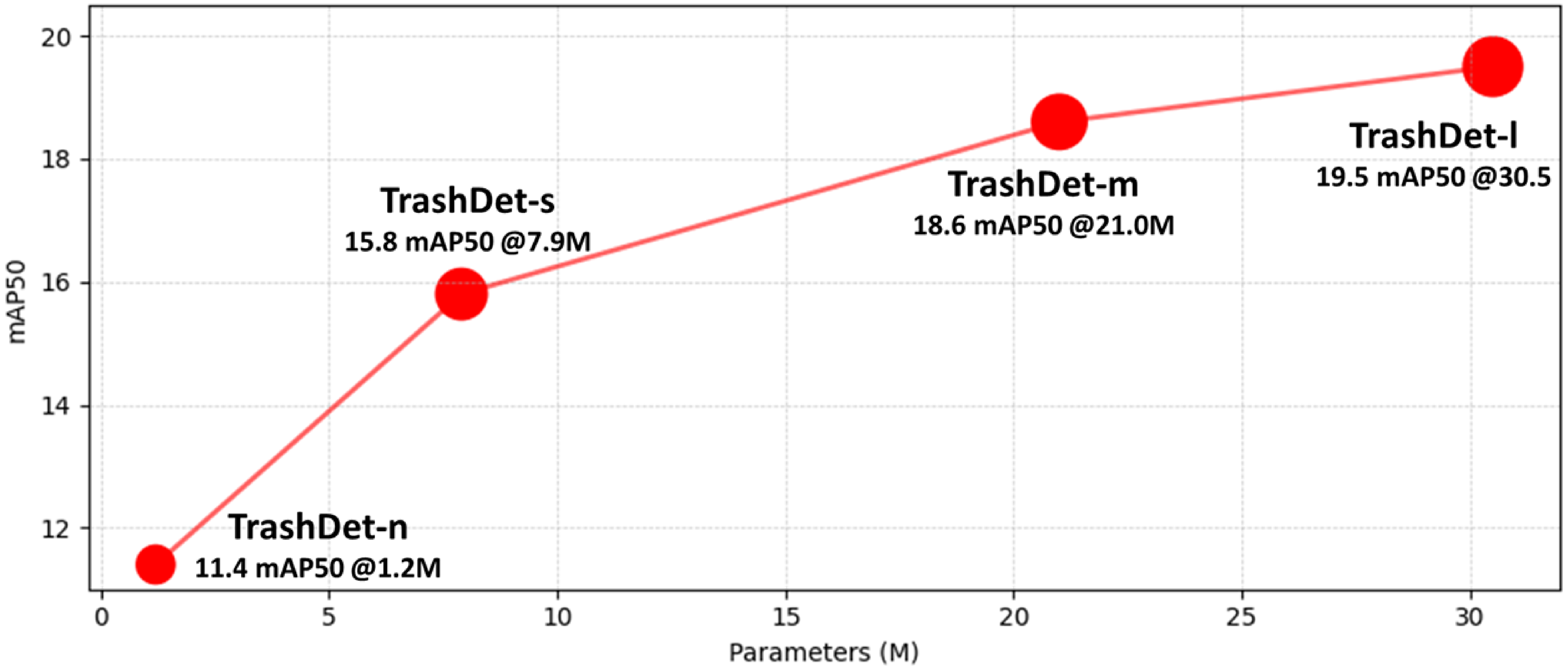
UNIVERSITY OF HOUSTON



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

Family of Scalable Waste Detectors

UNIVERSITY OF HOUSTON

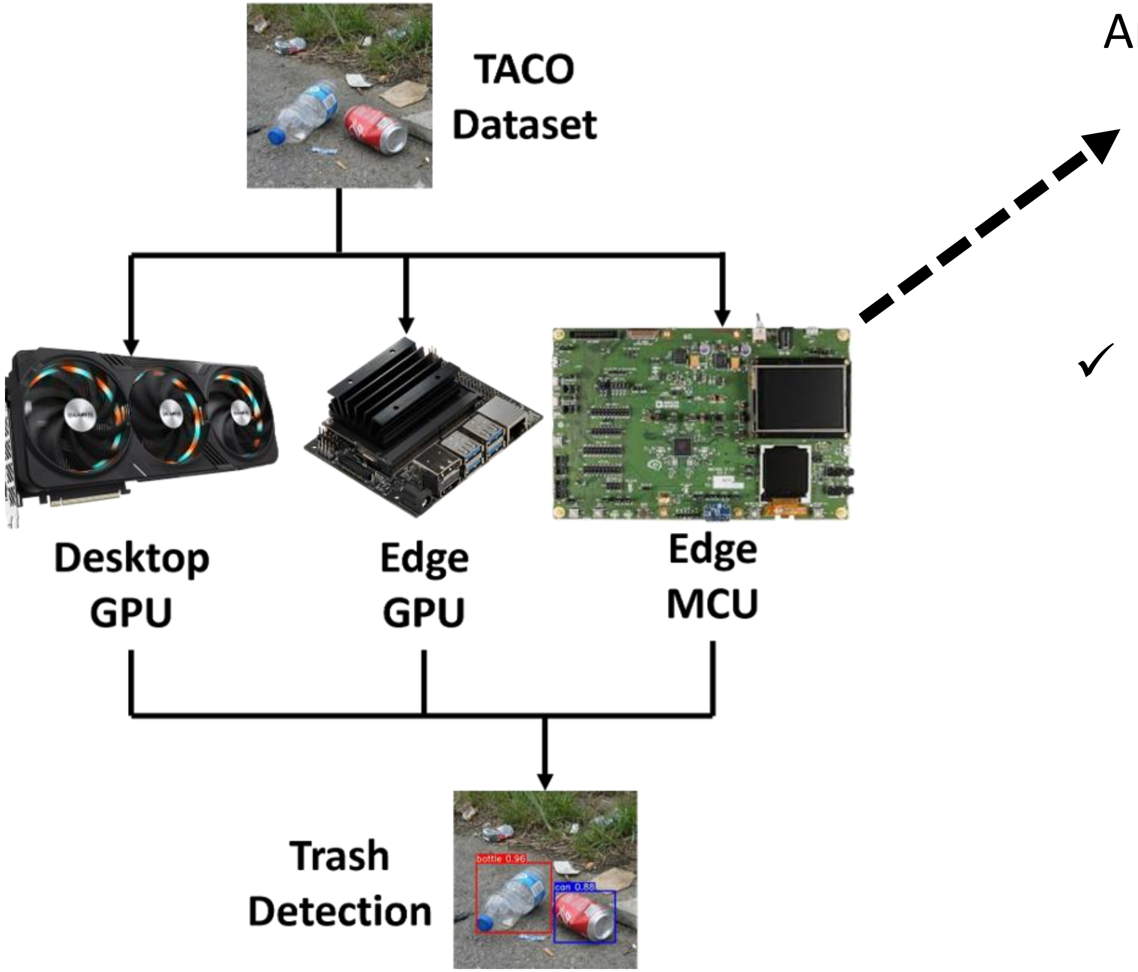


Our NAS Framework can Target a Range of Parameter Budgets,
Providing Practitioners with Scalable Options for Diverse Deployment Targets

Deployment on Resource Constrained MCUs

UNIVERSITY OF HOUSTON

Step 4: Deploy Tailored Models



<u>Target Device: MAX78002</u>

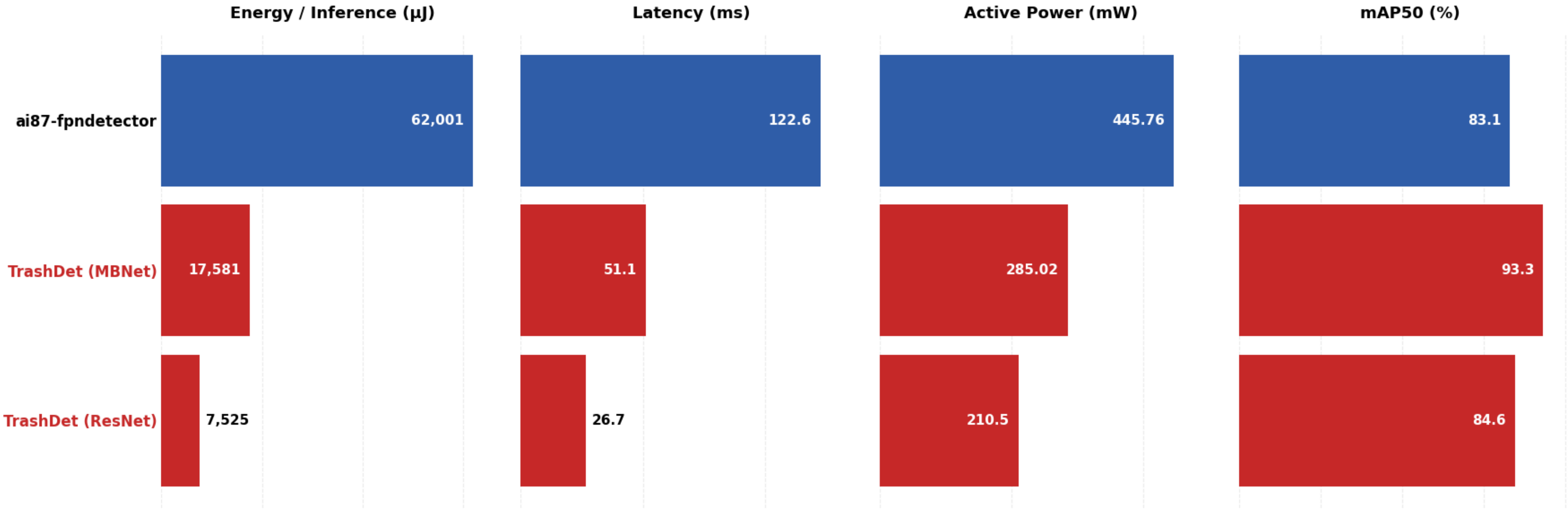
Analog Devices' Low-Power AI Microcontroller with Low-Power CNN Accelerator

<u>Hardware Constraints:</u>

- ✓ 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

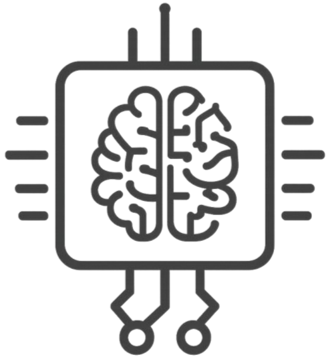
TrashDet Efficiency Comparison on MCU

Comparison Measured on TrashNet Dataset for Object Detection on MAX78002

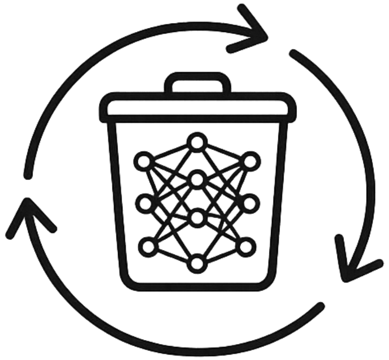


87.9%	**78.2%**	**52.8%**	**10.2%**
Energy Reduction Compared Against TrashDet (ResNet)	Latency Reduction Compared Against TrashDet (ResNet)	Power Reduction Compared Against TrashDet (ResNet)	mAP50 Increase Compared Against TrashDet (MBNet)

Conclusion



Contribution 1: We Define a Unified OFA Supernet for Waste Detection, Covering All Components of the Network (i.e. Backbone, Neck, Head), Filtered by Hardware Constraints.



Contribution 2: We Develop a Novel Iterative Evolutionary Search with Population Passthrough Mechanism, Enabling Efficient Modular Optimization for Waste Detection

Contribution 3: We Discover TrashDets, a Family of Models, Outperforming State-of-the-Art Baselines on TACO Benchmark and Demonstrating Significant Reductions in Energy, Latency, and Power on Resource Constrained Microcontrollers.

WACV 2026
TUCSON, AZ 3/6 - 3/10

UNIVERSITY OF HOUSTON

Thank You

Scan Here to
Download Paper

TrashDet: Iterative Neural Architecture Search for Efficient Waste Detection

Trash Detection Demo
on ModalAI Sentinel

UNIVERSITY OF HOUSTON

NAIL
Networked Autonomous Intelligent Learning

Backup Slides – Quantitative Comparison

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 1. 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

Backup Slides – Scalable Detectors

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 2. 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.

Backup Slides – On TrashNet MAX78002

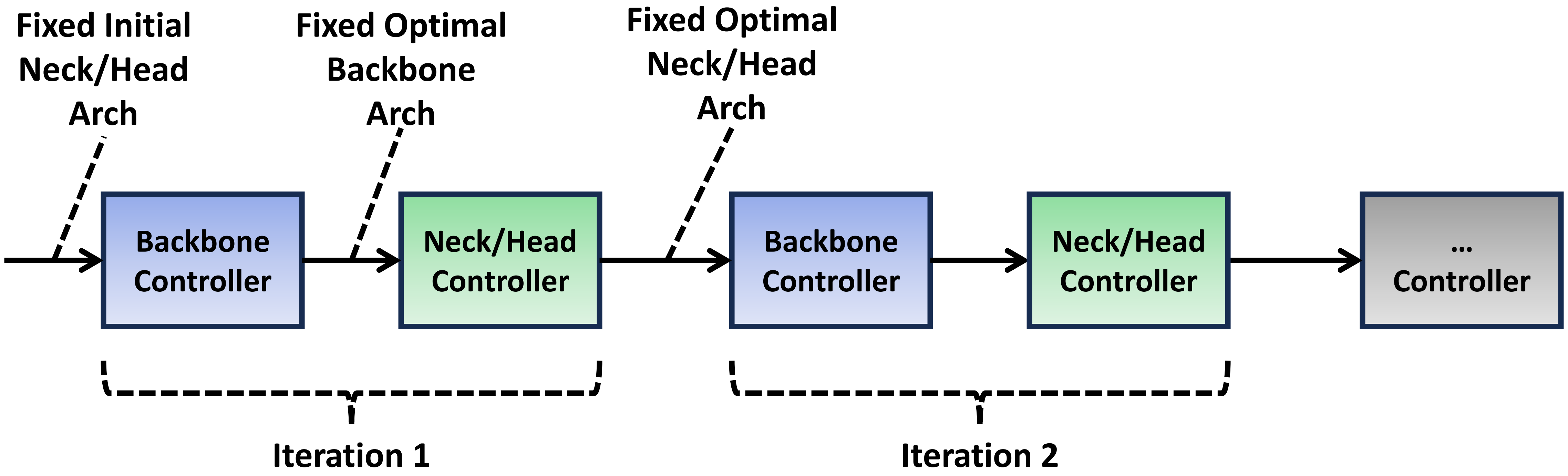
Model	Resolution	Dataset	Params	Energy (μJ)	Latency (ms)	Power (mW)	FPS	mAP50
ai87-fpndetector [1]	256×320	TrashNet	2.18M	62001	122.6	445.76	8.16	83.1
TrashDet - MBNet	224×224	TrashNet	1.32M	17581	51.1	285.02	19.57	**93.3**
TrashDet - ResNet	224×224	TrashNet	**1.08M**	**7525**	**26.7**	**210.5**	**37.45**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

Table 3. 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.

Backup Slides - Population Passthrough 1/2

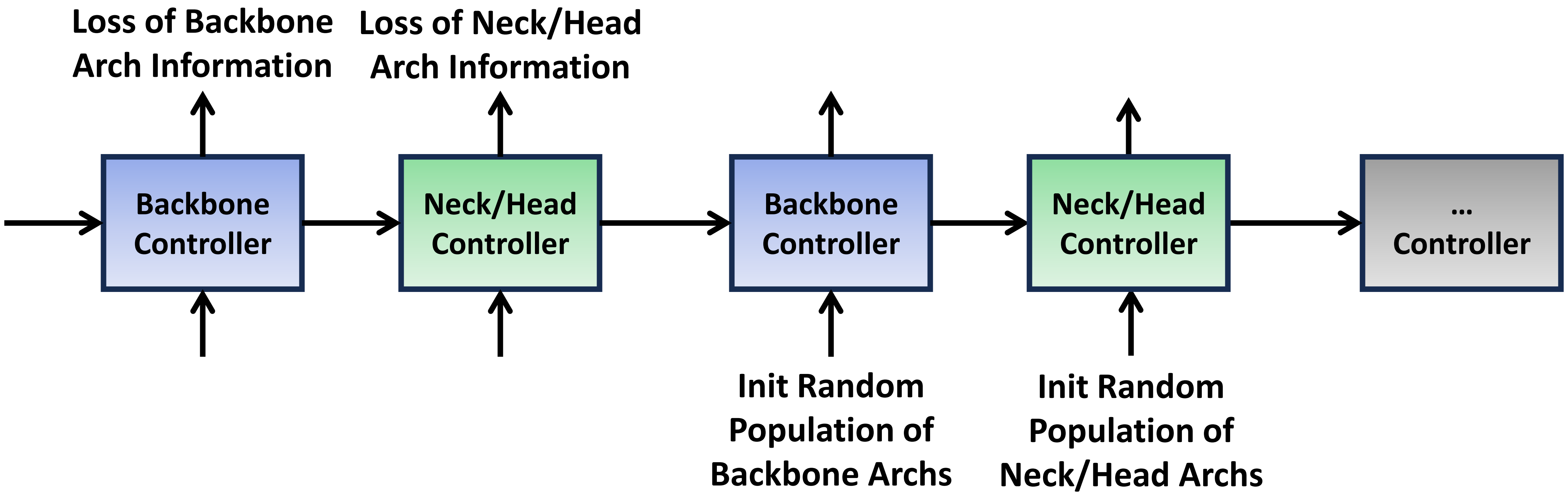
UNIVERSITY OF HOUSTON

Iterative Evolutionary Controller as a Sequence of Backbone and Neck/Head Controllers



Backup Slides - Population Passthrough 2/2

Iterative Evolutionary Controller as a Sequence of Backbone and Neck/Head Controllers



How can we Utilize Information from Previous Iterations to Guide Next Iterations?