ARCANE:
ADVERSARIAL ROBUSTNESS USING CLASS-CONDITIONAL GENERATIVE MODELS

Arian Tashakkor

Advisor: Dr. Mohammad Hossein Manshaei

Isfahan University of Technology

OVERVIEW

1. Motivation
2. Introduction to Adversarial Attacks
3. Existing Adversarial Defense Methods
4. Proposed Method: ARCANE
5. Experimental Results
6. Conclusion & Future Work

MOTIVATION

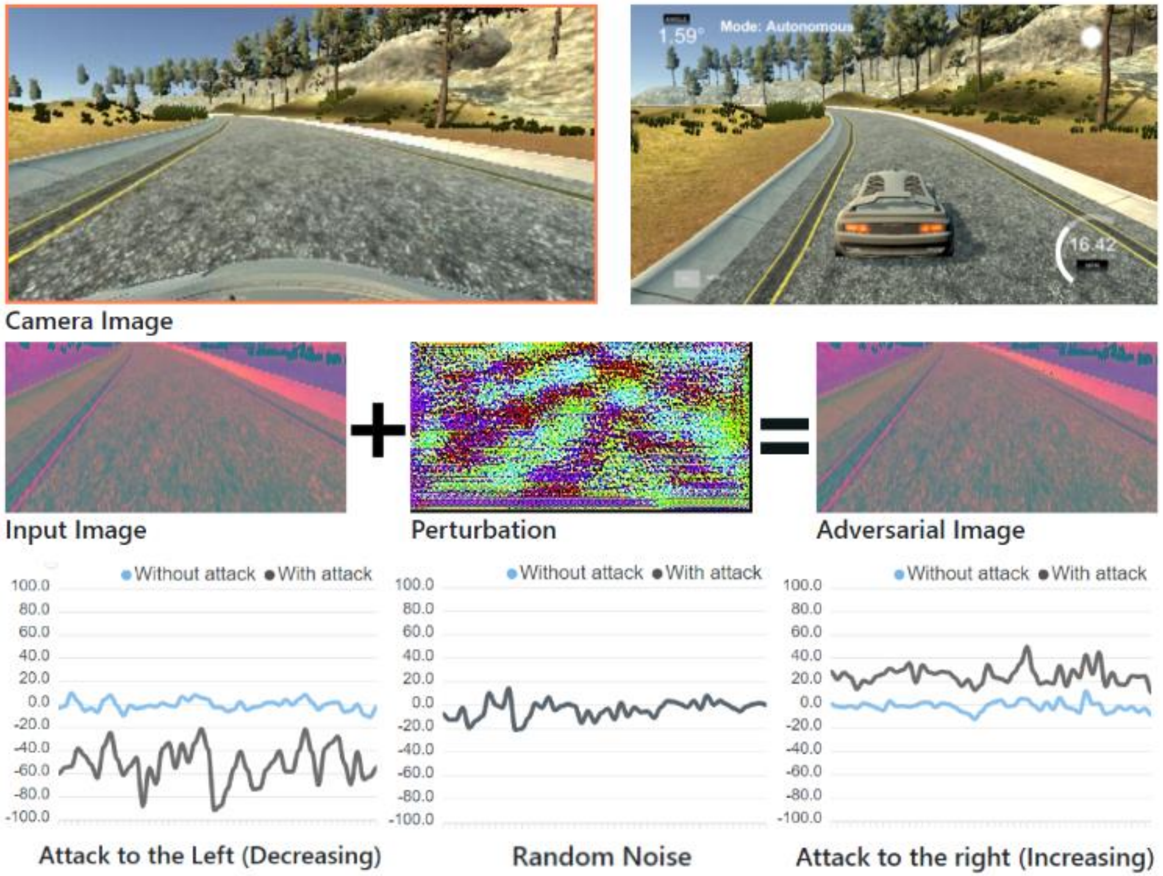
MOTIVATION

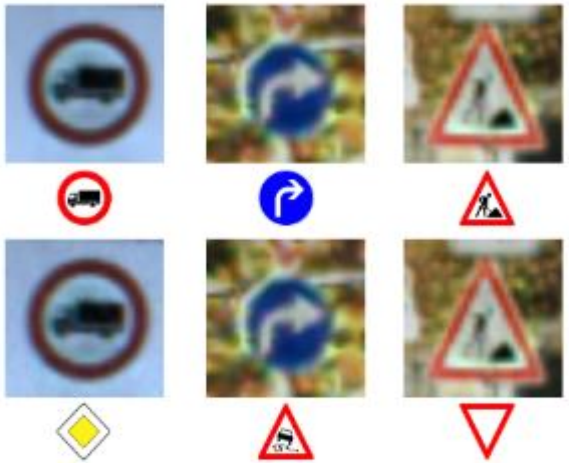
- **AI systems are rapidly advancing across industries**:
  - **77%** of the devices used worldwide include at least one AI feature.
  - In the US **50%** of the mobile users utilize voice search daily.
  - By **2030**, it’s estimated that **10%** of all vehicles will be self-driving.

MOTIVATION

- With AI so ingrained in our daily lives, it is important to ensure their safety against **potential cyberattacks.**
- A class of which, dubbed **Adversarial Attacks**, can manipulate AI models by **adding imperceptible amounts of noise** to an input.
- These attacks can affect a vast array of AI models:
    - **Autonomous Vehicles**
    - **Facial Recognition Systems**
    - **Intrusion Detection Systems**
    - **Large Language Models**
    - **…**

MOTIVATION


1.59°
Mode: Autonomous
16.42
Camera Image
+
=
Input Image
Perturbation
Adversarial Image
● Without attack ● With attack
100.0
80.0
60.0
40.0
20.0
0.0
-20.0
-40.0
-60.0
-80.0
-100.0
Attack to the Left (Decreasing)
Random Noise
Attack to the right (Increasing)



INTRODUCTION TO ADVERSARIAL ATTACKS

INTRODUCTION TO ADVERSARIAL ATTACKS

Adversarial Attacks:

> “An adversarial attack is a technique where imperceptible noise is added to the input of an AI model with the intent of deceiving it into producing incorrect predictions.”

Formally:

$$\hat{x} = x + \delta$$

$$s.t.\ \|\delta\| < \epsilon,$$

$$f_\theta(\hat{x}) \neq f_\theta(x) \qquad (confidence\, reduction)$$

$$or$$

$$\hat{y} = \arg\max_c f_\theta(\hat{x}) \neq y = \arg\max_c f_\theta(x) \qquad (misclassification)$$

INTRODUCTION TO ADVERSARIAL ATTACKS

- **White-box Attack:**
  - Attacker presumed to have total access to model, including it’s output logits, gradients, training data, etc.

- **Black-box Attack:**
  - Attacker presumed to have limited access to the model, potentially only to the output predictions or at most, the raw logits.

In this study we will be focusing on defense against white-box attacks.

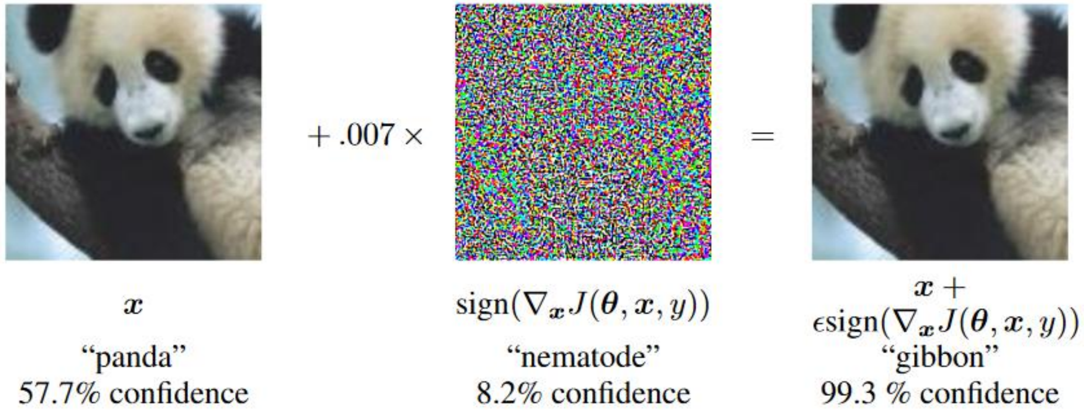
INTRODUCTION TO ADVERSARIAL ATTACKS (FGSM)

Fast Gradient Sign Method (FGSM):

- One of the earliest discovered adversarial attacks.
- Generates adversarial examples by **adding noise to input data in the direction of the gradient of the loss function with respect to the input**, aiming to **maximize the model's prediction error**.

Formally:

$$\hat{x} = x + \delta,$$
$$\boxed{\delta = \epsilon \cdot \mathrm{sgn}\big(\nabla_x J(f_\theta(x), y)\big)}$$


$+ .007 \times$
$=$
$\boldsymbol{x}$
"panda"
57.7% confidence
$\mathrm{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"nematode"
8.2% confidence
$\boldsymbol{x} +$
$\epsilon \mathrm{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"gibbon"
99.3 % confidence

INTRODUCTION TO ADVERSARIAL ATTACKS (CW)

Carlini-Wagner attack(CW):

- A strong optimization-based adversarial attack that generates adversarial examples by **minimizing the perturbation added to the input** while **ensuring the modified input misleads the model**.

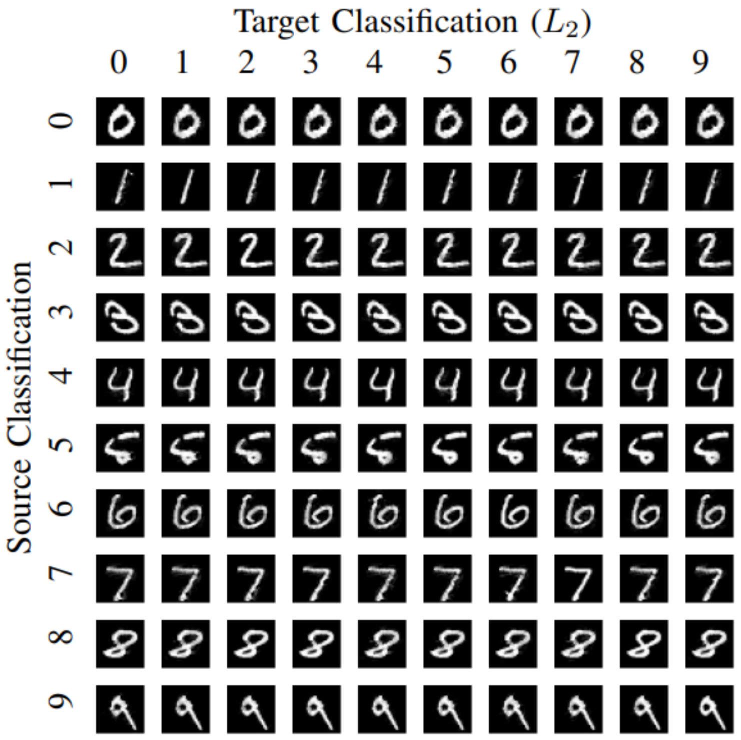
Formally:

$$\hat{x} = x + \delta,$$

$$\boxed{\delta\colon \arg\min_{\omega} \|\delta\|_{\mathrm{p}} + \mathrm{c} \cdot f(x + \delta),}$$

$$\delta = \frac{1}{2}(\tanh(\omega) + 1) - x,$$

$$f(x) = \max(\max\{Z(x)_i : i \neq t\} - Z(x)_t, -\kappa)$$



EXISTING ADVERSARIAL DEFENSE METHODS

EXISTING ADVERSARIAL DEFENSE METHODS

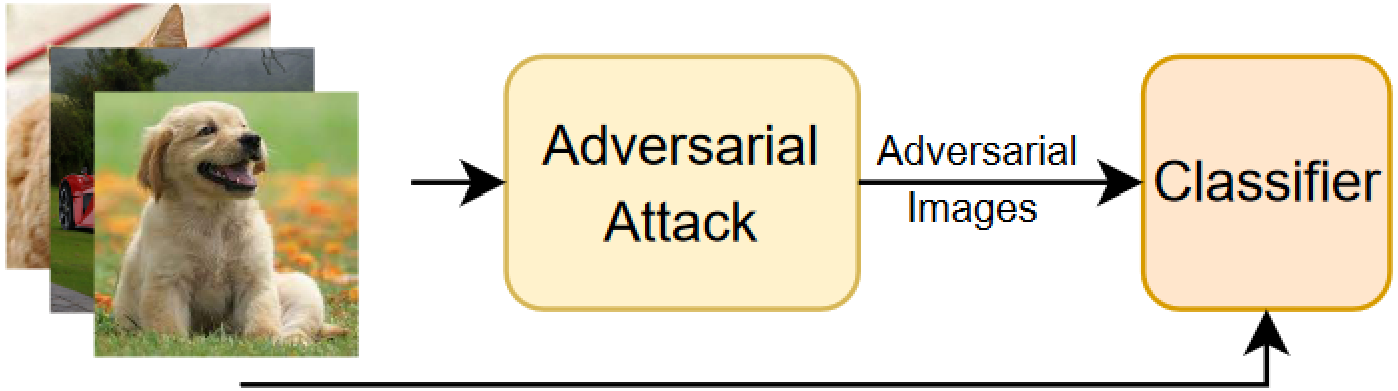
Adversarial Defense:

- **Detection:** Simply **detect** the existence of an attack.
- **Purification:** Make the **victim model more robust** so that it manages to deflect adversarial attacks or **use and auxiliary model to correct the prediction** output of the victim model.

Our focus: Detection + Purification using an auxiliary model

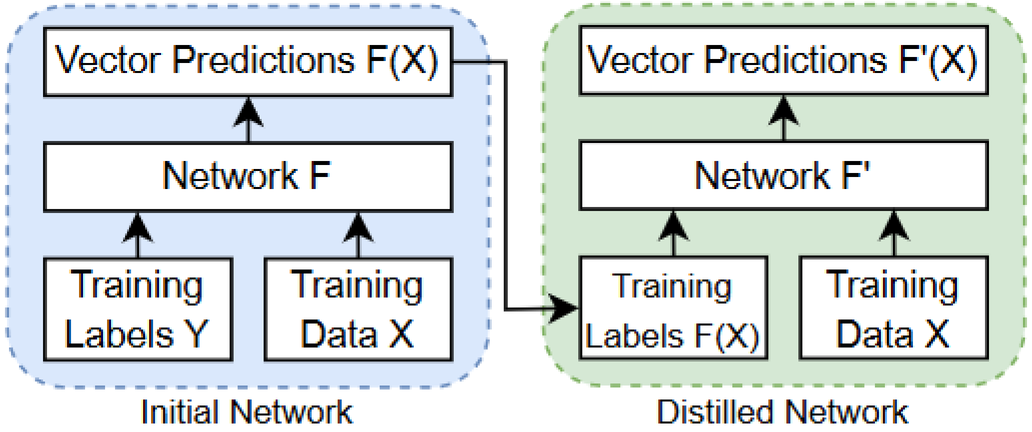
EXISTING ADVERSARIAL DEFENSE METHODS

- **Adversarial Training:**
  - Make adversarial samples.
  - Train model on a mix of normal and adversarial samples.



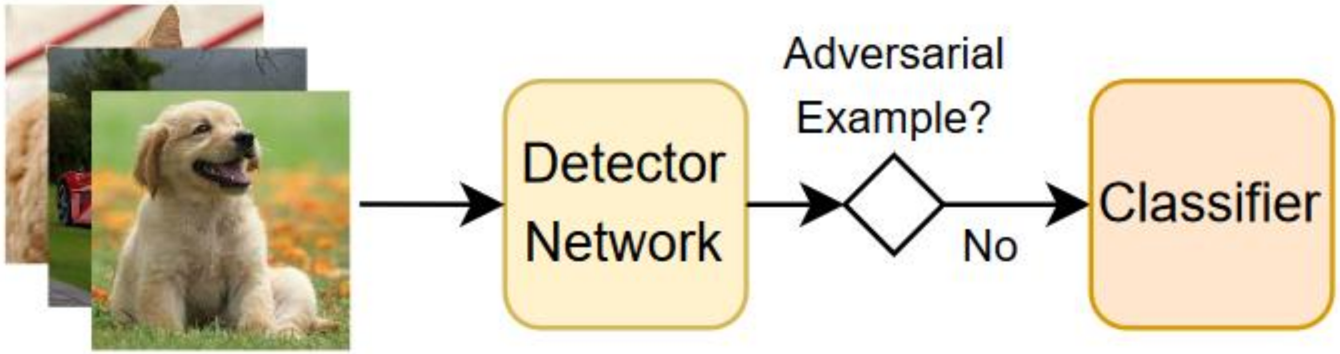
EXISTING ADVERSARIAL DEFENSE METHODS

- **Defensive Distillation**:
  - Train model F on dataset $D: (X, Y)$ to obtain predictions $F(X)$.
  - Train distilled model $F_{distilled}$ on $D': \left(X, F(X)\right)$.
  - Compact model with relaxed labels → More robust against attacks with less risk of overfitting.



EXISTING ADVERSARIAL DEFENSE METHODS

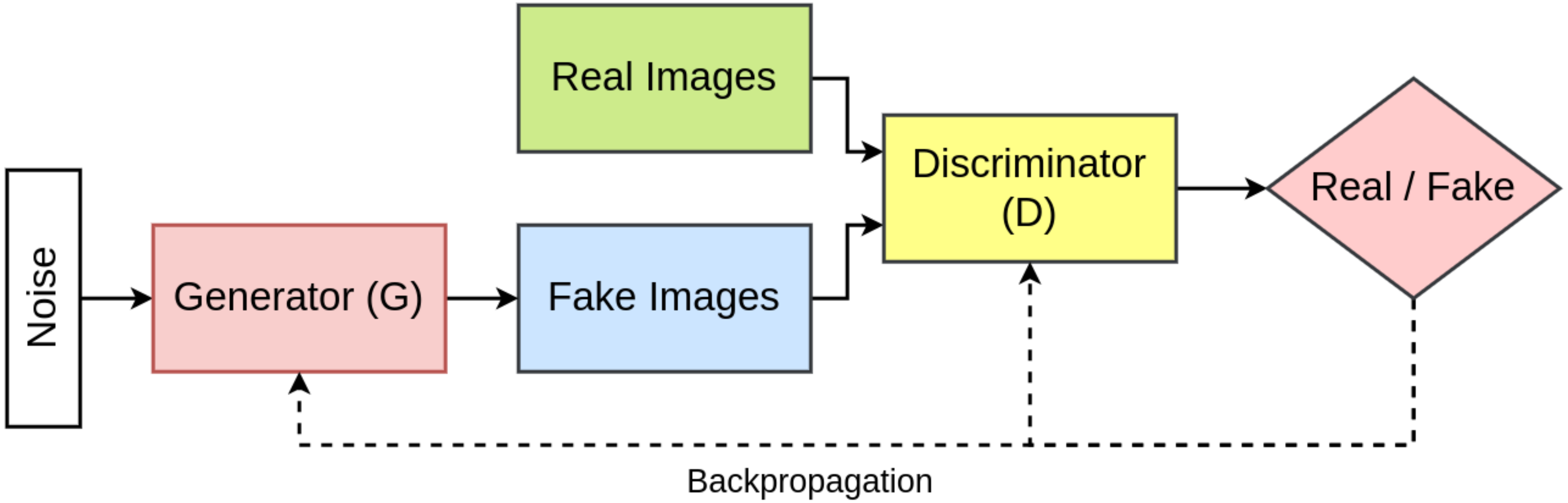
- **Detectors**:
  - Estimate the statistical features of clean samples with a mathematical model.
  - Decide if given sample is adversarial through the trained model.



DETOUR: GANS

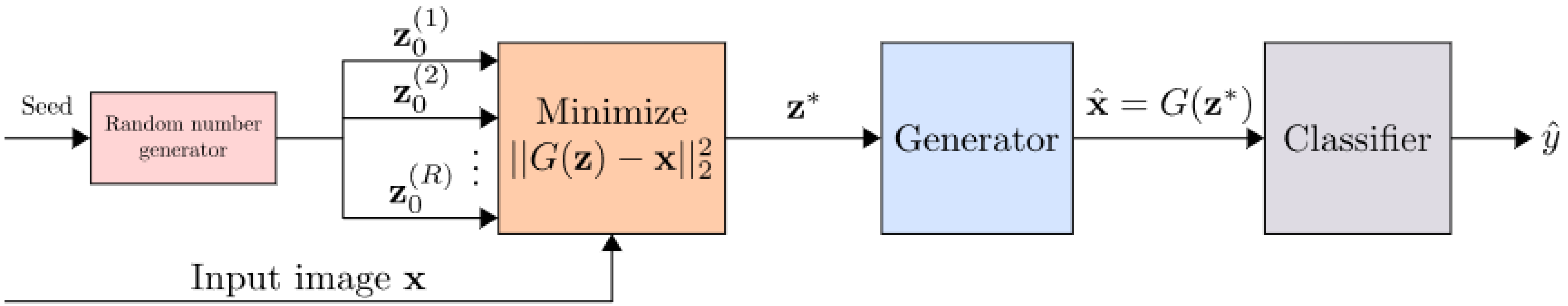
- **Generative Adversarial Networks (GANs):**

$$\min_{G}\max_{D} V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))]$$



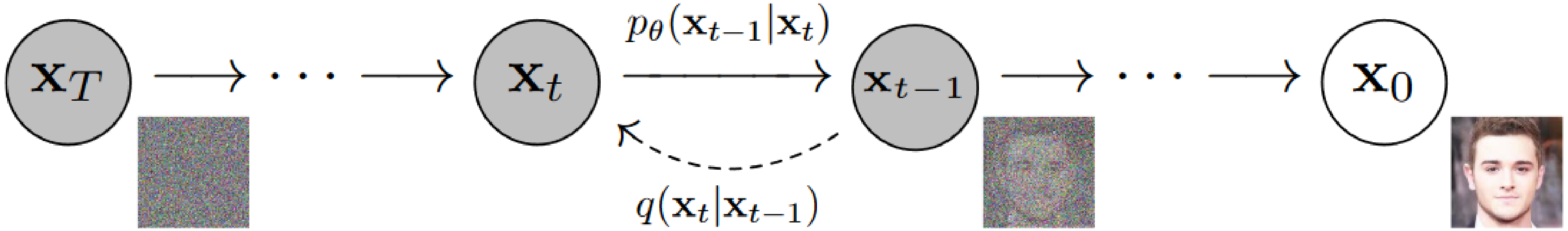
EXISTING ADVERSARIAL DEFENSE METHODS

- **Projection (Our Focus)**:
  - Train generative model G on clean samples.
  - At test time, project input image x onto the distribution learned by G.
  - I.e., $x_{clean} = \arg\min_{G(z)} \|x - G(z)\|_2^2$
  - Example: **Defense-GAN**



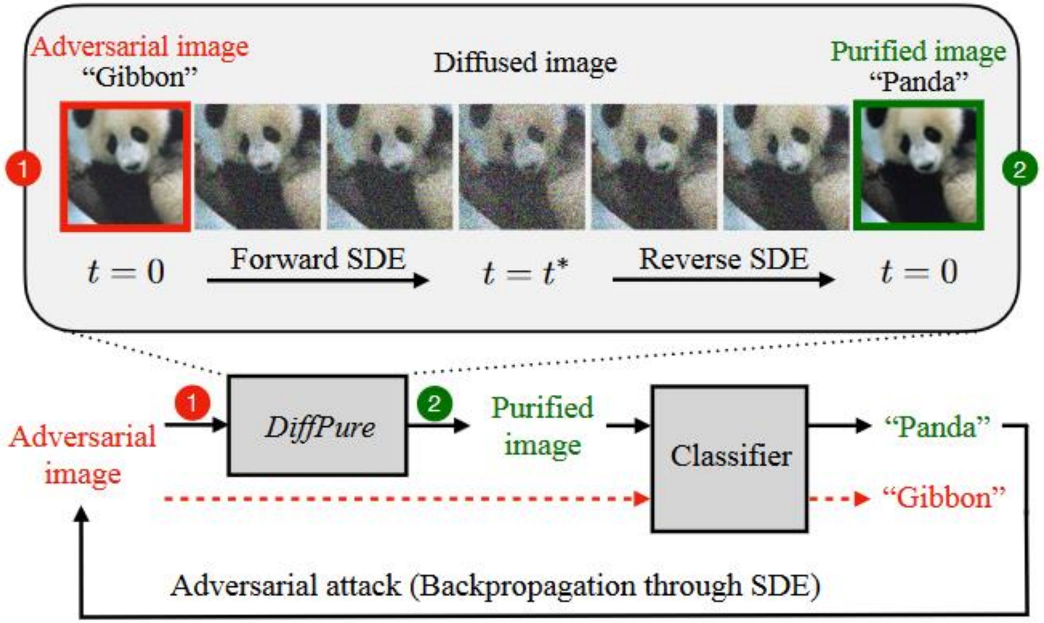
DETOUR: DIFFUSION MODELS

- **Diffusion Models:**



EXISTING ADVERSARIAL DEFENSE METHODS

- **Projection:**
  - Example: **DiffPure**
  - Train a generative diffusion model $Diff$ on the clean samples.
  - Forward diffusion for t^* timesteps.
  - Backward diffusion for t^* timesteps.



EXISTING ADVERSARIAL DEFENSE METHODS - PREDECESSOR

- **ACGAN-ADA:**
- An extension to the prior work, **Defense-GAN** which employs an **ACGAN** instead of the vanilla GAN.
- **Why class-conditional?** Provably easier to model conditioned distributions.
- Uses the **class label** of the samples as a way to guide the purification and detection procedures.
- Uses 3 criteria to decide whether a given sample is adversarial:
  - $\boldsymbol{S_C} = p_D(\hat{c}|x)$
  - $\boldsymbol{S_R} = D(x)$
  - $\boldsymbol{S_g} = \min_z \|x - G(z|\hat{c})\|_2^2$

DETOUR: ACGAN

- **Auxiliary Classifier Generative Adversarial Networks (ACGANs):**

$$L_S = -\left(\mathbb{E}\left[P(S = real|X_{real})\right] + \mathbb{E}\left[P(S = fake|X_{fake})\right]\right)$$

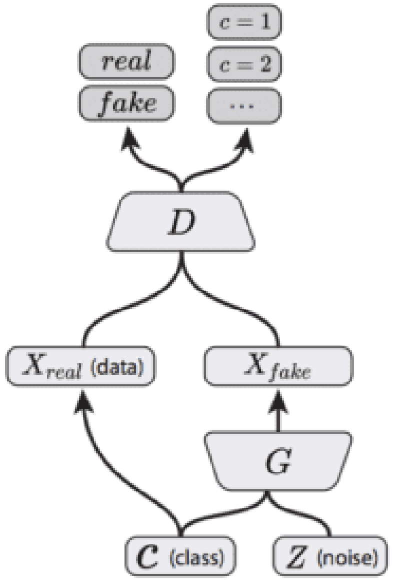
$$L_C = -\left(\mathbb{E}\left[P(C = c|X_{real})\right] + \mathbb{E}\left[P(C = c|X_{fake})\right]\right)$$

Generator Objective:

$$\min L_C - L_S$$

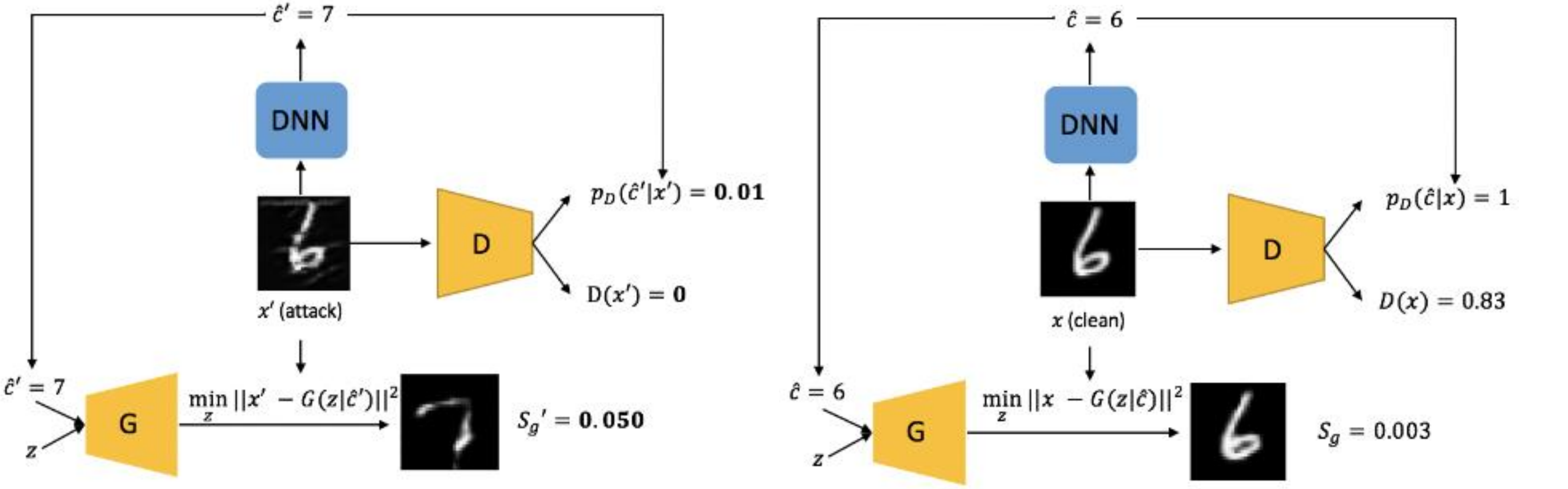
Discriminator Objective:

$$\min L_C + L_S$$



EXISTING ADVERSARIAL DEFENSE METHODS – PREDECESSOR

- **ACGAN-ADA:**



(a) Detection for an attack image

(b) Detection for a clean image

EXISTING ADVERSARIAL DEFENSE METHODS - PREDECESSOR

- **ACGAN-ADA Shortcomings:**
  - Poor performance when the dataset has **many modes.**
  - **Poor generation quality** for purification tasks.
  - Use of limited criteria with **manually tuned hyperparameters.**

PROPOSED METHOD: ARCANE

PROPOSED METHOD: ARCANE

- We aim to improve **ACGAN-ADA** by:
  1. Employing a more advanced ACGAN architecture (**ReACGAN**) that **boosts the performance in highly multi-modal settings**.
  2. Adding **new decision criteria** for adversarial sample detection and use of a trained **XGBoost classifier** instead of manually tuned thresholds.

- Moreover, we investigate the use of **Conditional Diffusion Models** instead of GANs to see how they can affect purification performance.

- Finally, we present a novel **purification regime** to make use of the conditional generative models to the fullest.

ARCANE: DETECTION

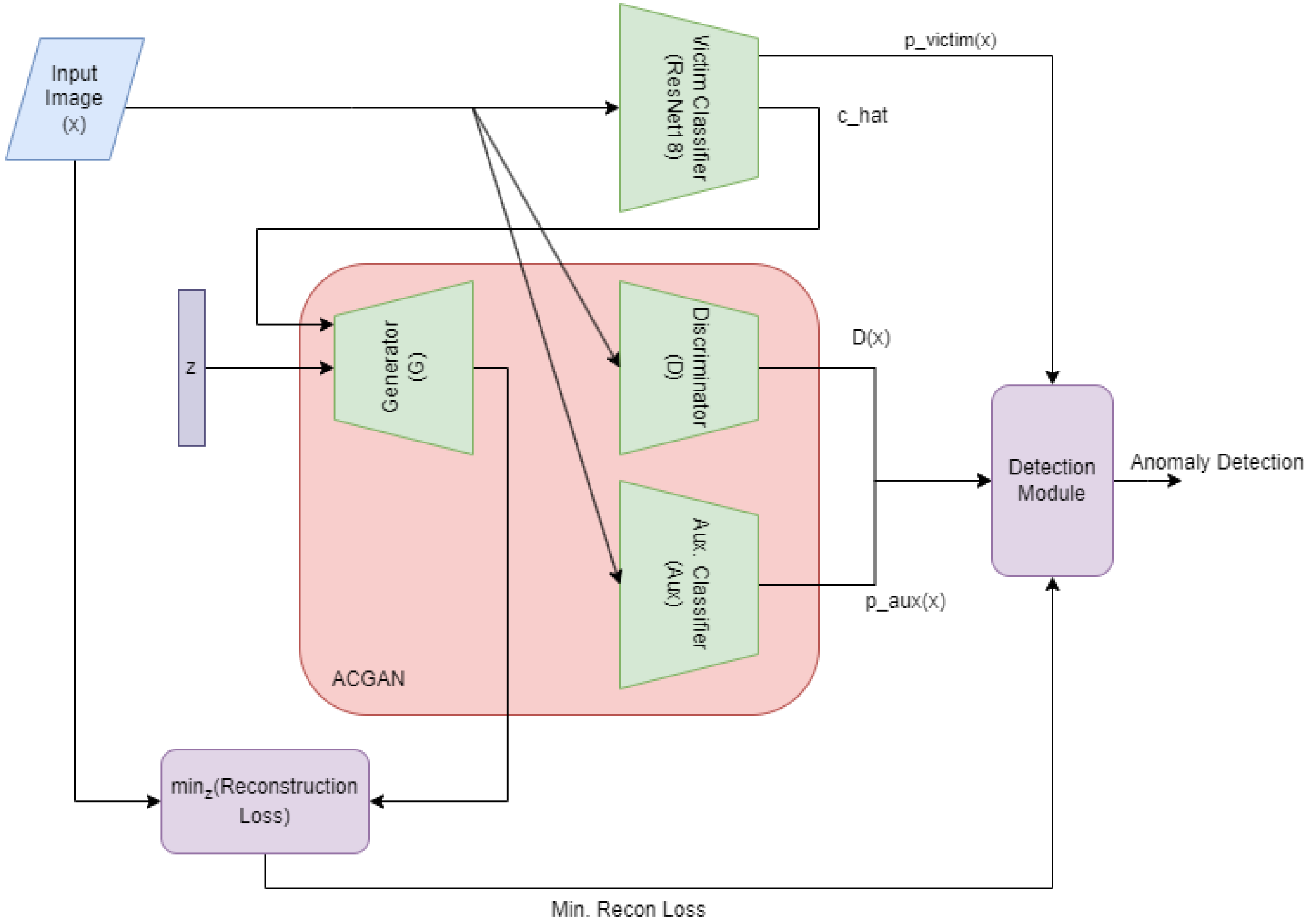
- Our proposed methods: **ARCANE-GAN** and **ARCANE-Diff**
- For **Detection** we add the following criteria to the ones previously used by **ACGAN-ADA:**

1. $S_C = p_D(\hat{c}|x)$
2. $S_R = D(x)$
3. $S_g = \min_z \|x - G(z|\hat{c})\|_2^2$
4. $\boldsymbol{p_{victim}(\hat{c}|x)}$
5. $\boldsymbol{JSD\left(p_D(x) \parallel p_{victim}(x)\right)}$
6. $\boldsymbol{\log\left(p_D(\hat{c}|x)\right) + \log\left(D(x)\right)}$

These 6 features are used to train a **XGBoost** classifier which then identifies whether a given sample is **adversarial or not.**

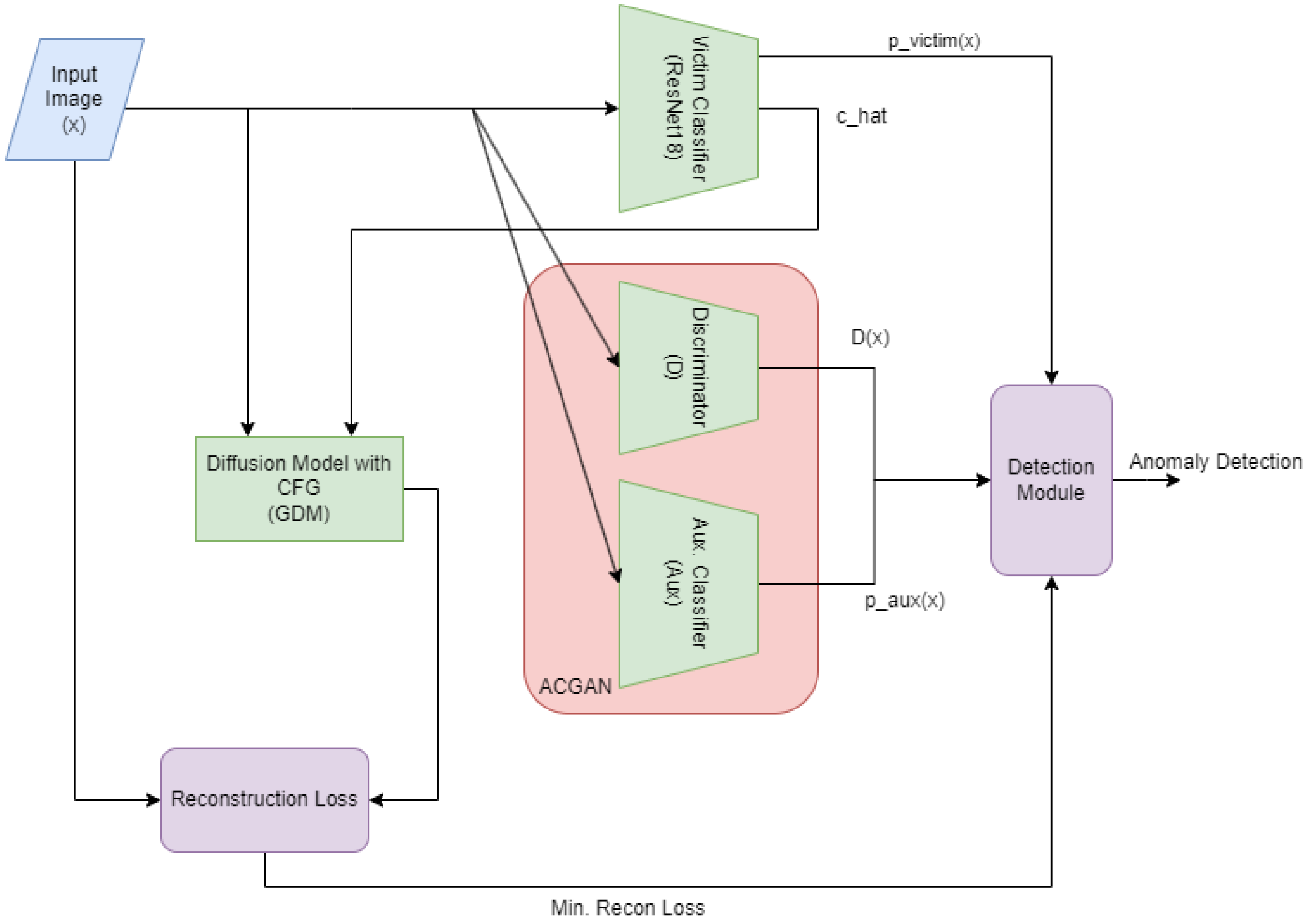
ARCANE: DETECTION

- **ARCANE-GAN**



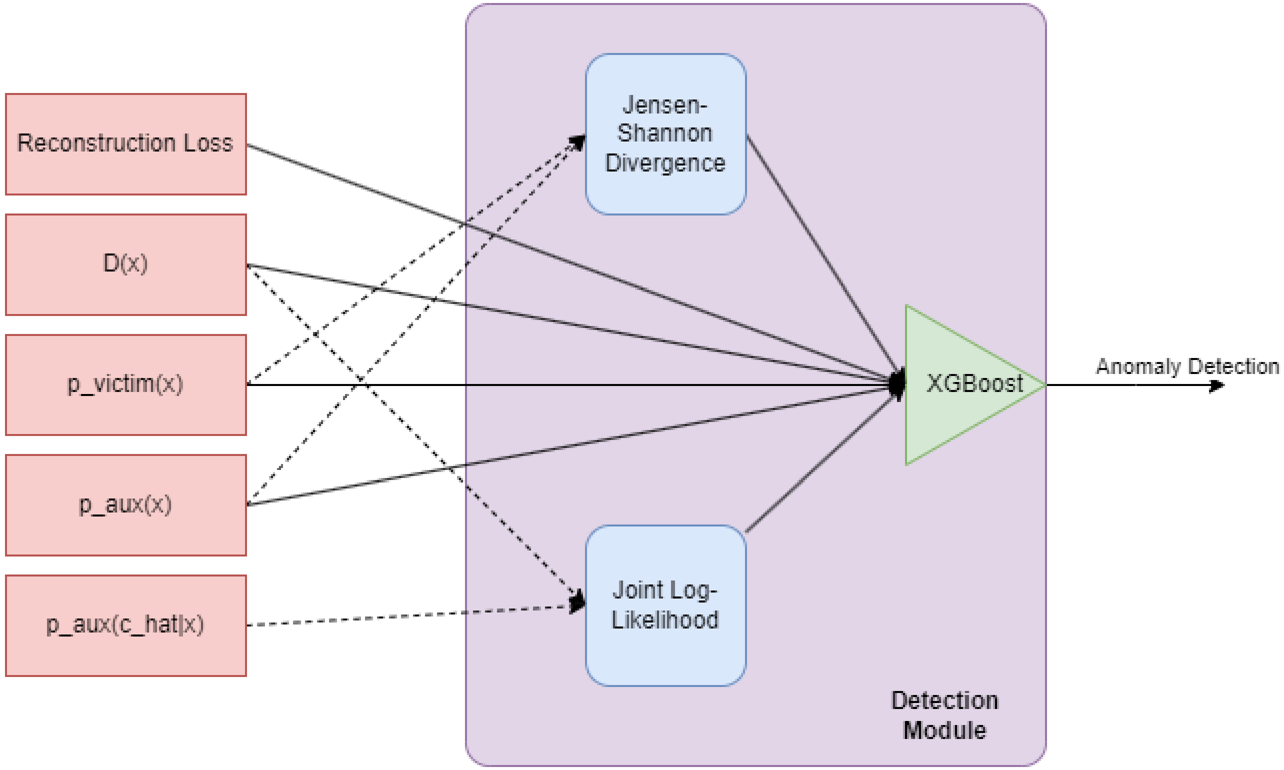
ARCANE: DETECTION

- **ARCANE-Diff**



ARCANE: DETECTION

- **Detection Module**

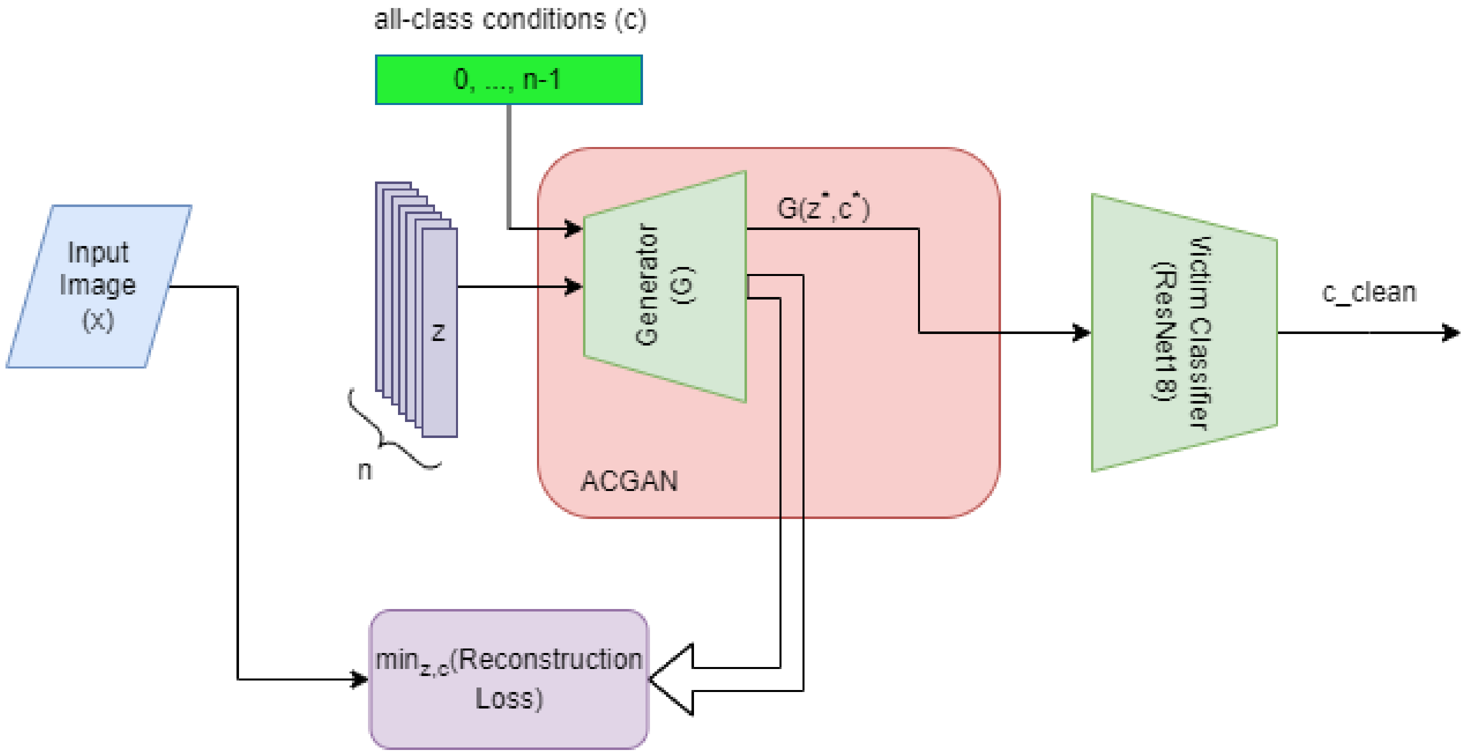


ARCANE: PURIFICATION

- For **purification** we aim to take advantage of the conditional generative models.
- Given an input sample x:
  - **For ARCANE-GAN:**
    - Cleaned Sampled: $G(z^*|c^*) = \arg\min_{z,c} \|x - G(z|c)\|_2^2$
  - **For ARCANE-Diff:**
    - For each class c:
      - Forward diffusion for t^* steps.
      - Backward diffusion for t^* steps.
    - Cleaned Sample: $GDM(z|c^*) = \arg\min_{c} \|x - GDM(z|c)\|_2^2$

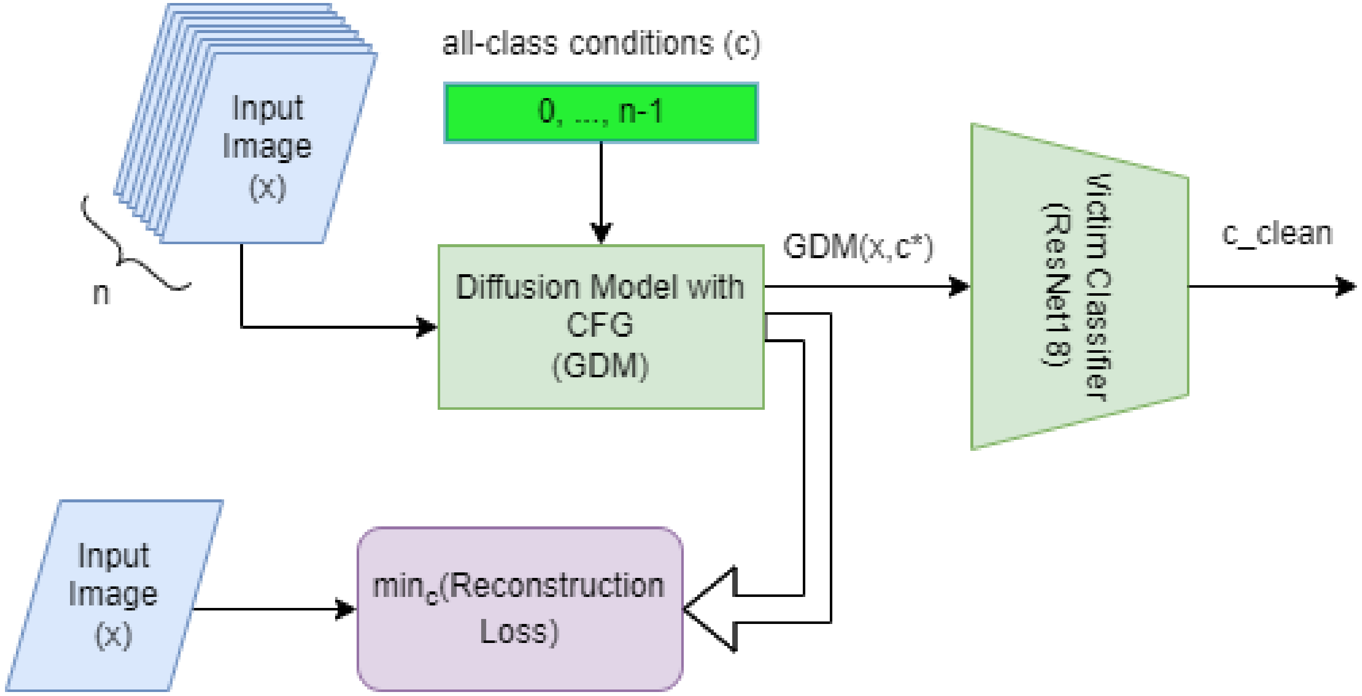
ARCANE: PURIFICATION

- **ARCANE-GAN**



ARCANE: PURIFICATION

- **ARCANE-Diff**



EXPERIMENTAL RESULTS

EXPERIMENTAL RESULTS

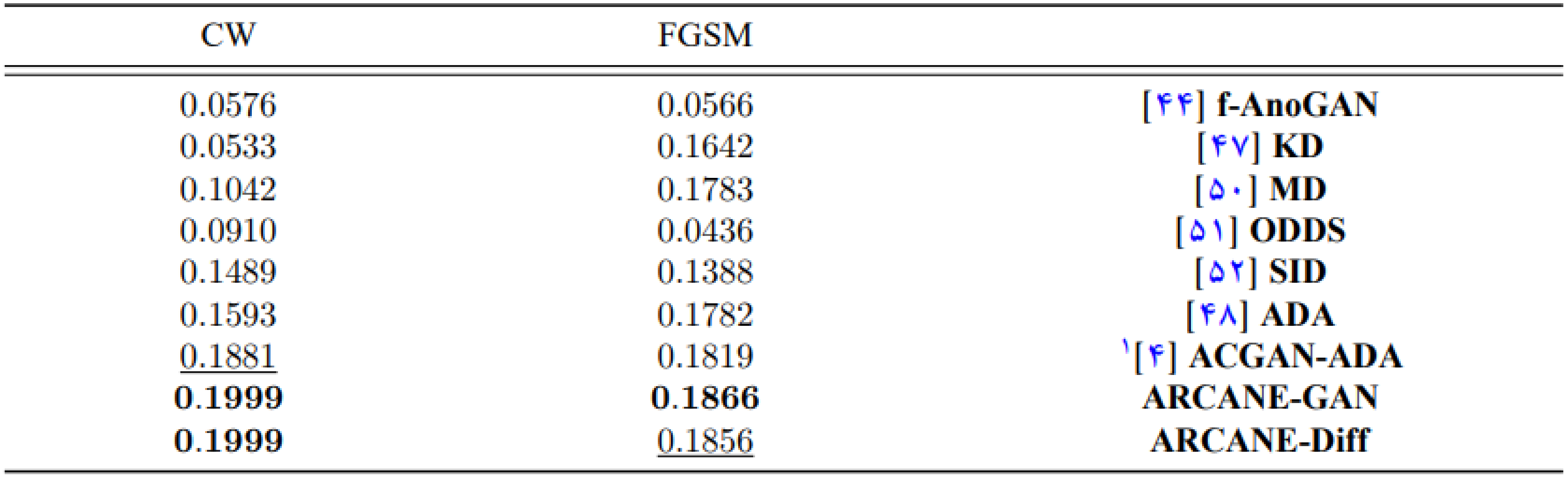
- **Implementation Overview:**
  - **Language:** Python
  - **Deep Learning Framework:** PyTorch
  - **GPU:** 1 x NVIDIA 3090 (courtesy of IUT)
  - **ACGAN Variant:** ReACGAN
  - **Diffusion Model Variant:** Label-conditioned Diffusion Model with CFG
  - **Detection Head Model:** XGBoost
  - **Attacks:** FGSM (*weak*), CW (*strong*)
  - **Datasets:** CIFAR10 (*easy*), TinyImageNet (*difficult*)
  - **Evaluation Metrics:**
    - **Detection:** Partial Area Under Curve @ False Positive Rate $\leq$ 0.2 (*pAUC-0.2*)
    - **Purification:** Matched-classification Accuracy

EXPERIMENTAL RESULTS

- **Training Method:**
  - **Generation:** Entire training splits of both datasets used to train the diffusion and GAN models.
  - **Detection:** In order to train the XGBoost classification head:
    1. **1000 clean images** balanced-sampled from each dataset.
    2. **1000 adversarial images** for each attack method (CW, FGSM) created.
    3. For each (dataset, attack) pair**, a total of 2000 samples** (1000 clean and 1000 adversarial) form the final training dataset.
    4. **5-fold cross-validation** across the dataset performed to tune the hyperparameters of the XGBoost model.
- **Evaluation Method:**
  - Same number of balanced samples as training (**2000**) used for each (attack, dataset) pair taken from the **test splits** of the datasets are used for evaluation.

EXPERIMENTAL RESULTS: DETECTION

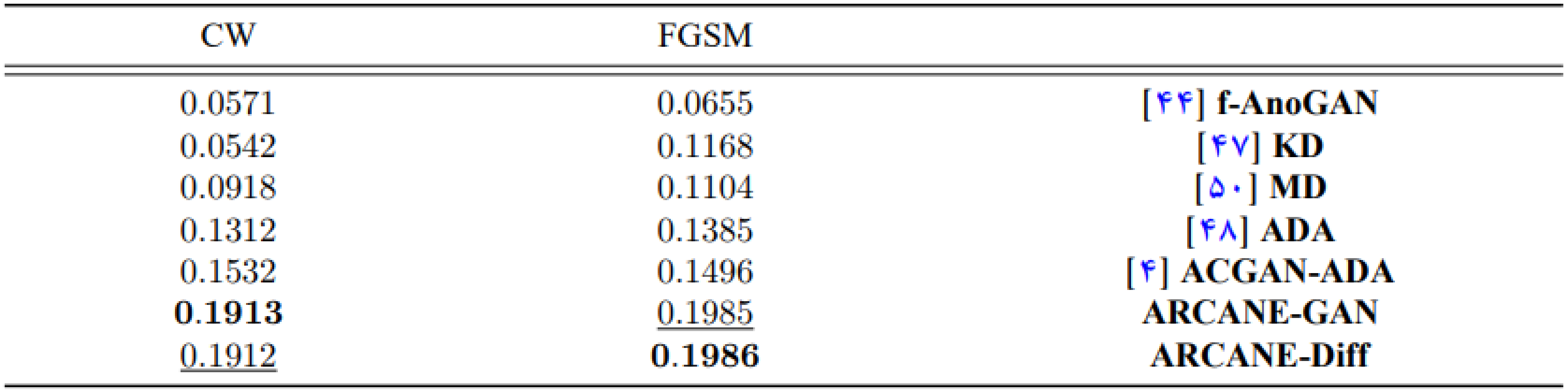
Detection Results on CIFAR10



CW	FGSM	
0.0576	0.0566	[۴۴] **f-AnoGAN**
0.0533	0.1642	[۴۷] **KD**
0.1042	0.1783	[۵۰] **MD**
0.0910	0.0436	[۵۱] **ODDS**
0.1489	0.1388	[۵۲] **SID**
0.1593	0.1782	[۴۸] **ADA**
0.1881	0.1819	۱[۴] **ACGAN-ADA**
0.1999	**0.1866**	**ARCANE-GAN**
0.1999	0.1856	**ARCANE-Diff**

EXPERIMENTAL RESULTS: DETECTION

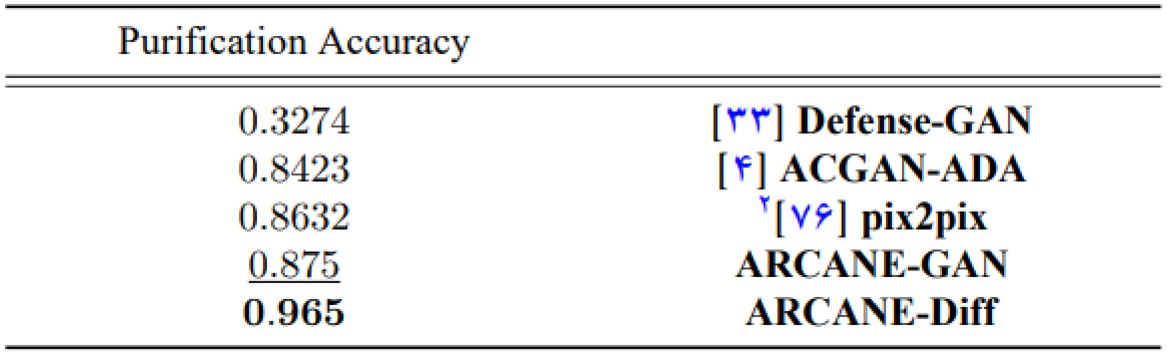
Detection Results on Tiny-ImageNet



CW	FGSM	
0.0571	0.0655	[۴۴] **f-AnoGAN**
0.0542	0.1168	[۴۷] **KD**
0.0918	0.1104	[۵۰] **MD**
0.1312	0.1385	[۴۸] **ADA**
0.1532	0.1496	[۴] **ACGAN-ADA**
0.1913	0.1985	**ARCANE-GAN**
0.1912	**0.1986**	**ARCANE-Diff**

EXPERIMENTAL RESULTS: PURIFICATION

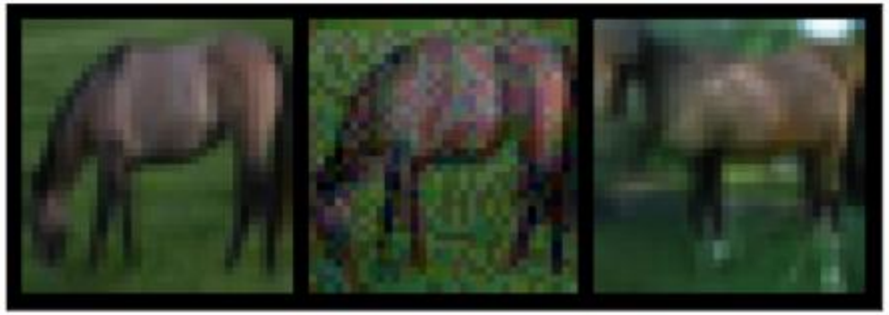
Purification Results on CIFAR10 + CW



Purification Accuracy	
0.3274	[۳۳] **Defense-GAN**
0.8423	[۴] **ACGAN-ADA**
0.8632	[۲][۷۶] **pix2pix**
0.875	**ARCANE-GAN**
0.965	**ARCANE-Diff**

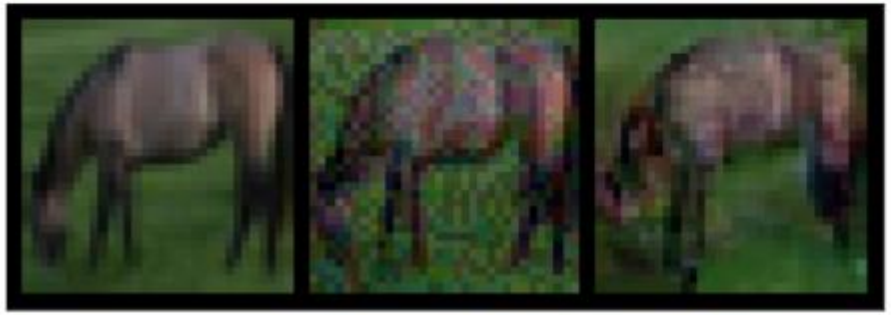
EXPERIMENTAL RESULTS: PURIFICATION

ARCANE-GAN



Clean Adversarial Purified

ARCANE-Diff



Clean Adversarial Purified

EXPERIMENTAL RESULTS: SUMMARY

- **On detection:**
  - **CIFAR10:** ARCANE beats the SOTA by **6.27%** and **2.58%** on CW and FGSM respectively.
  - **Tiny-ImageNet:** ARCANE beats the SOTA by **24.87%** and **32.75%** on CW and FGSM respectively.

- **On purification:**
  - **CIFAR10:** ARCANE beats the SOTA by **11.79%** on CW.

CONCLUSION & FUTURE WORK

CONCLUSION

- **In this work we presented ARCANE.**
- A novel adversarial robustness framework based on class-conditional generative modelling.
- We used two new generative models in **ReACGAN** and a **Conditional Diffusion Model**.
- We evaluated ARCANE on **CIFAR10** and **Tiny-ImageNet** dataset over **FGSM** and **CW** attacks.
- We have experimentally shown that ARCANE on average:
    - Performs **16.62% better on detection** than the current SOTA.
    - Performs **11.8% better on purification** than the current SOTA.

FUTURE WORK

- **Reconstruction Loss as a feature is not very impactful.** Potential improvements:
  - Use of a better reconstruction loss measure.

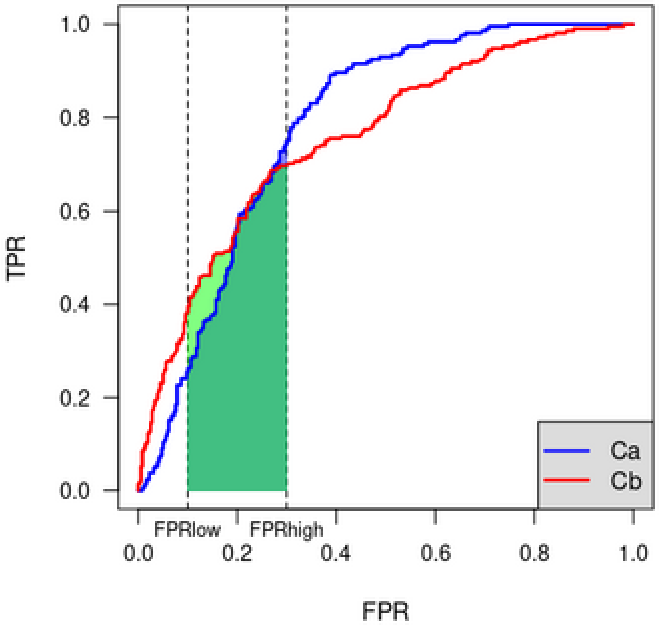
- **ARCANE is slow.** Potential improvements:
  - Fasters sampling techniques on the diffusion model.

THANKS FOR YOUR ATTENTION

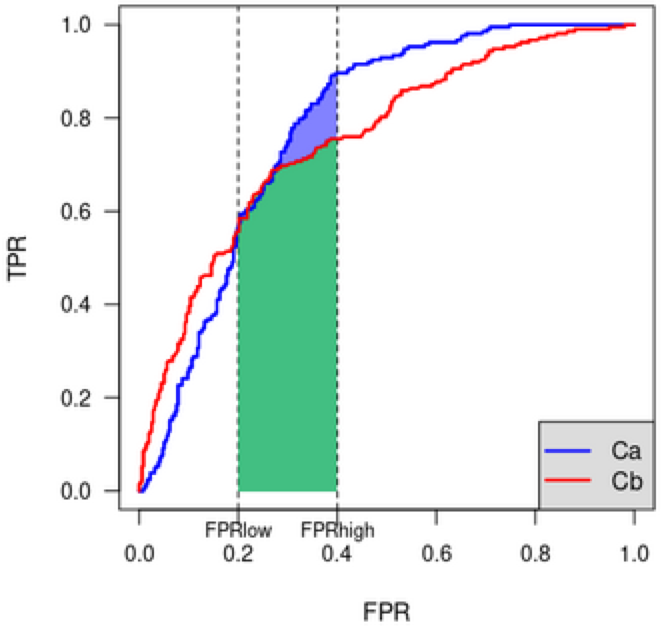
Q&A

ADDENDUM I: P-AUC

- Proposed with the goal of **restricting the evaluation of given ROC curves in the range of false positive rates that are considered interesting for diagnostic purposes.**
- The partial AUC is computed as the area under the ROC curve in the vertical band of ROC space where FPR is in the range $\left[FPR_{low}, FPR_{high}\right]$.

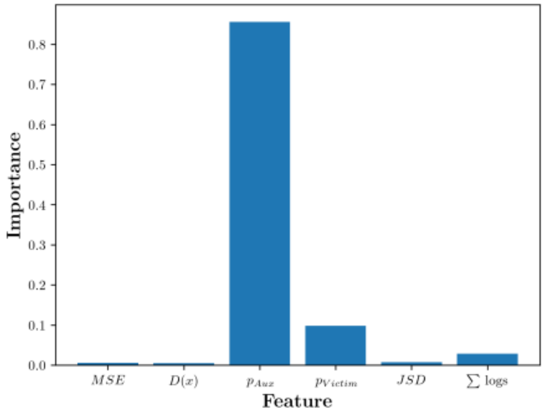


$FPR_{low} = 0.1, FPR_{high} = 0.3$

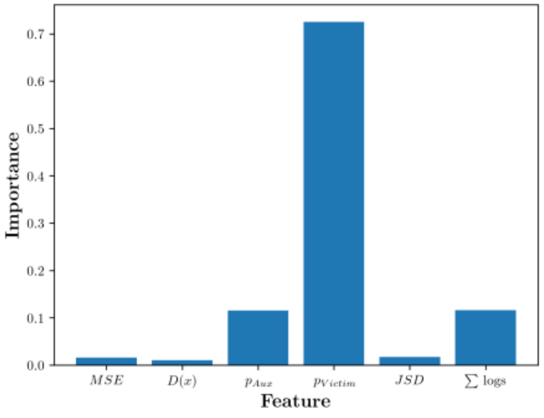


$FPR_{low} = 0.2, FPR_{high} = 0.4$

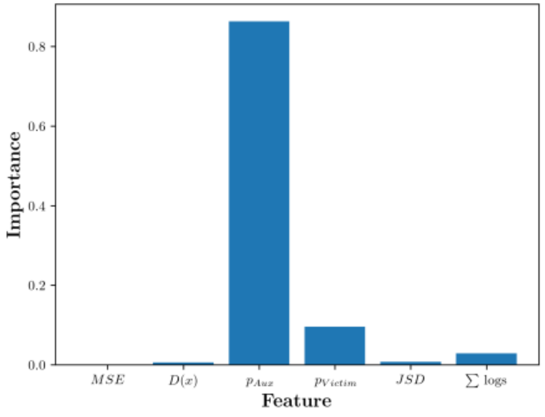
ADDENDUM II: FEATURE IMPORTANCE



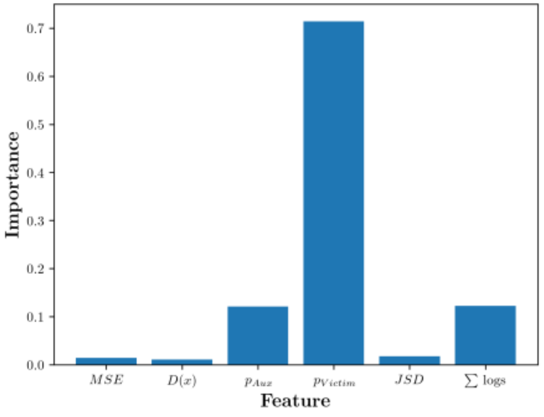
ARCANE-Diff, CIFAR10, CW (و)



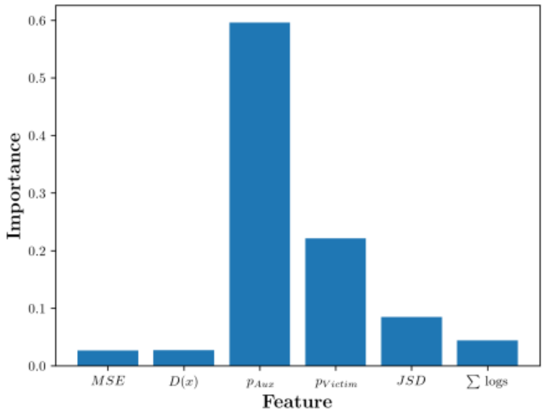
ARCANE-Diff, CIFAR10, FGSM (ه)



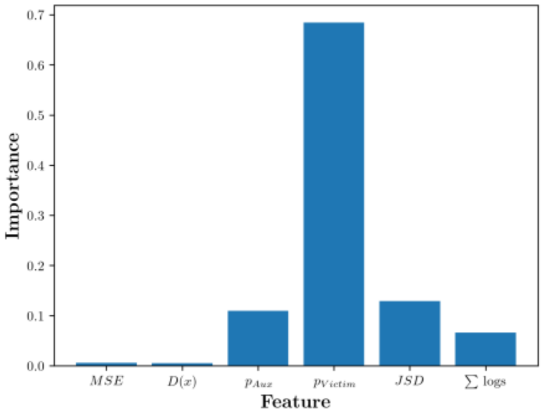
ARCANE-GAN, CIFAR10, CW (ب)



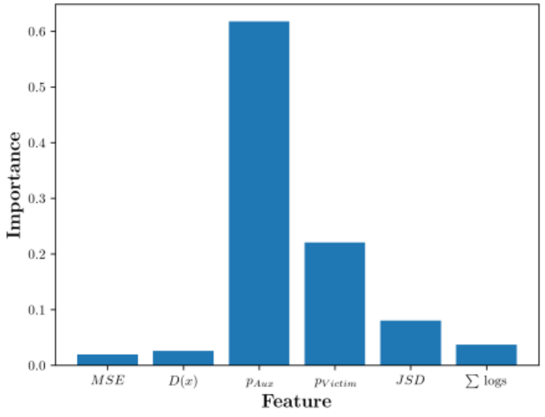
ARCANE-GAN, CIFAR10, FGSM (آ)



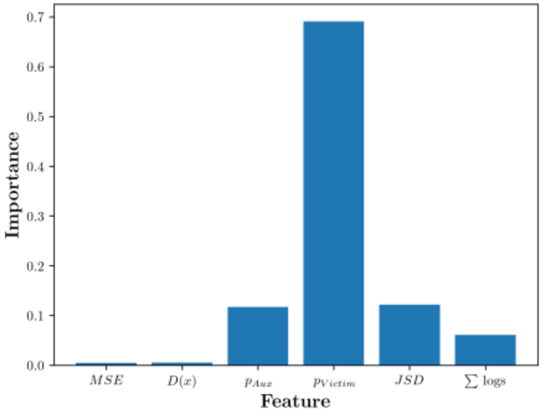
ARCANE-Diff, Tiny-ImageNet, CW (ح)



ARCANE-Diff, Tiny-ImageNet, FGSM (ز)

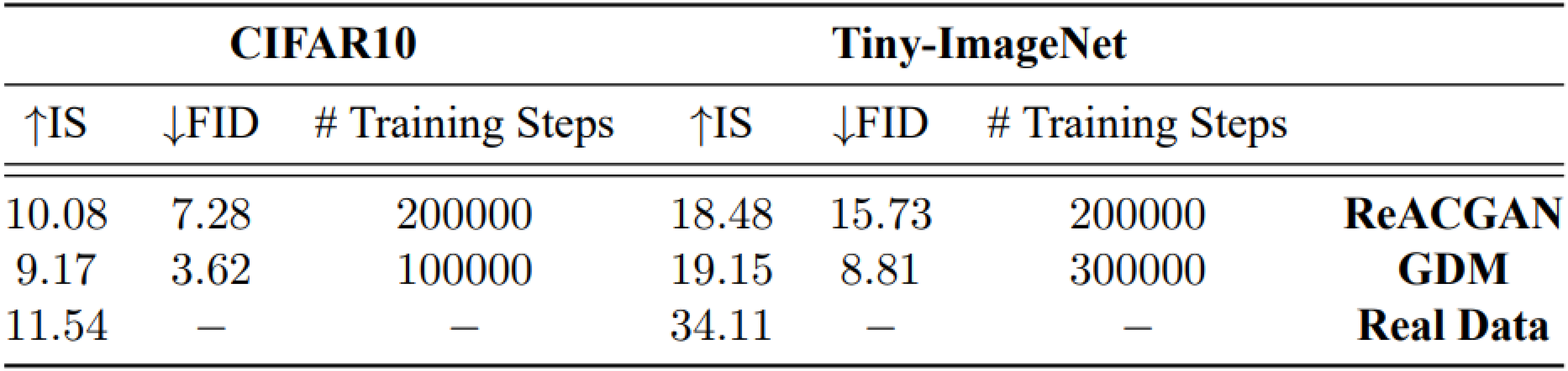


ARCANE-GAN, Tiny-ImageNet, CW (د)



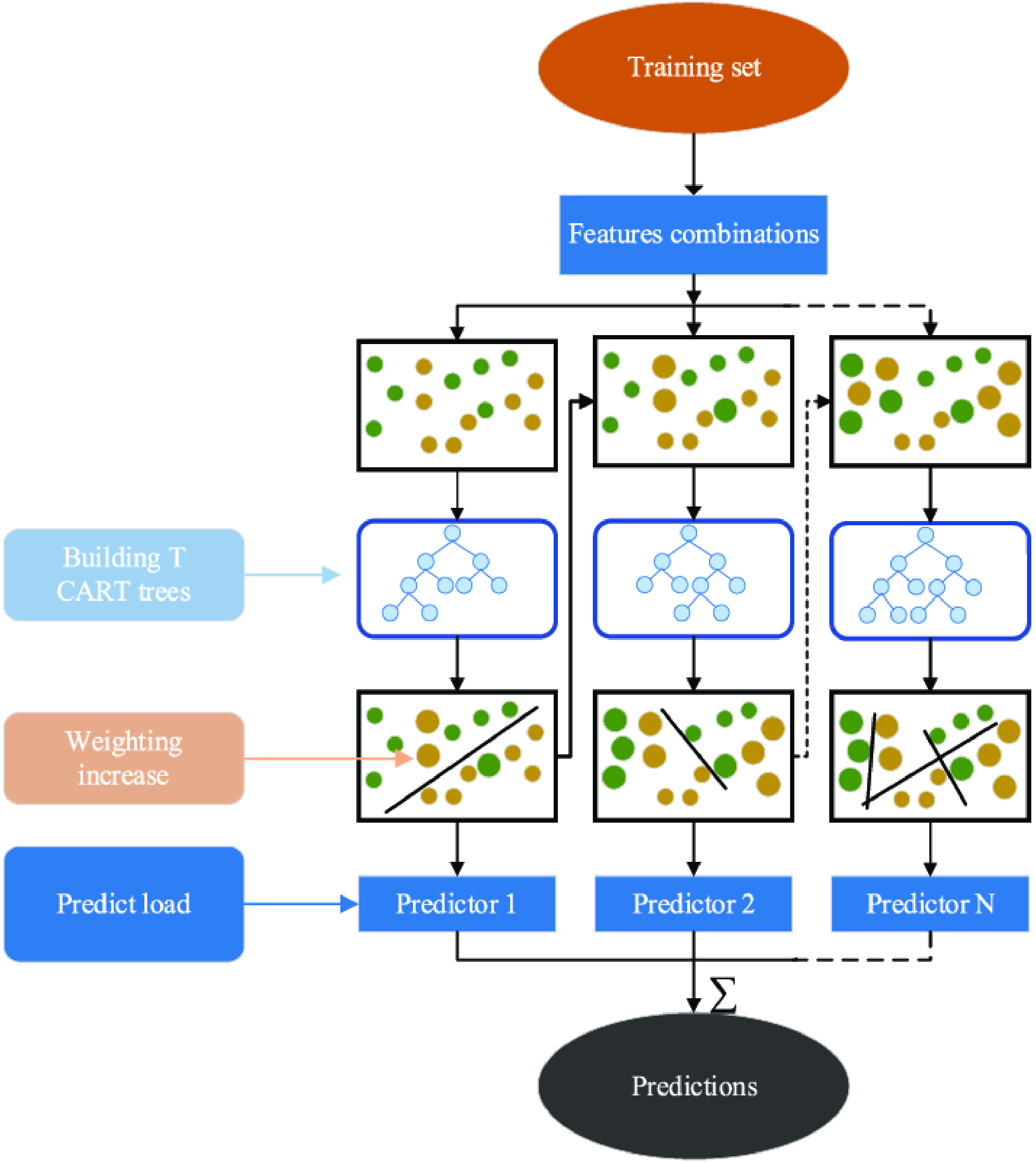
ARCANE-GAN, Tiny-ImageNet, FGSM (ج)

ADDENDUM III: GENERATION EVALUATION RESULTS



CIFAR10			Tiny-ImageNet			
↑IS	↓FID	# Training Steps	↑IS	↓FID	# Training Steps	
10.08	7.28	200000	18.48	15.73	200000	**ReACGAN**
9.17	3.62	100000	19.15	8.81	300000	**GDM**
11.54	–	–	34.11	–	–	**Real Data**

ADDENDUM IV: XGBOOST


Training set
Features combinations
Building T CART trees
Weighting increase
Predict load
Predictor 1
Predictor 2
Predictor N
Σ
Predictions