Anti Forensic Attacks Using Generative Adversarial Networks: A New Threat

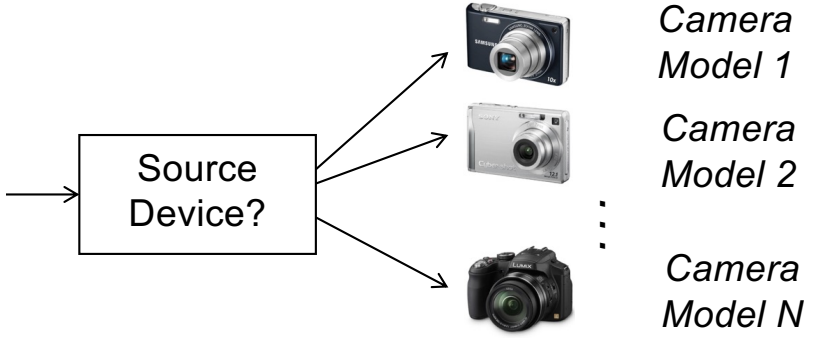
Matthew C. Stamm
Multimedia & Information Security Lab
Drexel University
mstamm@drexel.edu

Forensic Algorithms

- Deep learning has enabled dramatic advances in forensic algorithms

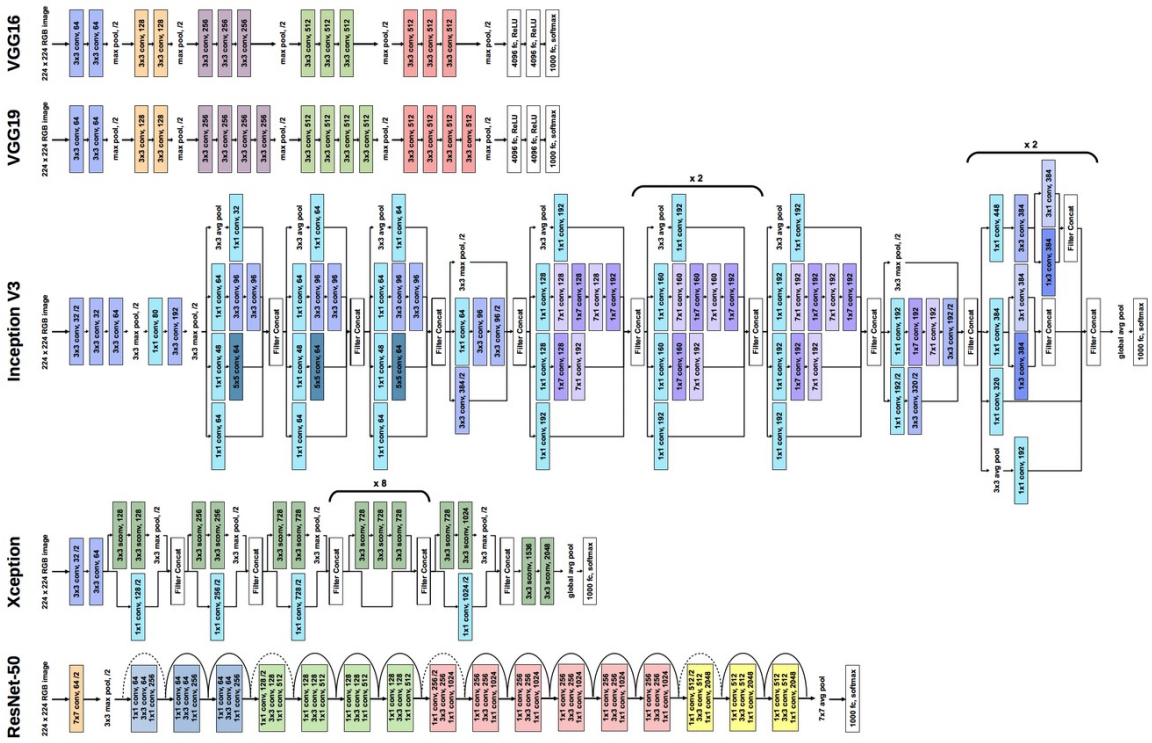
- Determine Authenticity
  - Detect fake & synthetic content
  - Detect manipulation and editing

- Identify Source
  - Camera
  - Distribution channel



Forensic Neural Networks

- Neural networks learn models of forensic traces directly from data
- Dramatically reduces design time
- Improves forensic accuracy



Anti-Forensics

- Intelligent attacker will use anti-forensic countermeasures
- Remove traces left by editing and falsification
- Falsify traces associated with source
- Difficult to attack neural networks using classical anti-forensic approaches

Deep Learning for Anti-Forensics

- Deep learning enables new anti-forensic threats
- Learned models of forensic traces can be used against forensic algorithms
- Create synthetic forensic traces using *generative adversarial networks*

Anti-Forensic Goals & Approaches

- Attack goals/requirements
  1) Fool forensic algorithm
  2) Fool human – visually convincing
  3) Don't undo intentional manipulations

- Attack approaches
  - Remove/synthesize fake forensic traces
    - Classical approach
    - GAN-based attacks
  - Exploit classifier vulnerabilities
    - Adversarial examples

Fooling Forensic Algorithms

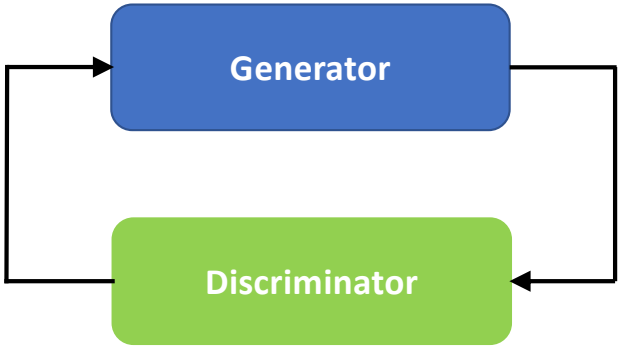
- Make forensic algorithm “useless”
  - Untargeted attacks
  - Reduce algorithm’s performance to random guess
  - Not necessary to produce wrong output all of the time

- Produce convincingly wrong decisions
  - Targeted attacks
  - Make forensic algorithm produce wrong output with high confidence
  - Harder to accomplish

Generative Adversarial Networks

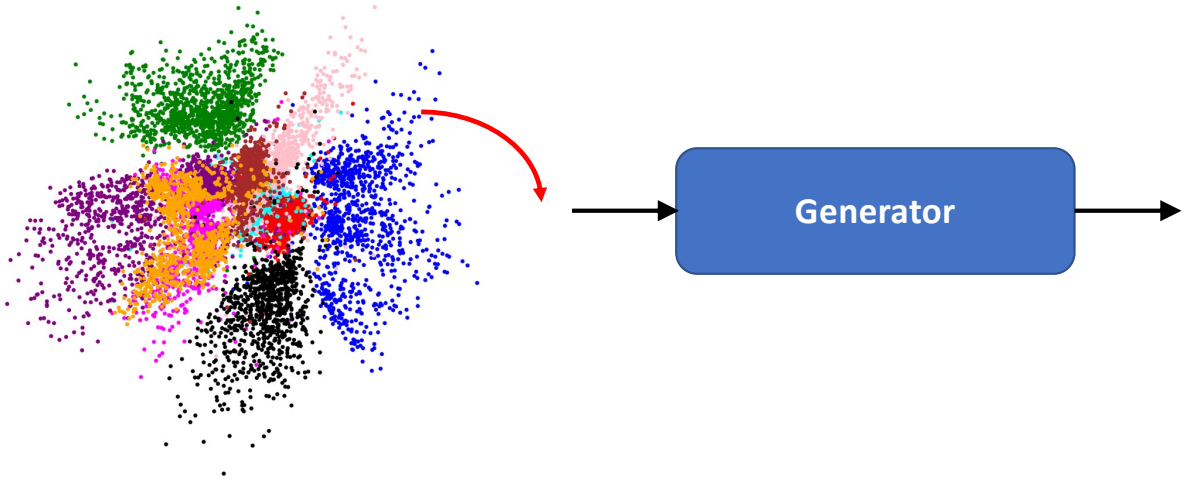
- GANs are used to create synthetic data

- Two main components:
  - Generator – creates synthetic data
  - Discriminator – detects synthetic data
- Learn through adversarial training



GAN-Generated Synthetic Data

Latent Space

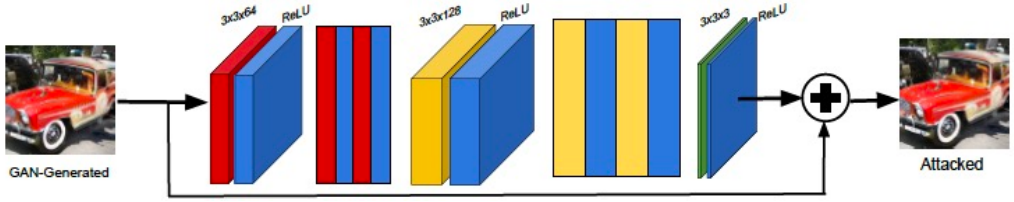


Synthetic Data

Taken from thispersondoesnotexist.com

GAN-Based Anti-Forensics

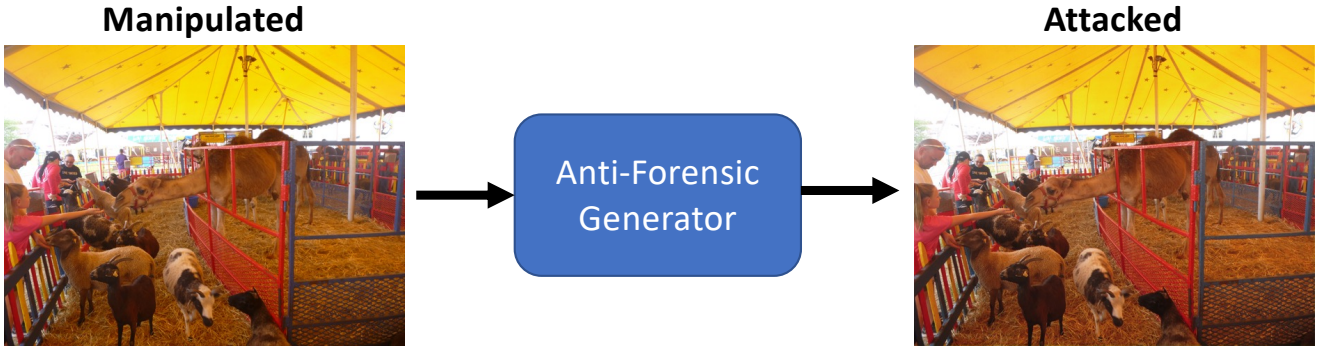
- Idea: Use GANs to generate *synthetic forensic traces*
- Attack workflow
  1. Adversarially train anti-forensic generator
  2. Save only generator
  3. Create attacked image by passing through pre-trained generator
- Generator is fully convolutional neural network



Chen et al. "MISLGAN: An anti-forensic camera model falsification framework using a generative adversarial network." IEEE ICIP (2018)

GAN-Based Anti-Forensics

- Anti-forensic generator
  - Synthesizes targeted forensic traces
  - Does not perceptually alter content
- Deploy attack by passing image through pre-trained generator



- Generator does not need to be re-trained for each image!

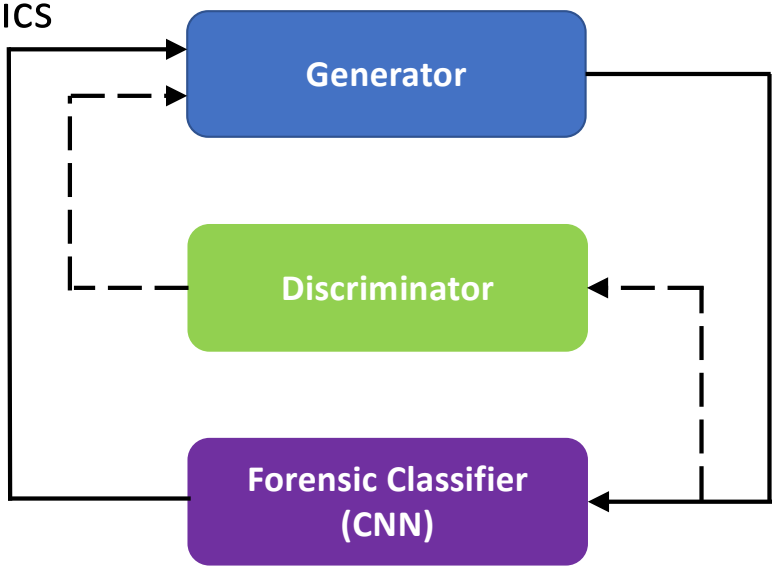
Adversarial Training

- Modify adversarial training process for anti-forensics
- Generator
  - Input – Image to attack
  - Output – Image with target synthetic traces
  - Loss function $\mathcal{L}_G = \alpha\mathcal{L}_p + \beta\mathcal{L}_c + \gamma\mathcal{L}_a$

Perceptual Loss (Distortion Penalty)

Classification Loss (Fools Forensic CNN)

Adversarial Loss (Fools Discriminator)

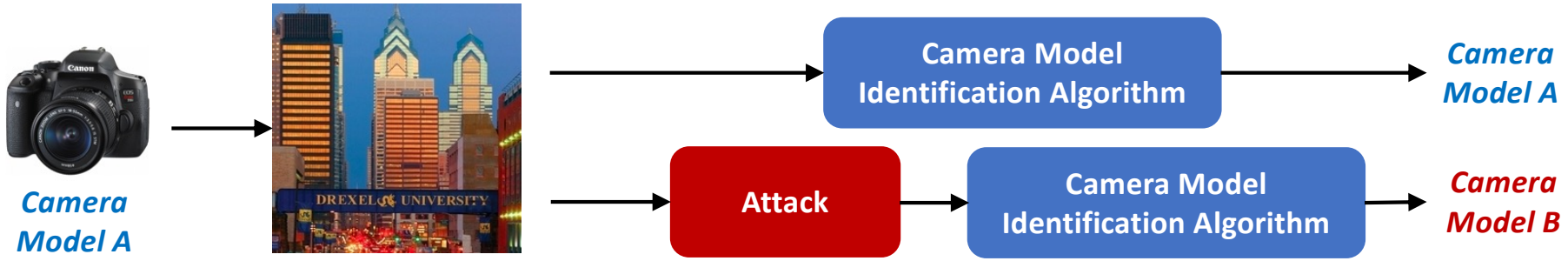


- Forensic Classifier (pre-trained)
- Discriminator
  - May not be needed

Chen et al. "Generative adversarial attacks against deep-learning-based camera model identification." IEEE TIFS (2019)

Example: Camera Model ID Falsification

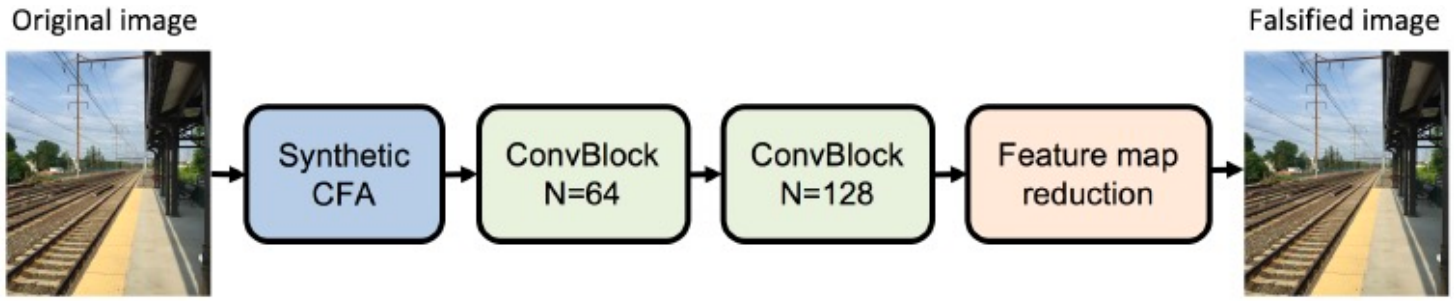
- Goal: Falsify image's source camera model



- Anti-forensic generator creates fake traces from target camera model
  - Input - Image from camera model A, Target camera model B
  - Output - Image that classifies as originating from camera model B

Chen et al. "Generative adversarial attacks against deep-learning-based camera model identification." IEEE TIFS (2019)

Generator



- Apply “software” CFA to image
  - Retains 1/3 of original pixels
- Use generator to “re-demosaic” image and falsify forensic traces
- Loss function

  $\mathcal{L}$ = α Mean Absolute Distortion + β Adversarial Loss + γ Camera Misclassification Loss

Camera Model Falsification Results

- Fools classifier with 98.5% likelihood
- Works even when the true source is *not* used to train the generator
- Human eye can't detect changes
  - PSNR > 45 dB
  - SSIM > 0.98

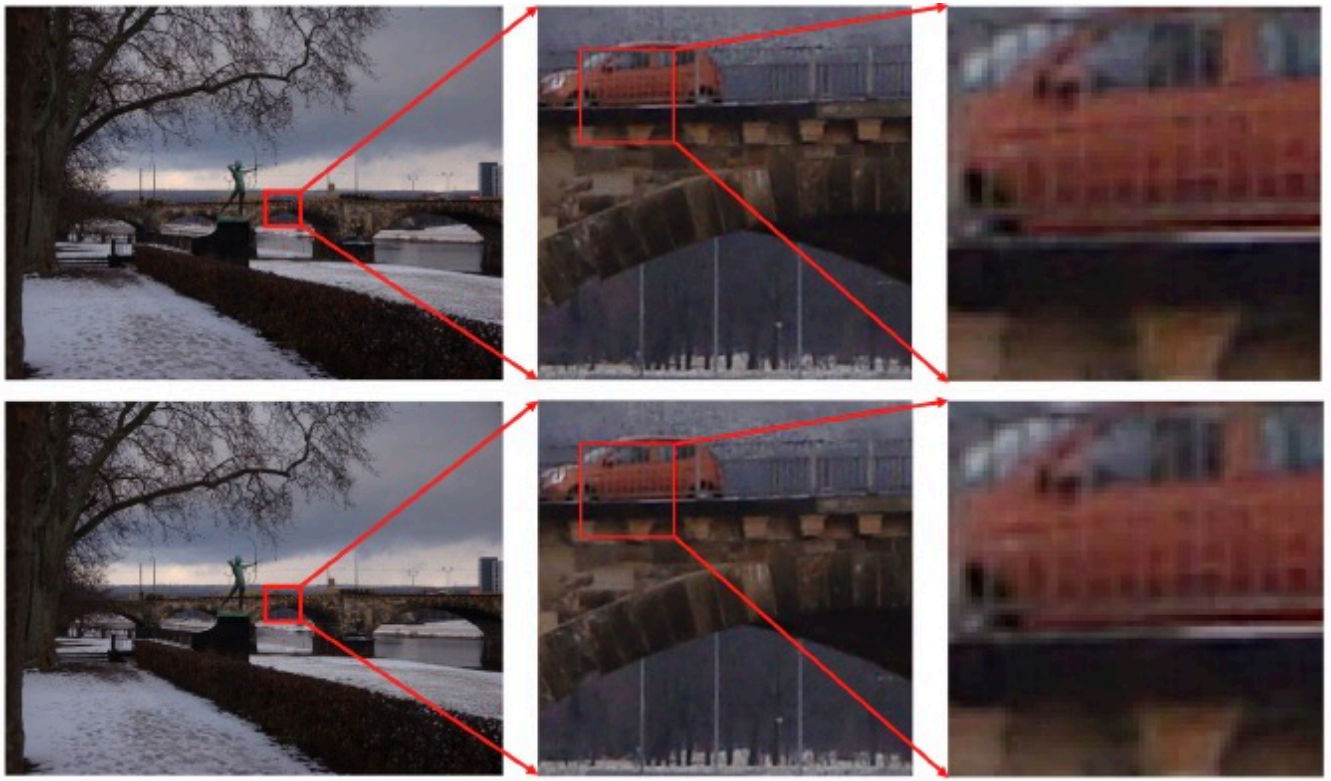
CNN Classifier		Bondi et al.[21]				Bayar & Stamm[22]				Tuama et al.[23]				Avg.
Target Model ID.		6	8	14	16	4	5	7	18	9	11	13	17	
Testing	STAR	95.70	98.35	97.14	89.19	94.51	98.08	98.00	90.91	92.97	96.36	97.62	97.37	**95.52**
	SUAR	98.69	99.42	99.56	95.59	97.27	99.43	99.22	97.61	98.02	98.89	99.43	99.36	**98.54**
	m-PSNR	45.44	45.01	45.09	44.34	45.96	46.28	46.44	44.91	45.38	46.04	45.84	46.96	**45.64**
	m-SSIM	0.987	0.988	0.986	0.981	0.988	0.989	0.989	0.983	0.989	0.988	0.988	0.990	**0.987**
Unseen	STAR	96.90	99.04	96.45	91.73	95.52	95.23	98.49	89.45	96.09	96.93	97.85	96.70	**95.87**
	m-PSNR	45.98	45.81	45.45	44.95	46.18	46.60	47.23	45.11	45.86	46.08	45.95	47.27	**46.04**
	m-SSIM	0.991	0.992	0.990	0.988	0.991	0.991	0.993	0.987	0.992	0.991	0.991	0.993	**0.991**

Chen et al. "Generative adversarial attacks against deep-learning-based camera model identification." IEEE TIFS (2019)

Source Camera Model Falsification Results

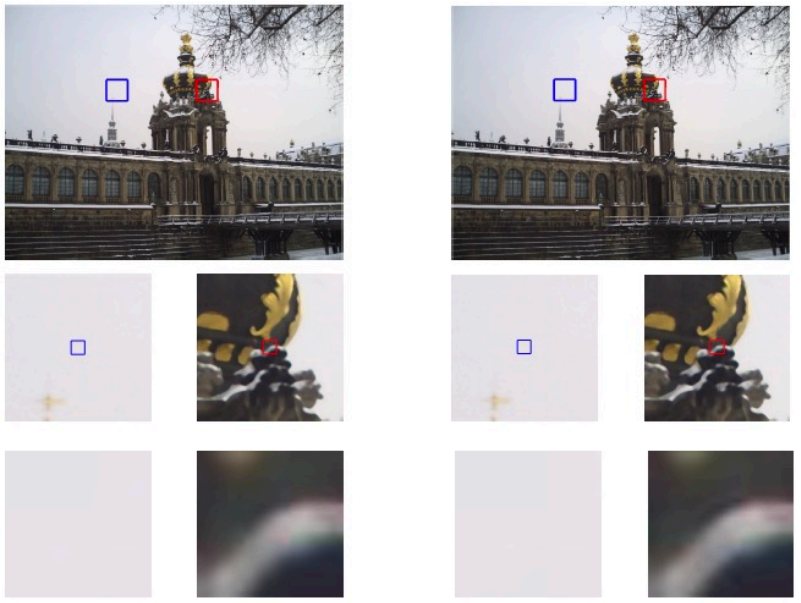
Authentic Image

Attacked Image



Example: Removing Manipulation Traces

- Attack can be adapted to remove multiple manipulation traces
- Slightly different generator
  - No synthetic CFA
- Strong results when attacker has full knowledge
- New problem related to class definitions arises



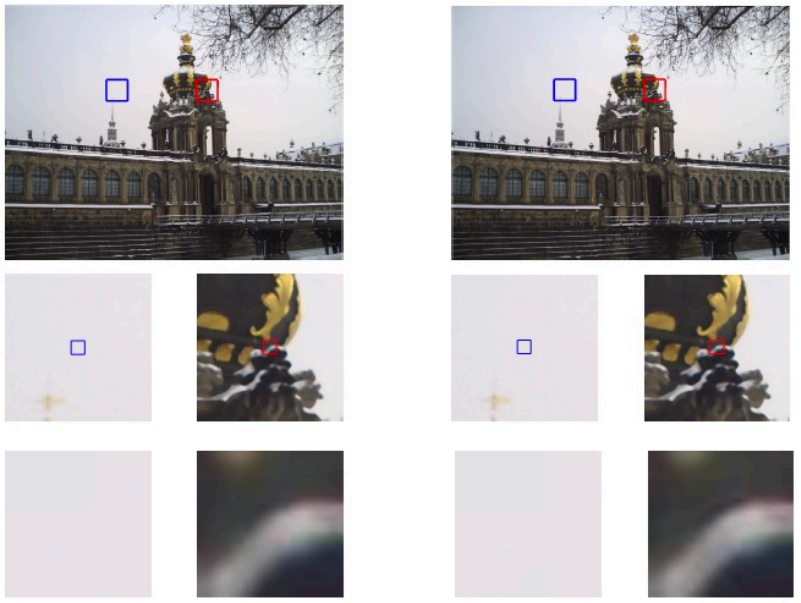
Zhao and Stamm "A Transferable Anti-Forensic Attack on Forensic CNNs Using A Generative Adversarial Network." arXiv (2021) https://arxiv.org/pdf/2101.09568.pdf

Example: Removing Manipulation Traces

- Attack can be adapted to remove multiple manipulation traces
  - Slightly different generator
- Strong results when attacker has full knowledge

CNN Architect.	ASR
MISLnet	1.00
TransferNet	1.00
PHNet	0.98
SRNet	0.93
DenseNet_BC	0.99
VGG-19	0.98
Avg.	**0.98**

- New problem related to class definitions arises



Median filtered images

Anti-forensically attacked images

Zhao and Stamm "A Transferable Anti-Forensic Attack on Forensic CNNs Using A Generative Adversarial Network." arXiv (2021) https://arxiv.org/pdf/2101.09568.pdf

Class Definition Problem

- Forensic CNNs can use different class definitions
  - Detectors, Classifiers, and Parameterizers

Baseline Results

	Successful Attack Rate		
CNN Architect.	**Manip. Detector**	**Manip. Classifier**	**Manip. Parameterizer**
MISLnet	0.55	0.95	0.84
TransferNet	0.81	0.84	0.98
PHNet	0.90	0.97	0.94
SRNet	0.88	0.90	0.82
DenseNet	0.90	0.94	0.94
VGG-19	0.71	0.97	0.96
Average	0.79	0.93	0.91

Transfer Results

	Successful Attack Rate	
CNN Architect.	**Manip. Classifier**	**Manip. Parameterizer**
MISLnet	0.004	0.045
TransferNet	0.008	0.005
PHNet	0.275	0.120
SRNet	0.420	0.000
DenseNet	0.005	0.010
VGG-19	0.020	0.090
Average	0.122	0.045

- Attacks don’t transfer well between class definitions!
  - Observed similar results for adversarial examples

Zhao and Stamm "The Effect Of Class Definitions On The Transferability Of Adversarial Attacks Against Forensic CNNs” Electronic Imaging (2020)

Attacker Knowledge Level

- Amount of information available to attacker has *strong* effect on attack design & feasibility

- Three knowledge scenarios
  - White-Box (Perfect Knowledge)
  - Black-Box (Limited Knowledge)
  - Zero Knowledge

White Box Attack

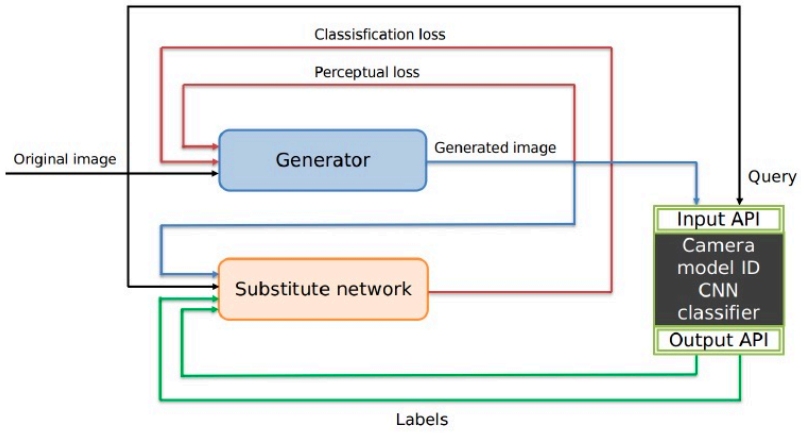
- Attacker has full knowledge of detection algorithm
  - Access to: Full algorithm details, Code/software implementation, Pre-trained detector, Detector training data
- Can directly train attack against detector
- Important Info:
  - Well studied in literature
  - Best case attacker, worst case for detector
  - Least realistic scenario

Black Box Attack

- Attacker doesn't have full access to detector
  - Can't see: Full algorithm details, code, possibly training data
- Attacker has *black box* access to detector
  - Can query input/output relationship
  - Provide images to detector and observe output
  - Leverage this information to build an attack
- Important Info
  - Studied in literature (research is ongoing)
  - More challenging for attacker, but still feasible
  - More realistic scenario

Making Black Box Attacks

- Query *victim* forensic neural network and observes output
- Train *substitute network* to reproduce the same decisions
- Train attack against substitute network
- Deploy trained attack against victim forensic neural network



Example: Black Box Camera Model ID Attack

- Use generic substitute architecture (e.g. DenseNet)
- Maintains high attack success rate and visual quality

CNN Classifier		Bondi et al.[21]				Bayar & Stamm[22]				Tuama et al.[23]				Avg.
Target Model ID		1	4	8	10	7	14	16	17	2	3	6	15	
Testing	STAR	72.85	84.23	94.08	91.84	96.78	90.07	93.16	96.71	93.59	82.35	87.93	87.76	**89.28**
	SUAR	94.89	94.72	98.07	97.92	98.61	97.34	97.47	98.69	98.55	95.86	96.80	97.88	**97.23**
	m-PSNR	45.11	44.78	46.70	45.91	46.35	46.18	46.19	46.68	46.66	46.28	46.16	44.94	**46.00**
	m-SSIM	0.984	0.981	0.989	0.988	0.989	0.988	0.988	0.990	0.989	0.989	0.988	0.989	**0.988**
Unseen	STAR	85.40	85.56	96.36	87.01	97.20	88.41	91.98	96.80	92.93	80.79	88.17	93.34	**90.33**
	m-PSNR	46.14	45.01	47.50	46.08	47.06	46.74	46.70	47.21	46.83	46.91	46.65	45.52	**46.53**
	m-SSIM	0.990	0.986	0.993	0.991	0.992	0.992	0.991	0.993	0.992	0.993	0.992	0.992	**0.991**

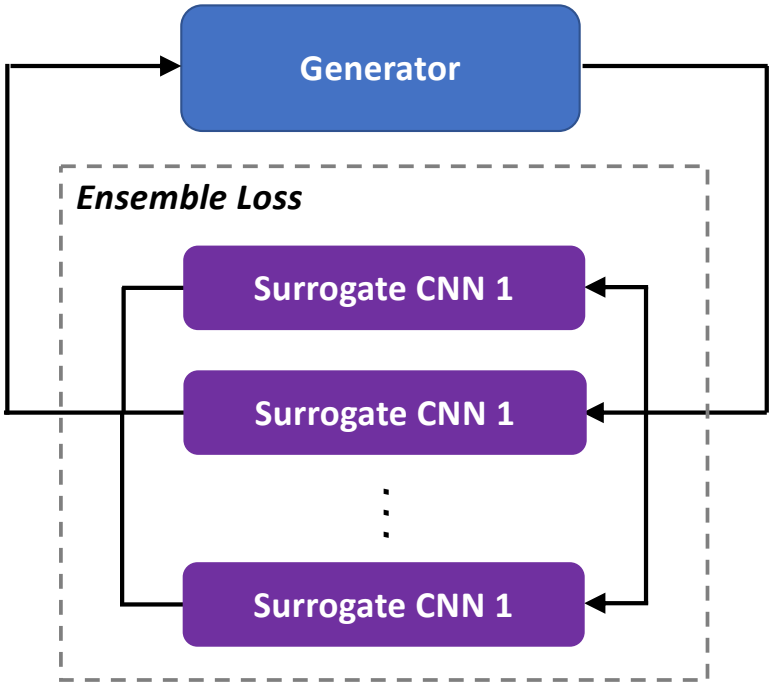
Chen et al. "Generative adversarial attacks against deep-learning-based camera model identification." IEEE TIFS (2019)

Zero Knowledge Attack

- Attacker only knows that forensic algorithm exists
  - Can't see: Full algorithm details, code, possibly training data, software implementation
  - Can't query algorithm like a black box
- Attacker relies entirely on transferability
  - Attack designed against stand-in algorithm/neural network
  - Hope that attack also works against unseen network
- Important Info
  - Least studied
  - Most realistic scenario
  - Most challenging for attacker

Achieving Transferability

- Attacker creates their own set of "surrogate" classifiers
- Train generator to fool ensemble of surrogate classifiers
- Generator synthesizes traces in intersection of surrogate decision regions
- Unseen detector likely has overlapping decision region



Zhao and Stamm "Making GAN-Generated Images Difficult To Spot: A New Attack Against Synthetic Image Detectors." arXiv (2021) https://arxiv.org/abs/2104.12069

Example: Fooling Synthetic Image Detectors

- Train anti-forensic generator to make GAN-generated images appear "real"

- White box performance

CNNs	StarGAN-v2	StyleGAN	StyleGAN2	Avg.	M_PSNR	M_SSIM
Xception	1.0000	1.0000	1.0000	**1.0000**	45.52	0.9875
ResNet-50	0.7590	0.8770	0.8010	**0.8123**	35.97	0.9578
DenseNet	0.9385	0.9770	0.9970	**0.9708**	54.28	0.9997
MISLNet	0.9965	0.9905	0.9950	**0.9940**	51.28	0.9925
PHNet	1.0000	0.8497	0.9985	**0.9494**	41.85	0.9753
SRNet	0.8080	0.8855	0.9480	**0.8805**	50.97	0.9922
ImageCNN	0.9854	0.9935	0.8505	**0.9431**	53.64	0.9928
CamID CNN	0.9285	0.9120	0.9870	**0.9425**	58.10	0.9988
Avg.	**0.9270**	**0.9357**	**0.9471**	**0.9366**	**48.95**	**0.9871**

Zhao and Stamm "Making GAN-Generated Images Difficult To Spot: A New Attack Against Synthetic Image Detectors." arXiv (2021) https://arxiv.org/abs/2104.12069

Example: Fooling Synthetic Image Detectors

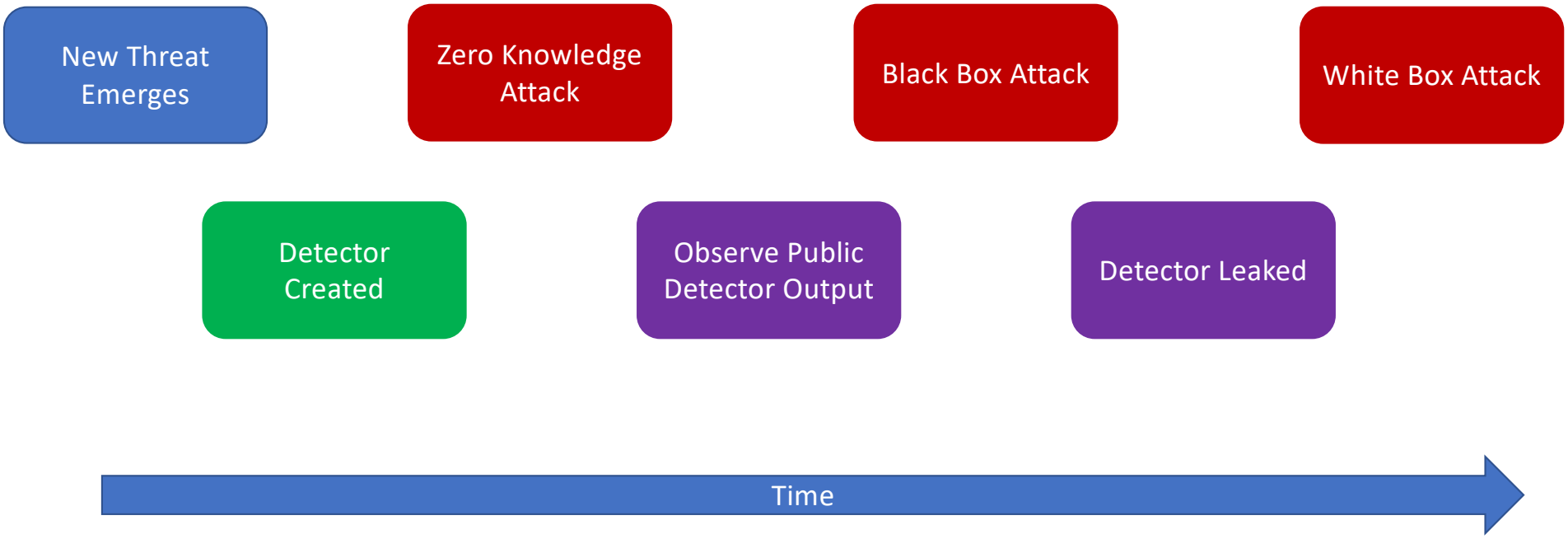
- Train using ensemble of surrogate forensic CNNs
- Zero knowledge performance

CNNs	StarGAN-v2	StyleGAN	StyleGAN2	Avg.	M_PSNR	M_SSIM
Xception	0.7855	0.9565	0.9900	**0.9107**	37.93	0.9766
ResNet-50	0.0695	0.3795	0.2815	**0.2435**	42.90	0.9765
DenseNet	0.8345	0.8325	0.9520	**0.8730**	38.79	0.9480
MISLNet	0.1250	0.2340	0.8350	**0.3980**	38.14	0.9742
PHNet	0.9925	0.7625	0.9935	**0.9162**	41.69	0.9704
SRNet	0.8495	0.8675	0.9465	**0.8878**	40.16	0.9709
Image CNN	0.8360	0.9595	0.8065	**0.8673**	41.27	0.9590
CamID CNN	0.7990	0.9480	0.9880	**0.9117**	42.77	0.9703
Avg.	**0.6614**	**0.7425**	**0.8491**	**0.7510**	**40.46**	**0.9682**

- Significant transferability!

Zhao and Stamm "Making GAN-Generated Images Difficult To Spot: A New Attack Against Synthetic Image Detectors." arXiv (2021) https://arxiv.org/abs/2104.12069

Threat Evolution Over Time



DARPA Hackathon 2 Anti-Forensic Challenge

Challenge - Detect GAN-generated images under anti-forensic attack

- Round 1: Drexel launches zero knowledge attacks
- Round 2: Drexel launches black box attack
  - Teams provide classifier outputs for ~ 2,000 query images
- Round 3: Teams deploy defensive measures
  - Drexel provides ~5,000 training examples of attacked images

DARPA Hackathon 2 Anti-Forensic Challenge

- Adversarial Examples
  - Carlini Wagner (CW)
  - Projected Gradient Descent (PGD)
  - Fast Gradient Sign Method (FGSM)

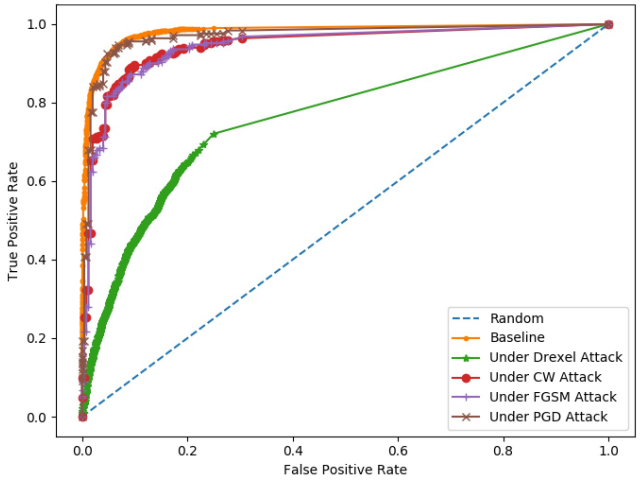
- Drexel's GAN-Based Anti-Forensic attack
  - Use adversarial generator to create synthetic "real" forensic traces
  - arXiv version (attack has improved some beyond this):

    Making GAN-Generated Images Difficult To Spot: A New Attack Against Synthetic Image Detectors

    https://arxiv.org/abs/2104.12069

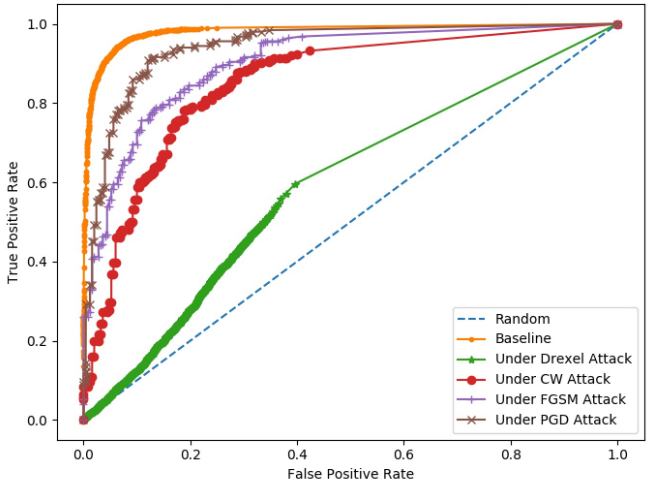
Sample Results - Team 1

Zero Knowledge Attack



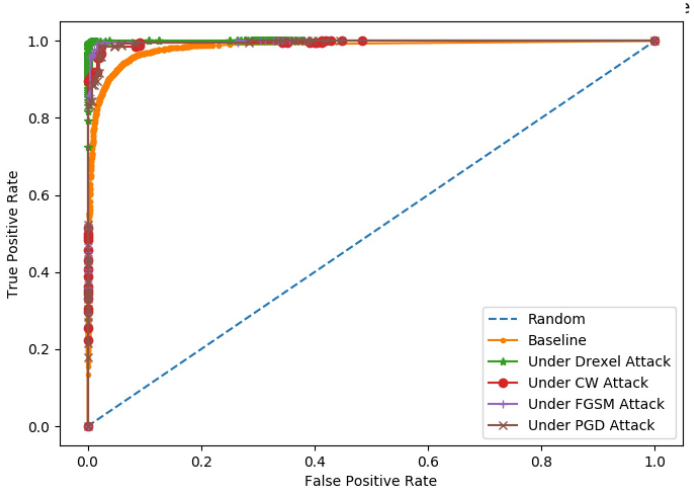
Drexel Attack, P_D at P_{FA}=0.1: **0.448**

Black Box Attack



Drexel Attack, P_D at P_{FA} =0.1: **0.1244**

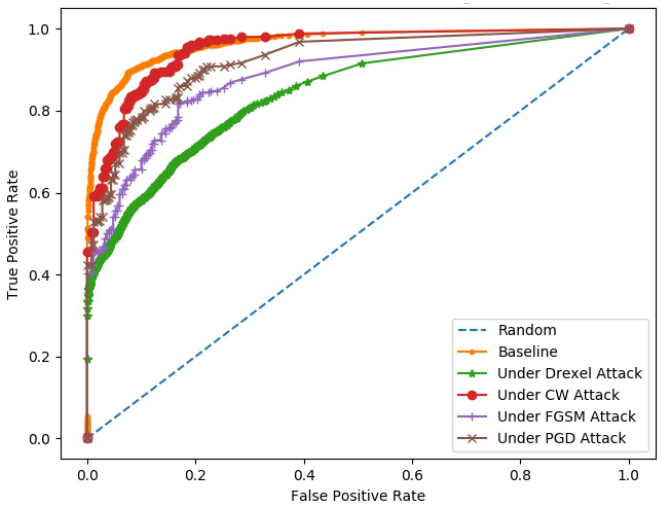
Black Box Attack With Defenses



PGD Attack, P_D at P_{FA} =0.1: **0.996**

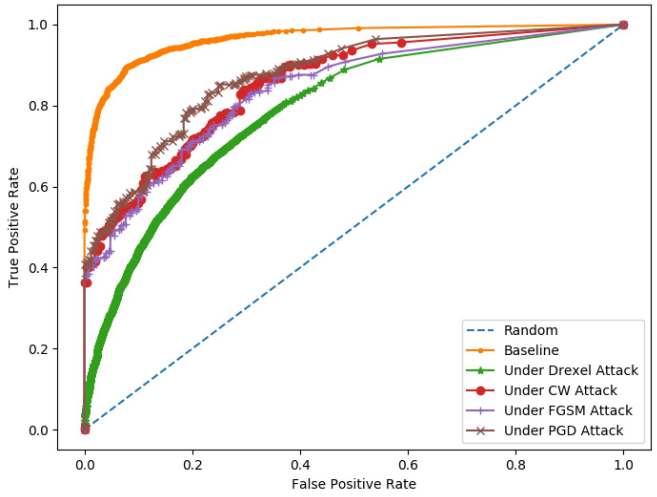
Sample Results - Team 2

Zero Knowledge Attack



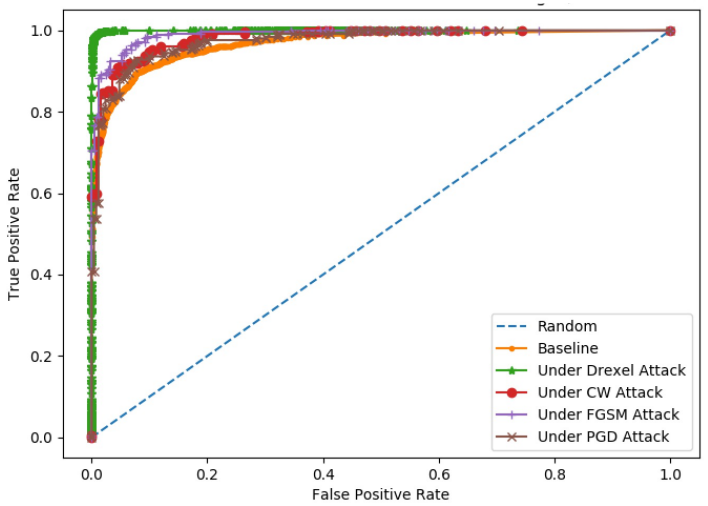
Drexel Attack, P_D at P_{FA}=0.1: **0.5816**

Black Box Attack



Drexel Attack, P_D at P_{FA} =0.1: **0.439**

Black Box Attack With Defenses



PGD Attack, P_D at P_{FA} =0.1: **0.928**

Summary

- Anti-forensic attacks can be designed to fool forensic neural networks
- GANs can be used to synthesize realistic forensic traces
- GAN-based attacks can
  - Falsify an image's source
  - Hide traces of editing
  - Disguise synthetic images
- Transferable attacks can be achieved through special training
- Further research needed to create defenses

Anti Forensic Attacks Using Generative Adversarial Networks: A New Threat

Matthew C. Stamm
Multimedia & Information Security Lab
Drexel University
mstamm@drexel.edu