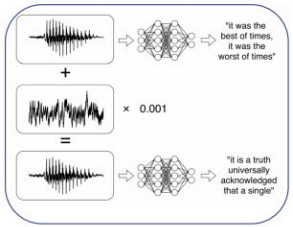
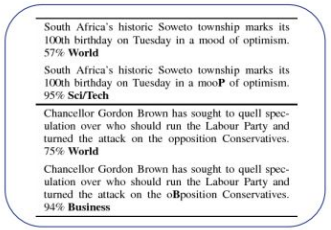
# DEFENSE USING GAN

# What are Adversarial Attacks ?

These are attacks to fool models by **designing** perturbations which when added to input seem legitimate but affect the model's performance tragically.




Adversarial Attack in Audio
([Carlini et al 2018])


South Africa's historic Soweto township marks its 100th birthday on Tuesday in a mood of optimism.
57% **World**

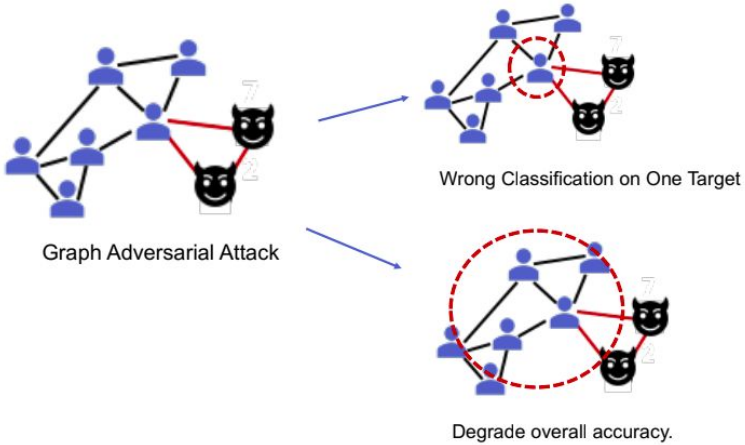
South Africa's historic Soweto township marks its 100th birthday on Tuesday in a moo**P** of optimism.
95% **Sci/Tech**

Chancellor Gordon Brown has sought to quell speculation over who should run the Labour Party and turned the attack on the opposition Conservatives.
75% **World**

Chancellor Gordon Brown has sought to quell speculation over who should run the Labour Party and turned the attack on the o**B**position Conservatives.
94% **Business**

Adversarial Attack in Text
([Ebrahimi et al 2017])

These attacks are not limited to just images, various attacks have also been tried on text models, even on graphs.



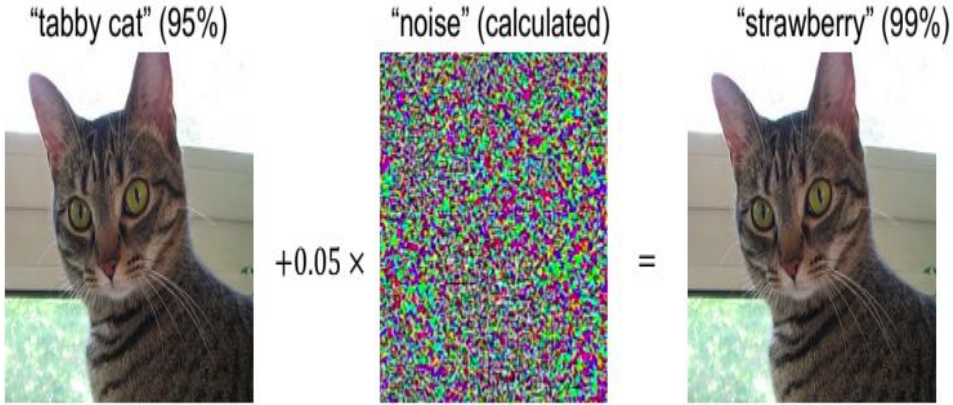
# Why do we care?

1 These Affect the model's performance very badly, for instance.

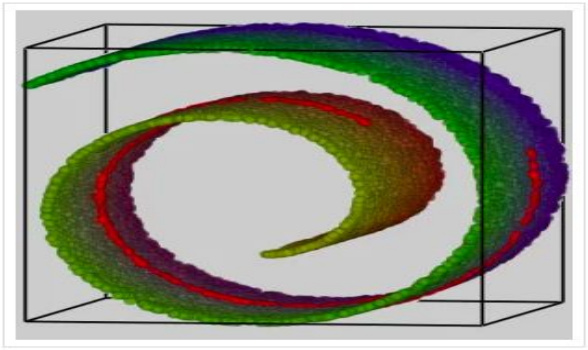
| | MNIST | | | | |
|---|---|---|---|---|---|
| | Clean | FGSM | CW | RP | BPDA |
| No Defense | 0.99 | 0.18 | 0.01 | 0.72 | - |
| Cowboy† | - | 0.78 | - | - | - |
| DefenseGAN | 0.98 | **0.83** | 0.96 | 0.92 | 0.79 |
| InvGAN | **0.99** | 0.78 | **0.99** | **0.92** | **0.87** |


2 Potentially Harmful Real Life Effects

3 Seemingly legitimate image after adding perturbation misclassified with more accuracy than accuracy of correct class

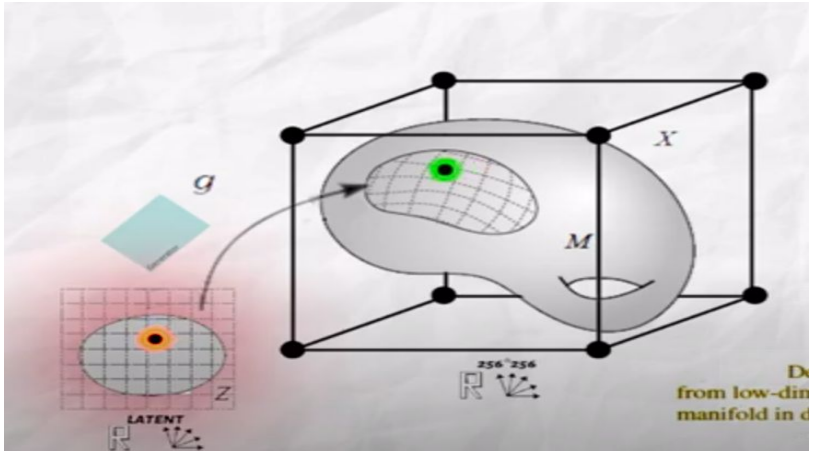


# Manifold Hypothesis



The data points of datasets although embedded in high dimensional space generally lie along a manifold like shown here.
The dimensionality of the points are artificially high but they all can be uniquely represented by moving on a low dimensional **surface/manifold** embedded in a high dimensional space.

A generator essentially tries to learn a mapping from the latent space to the manifold in this high dimensional space so that by changing z it can manoeuvre in this space



# DEFENCE GAN

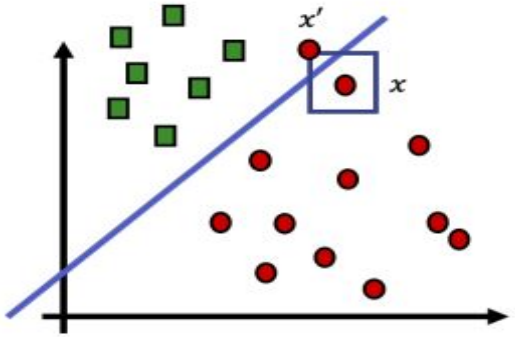
## ABSTRACT / INTRODUCTION

- A new framework which leverages the expressive capability of Gan’s to map the target model’s input data distribution in order to defend against adversarial attacks
- The assumption is that the the input data on which the target model is trained consists of clean data and it will naturally follow a different distribution than the adversarial samples, the gan is thus trained to model this clean distribution before the defense procedure starts.
- At test time we don’t directly input the test images (as they can be adversarially perturbed) directly, we use the gan to reconstruct these images which is then given to the classifier.

# Background

- White Box Attacks - White-box models assume that the attacker has complete knowledge of all the target model's parameters, i.e., network architecture and weights, as well as the details of any defense mechanism.

- Goal:
- Find $\delta$ such that $F(x+\delta) \neq y$
- Subject to $\left|\left|\delta\right|\right|_{\infty} \leq \epsilon$



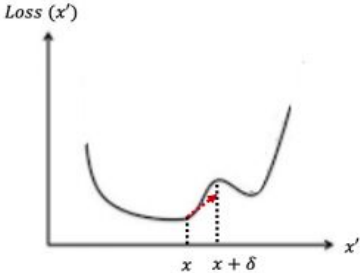
# FGSM

- Goal of Attack:
- $\max_{\delta} Loss\ (F(x+\delta;\theta), y)$
- Subject to $\left|\left|\delta\right|\right|_{\infty} \leq \epsilon$

- Finding Delta that maximizes the loss of classifier on the perturbed image.
- Delta constrained by some epsilon
- Higher the epsilon more liberty to manoeuvre

- Fast Gradient Sign Method (FGSM)
- $x + \delta = x + \epsilon \cdot \text{sign}\left(\nabla_x\ Loss\ (F(x;\theta), y)\right)$

One step of GD to maximise the loss.

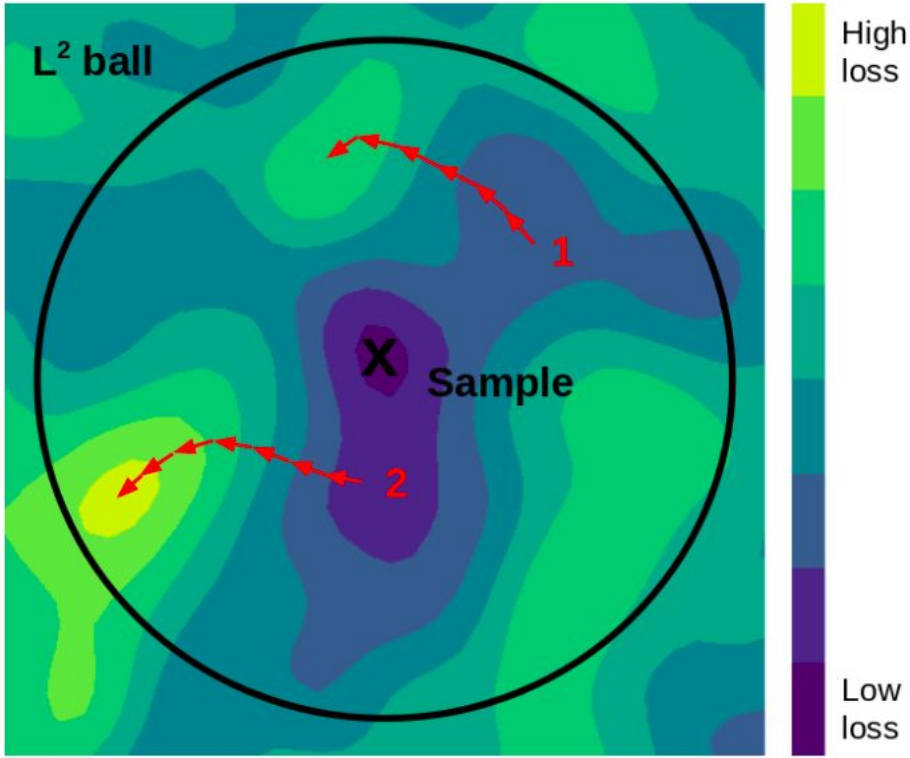


One Step Gradient Ascent to find bad $\delta$

# PGD

1 Start From a random point
2 Move in the direction of greatest loss but in a constrained way
3 Repeat 1 and 2 for some steps

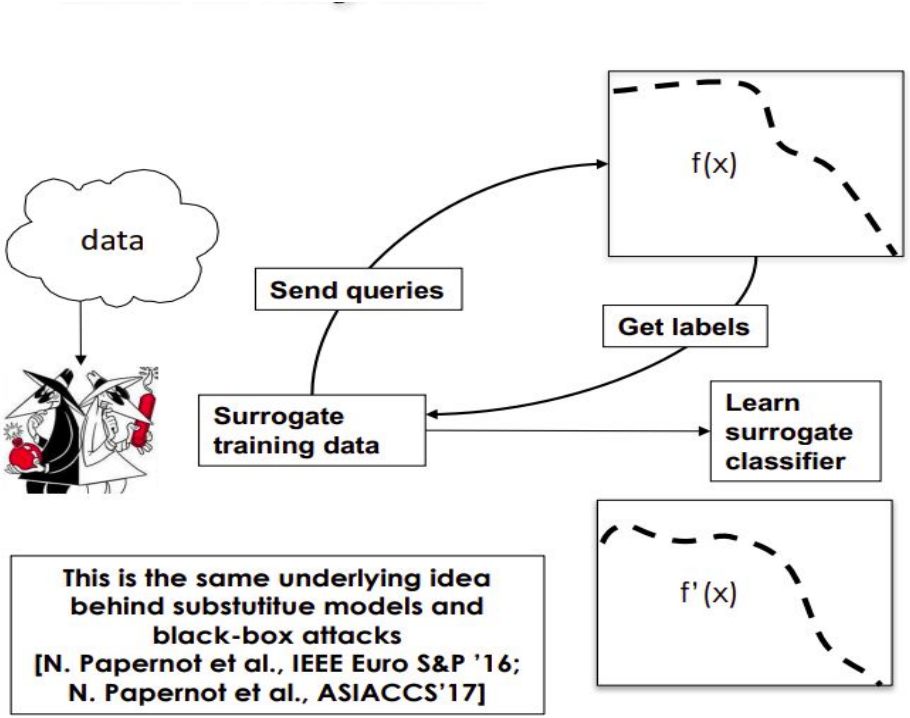
Iterative version of FGSM
More powerful




With random restarts like 1 and 2

# Black Box Attacks

Under the black-box attack model, the attacker does not have access to the classification model parameters



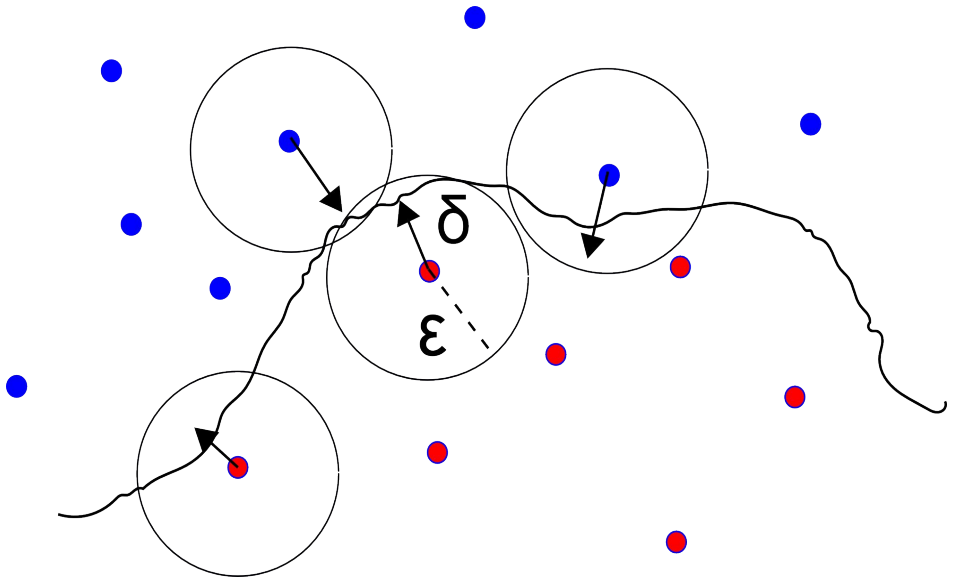
# Defense - Adversarial Training

## Goal

$$\min_{\theta} \sum_{(x,y)\sim D} \max_{||\delta|| \leq \epsilon} Loss\,(F(x+\delta;\theta), y)$$

1 First finds the adversarial examples by reconstructing the images basically attacking the model first

2 Using these to train to reform the decision boundary



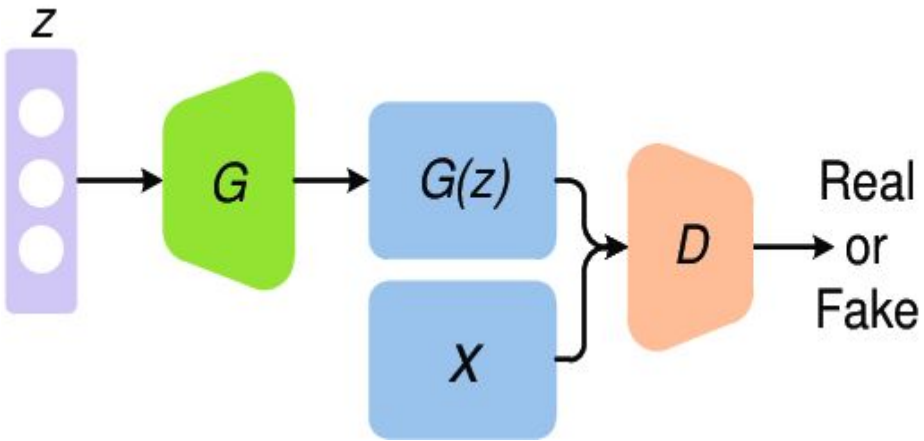
# GAN

$$\min_G \max_D V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))].$$

G wants to fool the D to believe Fake is real (same distribution) while D tries to tell them apart, both help in making the other learn from their inputs.

Generator Tries to minimize V while discriminator tries to maximize V

Trained in alternate fashion with one frozen while the other trained at each batch iteration, both are trained on each batch.



# GAN - difficult to train

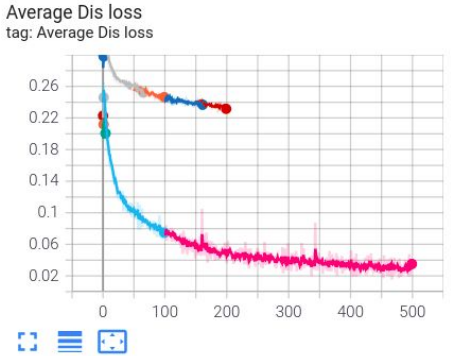
Quite Difficult to train as mostly the discriminator overpowers the generator, in this case the gradients become too sparse and the generator fails to update.

It is essential that both the models learn together the inputs of the discriminator help the generator to train and vice versa.

When a Gan is trained successfully it becomes difficult for the discriminator to distinguish between fake and real images.

Techniques like label smoothing and adding noise to D inputs helps the lower curve is after that.

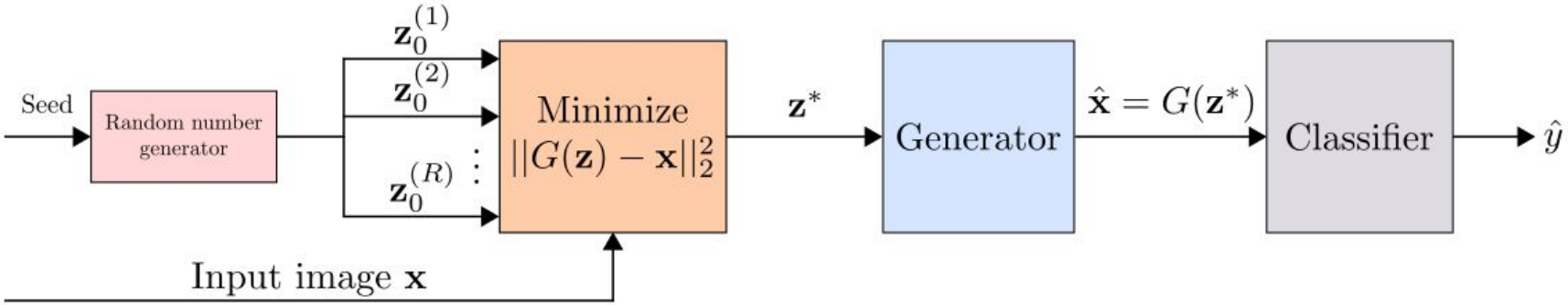
My training example




Average Gen loss

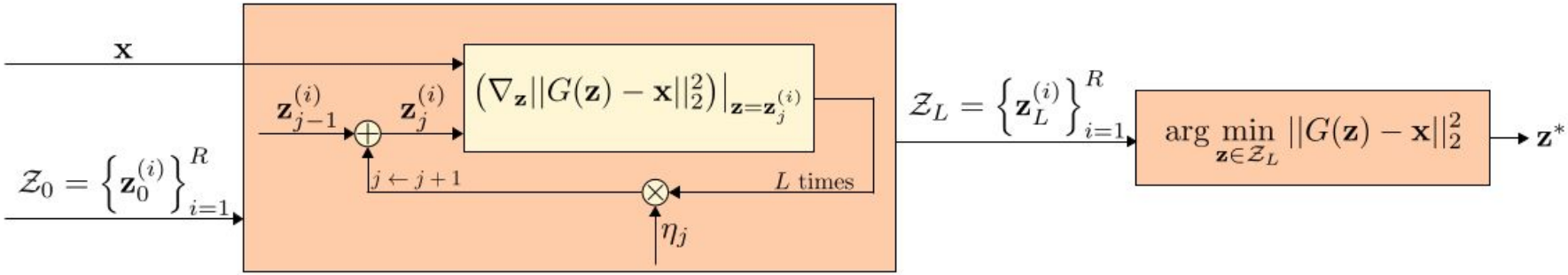

DCGAN on MNIST

# Proposed Defense




G is pretrained on the target model's input dataset
x is input image at test time
$z_i$ are the i random initialization of z vectors from the latent space (generally gaussian distribution)
classifier is the target classifier
z* is the initialization vector that has the least loss

# Proposed Defense




MSE loss is proposed to make the evaluate the reconstruction
L is the number of times we optimize z
R is the no of random vectors we take for initialization, two of the most important hyperparameters

# Experiments

Increasing L has the expected effect of improving performance when no attack is present. Interestingly, with an FGSM attack, the classification performance decreases after a certain L value. With too many GD iterations on the mean squared error

$$||G(\mathbf{z}) - (\mathbf{x} + \boldsymbol{\delta})||_2^2$$

some of the adversarial noise components are retained.

But Increasing R is always beneficial as due to the non convex nature of the MSE different R helps us sample different local minima.,

# Defense Gan as Attack Detector

The clean or unperturbed test images will lie closer to the modelled distribution of the gan while the adversarial images will be away hence we can use the MSE loss as a metric to determine if the input image is adversarially perturbed or not.

A threshold theta is decided

$$\left\|G(\mathbf{z}^*) - \mathbf{x}\right\|_2^2 \underset{\text{no attack}}{\overset{\text{attack}}{\gtrless}} \theta.$$

# Results

| Attack | Classifier Model | No Attack | No Defense | **Defense-GAN-Rec** | MagNet | Adv. Tr. $\epsilon = 0.3$ |
|---|---|---|---|---|---|---|
| FGSM $\epsilon = 0.3$ | A | 0.997 | 0.217 | **0.988** | 0.191 | 0.651 |
| | B | 0.962 | 0.022 | **0.956** | 0.082 | 0.060 |
| | C | 0.996 | 0.331 | **0.989** | 0.163 | 0.786 |
| | D | 0.992 | 0.038 | **0.980** | 0.094 | 0.732 |
| RAND+FGSM $\epsilon = 0.3, \alpha = 0.05$ | A | 0.997 | 0.179 | **0.988** | 0.171 | 0.774 |
| | B | 0.962 | 0.017 | **0.944** | 0.091 | 0.138 |
| | C | 0.996 | 0.103 | **0.985** | 0.151 | 0.907 |
| | D | 0.992 | 0.050 | **0.980** | 0.115 | 0.539 |
| CW $\ell_2$ norm | A | 0.997 | 0.141 | **0.989** | 0.038 | 0.077 |
| | B | 0.962 | 0.032 | **0.916** | 0.034 | 0.280 |
| | C | 0.996 | 0.126 | **0.989** | 0.025 | 0.031 |
| | D | 0.992 | 0.032 | **0.983** | 0.021 | 0.010 |

# MIMIC GAN

## ABSTRACT / INTRODUCTION

Claim is that Projections of input image to Gan space often fail to find the right z when the input is corrupted or perturbed.

Here corruptions refer to rotations, random crops, missing pixels and other non linear distributional shifts to input data.

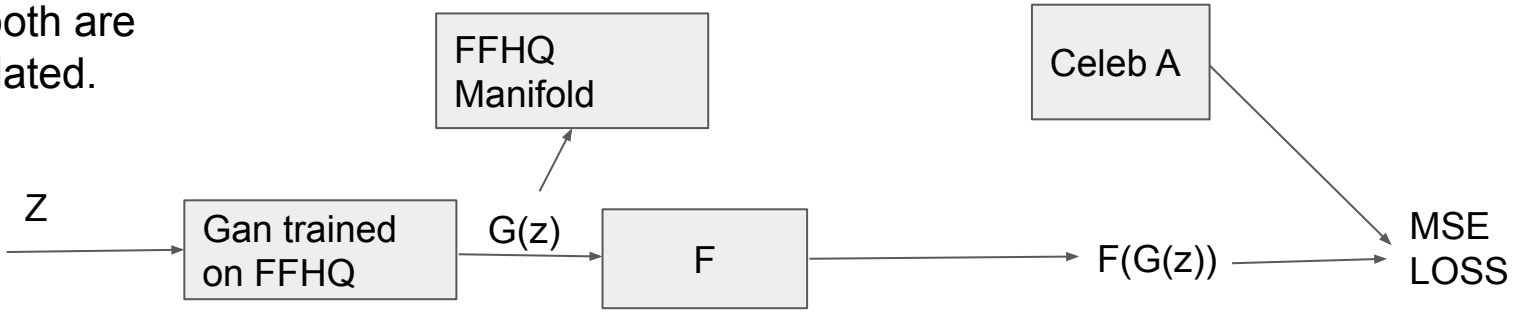
A surrogate network to mimic the corruption at test time with no additional data augmentation is proposed

Proposed Approach is also useful in domains like anomaly detection, adversarial defense

Unsupervised
Domain Adaptation

Mapping from one data distribution to another, both are closely related.

Example Mapping Celeb A to FFHQ



# Results

| **Attack** | No Defense | Cowboy [50] | Defense GAN [48] | MimicGAN *(Ours)* |
|---|---|---|---|---|
| BIM [32] | 05.60 | $66.00 \pm 3.47$ | $68.46 \pm 3.12$ | $\mathbf{72.20 \pm 3.80}$ |
| DF [42] | 06.20 | $71.43 \pm 4.72$ | $78.86 \pm 1.36$ | $\mathbf{82.40 \pm 3.81}$ |
| FGSM [21] | 11.60 | $61.60 \pm 4.99$ | $65.93 \pm 2.44$ | $\mathbf{67.60 \pm 2.83}$ |
| CWL [15] | 00.40 | $68.80 \pm 3.84$ | $71.67 \pm 3.16$ | $\mathbf{75.00 \pm 4.38}$ |
| PGDM [40] | 05.40 | $65.20 \pm 5.14$ | $69.90 \pm 2.17$ | $\mathbf{75.73 \pm 4.35}$ |
| Obfuscated [9] (attack includes GAN) | $26.93 \pm 4.29$ | $27.73 \pm 4.50$ | $27.26 \pm 2.49$ | $\mathbf{31.06 \pm 4.01}$ |

(b) Popular Adversarial Perturbations