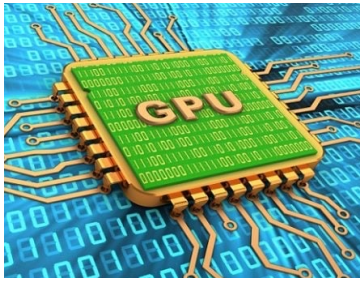
SoK: Efficiency Robustness of Dynamic Deep Learning Systems

Ravishka Rathnasuriya, Tingxi Li , Zexin Xu, Zihe Song, Mirazul Haque, Simin Chen, Wei Yang

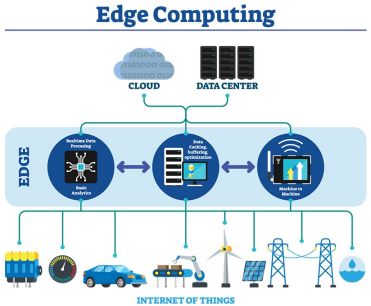
The University of Texas at Dallas

Efficiency is Important



Deployment in Computation Resources

Operate in Real time applications



Deploy in Edge Devices

Less Energy Consumption can be Environment-friendly

Motivation

Dynamic deep learning systems (DDLSs) optimize computation by adapting to input complexity.

Problem Statement

*The adaptivity in DDLSs introduces a new class of **security vulnerabilities**: adversaries can manipulate inputs to increase inference-time computational cost*

Formulation of Efficiency Attacks

The adversary seeks an optimal perturbation δ that maximizes computational cost

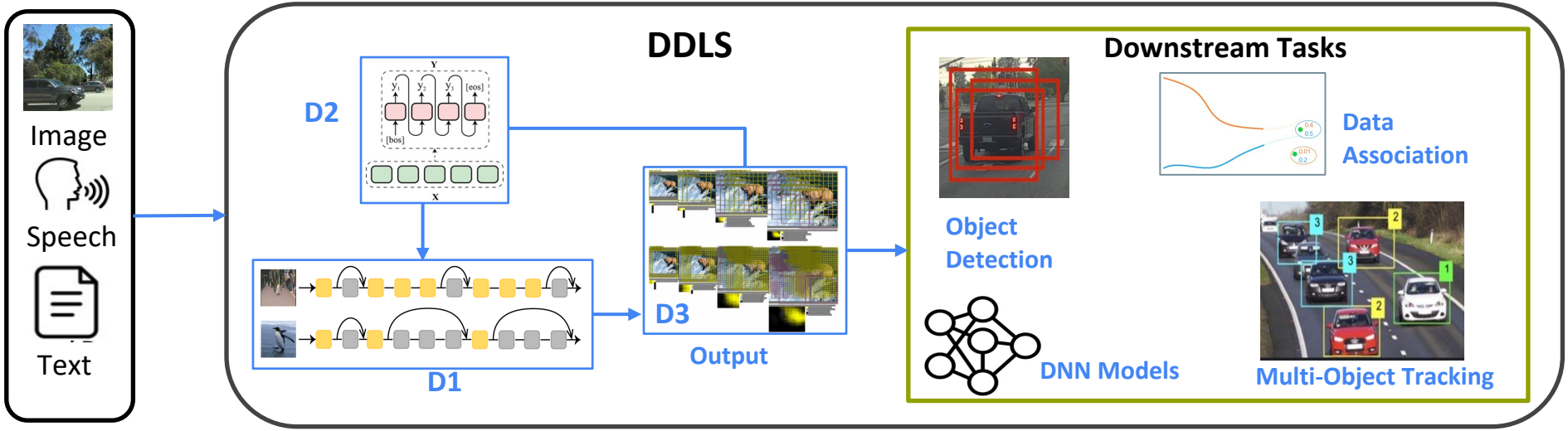
$$\text{maximize } C(N, H, \mathrm{x} + \delta)$$
$$\text{subject to } ||\delta|| \leq \epsilon, \mathrm{x} + \delta \in \mathrm{X}$$

A DDLS ***N* (·)** which takes an input **x** and hardware accelerator **H**

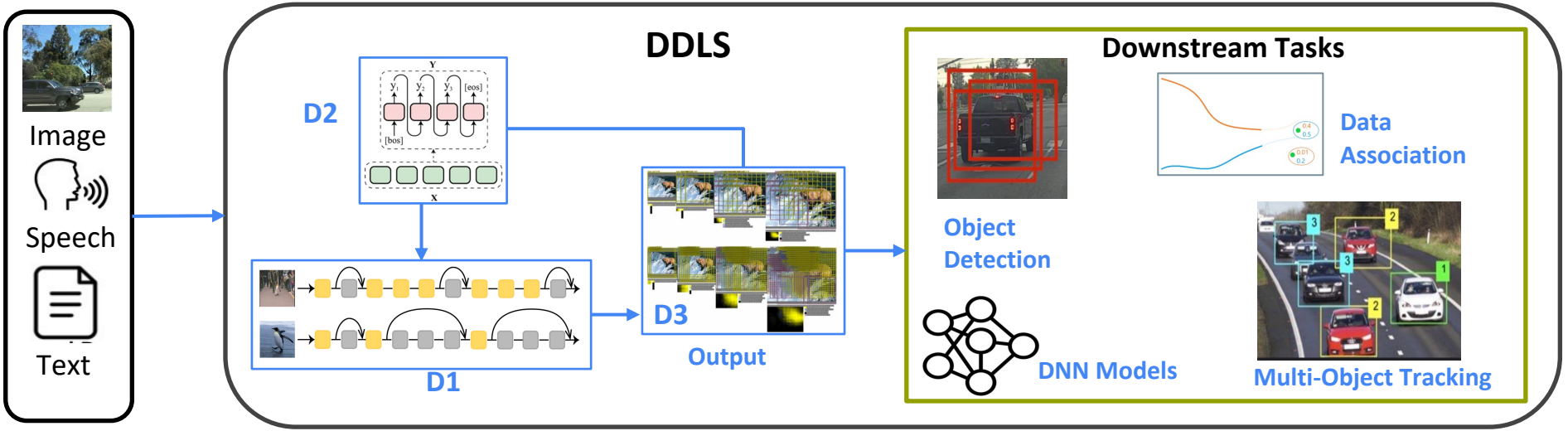
The computational costs **C** could be in terms of energy consumption, response latency, or computational floating-point operations

The objective is to increase computational cost without violating input constraints ($||\delta|| \leq \epsilon$), exposing a fundamental vulnerability in efficiency-optimized DDLSs

Dynamic Behaviors of DDLSs

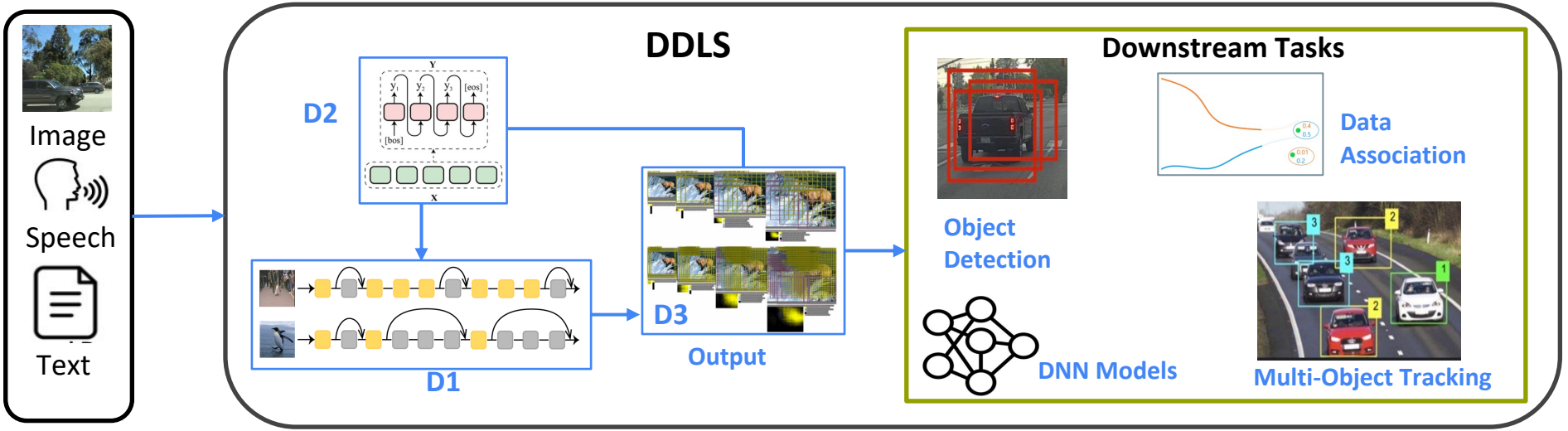


Dynamic Behaviors of DDLSs



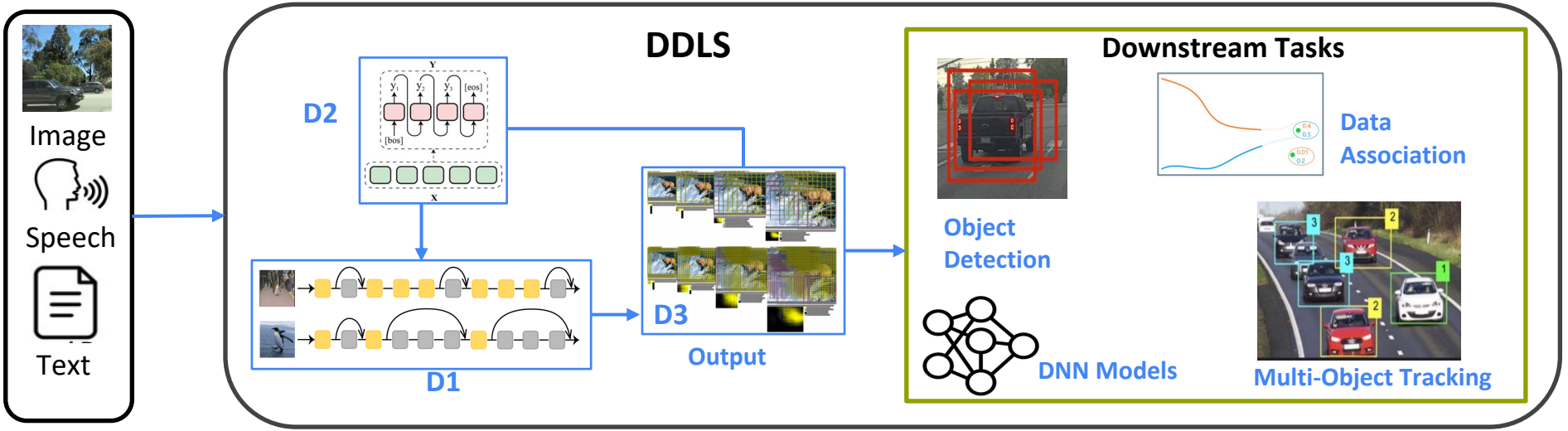
D1: Dynamic computation per inference - Varying internal execution paths during a single forward pass using mechanisms like early exits, conditional skipping, adaptive step sizing, and activation sparsity

Dynamic Behaviors of DDLSs



D2: Dynamic inference iterations -Varying how many steps inference runs in autoregressive and iterative generation models

Dynamic Behaviors of DDLSs

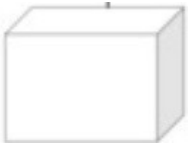


D3: Dynamic output production- Number of outputs varies at inference and influences downstream workload in systems like object detection

Threat Model

Adversary's Goal: (1) Maximize computational cost (e.g., latency, FLOPs), (2) Keep perturbations imperceptible to users, (3) Ensure adversarial inputs remain realistic and valid in deployment

Adversary's Knowledge and Capabilities: The adversary may act at inference or training time, with varying levels of model access



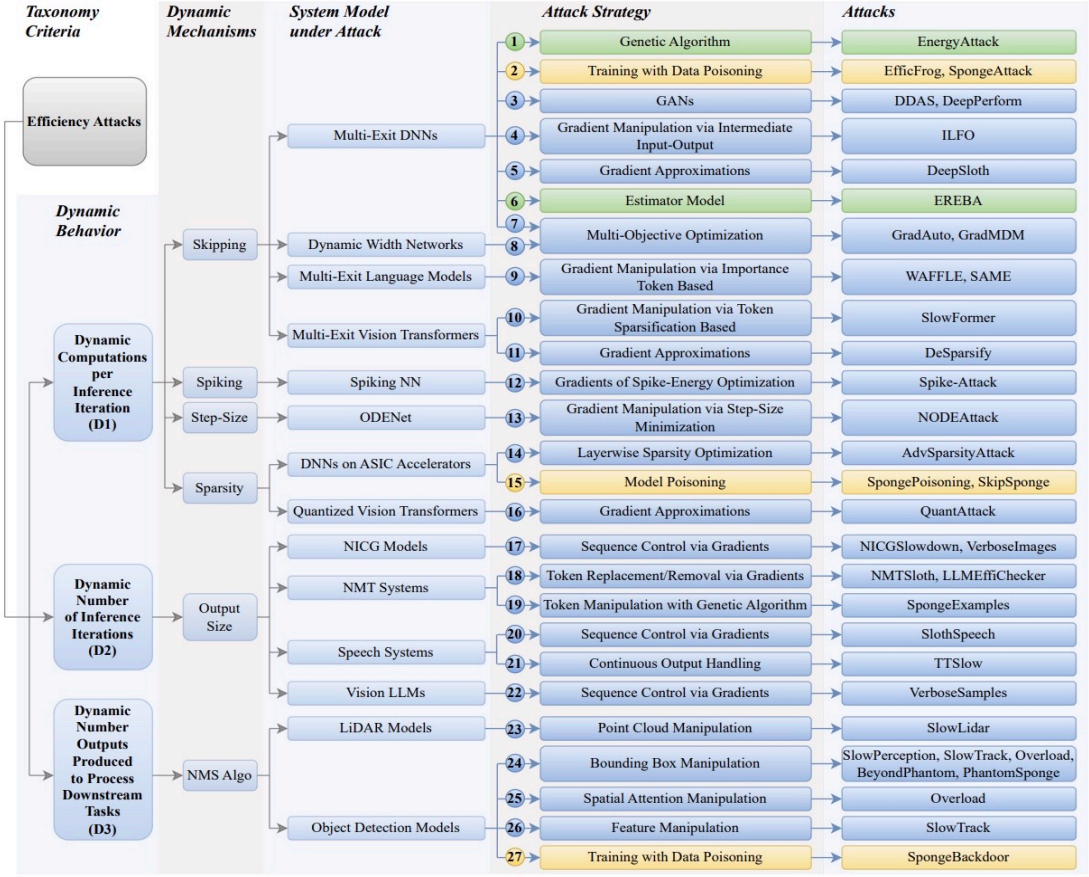
White-box evasion attacks

Black-box evasion attacks

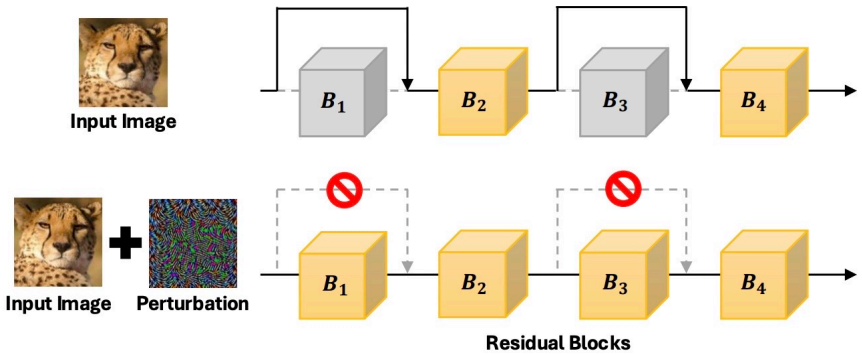
Data Poisoning

Model Poisoning

Taxonomy


Taxonomy Criteria
Dynamic Mechanisms
System Model under Attack
Attack Strategy
Attacks
Efficiency Attacks
Dynamic Behavior
Dynamic Computations per Inference Iteration (D1)
Dynamic Number of Inference Iterations (D2)
Dynamic Number Outputs Produced to Process Downstream Tasks (D3)
Skipping
Spiking
Step-Size
Sparsity
Output Size
NMS Algo
Multi-Exit DNNs
Dynamic Width Networks
Multi-Exit Language Models
Multi-Exit Vision Transformers
Spiking NN
ODENet
DNNs on ASIC Accelerators
Quantized Vision Transformers
NICG Models
NMT Systems
Speech Systems
Vision LLMs
LiDAR Models
Object Detection Models
1 Genetic Algorithm
EnergyAttack
2 Training with Data Poisoning
EfficFrog, SpongeAttack
3 GANs
DDAS, DeepPerform
4 Gradient Manipulation via Intermediate Input-Output
ILFO
5 Gradient Approximations
DeepSloth
6 Estimator Model
EREBA
7 Multi-Objective Optimization
8 GradAuto, GradMDM
9 Gradient Manipulation via Importance Token Based
WAFFLE, SAME
10 Gradient Manipulation via Token Sparsification Based
SlowFormer
11 Gradient Approximations
DeSparsify
12 Gradients of Spike-Energy Optimization
Spike-Attack
13 Gradient Manipulation via Step-Size Minimization
NODEAttack
14 Layerwise Sparsity Optimization
AdvSparsityAttack
15 Model Poisoning
SpongePoisoning, SkipSponge
16 Gradient Approximations
QuantAttack
17 Sequence Control via Gradients
NICGSlowdown, VerboseImages
18 Token Replacement/Removal via Gradients
NMTSloth, LLMEffiChecker
19 Token Manipulation with Genetic Algorithm
SpongeExamples
20 Sequence Control via Gradients
SlothSpeech
21 Continuous Output Handling
TTSlow
22 Sequence Control via Gradients
VerboseSamples
23 Point Cloud Manipulation
SlowLidar
24 Bounding Box Manipulation
SlowPerception, SlowTrack, Overload, BeyondPhantom, PhantomSponge
25 Spatial Attention Manipulation
Overload
26 Feature Manipulation
SlowTrack
27 Training with Data Poisoning
SpongeBackdoor

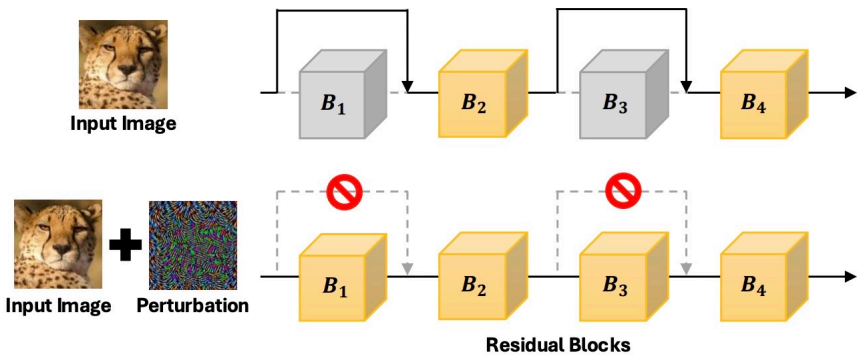
Attacks Overview on D1 - Skipping



System Models Under Attack

- Multi-Exit DNNs
- Dynamic Width Networks
- Multi-Exit Language Models
- Multi-Exit Vision Transformers

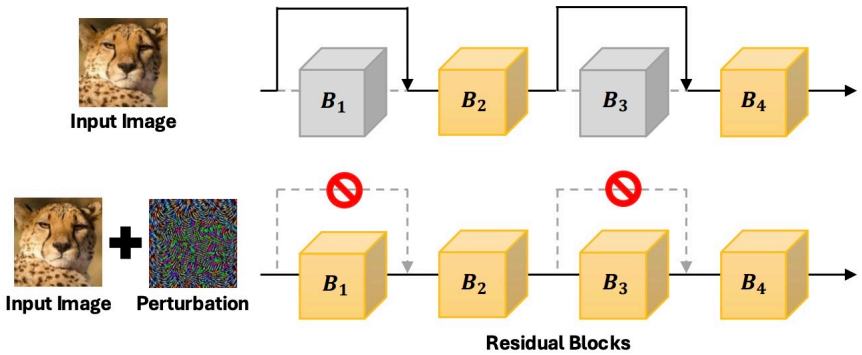
Attacks Overview on D1 - Skipping



White-Box Attack Strategies

- Gradient Manipulation via Intermediate Input-Output
- Multi-Objective Optimization
- Gradient Approximations
- Generative Adversarial Networks
- Gradient Manipulation via Importance Token Based

Attacks Overview on D1 - Skipping



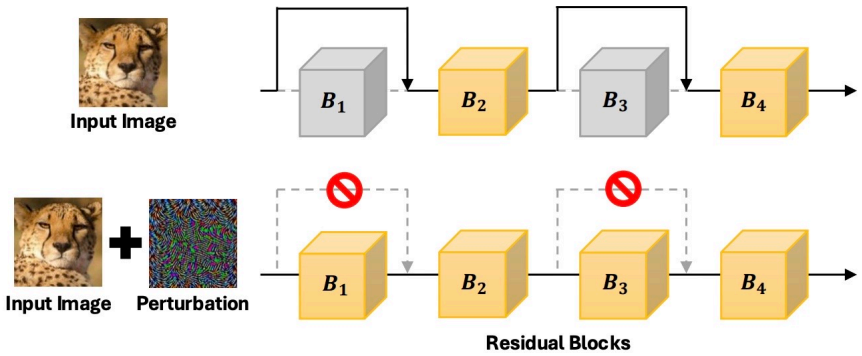
Black-Box Attack Strategies

- Gradient Manipulation via Intermediate Input-Output

Poisoning attacks

- Training with Data Poisoning

Attacks Overview on D1 - Skipping



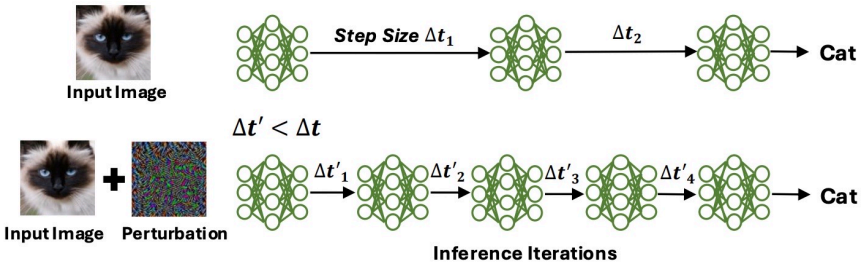
Real-World Impact

- Galaxy S9+, the number of successful inferences drops from approximately 10000 to between 3576
- IoT deployments resulting in a 1.5x to 5x delay

Implications

- Balance computations: Jointly optimize gate activation & inputs.
- New architectures: Extend attacks to MoE/MoD.
- Portability: Develop hardware-agnostic black-box models.

Attacks Overview on D1: Step-Size



System Models Under Attack

- ODENet

White-Box Attack Strategies

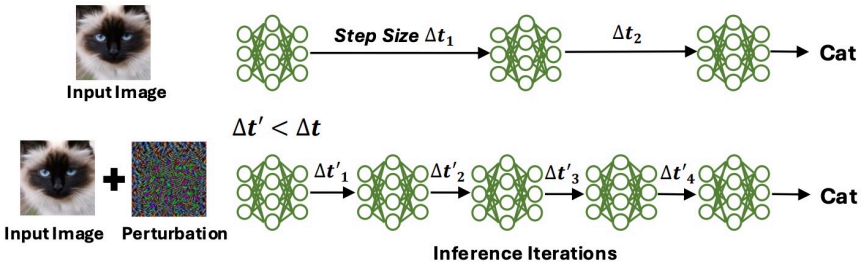
- Gradient Manipulation via Step-Size Minimization

Transferability

- Transferability on cross-solver attacks and cross-architecture attacks

No Blackbox or Poisoning Attacks

Attacks Overview on D1: Step-Size



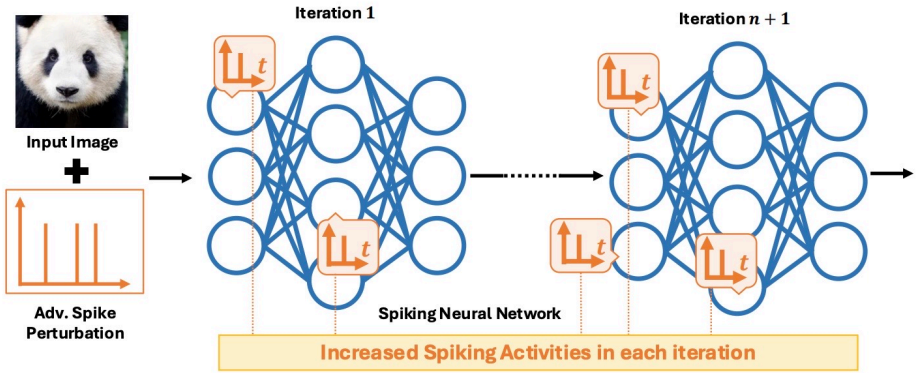
Real-World Impact

- In DNN compiler vulnerabilities, step-size manipulation cuts inference efficiency by ~50%

Implications

- Balance step size vs. energy efficiency in ODE solvers
- Develop black-box attacks across diverse ODE solvers

Attacks Overview on D1- Spiking



System Models Under Attack

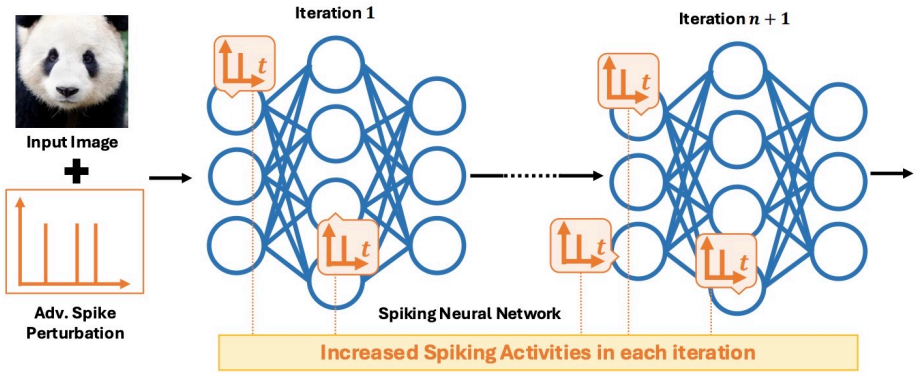
- Spiking Neural Networks

White-Box Attack Strategies

- Gradients of Spike-Energy Optimization

No Blackbox or Poisoning Attacks

Attacks Overview on D1- Spiking



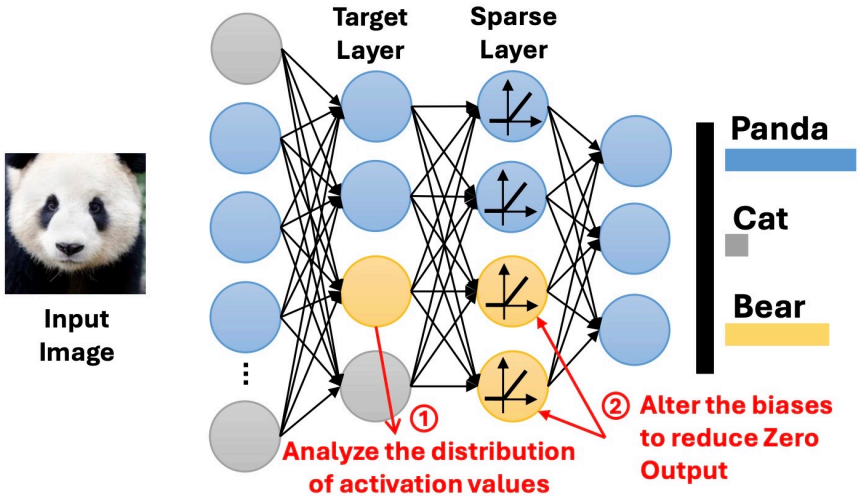
Real-World Impact

- Power budget collapse in neuromorphic/low-power systems like cameras, wearables, prosthetics
- Memory overload can fallback to cloud inference in edge devices.

Implications

- Extend to non-image data such as text, video, or temporal signals
- Develop black-box efficiency attacks for SNNs

Attacks Overview on D1: Sparsity



System Models Under Attack

- DNNs on ASIC Accelerators
- Quantized Vision Transformers

Blackbox attacks

- Reduce sparsity via input–output analysis & surrogate gradient estimation

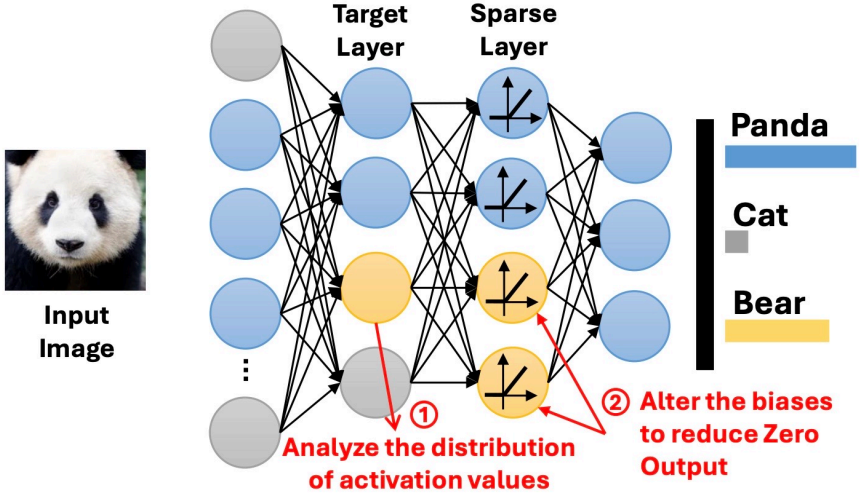
White-Box Attack Strategies

- Layerwise Sparsity Optimization
- Gradient Approximations

Poisoning attacks

- Model Poisoning

Attacks Overview on D1: Sparsity



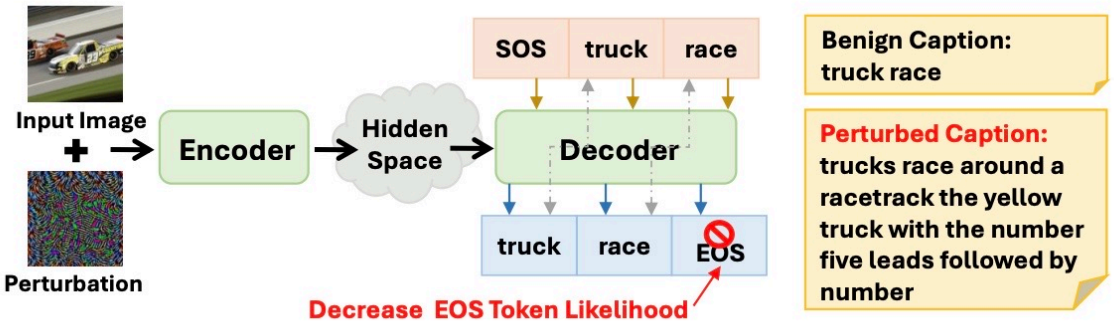
Real-World Impact

- Efficiency attacks affect safety-critical, IoT, and MLaaS systems
- Batch sensitivity when small batches hit hardest increasing 12% memory for 2 images vs. increasing 1.8% for 16 batch size

Implications

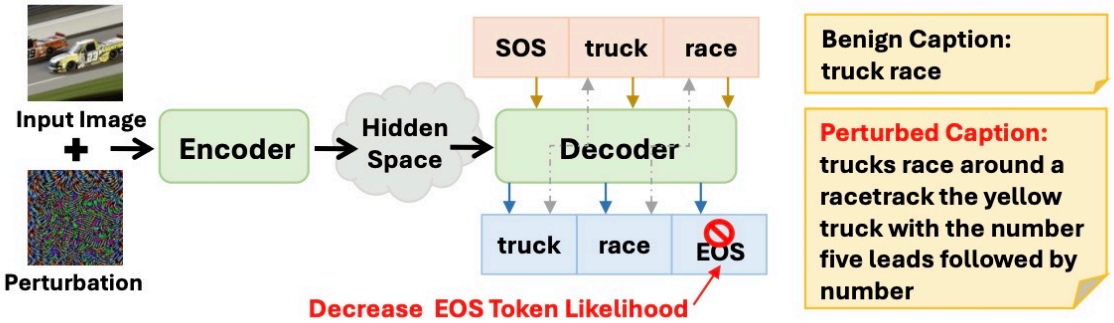
- Cost-effective strategies beyond sparse gradient reliance
- Balance accuracy vs. efficiency in black-box attacks
- Study impact on large-scale models with billions of parameters

Attack Overview on D2



Adversarial perturbations decrease the likelihood of the EOS token, forcing the decoder to generate excessively long captions, increasing computational cost

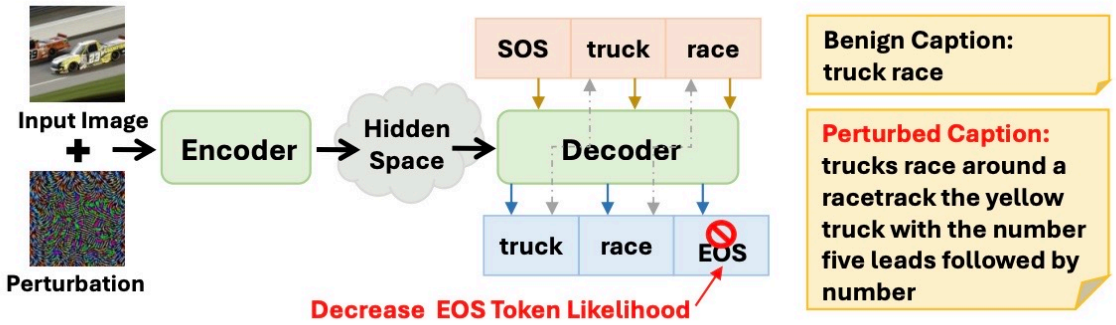
Attack Overview on D2



System Models Under Attack

- Neural Image Caption Generation
- Neural Machine Translation Systems
- Speech System
- Vision LLMs

Attack Overview on D2



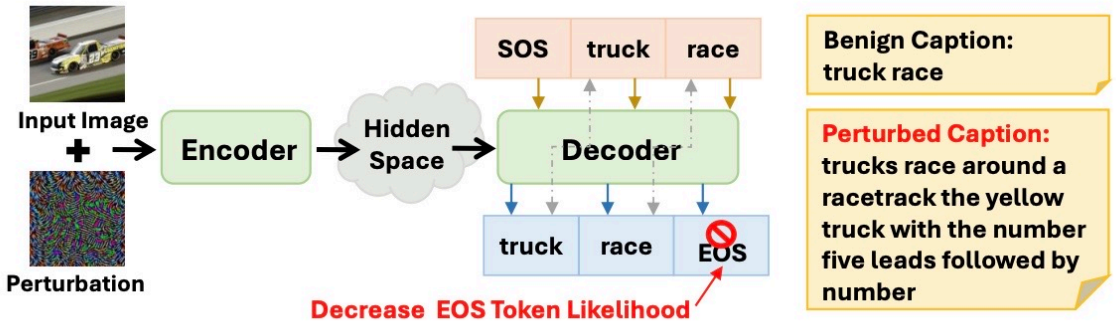
White-Box Attack Strategies

- Sequence Control via Gradients
- Token Replacement/Removal via Gradient
- Token Manipulation with Genetic Algorithm
- Continuous Output Handling

Black-Box Attack Strategies

- Leverage surrogate model

Attack Overview on D2



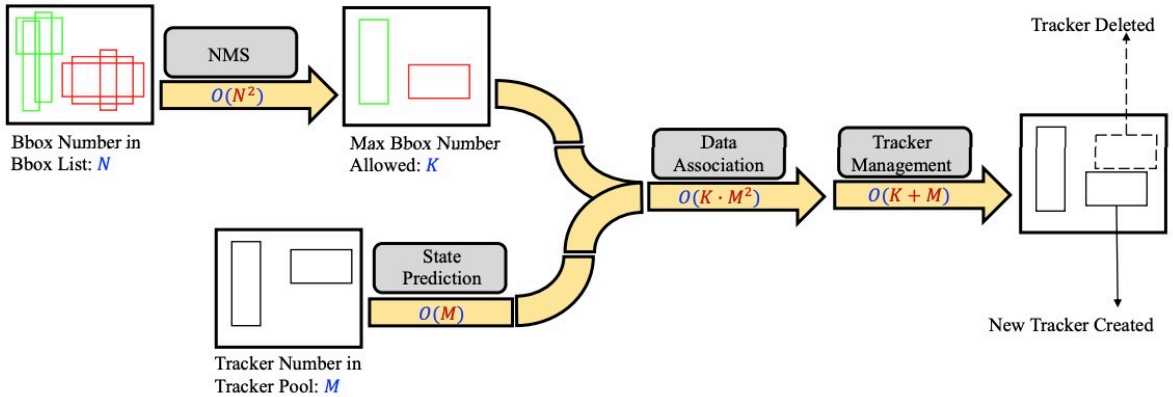
Real-World Impact

- In Mobile devices, NMT attack drained 30% battery in 300 runs compared to 1% with benign inputs
- In cloud services sponge examples on Azure Translation caused 6000× slowdown, straining energy/resources

Implications

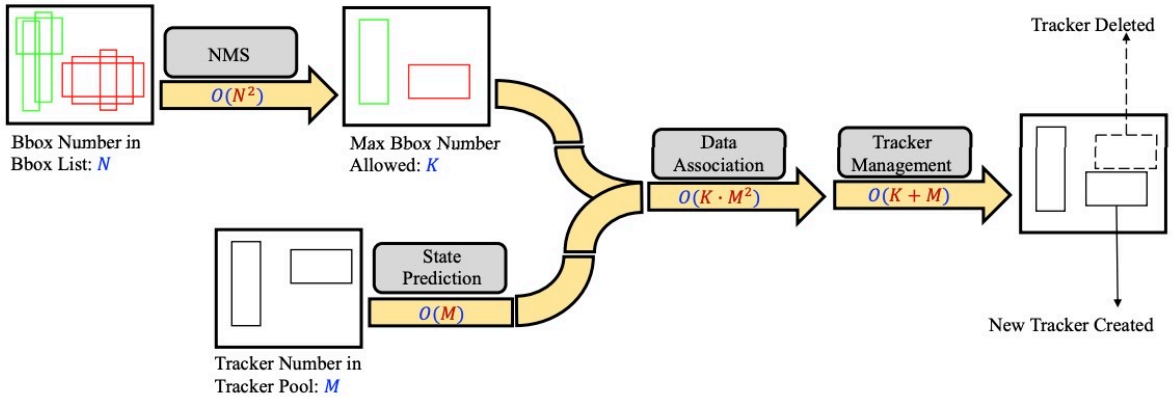
- Scalability: Attacks for long-sequence generative models such as 128K tokens
- Black-box focus : New strategies for closed-source models like GPT
- Efficiency : Optimize gradient/GA-based white-box attacks to cut overhead

Attack Overview on D3



The attacks aim to increase the computational load by generating more outputs than necessary, forcing the system to handle redundant bounding boxes and significantly increasing the complexity of downstream tasks

Attack Overview on D3



System Models Under Attack

- LiDAR Models
- Object Detection Models

White-Box Attack Strategies

- Point Cloud Manipulation
- Bounding Box Manipulation
- Spatial Attention Manipulation
- Feature Manipulation

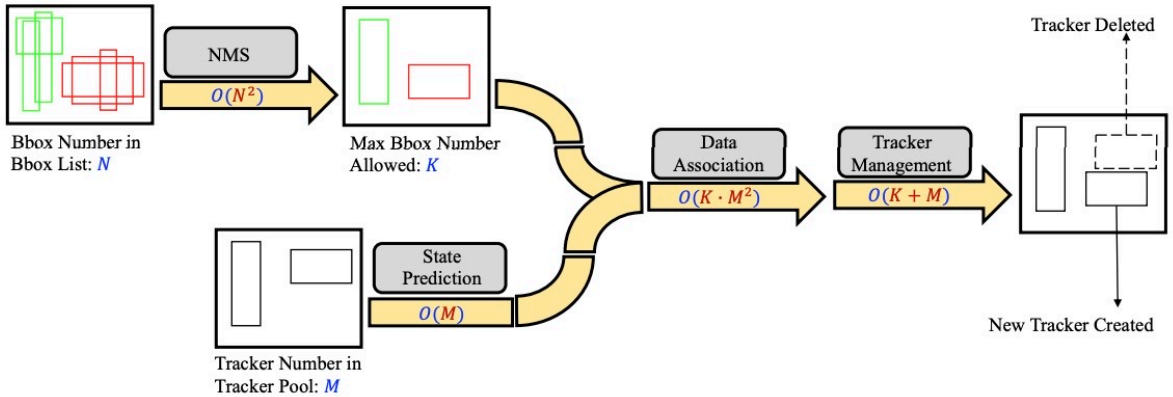
Transferability

- Ensemble learning

Poisoning Attack Strategies

- Training with Data Poisoning

Attack Overview on D3



Real-World Impact

- UAP attacks on video increases inference time by 251% and frame rate drops from 40 FPS to 16 FPS
- Threatens autonomous systems, surveillance, industrial vision with severe performance & reliability loss

Implications

- Stronger black-box attacks for real-world systems without model access
- Hybrid strategies combining partial knowledge with adaptive methods

Resilience of Existing Defenses

We evaluated two types of defenses.

- Detection: input validation with SVM classifiers.
- Mitigation: simple input transformations using JPEG compression and spatial smoothing

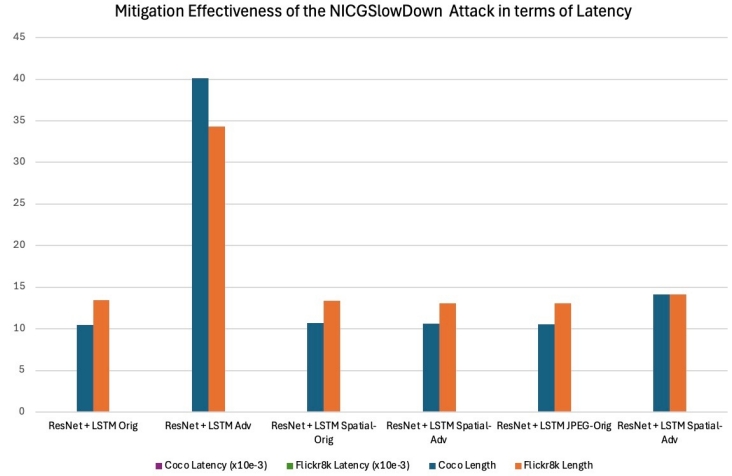
We test defenses under diverse efficiency attacks.

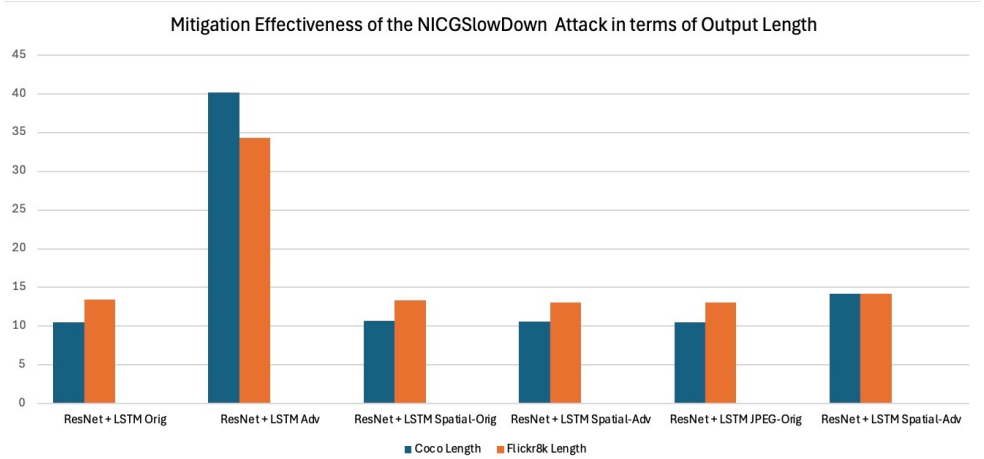
- Attacks Evaluated: NICGSlowDown, DeepSloth, SlowTrack.
- Evaluation Metrics: Accuracy BLEU, Latency, Efficacy.

Experimental Results

- Adversarial inputs increase output length from **10.7 to 62 tokens** and reduce BLEU from **0.16 to 0.0**
- JPEG compression restores output length to **11 tokens** and lowers latency, but BLEU remains low at **0.077**
- Spatial smoothing gives only small gains in length and latency, with almost no BLEU recovery
- On clean inputs, both JPEG and smoothing preserve efficiency and BLEU

JPEG partially restores efficiency but hurts fidelity and smoothing is largely ineffective

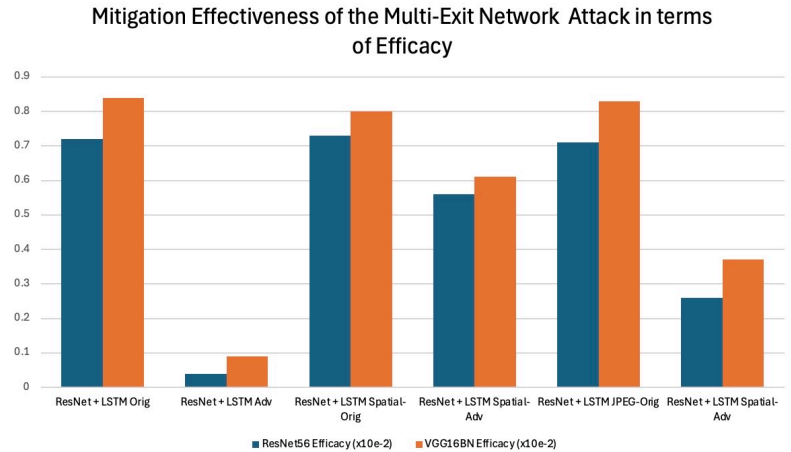


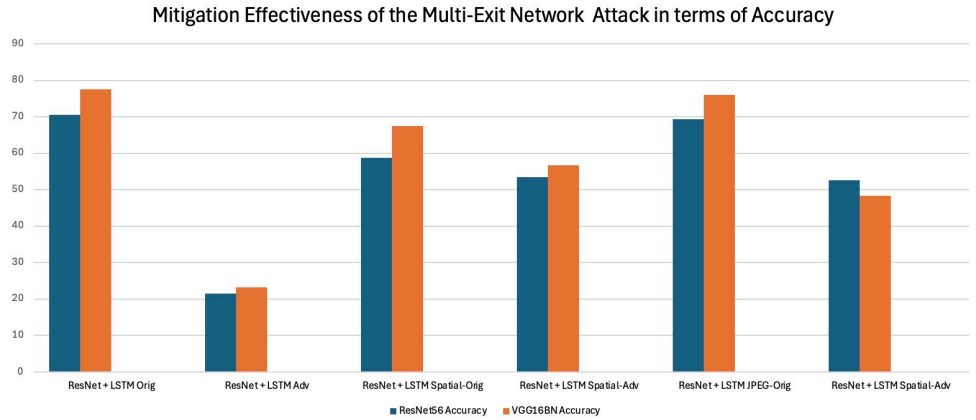


Experimental Results

- ResNet56 drops accuracy from **77.99% to 19.85%** and efficacy from **0.54 to 0.01**
- JPEG compression restores accuracy to **52.19%** and efficacy to **0.08**
- Spatial smoothing restores accuracy to **57.13%** and efficacy to **0.33**
- VGG16BN under attack improves from **18.29% to 45.14%** accuracy with JPEG, but efficacy gains are small.
- Clean inputs remain unaffected by JPEG or smoothing

JPEG and smoothing give partial recovery, but MENs remain highly vulnerable under L∞ attacks

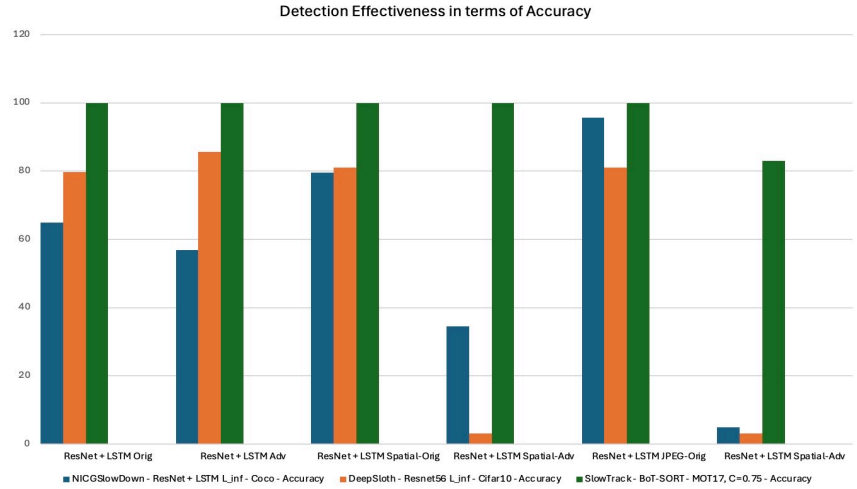


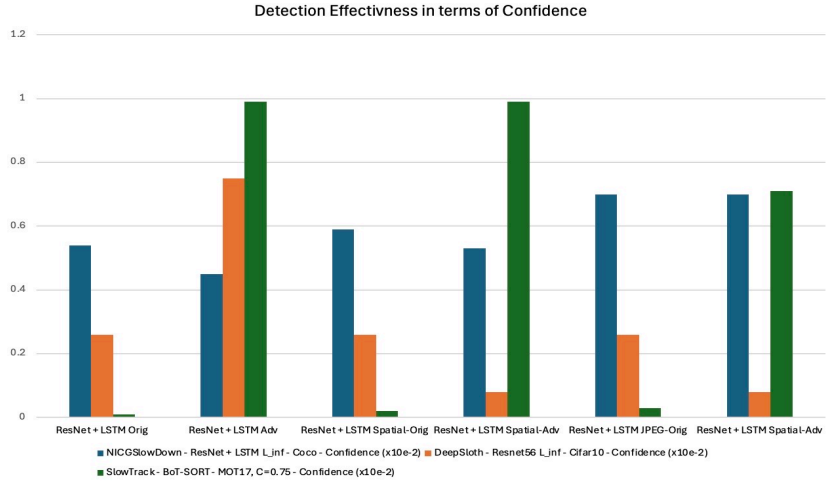


Experimental Results

- Detection accuracy ranges from **51% to 85.6%** on clean and adversarial inputs across NICG and MEN models
- JPEG on clean inputs improves detection in MobileNet+RNN on L2 from **64.6% to 87.3%**
- JPEG on adversarial inputs reduces detection in MobileNet+RNN on L2 from **64.5% to 29.1%** while confidence rises
- Spatial smoothing on clean inputs has little effect
- Spatial smoothing on adversarial inputs often collapses detection in ResNet+LSTM with **0.018 accuracy** and **0.909 confidence**

Detection works on clean/adversarial inputs but fails under JPEG and smoothing, showing overconfidence and poor separability





Takeaways

Real-world threat

Domains like autonomy, wearables, and IoT are especially vulnerable due to tight runtime and resource constraints

Attacks can force excessive computation leading to unexpected offloading and inflated cloud bills or device failures

Attacks can increase latency, energy consumption, or inference cost, often without triggering alerts

Limitations of attacks

Most existing attacks are evaluated on limited models and behaviors (e.g., only on ME-DNNs or early-exit models) and often use white-box assumptions

Few attacks target D2 or D3 mechanisms compared to the relatively well-explored D1

Most attack evaluations are on small or synthetic datasets and there is a lack of demonstration on large-scale, production-grade systems

Future research directions

Design black-box and query-efficient attacks that work with limited feedback (e.g., latency, cost) and unknown architectures

Extend efficiency attacks to large-scale modern models, including LLMs, mixture-of-experts, and diffusion models with complex adaptive behaviors

Create attack techniques that generalize across dynamic behaviors and architectures, transferring effectively across different DDLS types

Investigate stealthier hybrid attacks that combine poisoning and evasion, degrading both efficiency and accuracy to evade detection

GitHub: https://github.com/anonymous-sok/sok_submission
Zenodo: https://zenodo.org/records/15649771

Our Paper

Contact:
Ravishka Rathnasuriya
Ravishka.Rathnasuriya@UTDallas.edu