Who is this Pokemon?

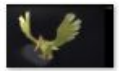
Orçun Ziylan

Dataset - Pokemon Generation ONE

- +10.000 images
- 149 classes
- ~60 images for each pokemon
- 1.2 GB

Dataset - Problems

- Duplicate images (Fearow 124 > 63)


0f8626c7a
4df46c0b0
23d6378e2
28f5b


1eaddc14e
de94ffcb69
e8d9671f6
f2a7


2f57ba2d6
f2940a98c
82a2e9ca1
2e3ef


4b779b44
422245a1a
a0f07e2f37
0a277


4fd4b90c1
ac94200a7
a9c41054f
33b2d


12492af1f8
d749ffbb4f
d856c9540
5b3


d61d33ad
678741e2a
ab071bf78
dc0003


e1a2a27d5
beb4da192
4740b216e
805fd

Dataset - Problems

- Wrong / Missing labelled images

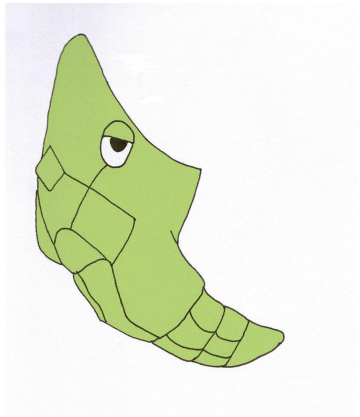
True: Mankey
Pred: Raticate 99.96%

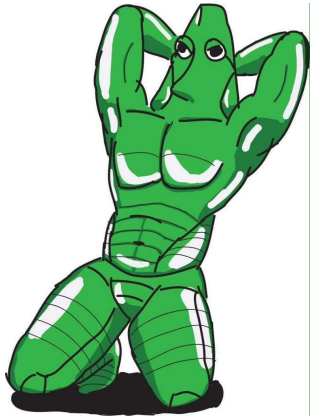
True: Nidoqueen
Pred: Nidoking 76.87%

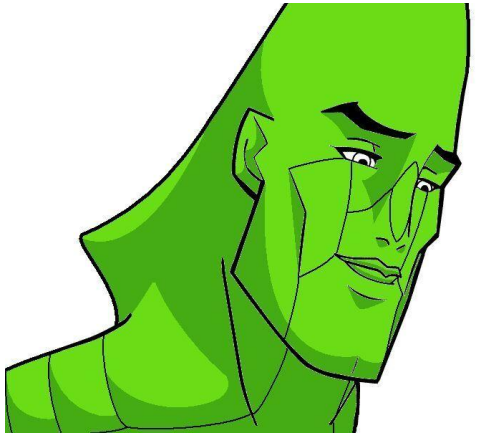
True: Squirtle
Pred: Bulbasaur 49.24%

Dataset - Problems

- Lots of weird images







Preprocessing - Building the dataset (1/2)

- Test size 10%
- Size of the classes - 35 ~ 300

Preprocessing - Building the dataset (2/2)

- Data augmentation
  - Random Contrast
  - Random Translation
  - Random Flip
  - Random Rotation
- Up to 200 images for each class
- Test size 20%

Preprocessing - Building the dataset (2/2)

- Cleaning opt.
  - Removing duplicates
  - Removing unsupported extensions (.gif , .svg , ...)
  - In total, +600 images

Base Model

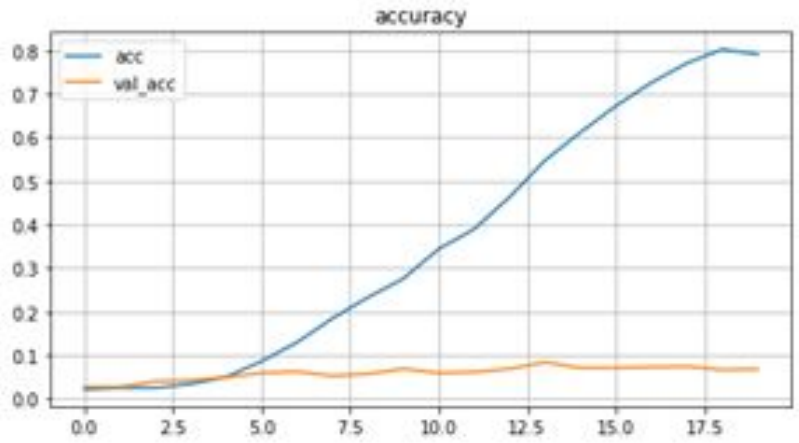
- Google CNN

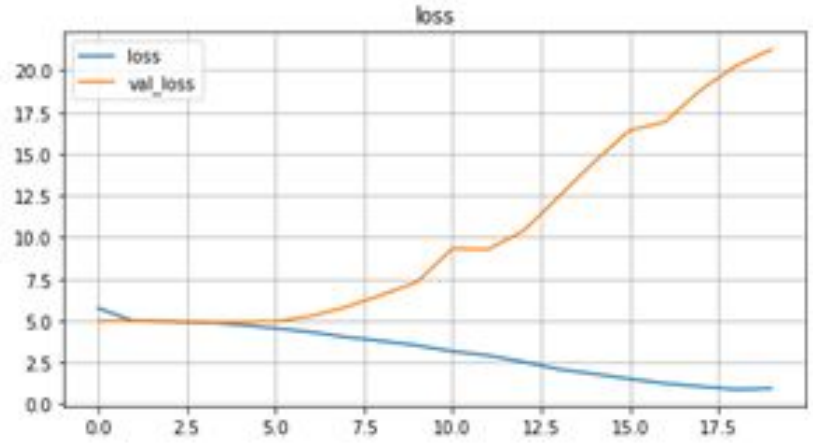
```
model = models.Sequential()
model.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32, 3)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
```

```
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(10))
```

Experiments - 1

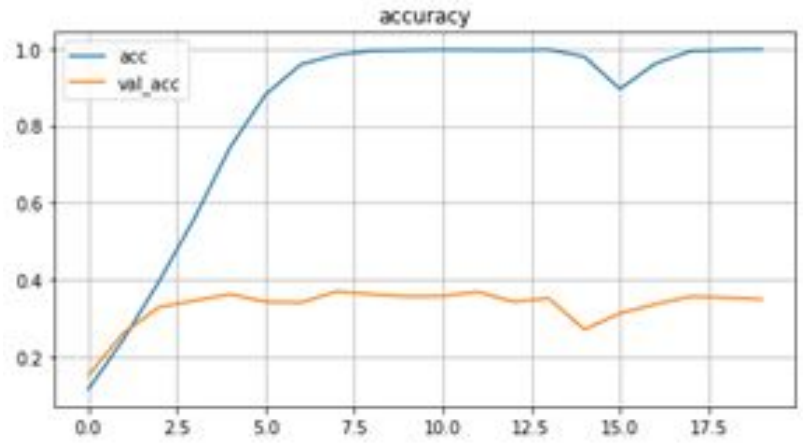
- Google CNN
- 20 Epochs - 96x96 Res. - 0.93 Training Acc. - 0.08 Validation Acc.

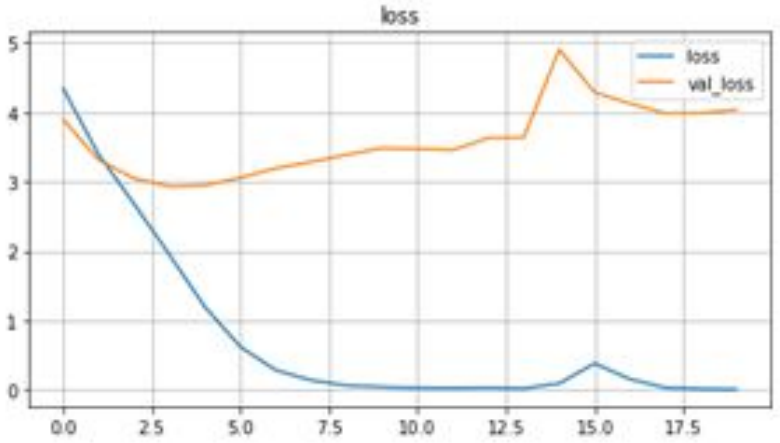




Experiments - 2

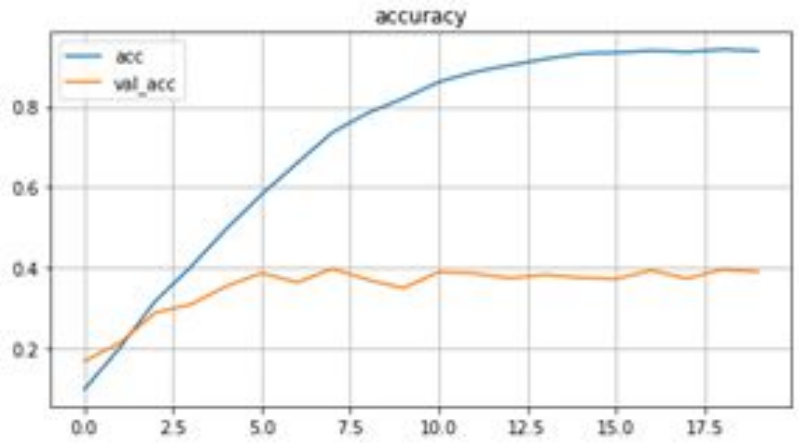
- Google CNN + Batch normalization
- 20 Epochs - 96x96 Res. - 0.99 Training Acc. - 0.36 Validation Acc.

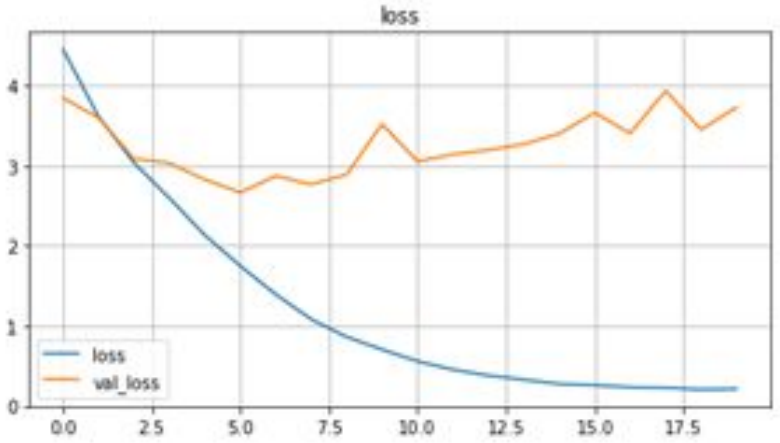




Experiments - 3

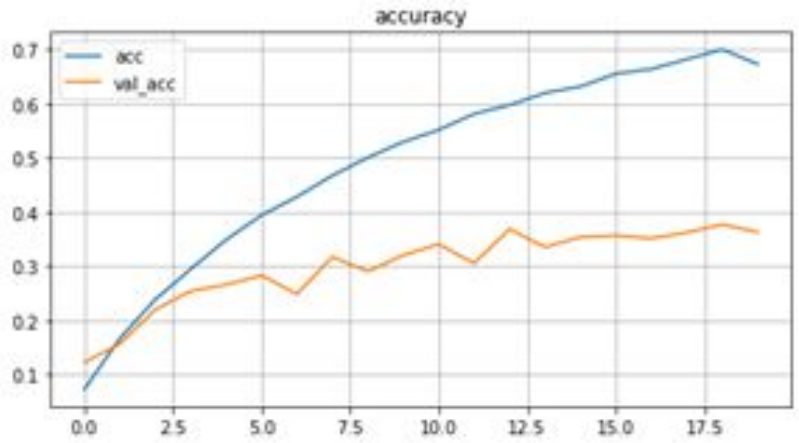
- Google CNN + Hidden layers with dropout 0.25
- 20 Epochs - 96x96 Res. - 0.93 Training Acc. - 0.39 Validation Acc.

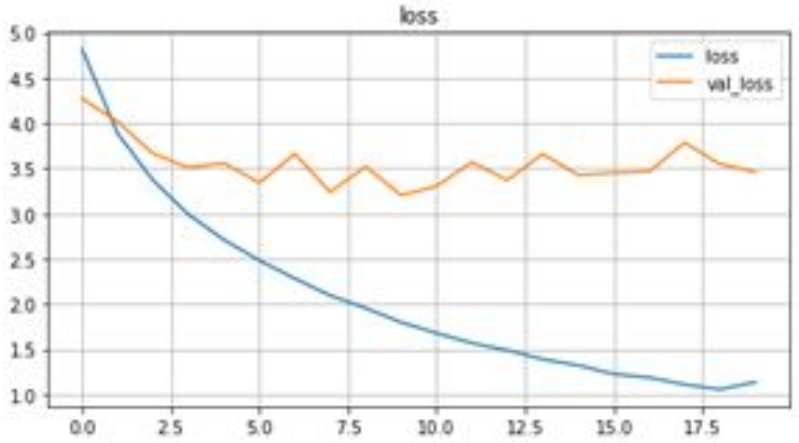




Experiments - 4

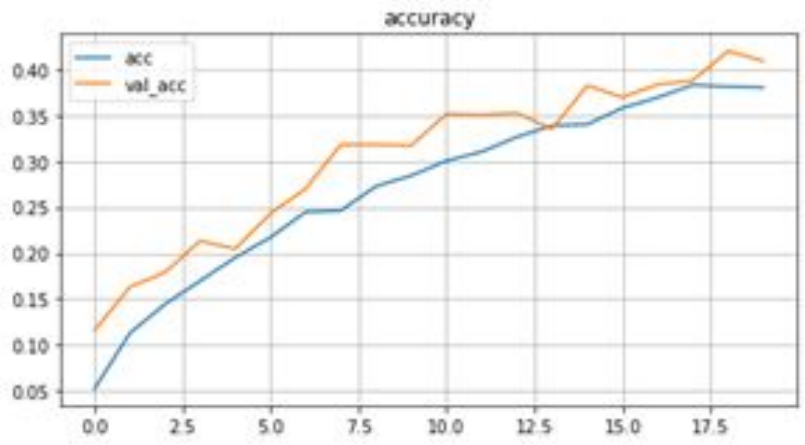
- Google CNN + Hidden layers with dropout 0.5
- 20 Epochs - 96x96 Res. - 0.67 Training Acc. - 0.37 Validation Acc.

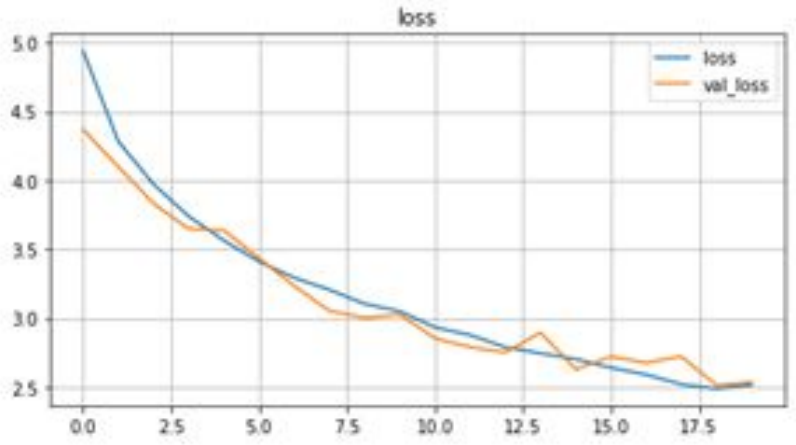




Experiments - 5

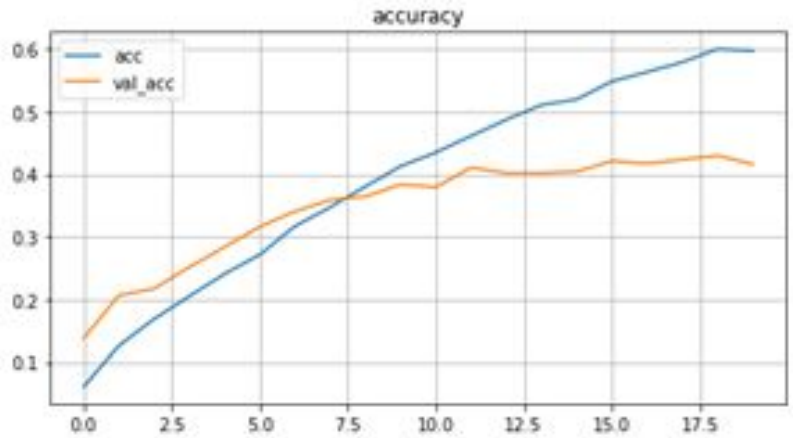
- Google CNN + Hidden layers and Output layers with dropout 0.5
- 20 Epochs - 96x96 Res. - 0.38 Training Acc. - 0.42 Validation Acc.

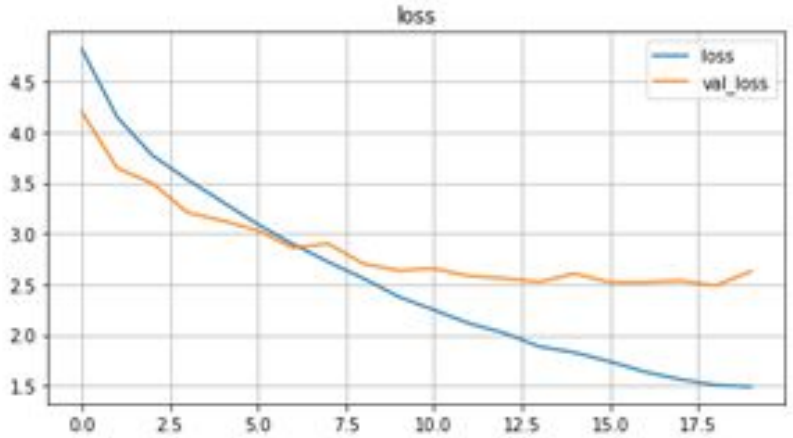




Experiments - 6

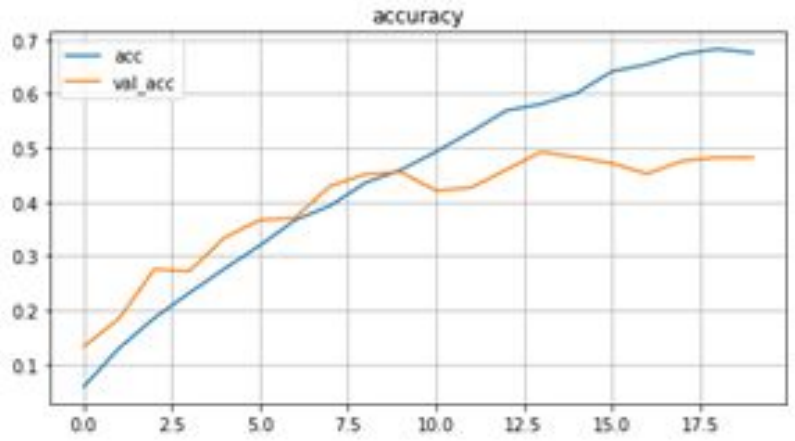
- Google CNN + Hidden w/ dropout 0.25 - Output w/ dropout 0.5
- 20 Epochs - 96x96 Res. - 0.59 Training Acc. - 0.43 Validation Acc.

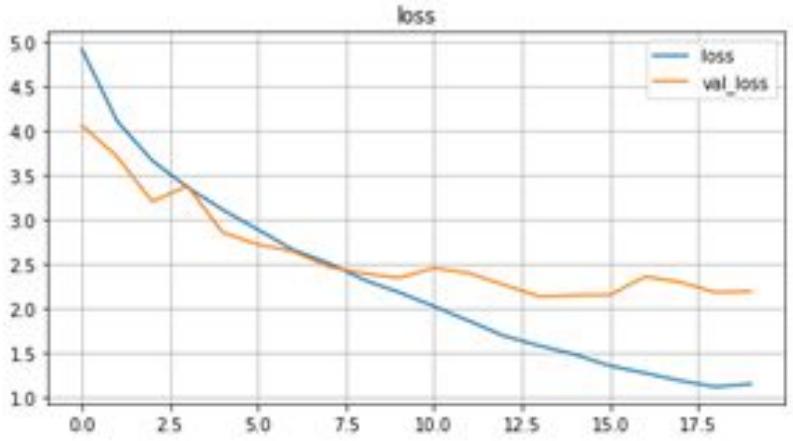




Experiments - 7

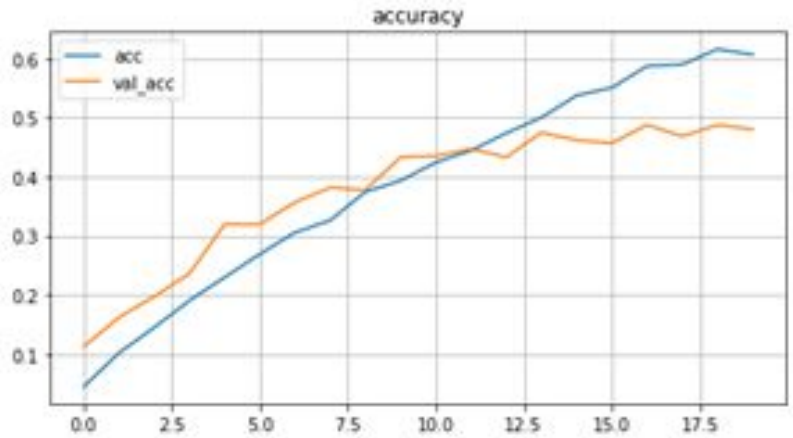
- Google CNN + New layer and Neuron count (32, 64, 128, 256)
- 20 Epochs - 96x96 Res. - 0.67 Training Acc. - 0.49 Validation Acc.

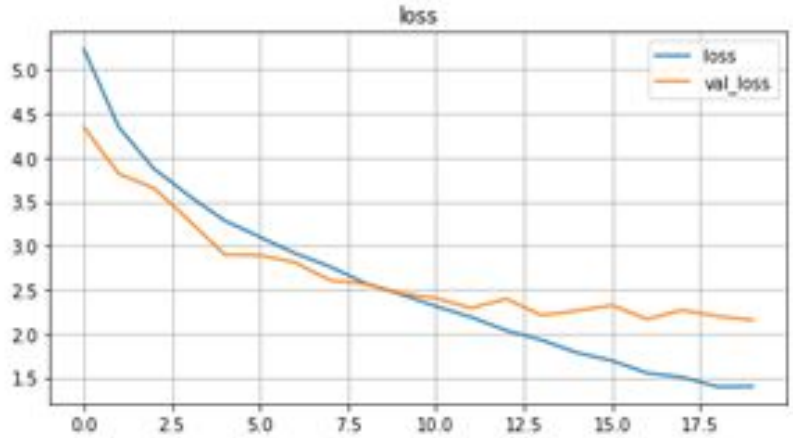




Experiments - 8

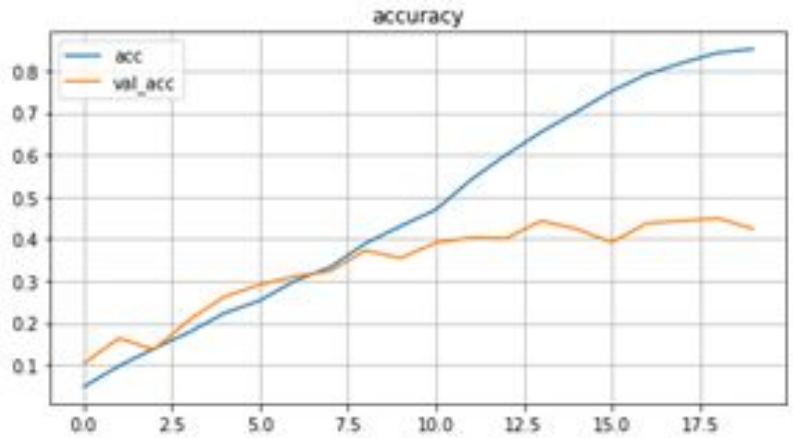
- Google CNN + New output layers (512, 256)
- 20 Epochs - 96x96 Res. - 0.60 Training Acc. - 0.48 Validation Acc.

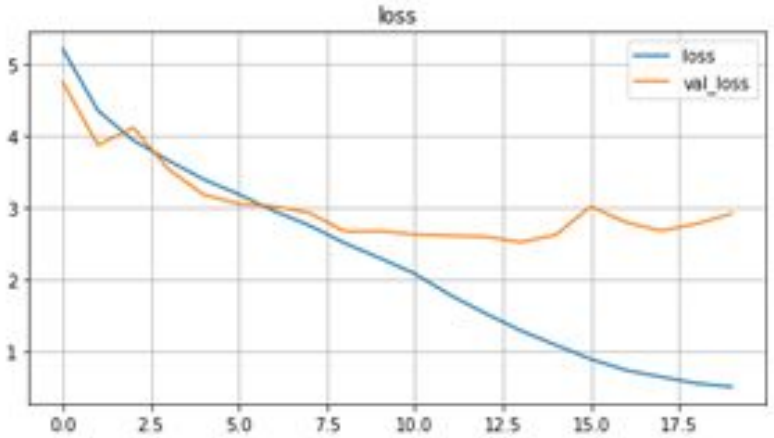




Experiments - 9

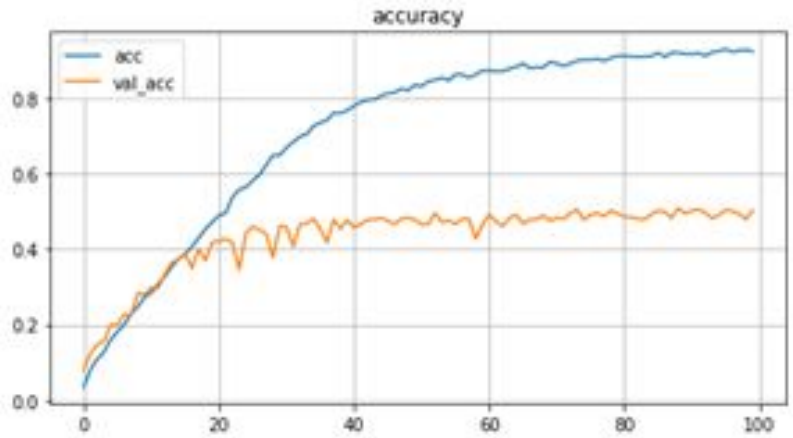
- Google CNN - Testing on high resolution (96 -> 256)
- 20 Epochs - 256x256 Res. - 0.85 Training Acc. - 0.45 Validation Acc.

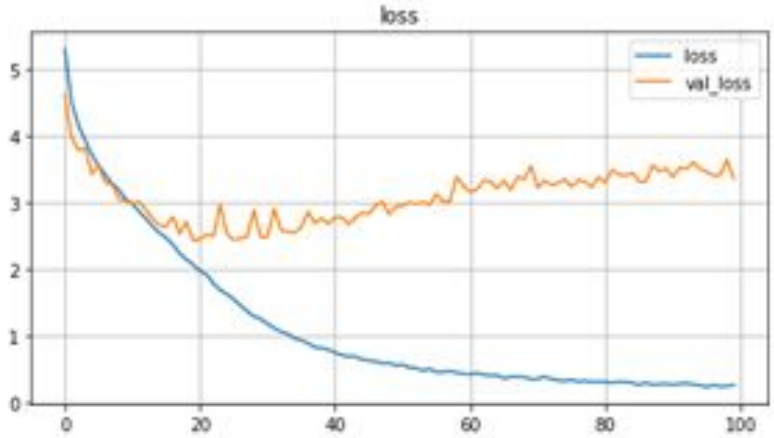




Experiments - 10

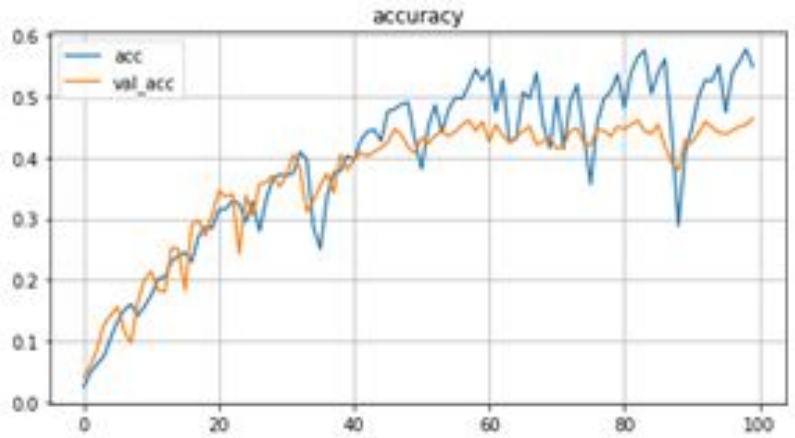
- Google CNN + Increased number of Epochs (20 -> 100)
- 100 Epochs - 96x96 Res. - 0.92 Training Acc. - 0.51 Validation Acc.

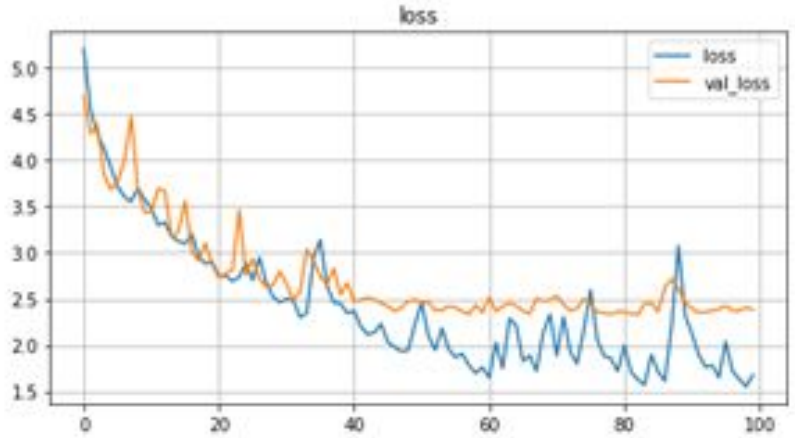




Experiments - 11 - Data Augmentation

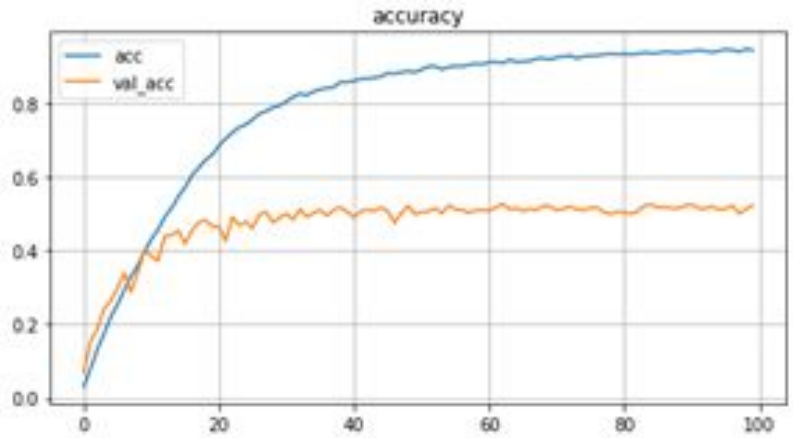
- Google CNN + each class has 100 train images but same test size
- 100 Epochs - 96x96 Res. - 0.54 Training Acc. - 0.46 Validation Acc.

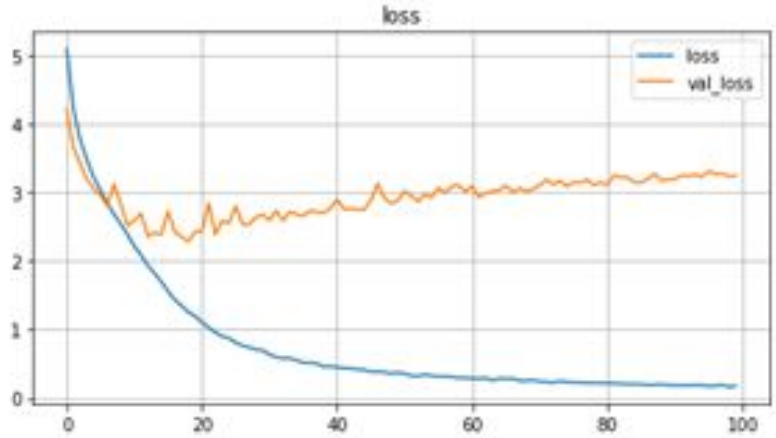




Experiments - 12 - Data Augmentation

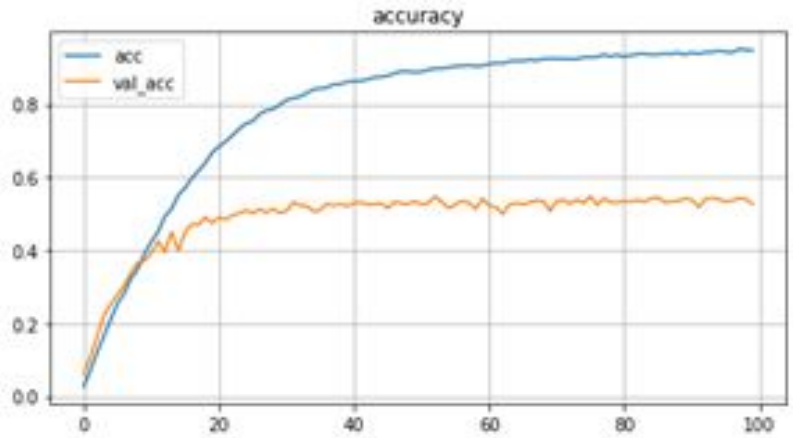
- Google CNN + 100 train images and 20 test images
- 100 Epochs - 96x96 Res. - 0.94 Training Acc. - 0.52 Validation Acc.

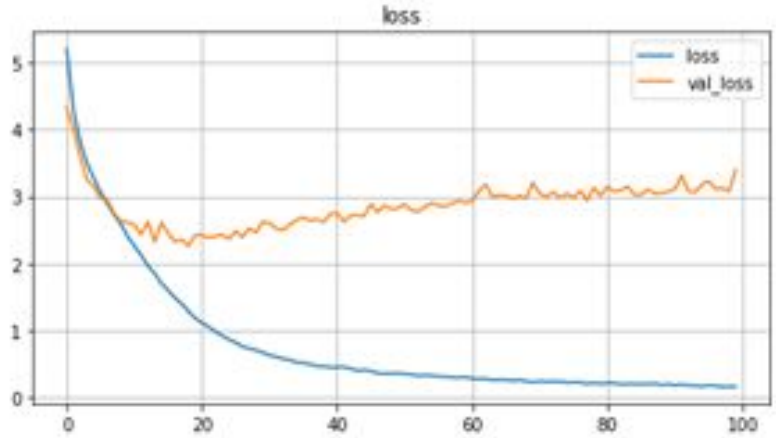




Experiments - 13 - Data Augmentation

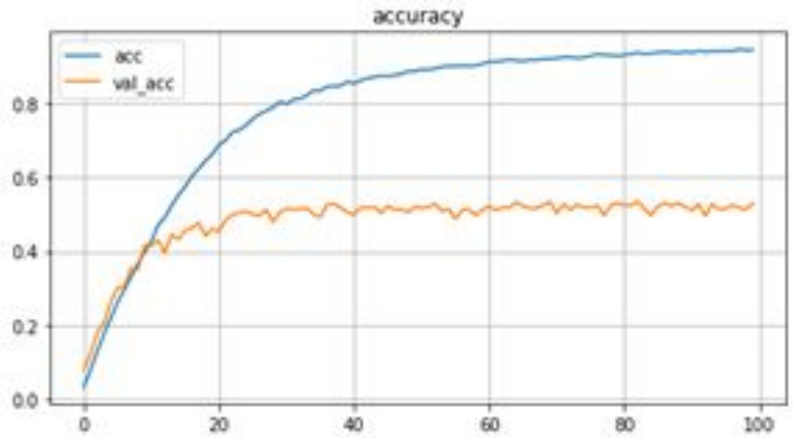
- Google CNN + Removed all duplicate and their original images
- 100 Epochs - 96x96 Res. - 0.94 Training Acc. - 0.54 Validation Acc.

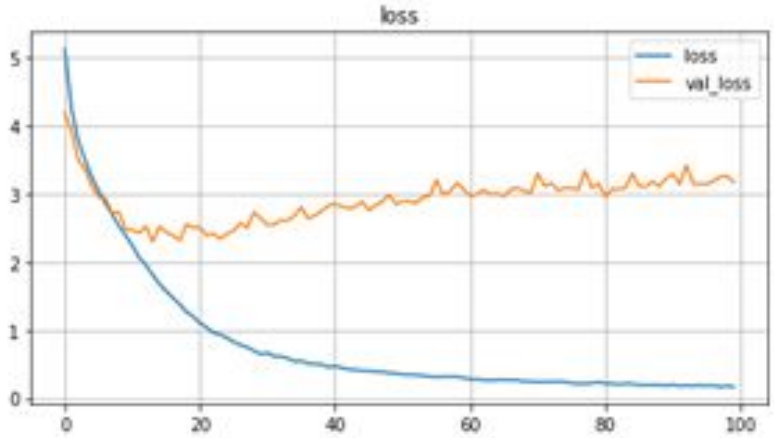




Experiments - 15 - Data Augmentation

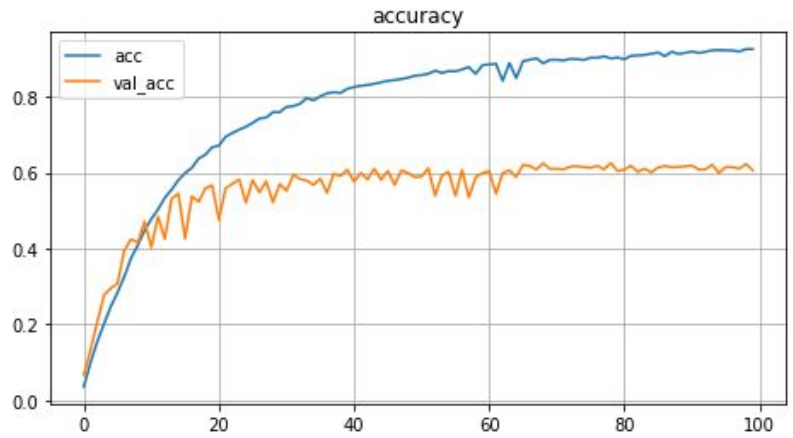
- Google CNN + Removed all duplicate in the same class
- 100 Epochs - 96x96 Res. - 0.94 Training Acc. - 0.53 Validation Acc.

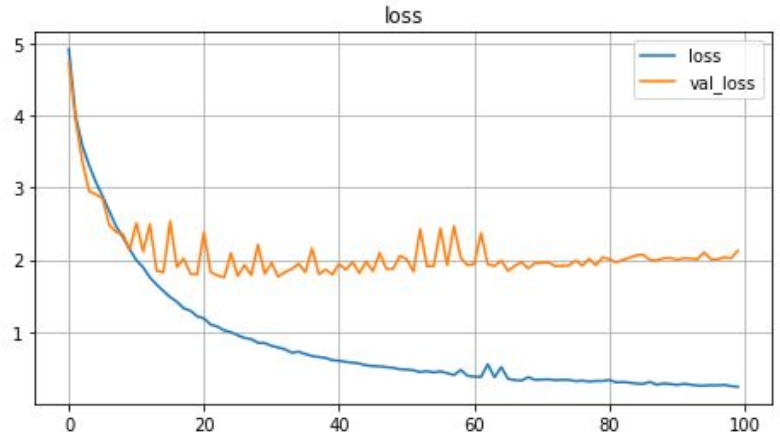




Experiments - 16 - Best Model

- Modified Google CNN + 200 images per class
- 100 Epochs - 96x96 Res. - 0.92 Training Acc. - 0.63 Validation Acc.





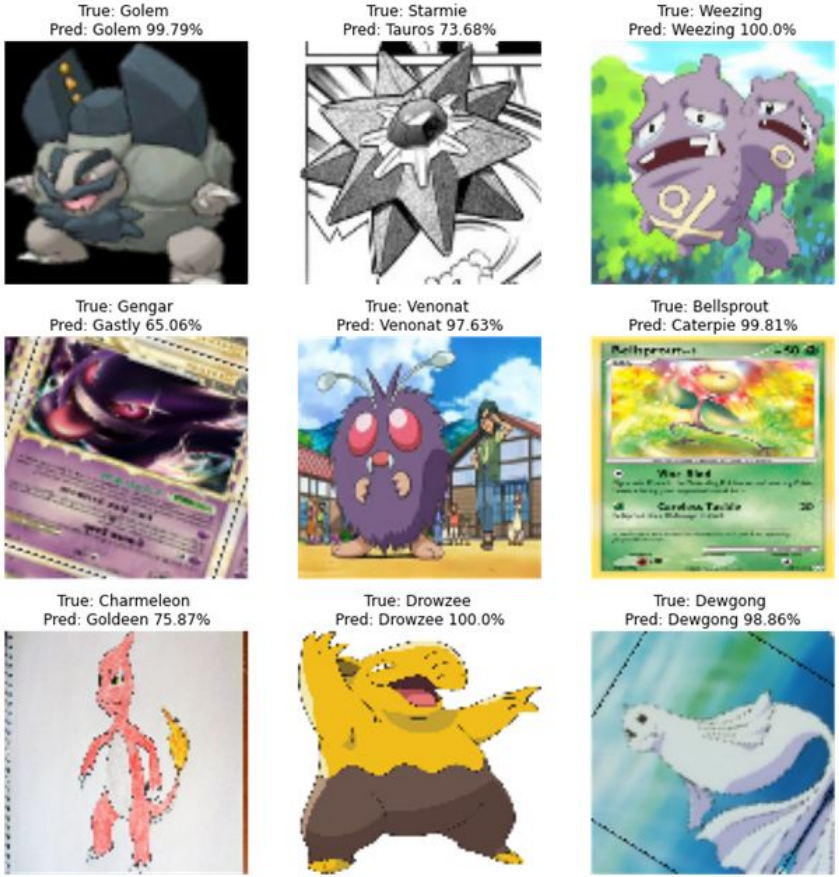
Experiments - Transfer Learning

Model	Epochs	Train Acc.	Test Acc.
Vgg16 - Flatten	20	0.9986	0.4231
Vgg16 + 256 Dense	20	0.6119	0.3979
Vgg16 + 512 Dense	20	0.4807	0.3977
Resnet - Flatten	20	0.9555	0.5391
Resnet + 256 Dense	20	0.9491	0.4262
Resnet + 512 Dense	20	0.8656	0.3957

Results


True: Koffing
Pred: Ditto 99.74%
True: Vileplume
Pred: Vileplume 99.96%
True: Drowzee
Pred: Drowzee 100.0%
True: Porygon
Pred: Porygon 97.52%
True: Dratini
Pred: Dratini 100.0%
True: Seadra
Pred: Seadra 99.23%
True: Zapdos
Pred: Zapdos 100.0%
True: Growlithe
Pred: Clefairy 49.43%
True: Pikachu
Pred: Pikachu 100.0%

Results


True: Golem
Pred: Golem 99.79%
True: Starmie
Pred: Tauros 73.68%
True: Weezing
Pred: Weezing 100.0%
True: Gengar
Pred: Gastly 65.06%
True: Venonat
Pred: Venonat 97.63%
True: Bellsprout
Pred: Caterpie 99.81%
True: Charmeleon
Pred: Goldeen 75.87%
True: Drowzee
Pred: Drowzee 100.0%
True: Dewgong
Pred: Dewgong 98.86%

Conclusion

- Classification with +100 classes is not an easy problem
- Need more images and a well prepared dataset
- Computational limitations