Image Recognition

Humpback Whale Identification Challenge

DATS 6203 SP2021
Group 2
Yuan Meng

Contents

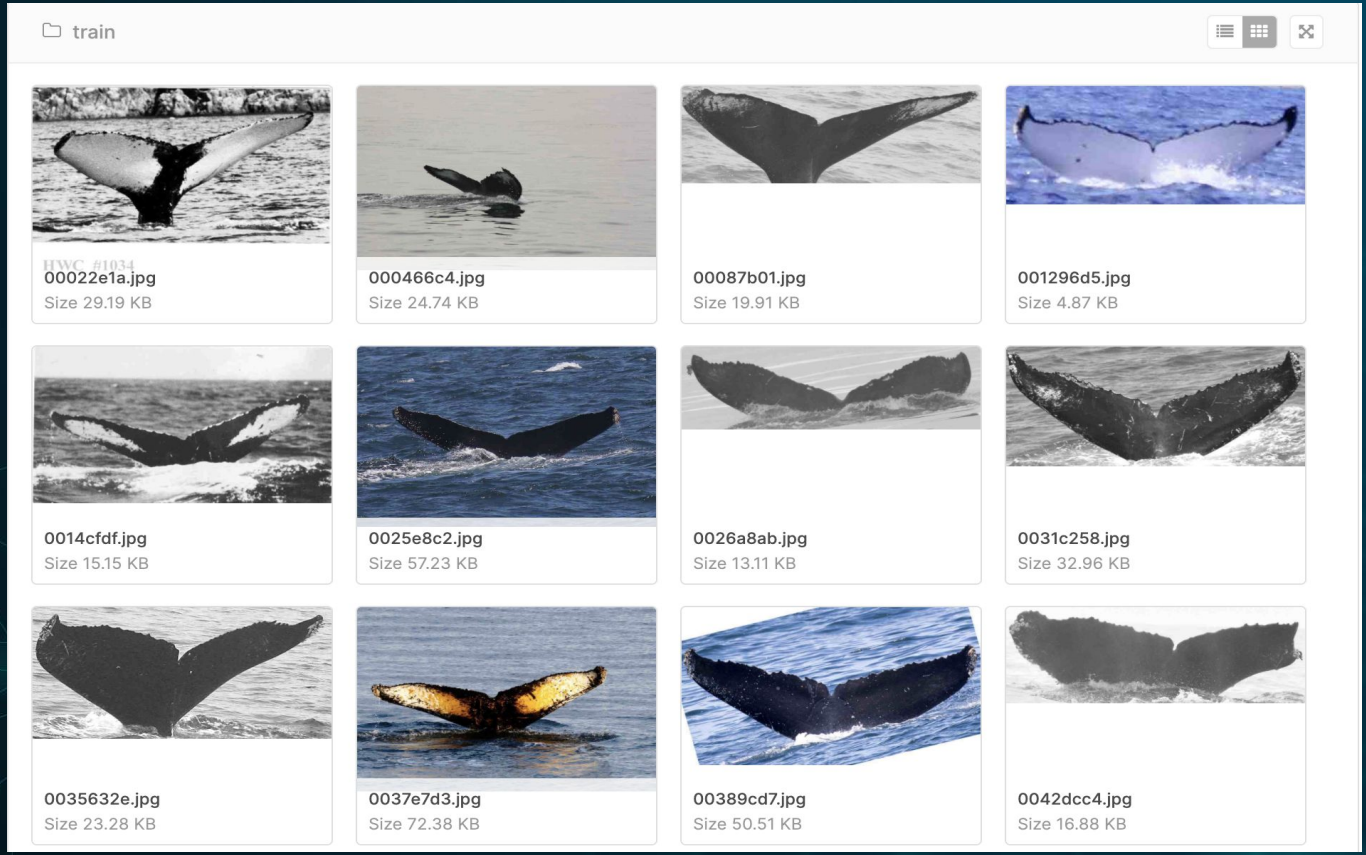
Problem & Dataset

- Shape of whales' tails
- Unique markings
- 25,000 images
  - Happy Whale
- 9850 training images
- 4246 IDs
  - W_1287fbc
  - new_whale

	Image	Id
0	00022e1a.jpg	w_e15442c
1	000466c4.jpg	w_1287fbc
2	00087b01.jpg	w_da2efe0
3	001296d5.jpg	w_19e5482
4	0014cfdf.jpg	w_f22f3e3
...	...	...
9845	ffe5c306.jpg	w_2ceab05
9846	ffeaa7a4.jpg	w_b067417
9847	ffecec63.jpg	w_8b56cb1
9848	fff04277.jpg	w_2dcbf82
9849	fffd4260.jpg	w_b9bfd4e

[9850 rows x 2 columns]

Some sample pictures:



Evaluation metric: Mean Average Precision (MAP)

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q}$$

Means

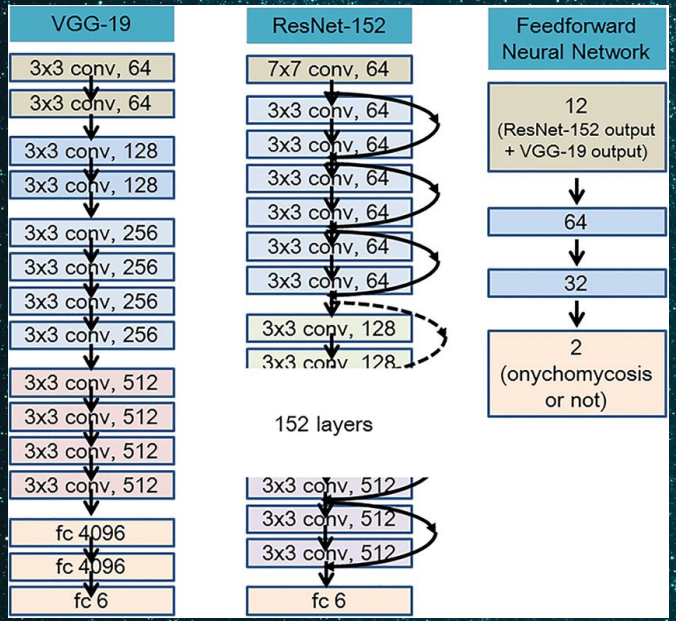
$$\frac{\text{Average number of precision per query}}{\text{Total number of queries}}$$

Here, Q is the number of queries.

	1st label	2nd label	3rd label	4th label	5th label
AP	1	1/2	1/3	1/4	1/5

Model Section

- ➔ Why transfer learning?
  - ◆ pre-trained networks
    - sufficient training datasets
    - reasonable architectures
- ➔ Why ResNets?
  - ◆ Residual Block
    - Skip Connections
    - Identity Mapping



Experiment Workflow

Step 1.

Data Processing

Step 2.

Network Setting

Step 3.

Parameter Tuning

Step 4.

Model Training

Step 5.

Model Testing

Step 6.

ID Predicting

Data Processing 1

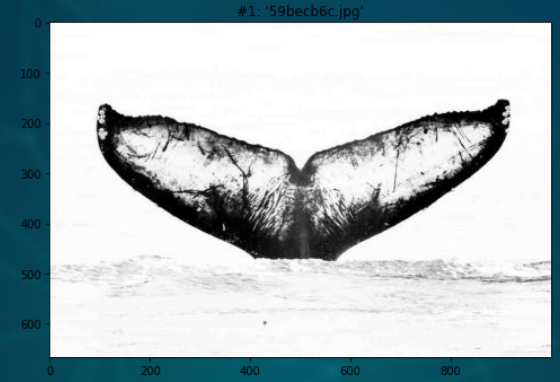
Explore data information

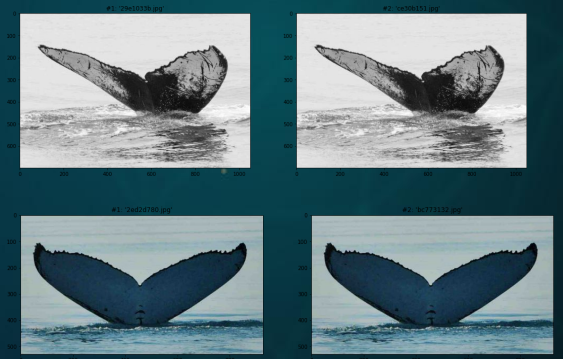
	Image	Id	Hash	Shape	Mode	New_Whale	Id_Count
0	00022e1a.jpg	w_e15442c	b362cc79b1a623b8	(699, 500)	L	False	1
1	000466c4.jpg	w_1287fbc	b3cccc3331cc8733	(1050, 700)	RGB	False	34
2	00087b01.jpg	w_da2efe0	bc4ed0f2a7e168a8	(1050, 368)	RGB	False	11
3	001296d5.jpg	w_19e5482	93742d9a2ab35b86	(397, 170)	RGB	False	1
4	0014cfdf.jpg	w_f22f3e3	d4a1dab1c49f6352	(700, 398)	L	False	2

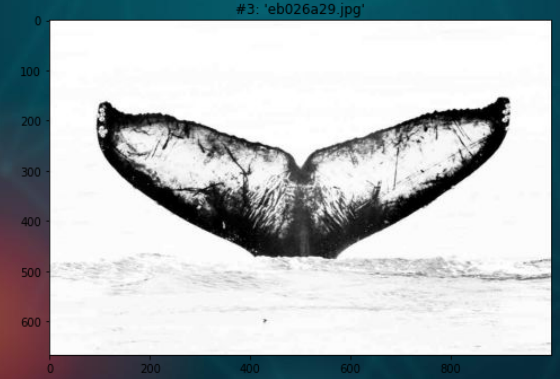
1. Duplicate Image Detection

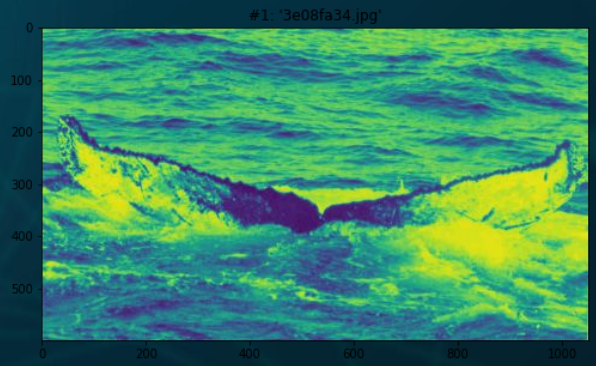
```
There are 780 duplicate images.
bb8ec43039cb663c    3
bcccd3346b342d0b    2
b619898ea6a6e1e9    2
e89a85e3b661d871    2
8f90e168da67b4c9    2
Name: Hash, dtype: int64
```

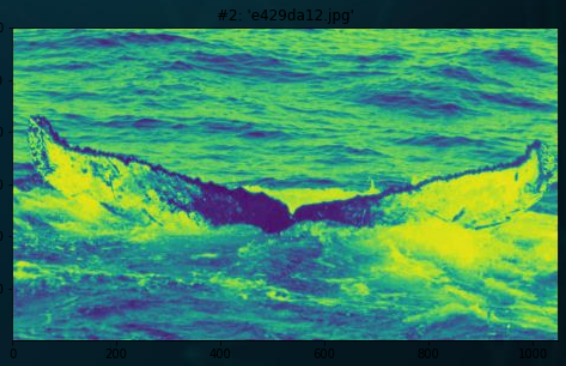
Duplicate Image Samples





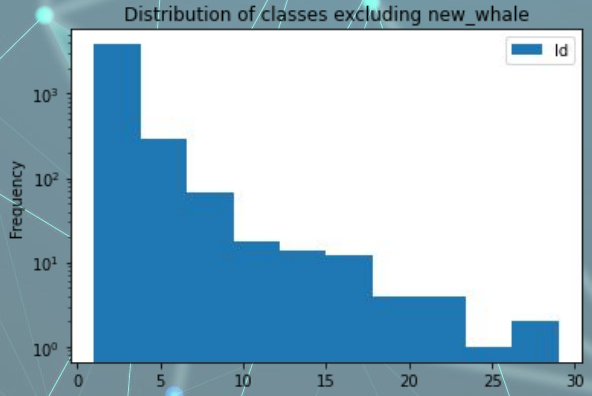




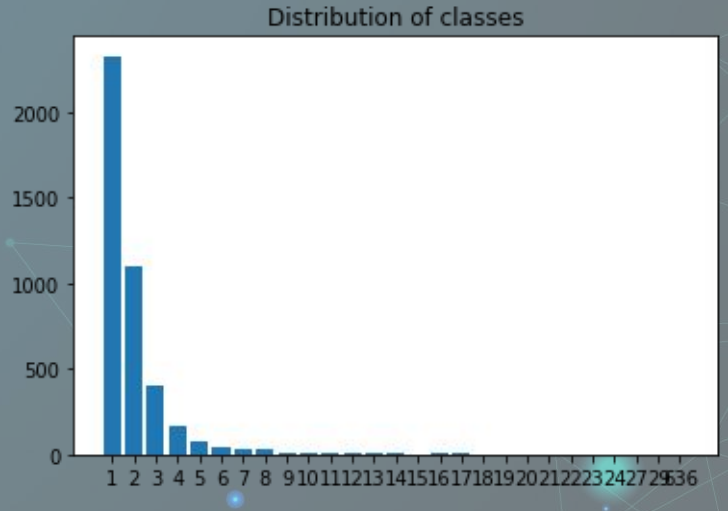


Data Processing 2

Class Distribution




```
new_whale    636
w_1287fbc     29
w_98baff9     27
w_7554f44     24
w_1eafe46     23
              ...
w_ab39440      1
w_7e8305f      1
w_f801078      1
w_c493795      1
w_3e9d82e      1
Name: Id, Length: 4246, dtype: int64
```

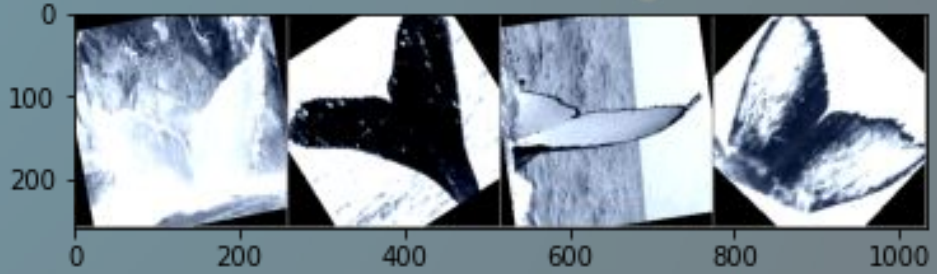


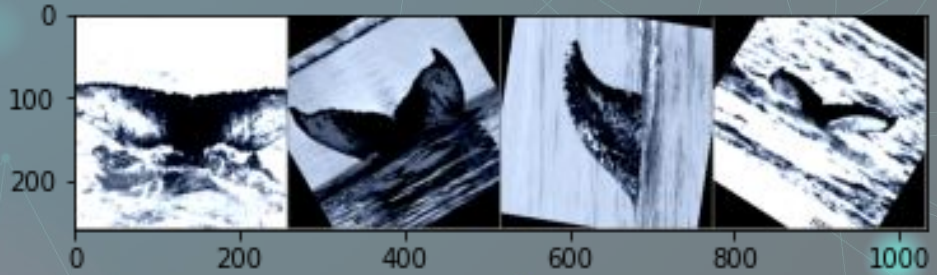
Method: Oversampling

Data Processing 3

Data Augmentation & Normalization

- Resize
- Central Crop
- Convert to Grayscale
- Random Horizontal Flip
- Random Vertical Flip
- Random Rotation
- To Tensor
- Normalize
  - per-channel mean [0.485, 0.456, 0.406]
  - per-channel standard deviation [0.229, 0.224, 0.225]





ResNet-50

Parameter Tuning

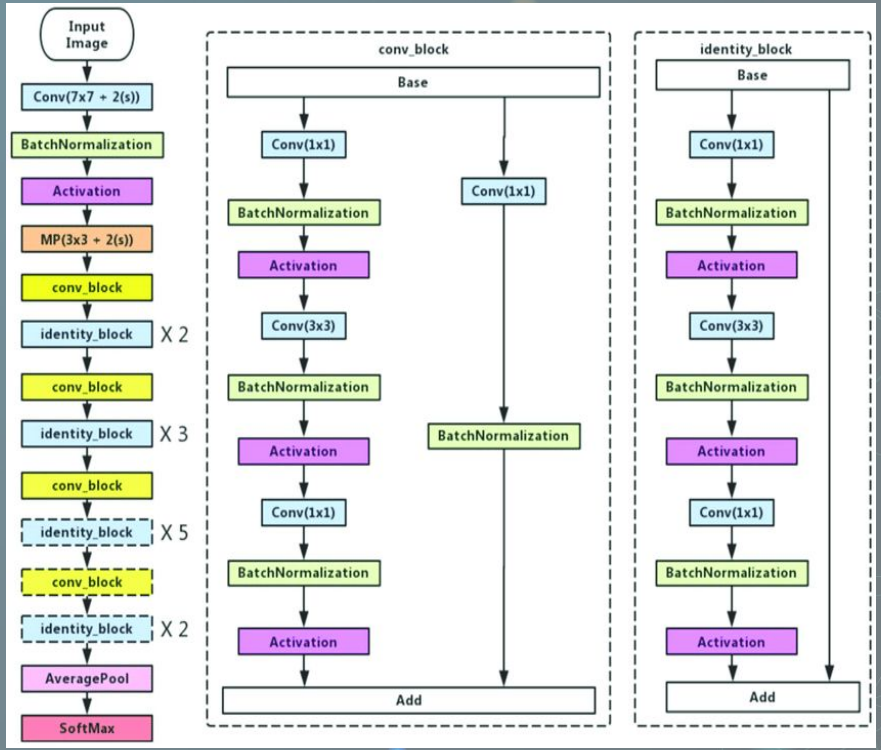
- Batch size = 128,
- Number of epochs = 20,
- Learning rate = 0.01
  - Annealed learning rate scheme
  - 0.001 after 17 epochs
  - 0.0001 after 19 epochs

Loss function & Optimizer

- Cross Entropy Loss function

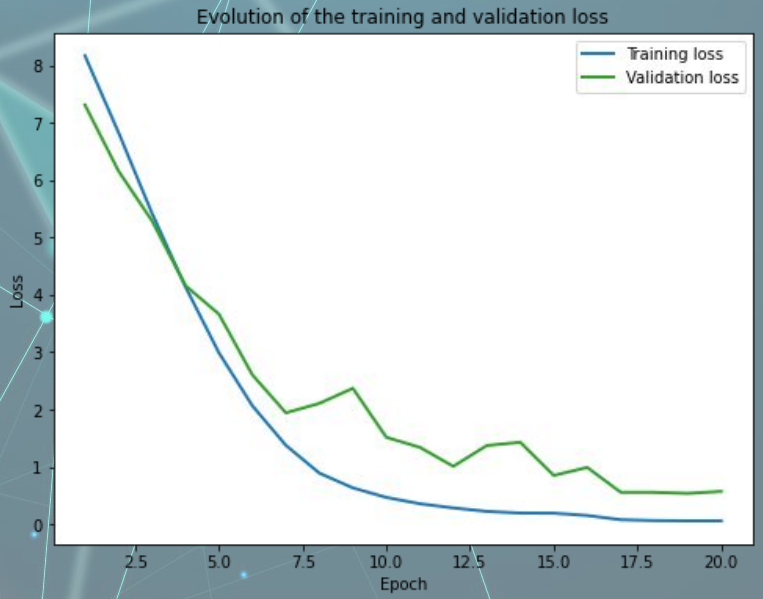
$$\text{loss}(x, class) = -\log\left(\frac{\exp(x[class])}{\sum_j \exp(x[j])}\right) = -x[class] + \log\left(\sum_j \exp(x[j])\right)$$

- SGD optimizer
  - 0.9 momenta
  - 1e-4 weight decay



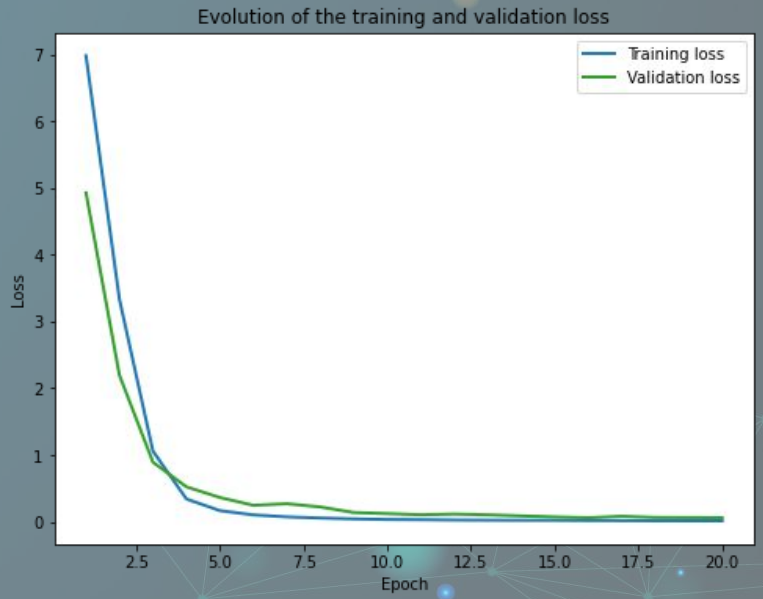
Loss & Mean Average Precision

Pretrained = False



- ★ Training Loss: 0.05
- ★ Validation Loss: 0.54
- ★ Mean average precision: 0.9344610007358328

Pretrained = True

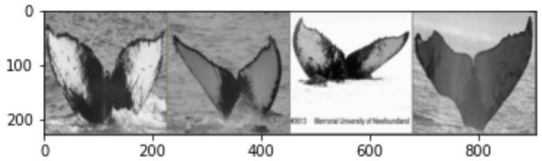


- ★ Training Loss: 0.01
- ★ Validation Loss: 0.06
- ★ Mean average precision: 0.9987398822663722

ID Prediction

Test in Validation data

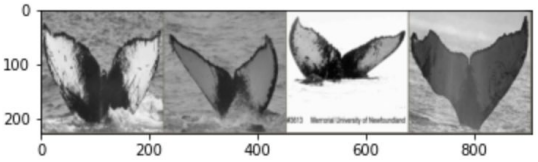
Ground truth:



w_5b99089 w_d19a884 w_8158b1f w_0d48a7d

Perfect Prediction

Predicted:



w_5b99089 w_d19a884 w_8158b1f w_0d48a7d

Final Submission

	Image	Id
0	00029b3a.jpg	w_711aaa1 w_67de30b w_d382236 w_fd1cb9d w_59349ea
1	0003c693.jpg	w_2863d51 w_b4369cc w_731e3aa new_whale w_89e159a
2	000bc353.jpg	w_da63cba w_1287fbc w_7311fe4 new_whale w_b34793e
3	0010a672.jpg	w_701972b w_8c605d2 w_9d65e55 w_e548eb7 w_9a4bd76
4	00119c3f.jpg	w_b6689cc w_9ceb05d w_44cccf6 w_bb2d34d w_abe383e
...	...	...
15605	fff31d26.jpg	w_aa16da4 w_d4eb9ed w_cae7677 w_c58b474 w_2f6a962
15606	fff3d049.jpg	new_whale w_fe87f0a w_987a36f w_03c84ef w_d88328d
15607	fffa5100.jpg	w_dad23fa w_654a5bb w_392bee3 w_63d1ea6 new_whale
15608	fffa6215.jpg	new_whale w_326e389 w_7e8b270 w_ec87420 w_06e47e3
15609	ffffdd7f.jpg	w_67de30b w_d6a9529 w_fd1cb9d w_fc7cc24 w_b0b275e

15610 rows × 2 columns

Result Analysis

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

- Customized Resnet
  - Layers not enough
- Resnet-152
  - Architecture too complex
- Resnet-50
  - Best fit

Conclusion

WHAT I LEARNED FROM THIS PROJECT

- Hands on experience
- “Residual Block” concept
- Data processing techniques

POTENTIAL FUTURE DIRECTION

- Different pre-trained networks
- Better oversampling methods

References

- https://www.kaggle.com/c/whale-categorization-playground/overview
- https://arxiv.org/pdf/1512.03385.pdf
- https://www.researchgate.net/figure/The-representation-of-model-architecture-image-for-ResNet-152-VGG-19-and-two-layered_fig2_322621180
- https://pytorch.org/hub/pytorch_vision_resnet/
- https://www.researchgate.net/figure/Left-ResNet50-architecture-Blocks-with-dotted-line-represents-modules-that-might-be_fig3_331364877
- https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html
- https://pytorch.org/docs/master/generated/torch.optim.SGD.html
- https://github.com/pytorch/vision/blob/master/torchvision/models/resnet.py
- https://www.kaggle.com/stehai/duplicate-images
- https://www.kaggle.com/stehai/duplicate-images-data-cleaning