Faking Sensor Noise Information

Presented by Justin Chang on 5/18/2022
Master's Defense
Advisor: Dr. Chris Pollett
Committee: Dr. Mark Stamp & Dr. Robert Chun

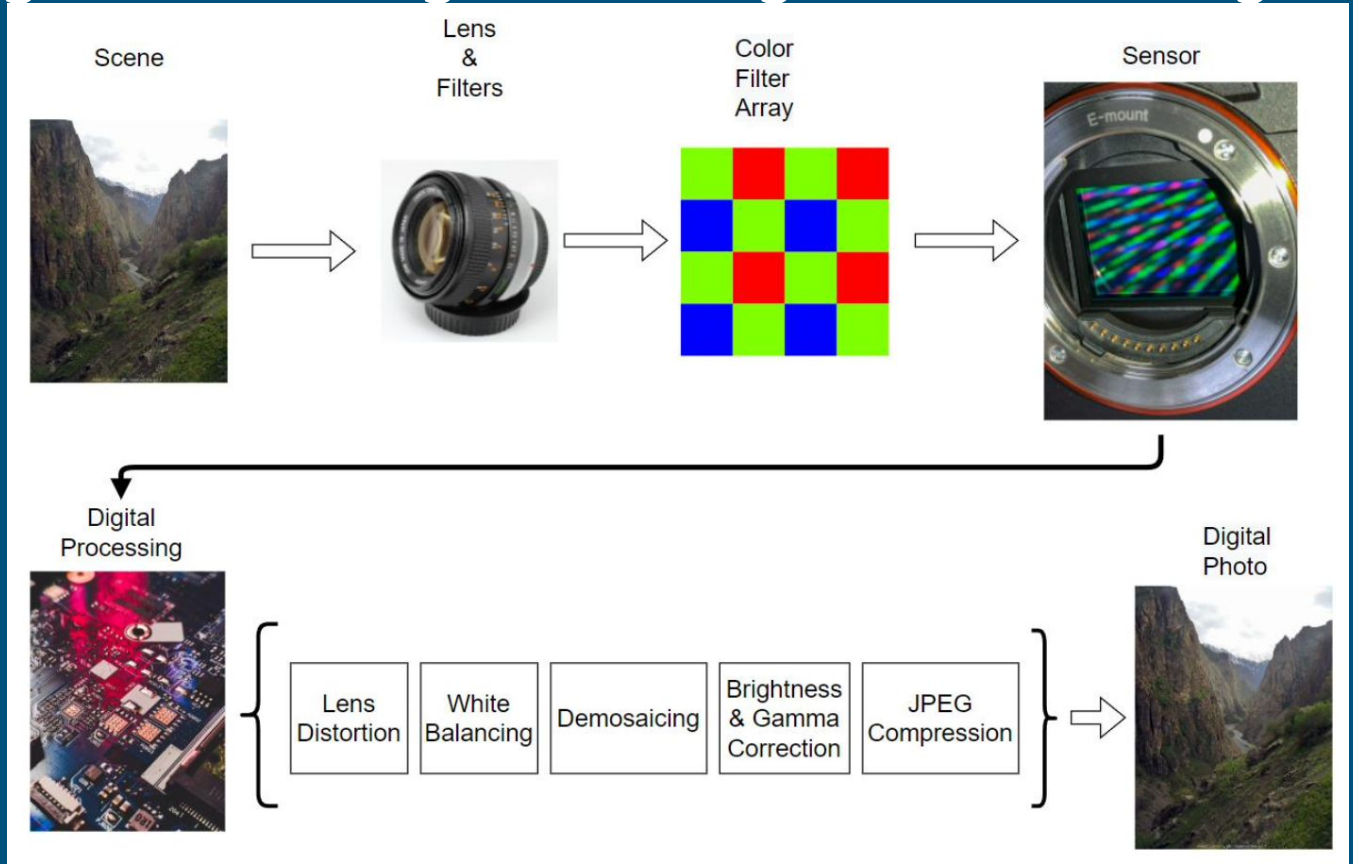
Outline

- Project goals
- Background
- Related works
- Dataset
- Model Design and Implementation
- Experiments and Results
- Recipes for Success
- Future work
- References

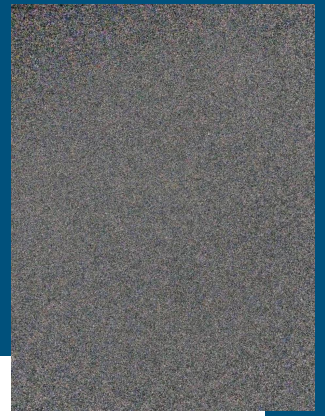
Project Goals

- Determine sensor noise patterns for camera models
- Spoof individual camera model sensor noise patterns
- Model sensor noise for individual cameras

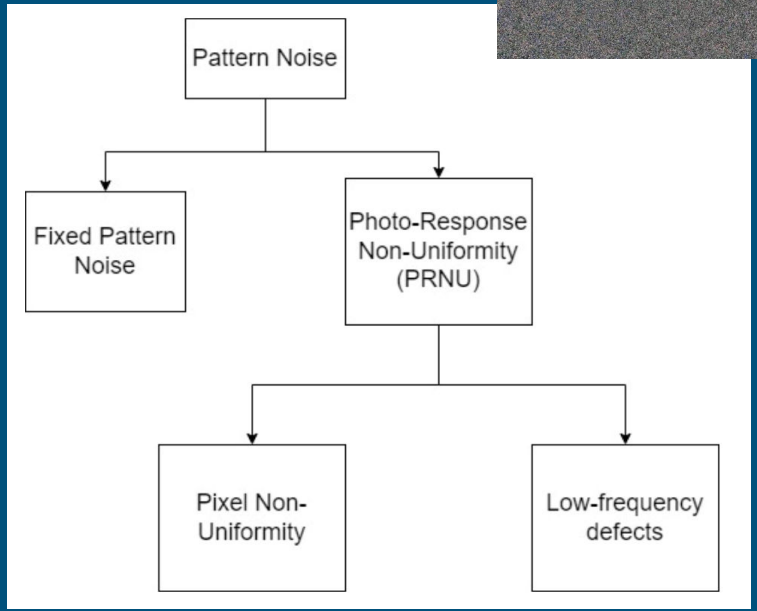
Background - Digital Image Processing



Background - What is a noiseprint?

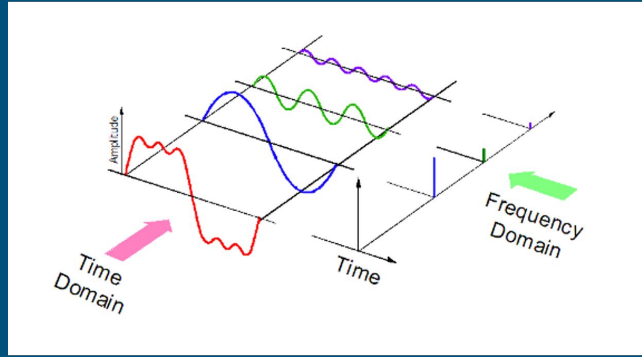


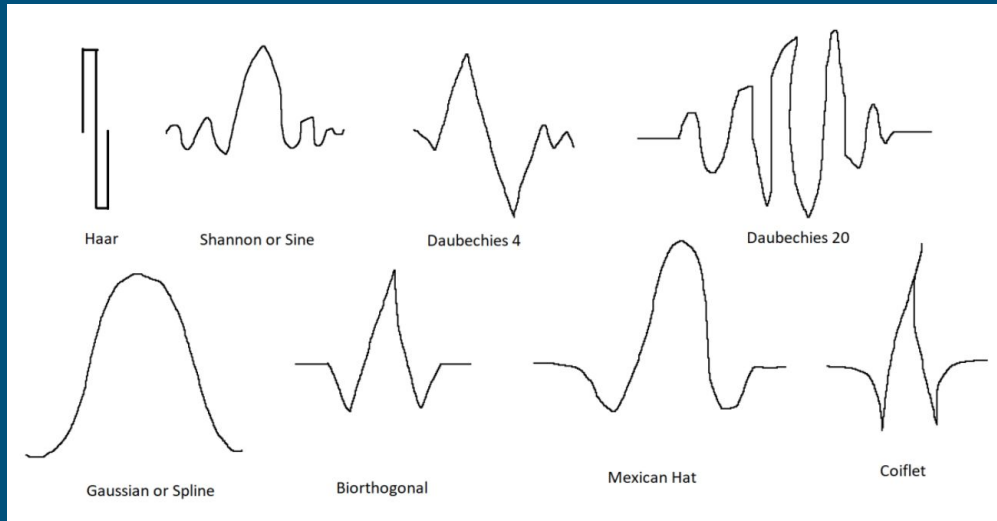
- Sensor noise pattern (“noiseprint”)
- Fixed Pattern Noise (FPN): Dark currents
- Photo-Response Non-Uniformity (PRNU): Pixel sensitivity
- Different qualities of noise
- Scene invariant noise is better



Background: Denoising Filters

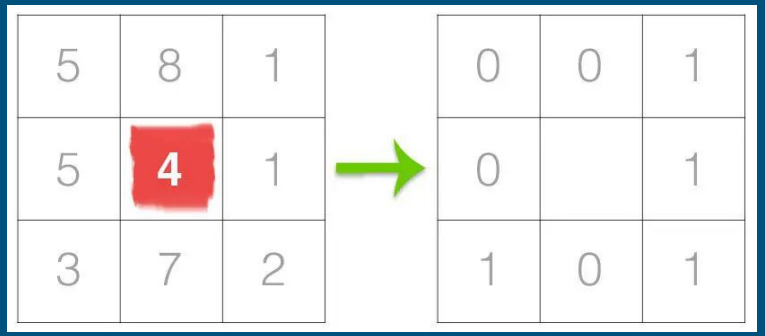
- Signal Waveform Decomposition
- Reconstruction
- Taylor Series
- Fourier Transform
- Wavelet Decomposition

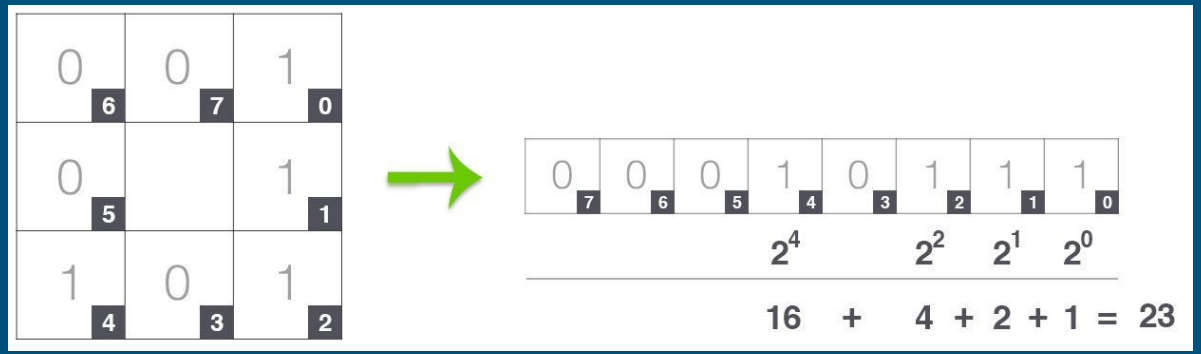


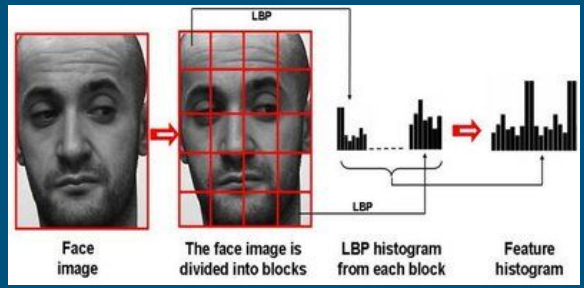


Background - Local Binary Patterns

- Texture Descriptor
- Compact and fast
- Sliding window

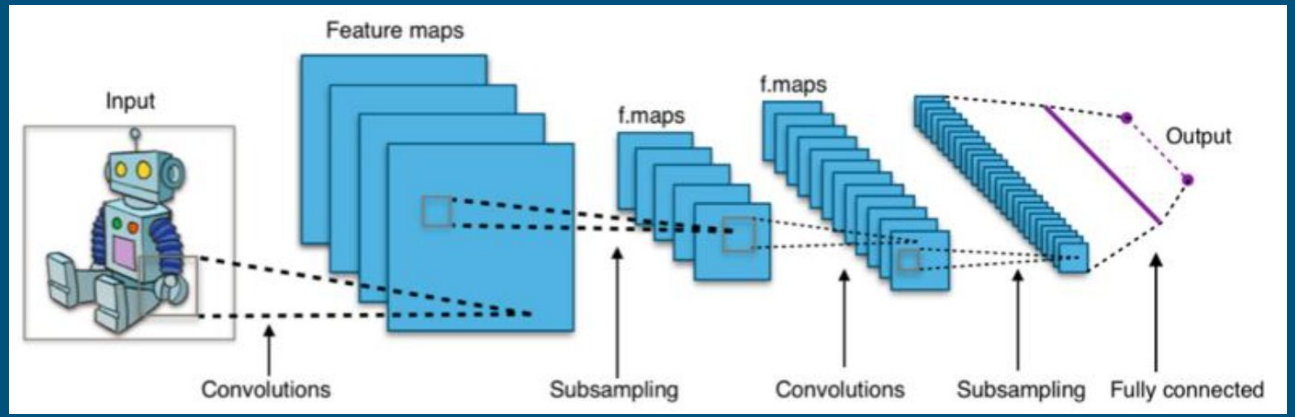






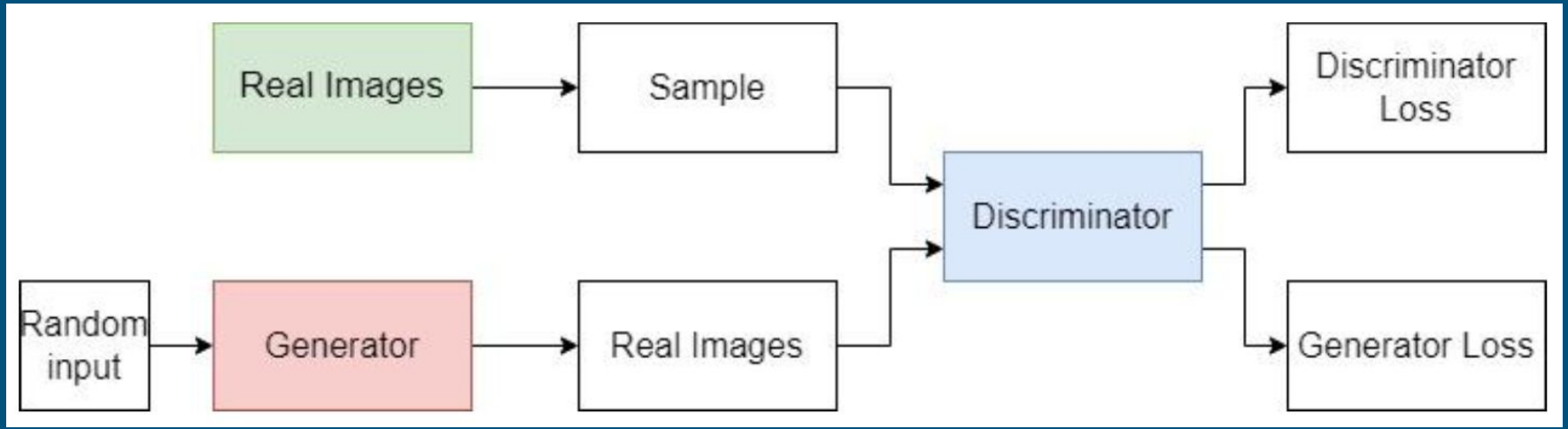
Background - Convolutional Neural Networks (CNNs)

- Sliding window
- Self Learning Filters
- Localized features



Background - Generative Adversarial Networks (GANs)

- Two competing models
- Real or Fake?



Related Works

- Sensor Pattern Noise discovery [Lucas et al.]
- Insertion of camera noiseprints onto artificially generated images [Cozzolino et al.]
- Arbitrary attacks on discriminative networks [Chen et al.]
- Tests in robustness of camera identification [Samaras et al.]
- PCA based denoising of noiseprints [Li et al.]

Dataset

- Simulate an attack on victims personal phone
- 3 Different Cameras (1 iPhone X and 2 Samsung Galaxy S8s)
- Old images and new images
- Brightness
- Indoors/Outdoors
- Temperatures
- Humidity
- Default settings

Dataset

- GAN Training: 240 images per class
- Classifier Training: 275 images for classifier, 30 test set

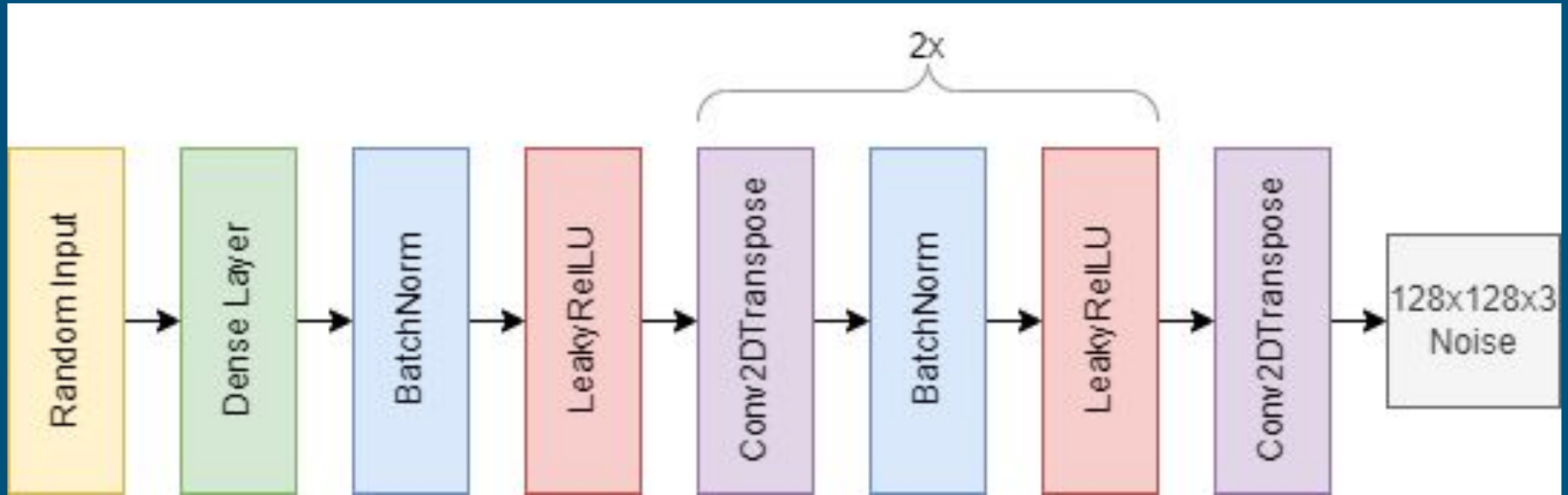
Implementation: Preprocessing GAN

- Center crop of 128x128x3 from dataset
- Multiple Regional Crops are better but increase training time
- Noise = Original Image - Denoised Image
- Arrays of sensor noise are small float values
- Saved as npz files readable by NumPy

GAN the model

Generator:

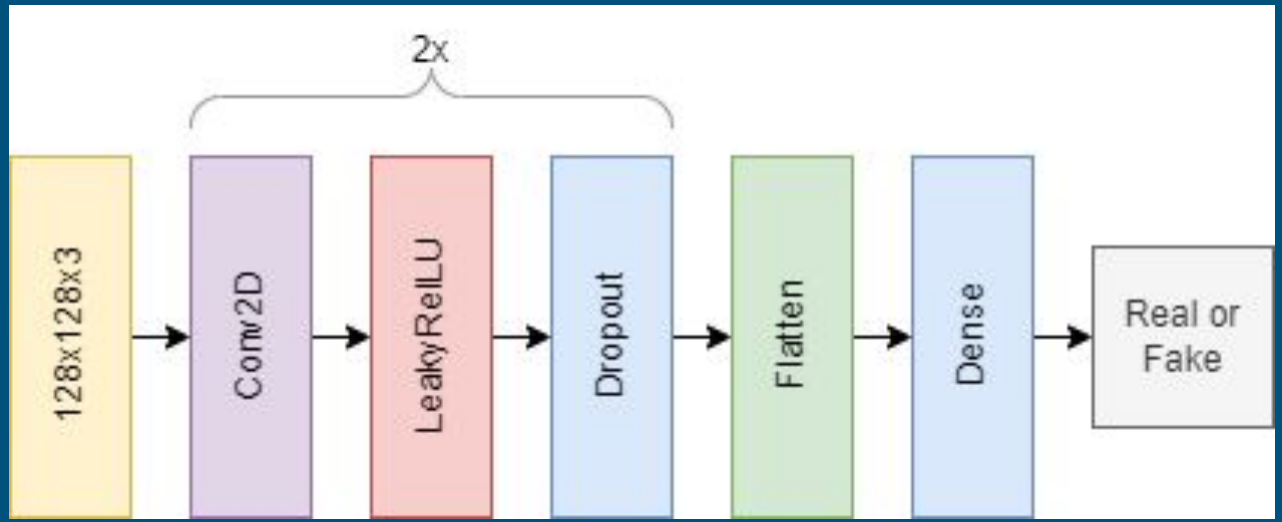
- Transforms Random Noise Vector into an 128 x 128 x 3 array



GAN the model

Discriminator:

- CNN based image classifier
- Classifies Noise from specific camera model as real or fake



GAN the model

- Generator and Discriminator is trained simultaneously
- Both models improve, but will only be using generator
- Model checkpoint and will be used later to generate fake noise

Preprocessing the Classifier

- Center crops for all images
- Input data is a 1-D vector of image features
- LBP [128x128] of noiseprint -> histogram [26x3]->input
- 3 RGB channels -> 3 Wavelet coefficients -> nth moment of the mean of wavelet coefficient matrix -> 3x3x9
- Input data of shape 26x3 + 3x3x9 = 159
- Test Images: Spoofed images + original
    - Denoise -> GAN generates np -> add np to denoised image

Training and testing the Classifier

- Multiple Models used
- Logistic Regression (train: 92%, val: 85%)
- K-means (train, val: 80%)
- MLP gave the best results (2 layers: 256 nodes and 128 nodes) (train: 92%, val: 88%)

Experiments and Ideal Results

Predicted \ Actual	cropped-iphone	cropped-galaxy8c
iphone	^	v
galaxy8c	v	^

Predicted \ Actual	denoised-iphone	denoised-galaxy8c
iphone	-	-
galaxy8c	-	-

Experiments and Ideal Results

Predicted	Actual	
	in-iphone	in-galaxy8c
iphone	^	^
galaxy8c	v	v

Predicted	Actual	
	gn-iphone	gn-galaxy8c
iphone	v	v
galaxy8c	^	^

Experiments and Ideal Results

Predicted	Actual: cropped-iphone	Actual: in-galaxy8c
iphone	^	^
galaxy8c	v	v

Predicted	Actual: cropped-galaxy8c	Actual: gn-iphone
iphone	v	v
galaxy8c	^	^

Double JPG compression

- Initially saved trained, test, and spoofed cropped images to jpgs
- Picture initially was jpg format
- Double compression lowered accuracy
- Classifier saving to png is reasonable to increase accuracy

Denoising methods

		Actual		Accuracy
		cropped-iphone	cropped-galaxy8c	
Predicted	iphone	26	7	0.816667
	galaxy8c	4	23	

		Actual		
		dn-bi-iphone	bi-galaxy8c	
Predicted	iphone	23	6	0.783333
	galaxy8c	7	24	

		Actual		
		dn-bior35-iphone	dn-bior35-galaxy8c	
Predicted	iphone	25	6	0.816667
	galaxy8c	5	24	

		Actual		
		dn-bior44-iphone	dn-bior44-galaxy8c	
Predicted	iphone	23	5	0.8
	galaxy8c	7	25	

		Actual		
		dn-coif4-iphone	dn-coif4-galaxy8c	
Predicted	iphone	23	6	0.783333
	galaxy8c	7	24	

		Actual		
		dn-coif8-iphone	dn-coif8-galaxy8c	
Predicted	iphone	24	7	0.783333
	galaxy8c	6	23	

		Actual		
		dn-db4-iphone	dn-db4-galaxy8c	
Predicted	iphone	23	5	0.8
	galaxy8c	7	25	

		Actual		
		dn-db8-iphone	dn-db8-galaxy8c	
Predicted	iphone	23	3	0.833333
	galaxy8c	7	27	

		Actual		
		dn-gaus-iphone	dn-gaus-galaxy8c	
Predicted	iphone	17	24	0.383333
	galaxy8c	13	6	

		Actual		
		dn-median-iphone	dn-median-galaxy8c	
Predicted	iphone	25	25	0.5
	galaxy8c	5	5	

		Actual		
		dn-n1-iphone	dn-n1-galaxy8c	
Predicted	iphone	27	25	0.533333
	galaxy8c	3	5	

Denoising methods

- Went with denoise_tv_chambolle denoising method (variation total denoising)
- Closest to 50% accuracy on denoised images
- less bias than median denoising

		Actual		
		dn-sym4-iphone	dn-sym4-galaxy8c	
Predicted	iphone	23	3	0.833333
	galaxy8c	7	27	

		Actual		
		dn-sym8-iphone	dn-sym8-galaxy8c	
Predicted	iphone	23	4	0.816667
	galaxy8c	7	26	

		Actual		
		dn-tv-iphone	dn-tv-galaxy8c	
Predicted	iphone	20	21	0.483333
	galaxy8c	10	9	

Rounding

- Can’t inject noise in the model as float for realistic attack
- Task: Inject noise in images and save them
- Problem: Conversion of floats to ints
- Made model slightly more biased to iphone noise
- Deemed necessary since it would be necessary to more accurately represent noise

Rounding

Before rounding

		Actual	
		in-iphone	in-galaxy8c
Predicted	iphone	13	6
	galaxy8c	17	24

		Actual	
		gn-iphone	gn-galaxy8c
Predicted	iphone	14	6
	galaxy8c	16	24

		Actual	
		cropped-iphone	in-galaxy8c
Predicted	iphone	26	6
	galaxy8c	4	24

		Actual	
		cropped-galaxy8c	gn-iphone
Predicted	iphone	7	14
	galaxy8c	23	16

After rounding

		Actual	
		in-iphone	in-galaxy8c
Predicted	iphone	13	9
	galaxy8c	17	21

		Actual	
		gn-iphone	gn-galaxy8c
Predicted	iphone	16	10
	galaxy8c	14	20

		Actual	
		cropped-iphone	in-galaxy8c
Predicted	iphone	26	9
	galaxy8c	4	21

		Actual	
		cropped-galaxy8c	gn-iphone
Predicted	iphone	7	16
	galaxy8c	23	14

Model Persistent Changes

- Saving picture in lossless format
- Using total variation denoising
- Rounding

Weighting noiseprints

		Actual						
		in-iphone	in-galaxy8c	in-iphone0.01	in-iphone0.05	in-iphone0.1	in-iphone0.3	in-iphone0.5
Predicted	iphone	12	8	20	21	20	19	15
	galaxy8c	18	22	10	9	10	11	15

		Actual						
		gn-iphone	gn-galaxy8c	gn-galaxy8c0.01	gn-galaxy8c0.05	gn-galaxy8c0.1	gn-galaxy8c0.3	gn-galaxy8c0.5
Predicted	iphone	16	8	21	21	20	19	18
	galaxy8c	14	22	9	9	10	11	12

		Actual						
		cropped-iphone	in-galaxy8c	in-galaxy8c0.01	in-galaxy8c0.05	in-galaxy8c0.1	in-galaxy8c0.3	in-galaxy8c0.5
Predicted	iphone	26	8	21	20	20	16	13
	galaxy8c	4	22	9	10	10	14	17

		Actual						
		cropped-galaxy8c	gn-iphone	gn-iphone0.01	gn-iphone0.05	gn-iphone0.1	gn-iphone0.3	gn-iphone0.5
Predicted	iphone	7	16	20	20	20	20	18
	galaxy8c	23	14	10	10	10	10	12

Weighting Noiseprints

Iphone noiseprints fool model better at lower weights

Galaxy8 noiseprints fool model better at higher weights

in-iphone0.7	in-iphone1.05	in-iphone1.1	in-iphone1.25	in-iphone1.5	in-iphone2.0	in-iphone2.2
14	10	11	12	12	13	15
16	20	19	18	18	17	15
gn-galaxy8c0.7	gn-galaxy8c1.05	gn-galaxy8c1.1	gn-galaxy8c1.25	gn-galaxy8c1.5	gn-galaxy8c2.0	gn-galaxy8c2.2
16	10	10	7	4	5	5
14	20	20	23	26	25	25
in-galaxy8c0.7	in-galaxy8c1.05	in-galaxy8c1.1	in-galaxy8c1.25	in-galaxy8c1.5	in-galaxy8c2.0	in-galaxy8c2.2
10	9	7	5	5	8	6
20	21	23	25	25	22	24
gn-iphone0.7	gn-iphone1.05	gn-iphone1.1	gn-iphone1.25	gn-iphone1.5	gn-iphone2.0	gn-iphone2.2
17	15	15	13	11	12	10
13	15	15	17	19	18	20

Conclusion: The model associates smooth images with iphones and noisy images with galaxies

Random Noise Injection

		Actual							
		cropped-iphone	cropped-galaxy8c	iphone(-1,2)	galaxy8c(-1,2)	iphone(-2,3)	galaxy8c(-2,3)	iphone(-4,5)	galaxy8c(-4,5)
Predicted	iphone	26	7	20	4	18	5	17	4
	galaxy8c	4	23	10	26	12	25	13	26

iphone(-8,9)	galaxy8c(-8,9)	iphone(-16,17)	galaxy8c(-16,17)	iphone(-24,25)	galaxy8c(-24,25)	iphone(-35,36)	galaxy8c(-35,36)	iphone(-55,56)	galaxy8c(-55,56)
19	7	16	10	18	14	19	20	24	28
11	23	14	20	12	16	11	10	6	2

iphone(-65,66)	galaxy8c(-65,66)	iphone(-78,79)	galaxy8c(-78,79)	iphone(-88,89)	galaxy8c(-88,89)	iphone(-100,101)	galaxy8c(-100,101)
24	27	25	28	26	29	29	29
6	3	5	2	4	1	1	1

- Initially thought that that due to previous experiment, a large amount of random noise would classify the image as galaxy8.
- This could be due to the uniform distribution implying that the model thinks the iphone noise is more uniform.

Random Pixel Test

- Purely Random RGB values between 0 to 255
- All 100 generated pictures were classified as iphone
- Random distribution was set to uniform
- Confirmed theory from previous experiment

Inter-Model Classification

- Do different phones of the same model have distinguishable noiseprints?
- Novel problem that hasn’t been studied
- 3 classes: iphone, galaxy8c, galaxy8l
- 3 GANs

		Actual		
		cropped-iphone	cropped-galaxy8c	cropped-galaxy8l
Predicted	iphone	18	3	3
	galaxy8c	8	20	13
	galaxy8l	4	7	14

		Actual		
		denoised-iphone	denoised-galaxy8c	cropped-galaxy8l
Predicted	iphone	7	4	8
	galaxy8c	9	9	6
	galaxy8l	14	17	16

		Actual		
		in-iphone	in-galaxy8c	in-galaxy8l
Predicted	iphone	4	2	3
	galaxy8c	13	15	17
	galaxy8l	13	13	10

		Actual		
		8cn-iphone	8cn-galaxy8c	8cn-galaxy8l
Predicted	iphone	3	3	4
	galaxy8c	12	10	10
	galaxy8l	15	17	16

		Actual		
		8ln-iphone	8ln-galaxy8c	8ln-galaxy8l
Predicted	iphone	12	4	2
	galaxy8c	2	8	6
	galaxy8l	16	18	22

Random Noise Injection into Training Set

Original

		Actual		
		cropped-iphone	cropped-galaxy8c	cropped-galaxy8l
Predicted	iphone	18	3	3
	galaxy8c	8	20	13
	galaxy8l	4	7	14

		Actual		
		denoised-iphone	denoised-galaxy8c	cropped-galaxy8l
Predicted	iphone	7	4	8
	galaxy8c	9	9	6
	galaxy8l	14	17	16

		Actual		
		in-iphone	in-galaxy8c	in-galaxy8l
Predicted	iphone	4	2	3
	galaxy8c	13	15	17
	galaxy8l	13	13	10

		Actual		
		8cn-iphone	8cn-galaxy8c	8cn-galaxy8l
Predicted	iphone	3	3	4
	galaxy8c	12	10	10
	galaxy8l	15	17	16

		Actual		
		8ln-iphone	8ln-galaxy8c	8ln-galaxy8l
Predicted	iphone	12	4	2
	galaxy8c	2	8	6
	galaxy8l	16	18	22

Delta2

		Actual		
		cropped-iphone	cropped-galaxy8c	cropped-galaxy8l
Predicted	iphone	18	7	6
	galaxy8c	6	15	7
	galaxy8l	6	8	17

		Actual		
		denoised-iphone	denoised-galaxy8c	cropped-galaxy8l
Predicted	iphone	13	16	10
	galaxy8c	6	4	5
	galaxy8l	11	10	15

		Actual		
		in-iphone	in-galaxy8c	in-galaxy8l
Predicted	iphone	15	13	10
	galaxy8c	7	7	6
	galaxy8l	8	10	14

		Actual		
		8cn-iphone	8cn-galaxy8c	8cn-galaxy8l
Predicted	iphone	14	16	14
	galaxy8c	4	6	6
	galaxy8l	12	8	10

		Actual		
		8ln-iphone	8ln-galaxy8c	8ln-galaxy8l
Predicted	iphone	18	10	3
	galaxy8c	2	5	5
	galaxy8l	10	15	22

Random Noise Injection into Training Set

- Attempt at regularization of the model
- Comparable Results for original test images
- iphone noise spoofing was more successful
- galaxy8c and 8l noise spoofing was less successful
- Two possible conclusions:
  - The model is now better able to model larger changes between iphone and galaxy (more success between distinguishing between the two models) at the cost of lower intermodel accuracy (due to smaller variation between classes being overwritten in training)
  - The uniform random noise injection contributed towards the model leaning towards iphone due to distribution shape association

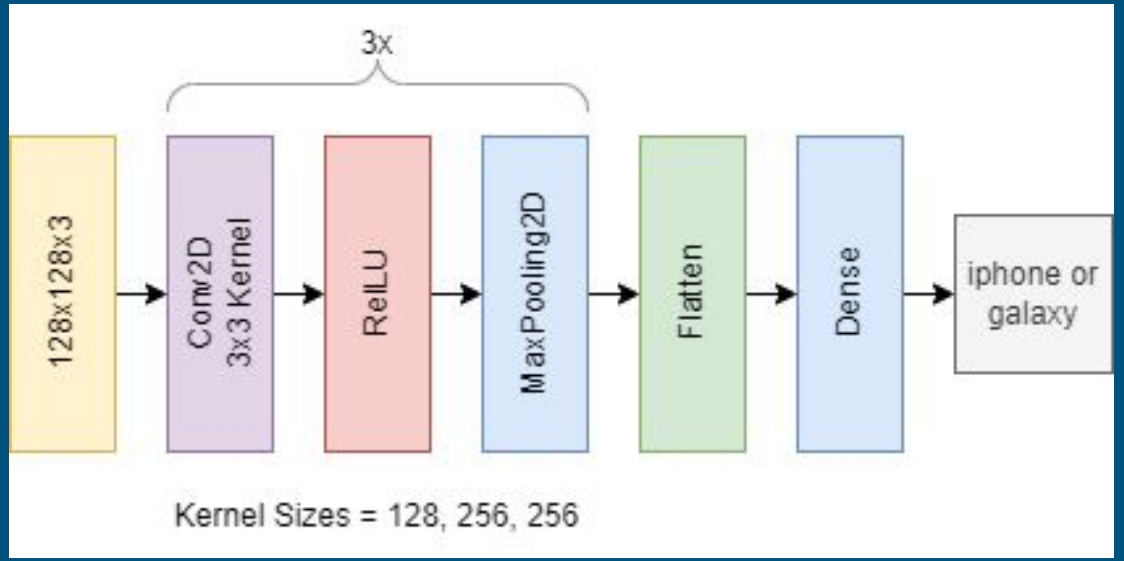
2D-CNN classifier

Tested on Raw images

Tested on NoisePrints

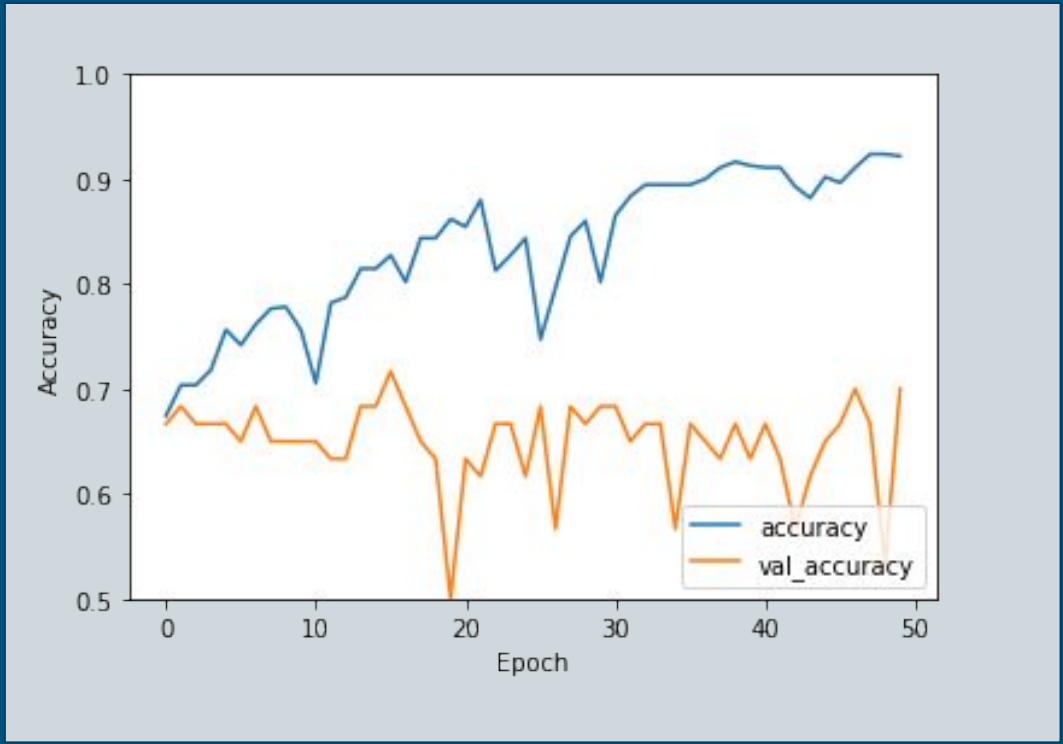
Tested on LBP data

Results were subpar

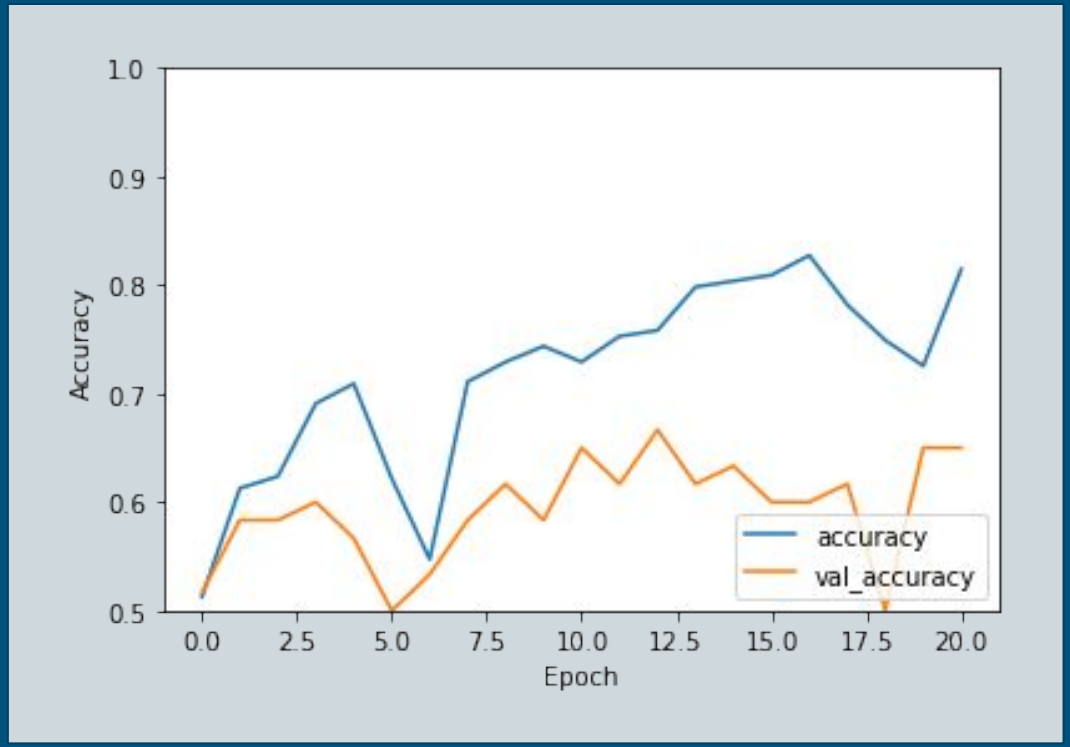


Demonstrated the strength of wavelet coefficients and histograming

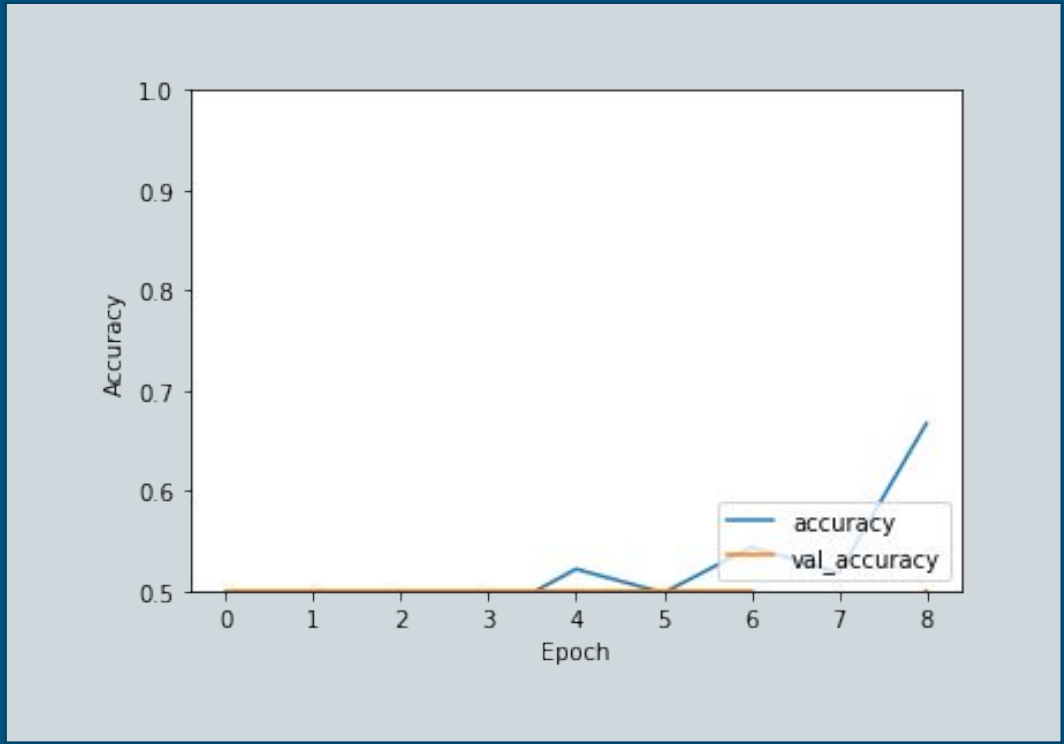
2D-CNN - Raw images



2D-CNN - Noiseprints

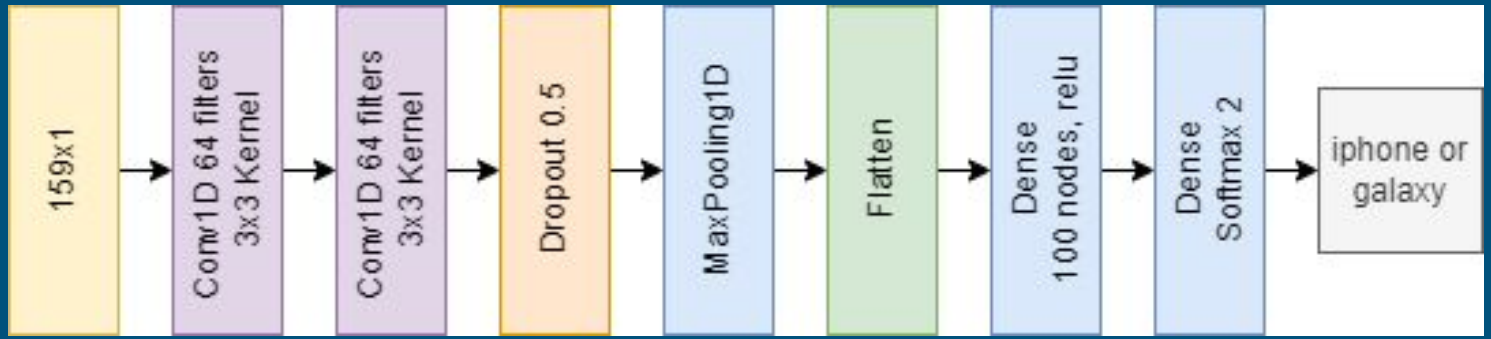


2D-CNN - LBP


1.0
0.9
0.8
0.7
0.6
0.5
Accuracy
0
1
2
3
4
5
6
7
8
Epoch
accuracy
val_accuracy

1D-CNN classifier

Results were very similar to MLP model



1D-CNN classifier

		Actual	
		cropped-iphone	cropped-galaxy8c
Predicted	iphone	26	7
	galaxy8c	4	23

		Actual	
		denoised-iphone	denoised-galaxy8c
Predicted	iphone	20	21
	galaxy8c	10	9

		Actual	
		in-iphone	in-galaxy8c
Predicted	iphone	13	11
	galaxy8c	17	19

		Actual	
		gn-iphone	gn-galaxy8c
Predicted	iphone	16	6
	galaxy8c	14	24

		Actual	
		cropped-iphone	in-galaxy8c
Predicted	iphone	26	11
	galaxy8c	4	19

		Actual	
		cropped-galaxy8c	gn-iphone
Predicted	iphone	7	16
	galaxy8c	23	14

MLP

1DCNN

		Actual	
		cropped-iphone	cropped-galaxy8c
Predicted	iphone	26	7
	galaxy8c	4	23

		Actual	
		denoised-iphone	denoised-galaxy8c
Predicted	iphone	18	19
	galaxy8c	12	11

		Actual	
		in-iphone	in-galaxy8c
Predicted	iphone	17	10
	galaxy8c	13	20

		Actual	
		gn-iphone	gn-galaxy8c
Predicted	iphone	17	12
	galaxy8c	13	18

		Actual	
		cropped-iphone	in-galaxy8c
Predicted	iphone	26	10
	galaxy8c	4	20

		Actual	
		cropped-galaxy8c	gn-iphone
Predicted	iphone	7	17
	galaxy8c	23	13

Cross Validation MLP

		Actual		
		cropped-iphone	cropped-galaxy8c	cropped-galaxy8l
Predicted	iphone	18	3	3
	galaxy8c	8	20	13
	galaxy8l	4	7	14

		Actual		
		denoised-iphone	denoised-galaxy8c	cropped-galaxy8l
Predicted	iphone	7	4	8
	galaxy8c	9	9	6
	galaxy8l	14	17	16

		Actual		
		in-iphone	in-galaxy8c	in-galaxy8l
Predicted	iphone	4	2	3
	galaxy8c	13	15	17
	galaxy8l	13	13	10

		Actual		
		8cn-iphone	8cn-galaxy8c	8cn-galaxy8l
Predicted	iphone	3	3	4
	galaxy8c	12	10	10
	galaxy8l	15	17	16

		Actual		
		8ln-iphone	8ln-galaxy8c	8ln-galaxy8l
Predicted	iphone	12	4	2
	galaxy8c	2	8	6
	galaxy8l	16	18	22

No noticeable changes

NoCV CV

		Actual		
		cropped-iphone	cropped-galaxy8c	cropped-galaxy8l
Predicted	iphone	15	3	7
	galaxy8c	5	18	9
	galaxy8l	10	9	14

		Actual		
		denoised-iphone	denoised-galaxy8c	cropped-galaxy8l
Predicted	iphone	5	4	2
	galaxy8c	4	6	3
	galaxy8l	21	20	25

		Actual		
		in-iphone	in-galaxy8c	in-galaxy8l
Predicted	iphone	13	4	7
	galaxy8c	8	15	13
	galaxy8l	9	11	10

		Actual		
		8cn-iphone	8cn-galaxy8c	8cn-galaxy8l
Predicted	iphone	8	4	8
	galaxy8c	9	15	10
	galaxy8l	13	11	12

		Actual		
		8ln-iphone	8ln-galaxy8c	8ln-galaxy8l
Predicted	iphone	13	6	3
	galaxy8c	2	4	4
	galaxy8l	15	20	23

Recipes for success

- Poison attack:
  - Simple Denoising, least noticable
  - Weighted noise, Random noise makes the model misclassify
  - Be careful of SSIM values
- Evasion:
  - Test classification to identify bias in network and attempt to hand craft noise to spoof network

Future Work

- Larger crop windows (either full size or regional crops)
- Machine Learning based PRNU extraction
- More hyperparameter tuning of models
- Auxiliary Classifier Generative Adversarial Network (AC-GAN)

References

J. Lukas, J. Fridrich and M. Goljan, "Digital camera identification from sensor pattern noise," in IEEE Transactions on Information Forensics and Security, vol. 1, no. 2, pp. 205-214, June 2006, doi: 10.1109/TIFS.2006.873602.

R. Li, Y. Guan and C. Li, "PCA-based denoising of Sensor Pattern Noise for source camera identification," 2014 IEEE China Summit & International Conference on Signal and Information Processing (ChinaSIP), 2014, pp. 436-440, doi: 10.1109/ChinaSIP.2014.6889280.

D. Cozzolino, J. Thies, A. Rössler, M. Nießner and L. Verdoliva, "SpoC: Spoofing Camera Fingerprints," 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2021, pp. 990-1000, doi: 10.1109/CVPRW53098.2021.00110.

C. Chen, X. Zhao and M. C. Stamm, "Generative Adversarial Attacks Against Deep-Learning-Based Camera Model Identification," in IEEE Transactions on Information Forensics and Security, doi: 10.1109/TIFS.2019.2945198.

S. Samaras, V. Mygdalis and I. Pitas, "Robustness in blind camera identification," 2016 23rd International Conference on Pattern Recognition (ICPR), 2016, pp. 3874-3879, doi: 10.1109/ICPR.2016.7900239.

Thank You!