Convolutional Neural Networks and you

Does that Tesla see me?

Datascience Workflow

1. Definition
2. Data harvesting
3. Exploration
4. Modeling
5. Evalutating
6. Answers

Problem

Can a computer tell a motorcycle from a car?

Free Range Data Hunting

MIOvision Traffic Camera Data

Free Range Data Hunting

MIOvision Traffic Camera Data

519, 982 images in 11 categories

Free Range Data Hunting

MIOvision Traffic Camera Data

519, 982 images in 11 categories

Training and Testing Images

Look! Pictures!

Articulated Truck	10,346
background	160,000
bicycle	2284
bus	10,316
car	260,518
motorcycle	1982
non-motorized vehicle	1751
pedestrian	6262
pick-up truck	50,906
single-unit truck	5120
work van	9697

One model to rule them all...

Binary

Input (input size based on image size)

15 hidden layers

Output (sigmoid activation)

Multiclass

Input (input size based on image size)

15 hidden layers

Output (softmax activation)

Does it work?

NO

Not even close

Data Issues

Issues

1. Out of balance
2. Different sizes
3. Different shapes
4. Size (3GB)

Solutions

1. Selective image imports
2. Resize
3. Reshape
4. Bigger/better computer

Libraries:

1. Pillow - friendly version of Pythom Image Library (PIL)
2. keras ImageDataGenerator

Modeling Issues

Get your geek on!

4 convolution layers

3 pooling layers

4 dropout layers

1 flatten layer

4 dense layers

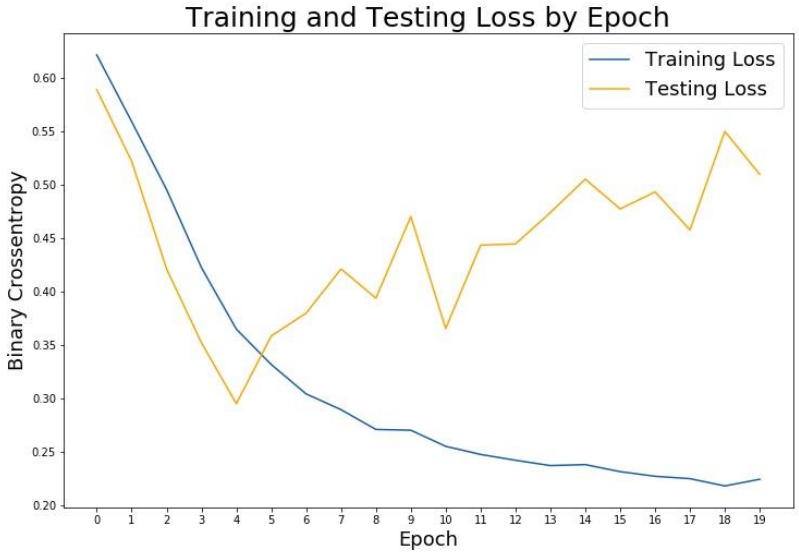
Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 126, 126, 112)	1120
max_pooling2d_1 (MaxPooling2	(None, 63, 63, 112)	0
dropout_1 (Dropout)	(None, 63, 63, 112)	0
conv2d_2 (Conv2D)	(None, 61, 61, 56)	56504
max_pooling2d_2 (MaxPooling2	(None, 30, 30, 56)	0
dropout_2 (Dropout)	(None, 30, 30, 56)	0
conv2d_3 (Conv2D)	(None, 29, 29, 28)	6300
max_pooling2d_3 (MaxPooling2	(None, 14, 14, 28)	0
dropout_3 (Dropout)	(None, 14, 14, 28)	0
conv2d_4 (Conv2D)	(None, 13, 13, 14)	1582
dropout_4 (Dropout)	(None, 13, 13, 14)	0
flatten_1 (Flatten)	(None, 2366)	0
dense_1 (Dense)	(None, 112)	265104
dense_2 (Dense)	(None, 56)	6328
dense_3 (Dense)	(None, 28)	1596
dense_4 (Dense)	(None, 11)	319

Total params: 338,853
Trainable params: 338,853
Non-trainable params: 0

Models

all the models

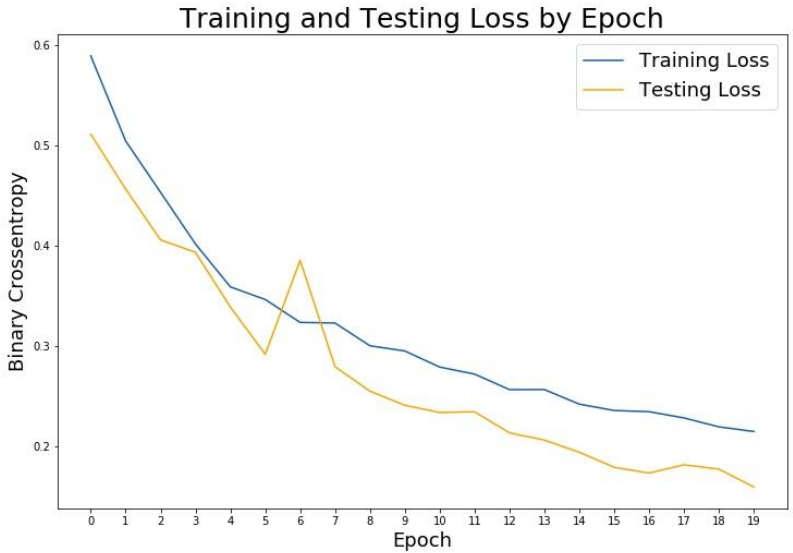
Unbalanced, Binary, 28x28 pixels



Accuracy: 75.04%

	pred car	pred mc
actual car	4505	3498
actual mc	5	3959

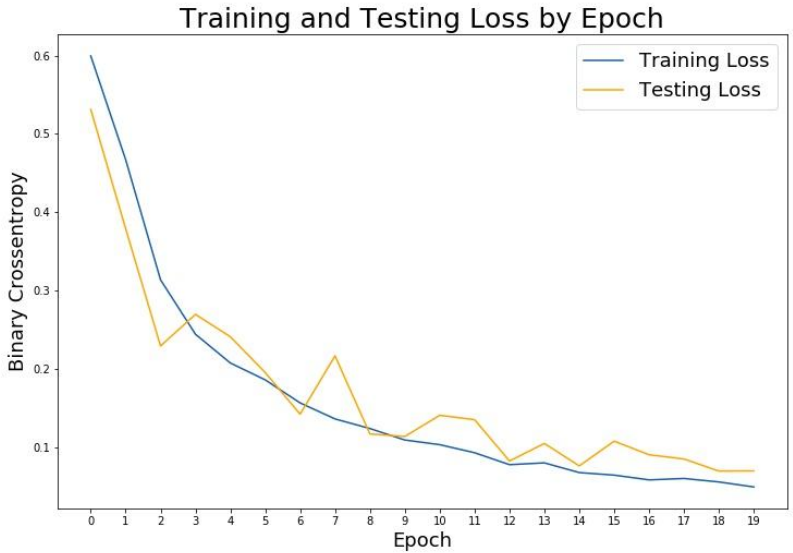
Unbalanced, Binary, 64x64 pixels



Accuracy: 94.92%

	pred car	pred mc
actual car	7726	277
actual mc	436	3528

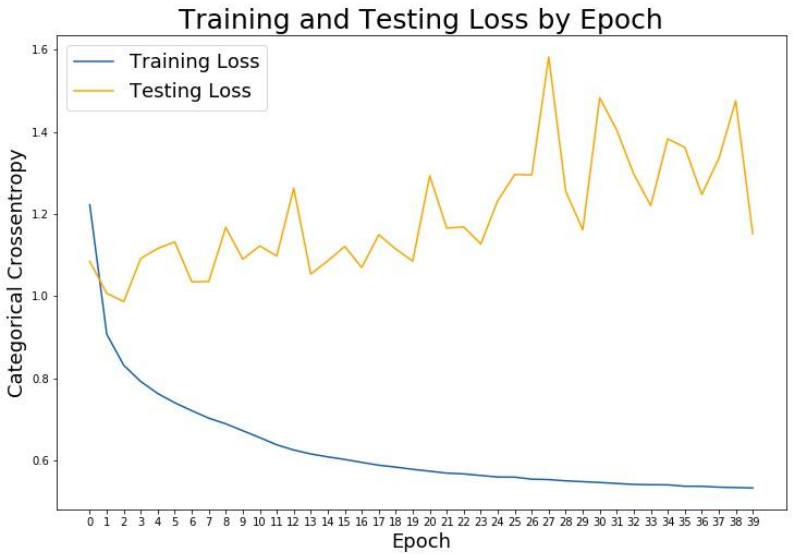
Unbalanced, Binary, 128x128 pixels



Accuracy: 97.91%

	pred car	pred mc
actual car	7770	233
actual mc	59	3905

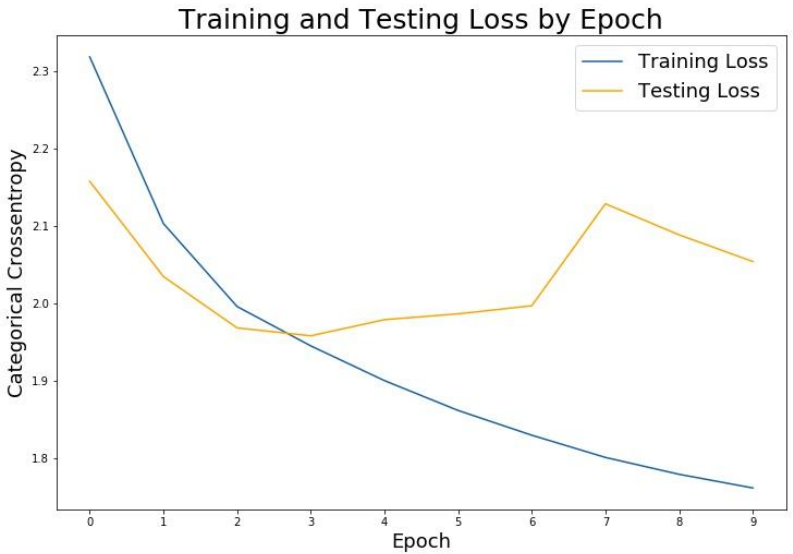
Full Dataset, 28x28



Full Dataset, 28x28

	articulated_truck	background	bicycle	bus	car	motorcycle	non-motorized_vehicle	pedestrian	pickup_truck	single_unit_truck	work_van
articulated_truck	555	127	341	88	445	138	90	741	580	689	118
background	10	3026	51	41	57	26	59	506	43	115	9
bicycle	36	15	2857	2	44	290	2	555	148	14	7
bus	74	56	51	1419	544	36	57	365	844	310	198
car	101	41	428	91	650	430	6	593	1480	151	39
motorcycle	29	5	1642	3	27	1686	7	306	236	24	2
non-motorized_vehicle	166	223	203	58	335	195	438	598	449	591	225
pedestrian	47	72	1334	10	125	136	13	2035	153	59	1
pickup_truck	66	67	385	57	450	502	4	401	1892	137	46
single_unit_truck	230	85	252	134	497	146	152	445	718	1092	219
work_van	158	78	87	269	568	79	137	349	945	725	566

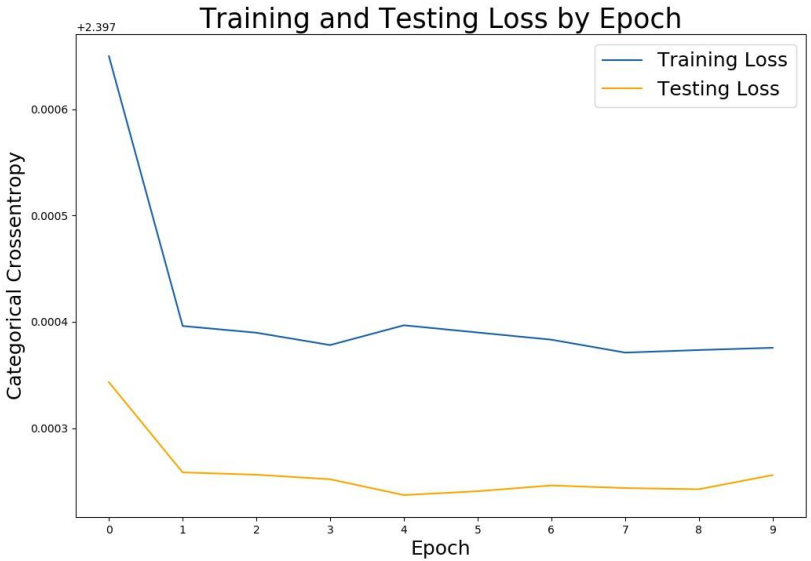
Balanced Full Dataset 28x28



Balanced Full Dataset 28x28

	articulated_truck	background	bicycle	bus	car	motorcycle	non-motorized_vehicle	pedestrian	pickup_truck	single_unit_truck	work_van
articulated_truck	434	20	172	2	567	922	101	1657	7	28	0
background	54	1530	73	0	31	69	39	2152	0	1	0
bicycle	4	0	2759	0	24	810	0	373	0	1	0
bus	270	2	3	102	2458	384	85	517	19	110	10
car	64	0	84	6	1412	1844	0	587	1	8	0
motorcycle	6	0	1309	0	7	2488	1	121	0	0	0
non-motorized_vehicle	447	22	88	1	385	639	489	1281	5	103	2
pedestrian	13	0	1935	0	49	719	5	1208	0	0	0
pickup_truck	66	0	84	6	1638	1556	3	613	7	6	0
single_unit_truck	746	1	58	3	981	526	297	1311	18	121	6
work_van	538	1	24	26	1601	353	379	721	26	296	11

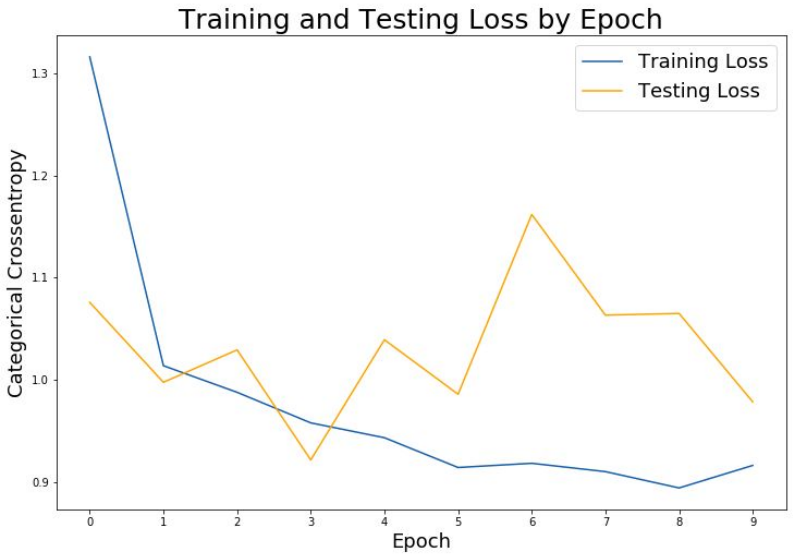
Balanced Full Dataset 64x64



Balanced Full Dataset 64x64

	articulated_truck	background	bicycle	bus	car	motorcycle	non-motorized_vehicle	pedestrian	pickup_truck	single_unit_truck	work_van
articulated_truck	943	108	104	388	552	230	301	285	253	615	131
background	41	3305	48	29	50	47	85	290	5	47	2
bicycle	22	73	2183	7	124	931	16	583	19	8	5
bus	216	59	63	2371	373	96	93	86	180	228	195
car	79	22	153	145	2645	358	21	119	317	57	90
motorcycle	33	22	434	7	147	2910	38	320	18	1	2
non-motorized_vehicle	419	256	140	190	290	389	960	344	51	262	161
pedestrian	44	243	403	16	32	505	56	2625	5	0	0
pickup_truck	104	17	55	380	1554	266	22	87	1243	109	142
single_unit_truck	334	47	43	357	442	74	225	62	260	2003	221
work_van	158	64	39	465	671	67	196	90	658	530	1038

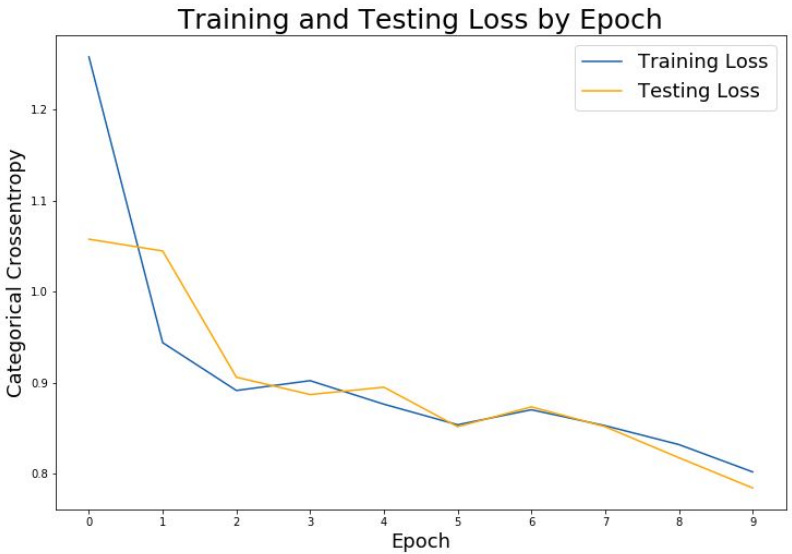
ImageDataGenerator, 28x28



ImageDataGenerator, 28x28

	articulated_truck	background	bicycle	bus	car	motorcycle	non-motorized_vehicle	pedestrian	pickup_truck	single_unit_truck	work_van
articulated_truck	0	1084	0	0	1502	0	0	0	0	0	0
background	0	17014	0	0	22986	0	0	0	0	0	0
bicycle	0	251	0	0	320	0	0	0	0	0	0
bus	0	1105	0	0	1474	0	0	0	0	0	0
car	0	27441	0	0	37688	0	0	0	0	0	0
motorcycle	0	209	0	0	286	0	0	0	0	0	0
non-motorized_vehicle	0	195	0	0	242	0	0	0	0	0	0
pedestrian	0	631	0	0	934	0	0	0	0	0	0
pickup_truck	0	5392	0	0	7334	0	0	0	0	0	0
single_unit_truck	0	551	0	0	729	0	0	0	0	0	0
work_van	0	1015	0	0	1404	0	0	0	0	0	0

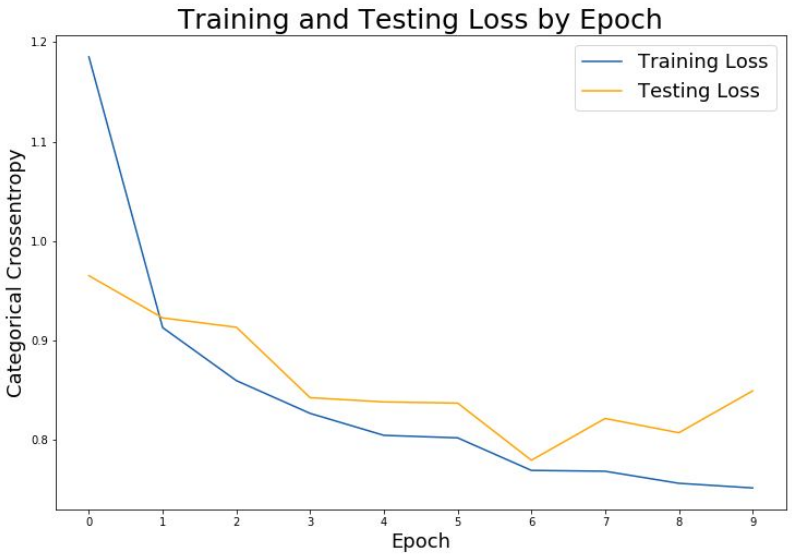
ImageDataGenerator, 64x64



ImageDataGenerator, 64x64

	articulated_truck	background	bicycle	bus	car	motorcycle	non-motorized_vehicle	pedestrian	pickup_truck	single_unit_truck	work_van
articulated_truck	0	715	0	0	1843	0	0	0	28	0	0
background	0	11660	0	4	27922	0	0	0	414	0	0
bicycle	0	174	0	0	390	0	0	0	7	0	0
bus	0	726	0	1	1826	0	0	0	26	0	0
car	0	18823	0	4	45689	0	0	0	613	0	0
motorcycle	0	160	0	0	330	0	0	0	5	0	0
non-motorized_vehicle	0	138	0	0	295	0	0	0	4	0	0
pedestrian	0	491	0	0	1060	0	0	0	14	0	0
pickup_truck	0	3600	0	0	8994	0	0	0	132	0	0
single_unit_truck	0	386	0	0	881	0	0	0	13	0	0
work_van	0	703	0	0	1690	0	0	0	26	0	0

ImageDataGenerator, 128x128



ImageDataGenerator, 128x128

	articulated_truck	background	bicycle	bus	car	motorcycle	non-motorized_vehicle	pedestrian	pickup_truck	single_unit_truck	work_van
articulated_truck	12	935	2	26	1567	0	0	19	24	0	1
background	159	13944	18	317	25133	0	0	163	227	0	39
bicycle	2	188	0	4	374	0	0	1	2	0	0
bus	5	875	1	15	1652	0	0	16	13	0	2
car	221	22678	22	515	40987	0	0	276	376	0	54
motorcycle	2	157	0	3	331	0	0	2	0	0	0
non-motorized_vehicle	0	148	1	4	279	0	0	1	4	0	0
pedestrian	4	536	0	14	994	0	0	9	8	0	0
pickup_truck	54	4385	11	98	8039	0	0	65	66	0	8
single_unit_truck	5	449	2	16	795	0	0	8	5	0	0
work_van	13	799	3	19	1557	0	0	12	12	0	4

Got CUDA?

Affect of CUDA enabled GPU*

	CPU s/epoch	GPU s/epoch
Binary 28	59	2.65
Binary 64	130	9
Binary 128	1548	37
Full Dataset 28	350	19
Full Dataset 64		97.3
ImageGen 28	310	
ImageGen 64	405	
ImageGen 128	760	

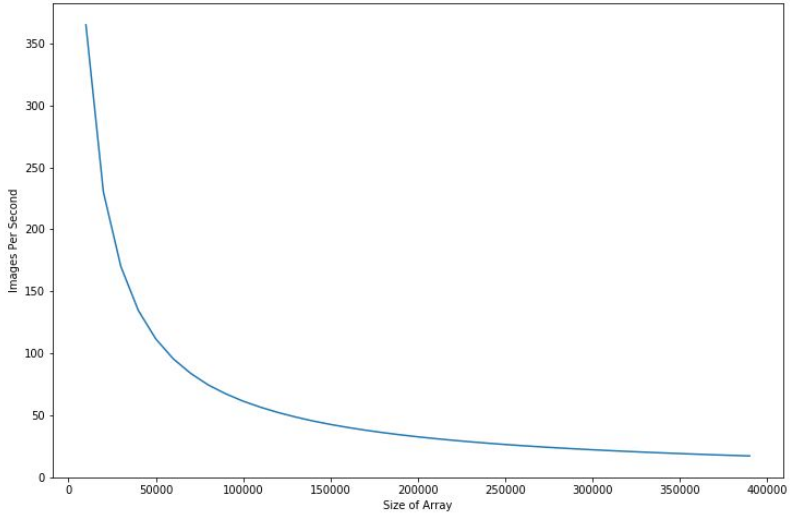
*This slide brought to you by my long suffering CFO

Affect of Image Size

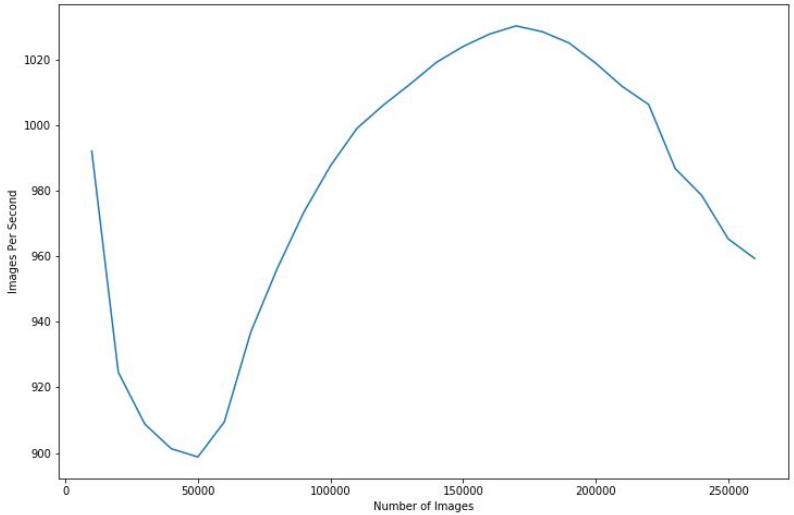
	Accuracy Score		
	28x28	64x64	128x128
Binary Unbalanced	75.04	94.92	97.91
ImageGen	37.90	43.20	42.40
Full Data	76.38	88.26	NaN

Spoiler: bigger is better

Pre fix Image Processing



Post fix Image Processing



Issues.

What’s the big deal? We all have them…

Some more than others

Issues

1. ImageDataGenerator CPU block
2. 128x128 full test memory issues
3. Image Processing time

Solutions

1. Get a bigger CPU, workers=4 didn't help
2. AWS, pickles
3. Solved! Changed code

Moving Forward

1. Understand the affect of Pooling and Dropout layers
2. Understand the affect of batch size and epoch steps
3. Understand and modify the weights
4. Vizualize the layers via keras and tensorboard
5. AWS
6. Ubuntu install