USING MOBILE NEURAL NETWORK FOR PETS CLASSIFICATION

Jie Wen 1155092211

Supervised by Prof. Michael R. LYU

MAIN TOPIC IN THE 2ND SEMESTER

1. Similar Species Classification

2. Object Detection

3. Simple UI Improve & Explore

MOTIVATION

Why Mobile

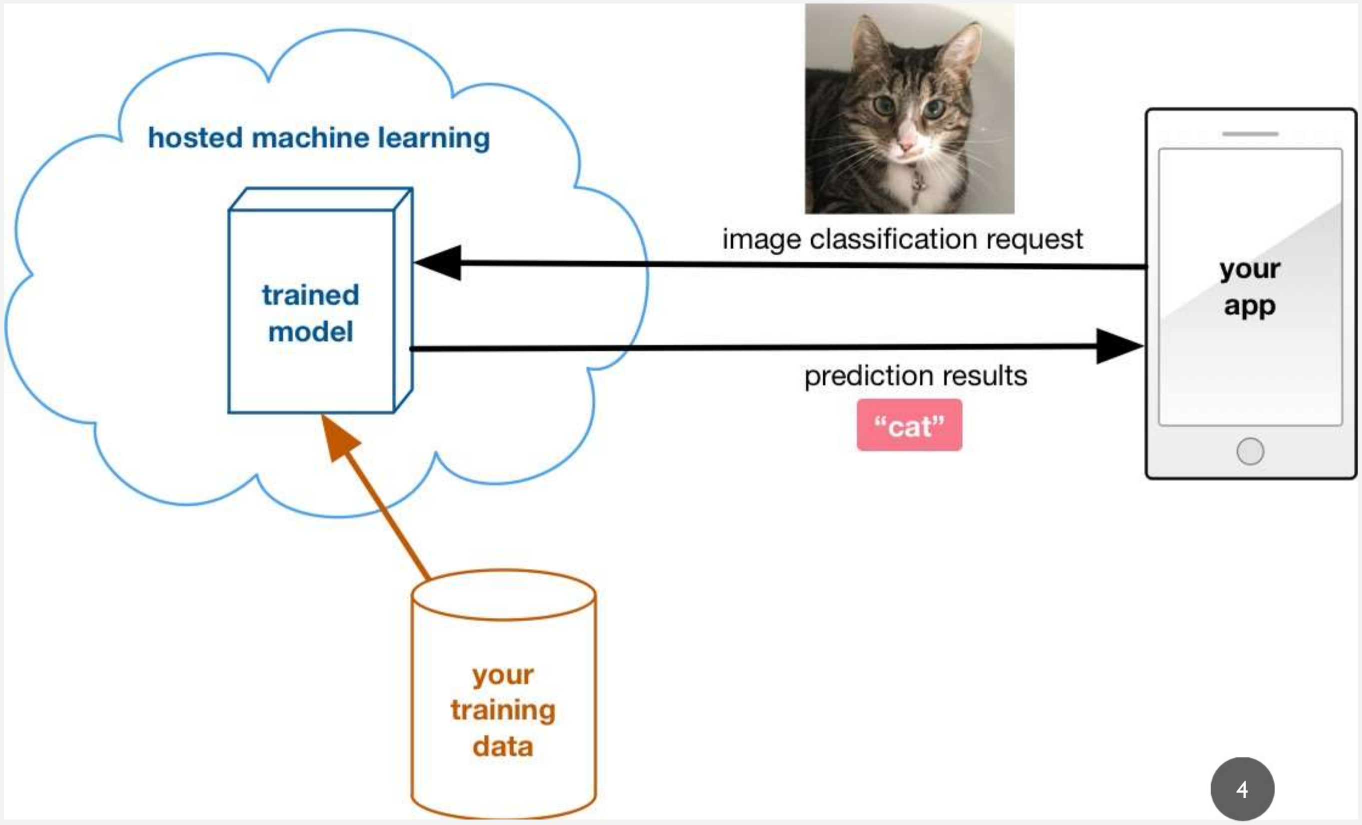
- General public
- Number of User
- Cost of time/money
- Wide usage scenario

MOTIVATION

Mobile implementation

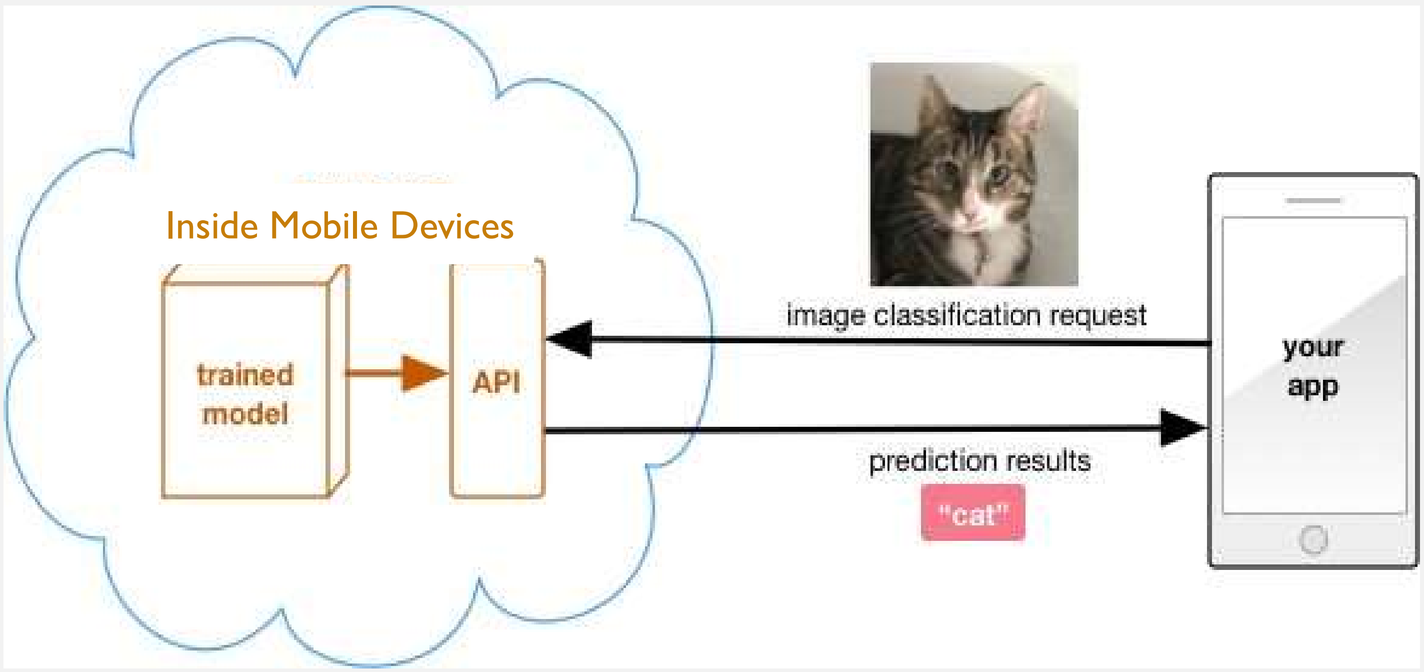
Need cloud server

Need Network



MOTIVATION

Mobile implementation



DATASET

Last term:

BotanWiki -> AnimalWiki
(Like ImageNet: dog, cat, cow, bird…)

AnimalWiki -> PetWiki
(Shiba, Husky, Scottish fold…)

DATASET

This term:

Crawled Dataset from Internet (22 species)

\+ Stanford Dogs Dataset (120 species)

= Dataset 133 (133 species, for similar species classification)

Oxford-IIIT Pets Dataset (37 species, for object detection)

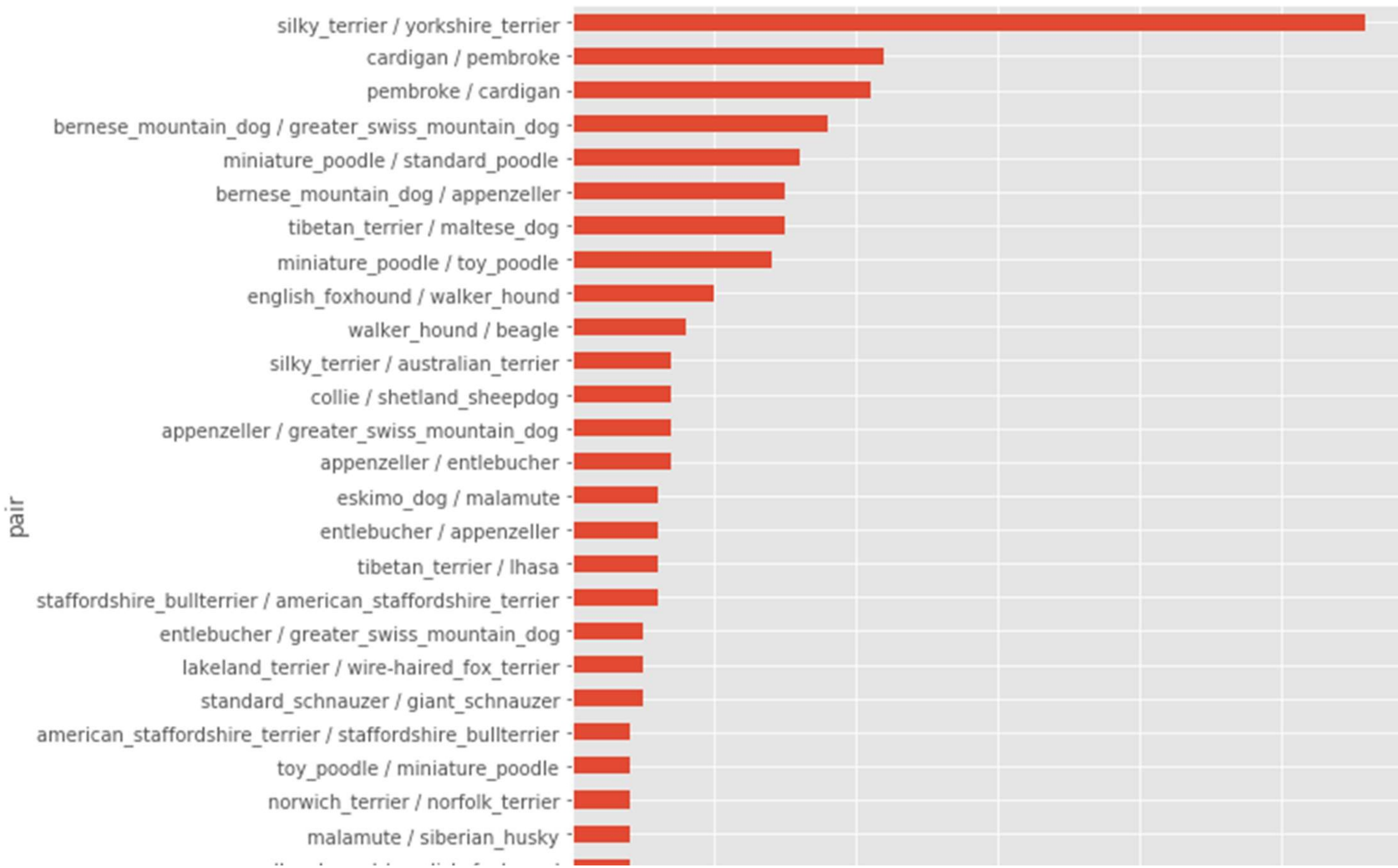
DATASET 133

About 23,500 images for 133 species (dogs)

129 dogs and 4 cats

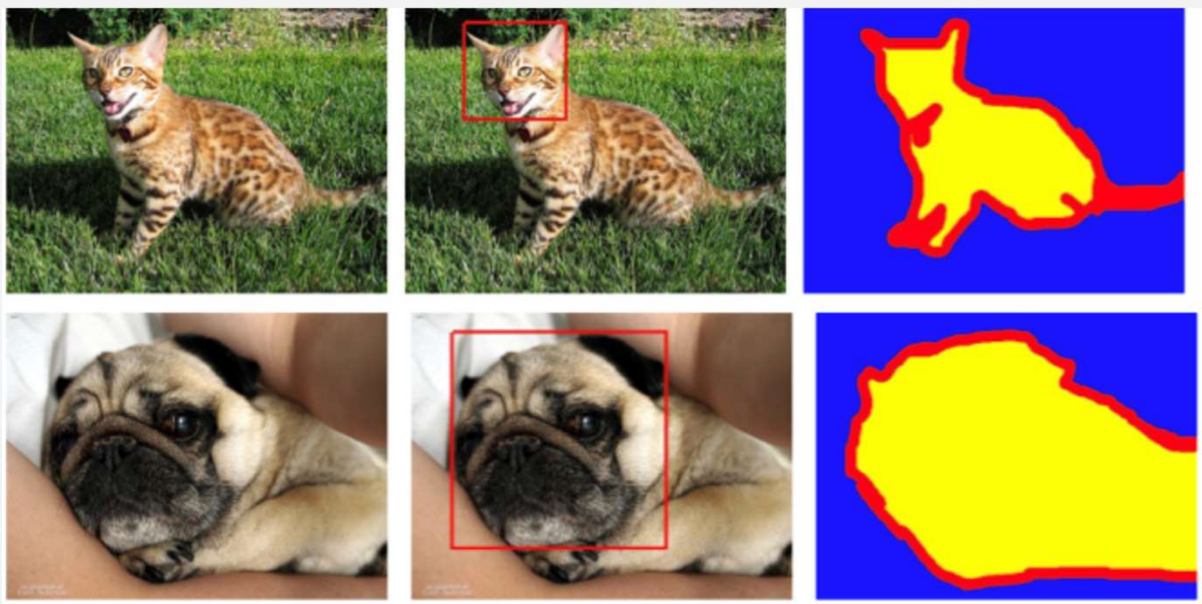
Large dog (31): German Shepherd, Greyhound, Saint Bernard,
Tibetan Mastiff, Samoyed, Scotch Collie, Husky…
Mid-size dog (53): Shiba, Black Shiba, Border Collie, Dalmatian,
Shar Pei, Pug. ..
Small dog (45): Bichon frise, Chihuahua, Corgi, Poodle, Schnauzer…

Cat (4): Bobcat, Persian Cat, Scottish Fold, Siamese Cat.


pair
silky_terrier / yorkshire_terrier
cardigan / pembroke
pembroke / cardigan
bernese_mountain_dog / greater_swiss_mountain_dog
miniature_poodle / standard_poodle
bernese_mountain_dog / appenzeller
tibetan_terrier / maltese_dog
miniature_poodle / toy_poodle
english_foxhound / walker_hound
walker_hound / beagle
silky_terrier / australian_terrier
collie / shetland_sheepdog
appenzeller / greater_swiss_mountain_dog
appenzeller / entlebucher
eskimo_dog / malamute
entlebucher / appenzeller
tibetan_terrier / lhasa
staffordshire_bullterrier / american_staffordshire_terrier
entlebucher / greater_swiss_mountain_dog
lakeland_terrier / wire-haired_fox_terrier
standard_schnauzer / giant_schnauzer
american_staffordshire_terrier / staffordshire_bullterrier
toy_poodle / miniature_poodle
norwich_terrier / norfolk_terrier
malamute / siberian_husky

OXFORD-IIIT PETS DATASET

37 species with roughly 200 images for each class



OXFORD-IIIT PETS DATASET

Oxford VS Stanford
Why use Oxford Dataset in object detection:

1. Complexity of using TensorFlow-gpu in CSE server without root/sudo access: TensorFlow/CuDNN outdated -> Anaconda exceed disk quota -> Miniconda can't find $PATH. Without multiple GPU, Stanford dataset would be painful.

Basically there'll be a lot of problems once TensorFlow/cuda/cudnn is outdated. Finally I moved all my data to Google Cloud.

OXFORD-IIIT PETS DATASET

Oxford VS Stanford
Why use Oxford Dataset:

2. Limitation of computation power.

37 species with a simple MobileNet-v1 structure took me a whole day to get the result on Google Cloud. It would be too time-consuming to use Stanford dataset.

MODEL

Wchich model to choose?

Inception (V3)

MobileNet

Faster-RCNN

Mask-RCNN

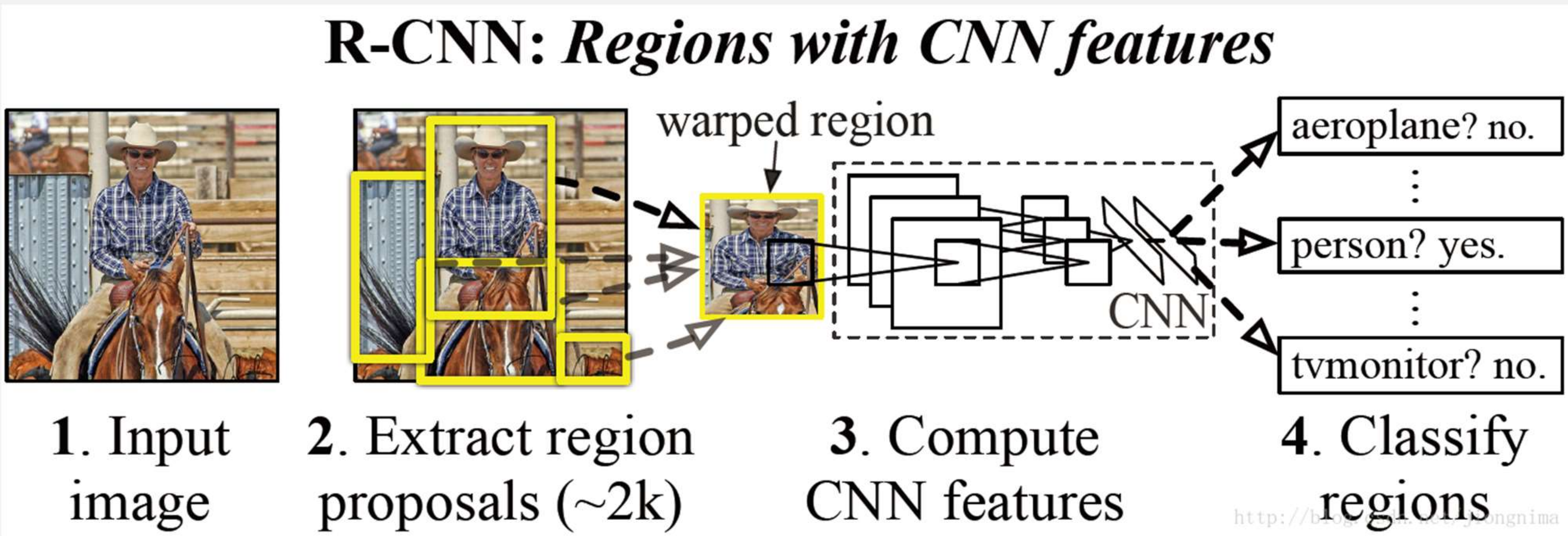
MODEL

R-CNN(CVPR 2014) -> Fast R-CNN(2015)

-> Faster R-CNN(2016) -> Mask-RCNN(ICCV 2017)

MODEL

R-CNN


R-CNN: Regions with CNN features
warped region
aeroplane? no.
person? yes.
tvmonitor? no.
CNN
1. Input image
2. Extract region proposals (~2k)
3. Compute CNN features
4. Classify regions

MODEL

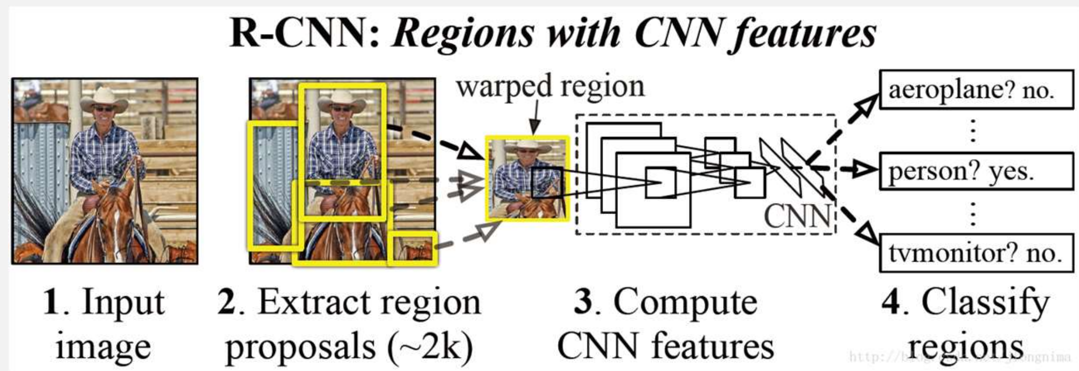
R-CNN

1. Extract 2k regions

2. Each of them go through CNN one by one to extract features.

3. Use SVM to classify regions.

4. Adjust the region through bounding box regression

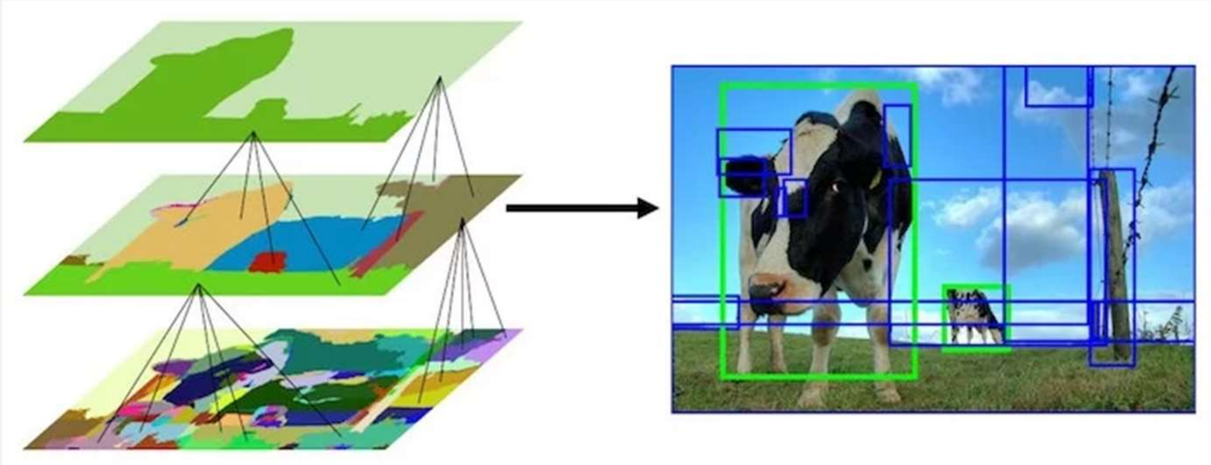


MODEL

R-CNN

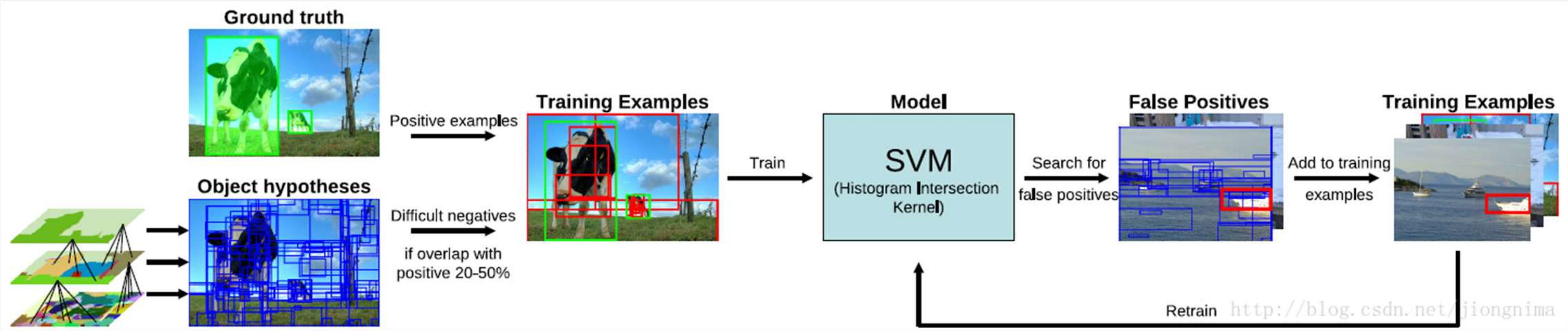
How to extract regions:
Selective Search

a) Based on traditional methods to segment images
b) Combine segments based on similarity and then go back to a).
c) Keep doing this and we will have the result at the right.



MODEL

R-CNN



Ground truth
Object hypotheses
Positive examples
Difficult negatives
if overlap with positive 20-50%
Training Examples
Train
Model
SVM
(Histogram Intersection Kernel)
Search for false positives
False Positives
Add to training examples
Training Examples
Retrain
http://blog.csdn.net/jiongnima

MODEL

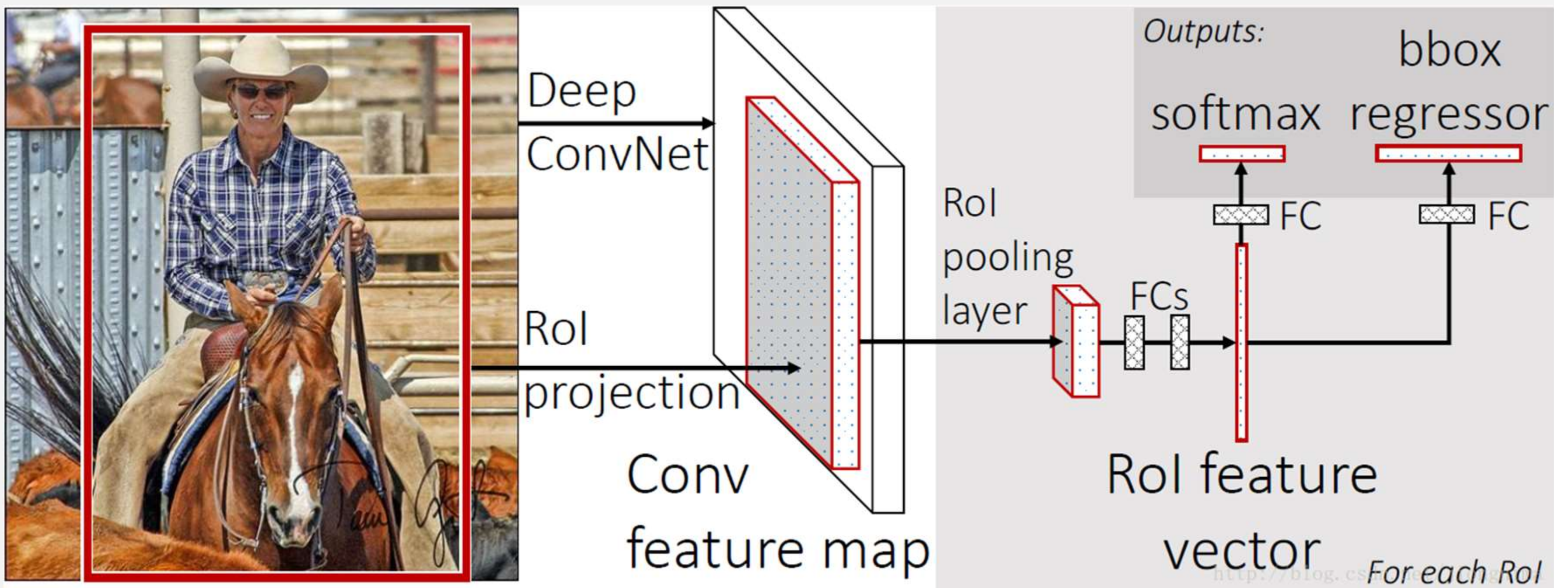
R-CNN

Pros: Use CNN to extract features.
Use bounding box regression to adjust final result.

Cons: Selective Search is time-consuming.
(Series) CNN forward propagation is time-consuming.
Each parts trained separately, waste of time & space.

MODEL

Fast R-CNN

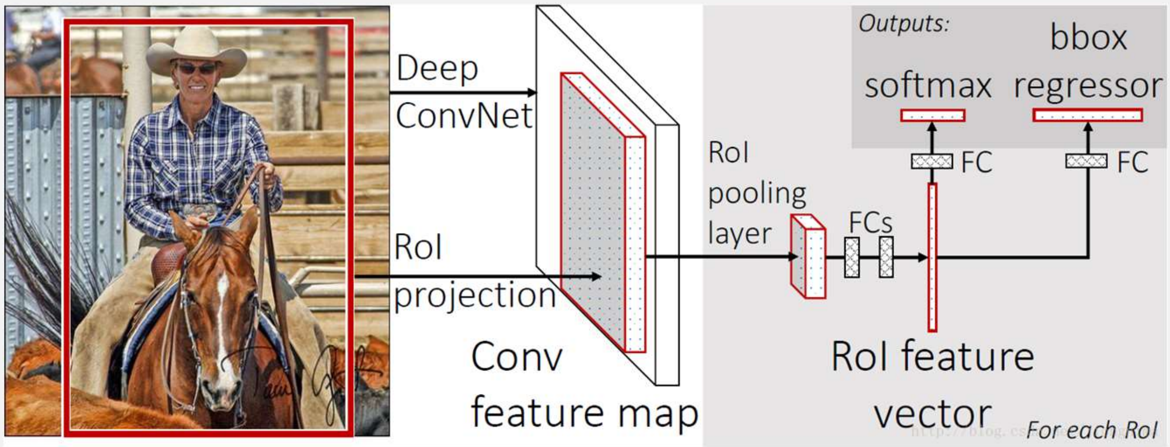


MODEL

Fast R-CNN

Still use selective search, and a neural network to extract features on the whole graph.

After that, an RoI Pooling Layer will be used to extract features from feature map and pass to FC Layer for correction.



MODEL

Fast R-CNN

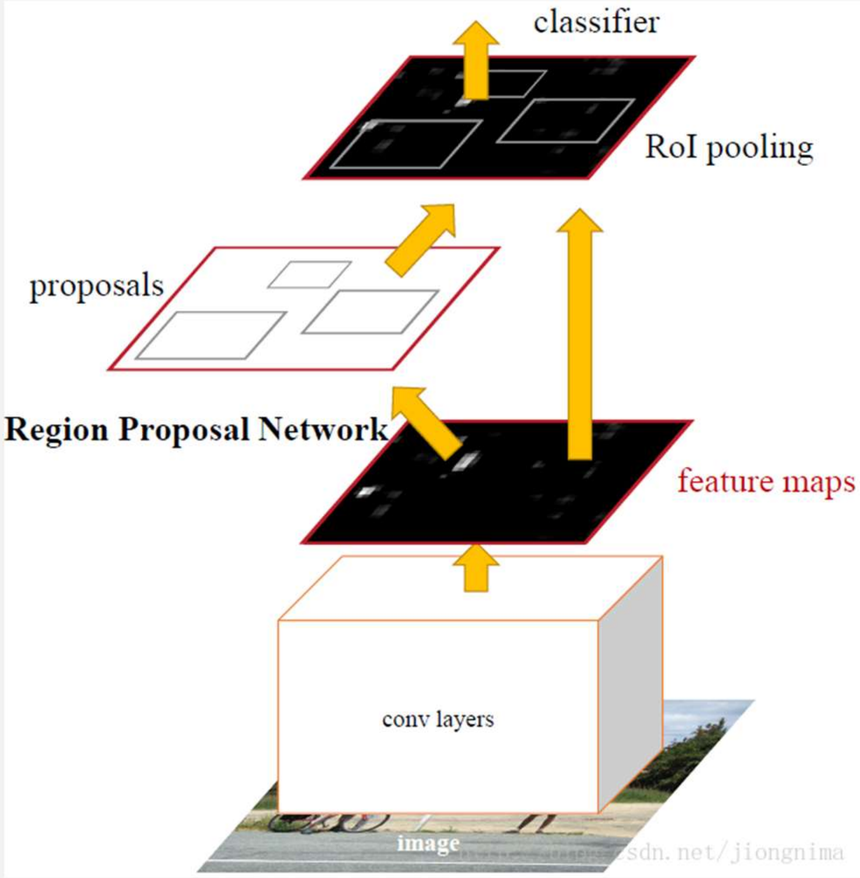
Pros: Use a NN to extract features based on the whole image instead of doing it one by one.
The other parts could be combined during tanning except for selective search.

Cons: Selective Search is still there.

MODEL

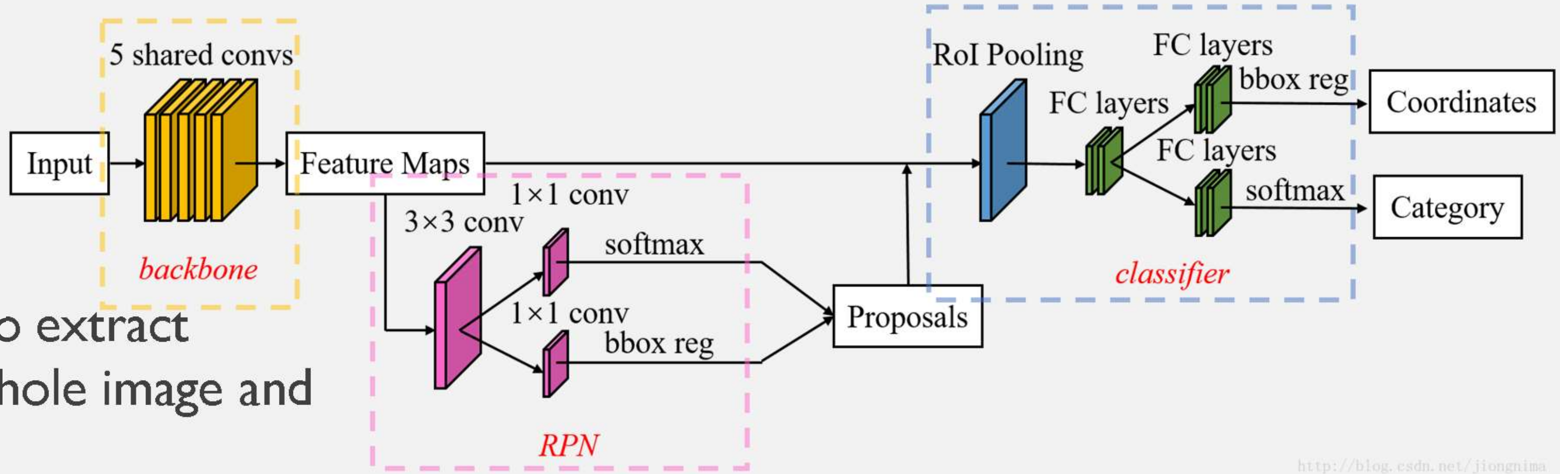
Faster R-CNN

Use Regional Proposal Network(RPN) to replace selective search.



MODEL

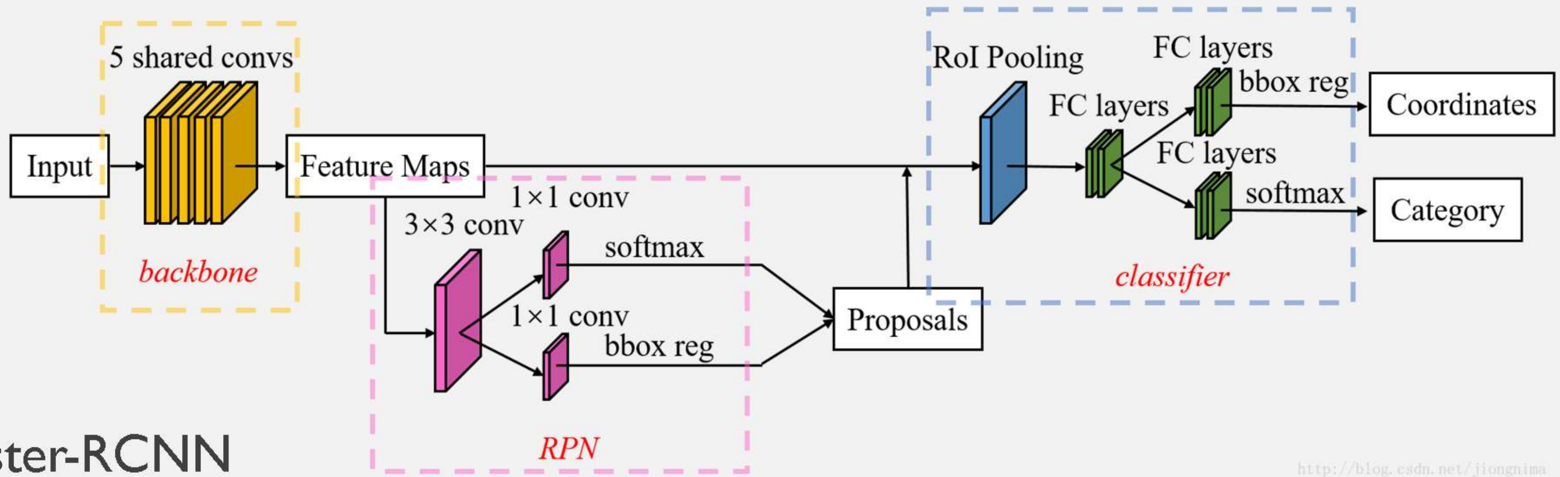
Faster R-CNN



1. Use backbone to extract features for the whole image and pass to RPN.
2. RPN will generate bounding box of ROI and slightly fix it.
3. ROI polling layer will select feature for each ROI and FC layer will do classification.

MODEL

Faster R-CNN



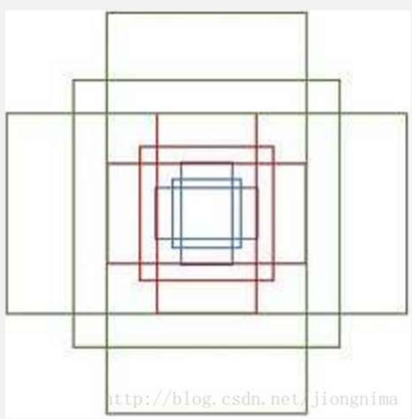
As we can see, Faster-RCNN mainly consists of 3 parts: backbone, RPN and classifier. Let's loot at RPN first.

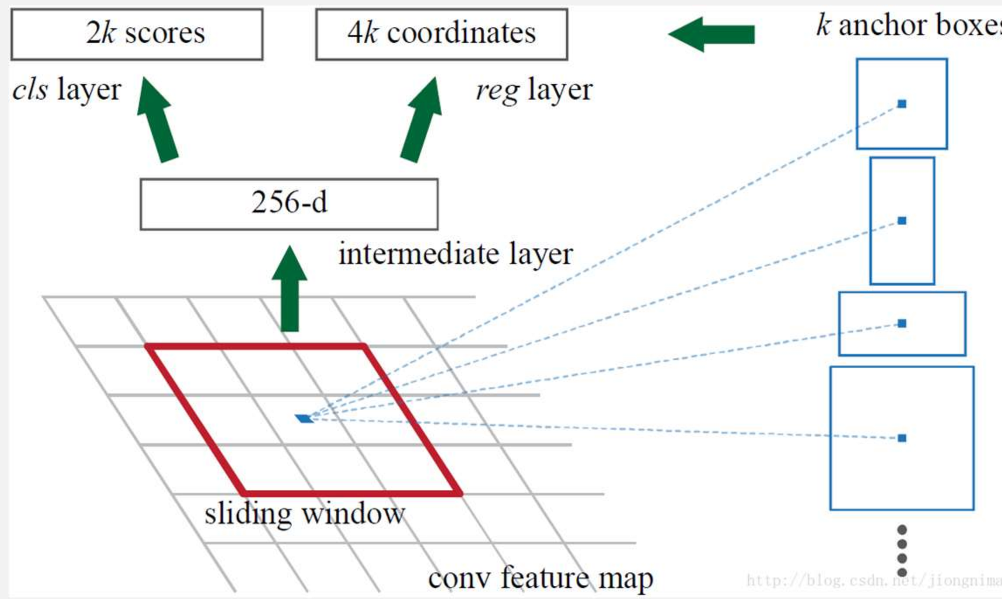
MODEL

Faster R-CNN

RPN relies on the sliding window to generate 9 pre-defined anchors for each position.

The 9 pre-defined anchors can be shown as:
(3 sizes, 128*128,
256*256, 512*512.
3 width-height ratios:
1:1, 1:2, 2:1)





MODEL

Faster R-CNN

RPN

There will be 40*60 shared feature map, and thus 40*60*9(~20000) anchors.
For each anchor, RNP needs to decide whether it is front or back(cover the object?) and adjust it if it's front.

For the 1st question, RPN use SoftmaxLoss to train.
For the 2nd question, RPN use SmmothL1Loss to train.

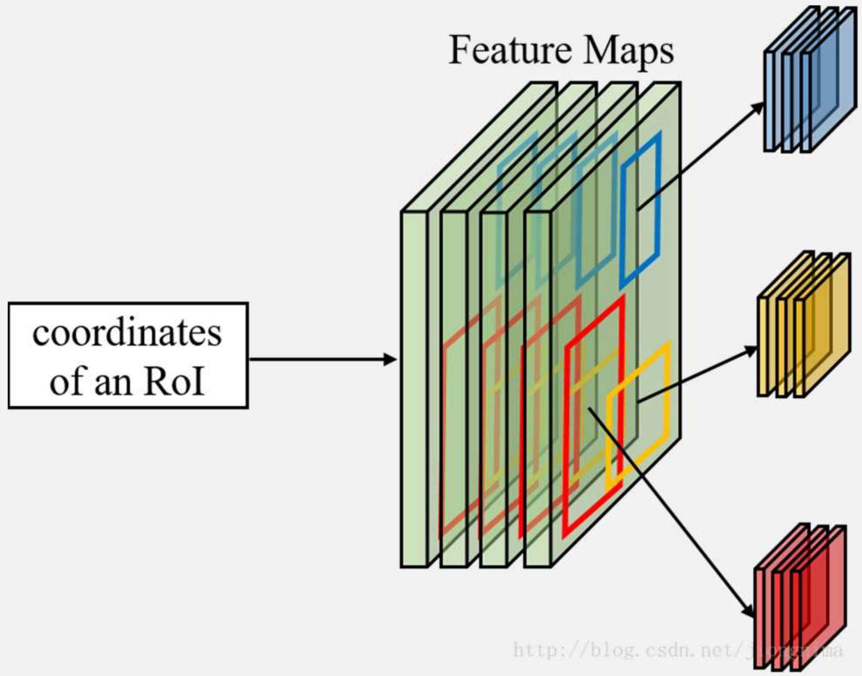
MODEL

Faster R-CNN

ROI Polling

For each ROI, we need to get it from the combined feature map and send it to classifier.

ROI Polling will select feature for each ROI and convert the dimension to meet the FC layer requirement.
E.g.: ROI polling in the right will pick out the feature for each ROI and convert it to 6*6.



MODEL

Faster R-CNN

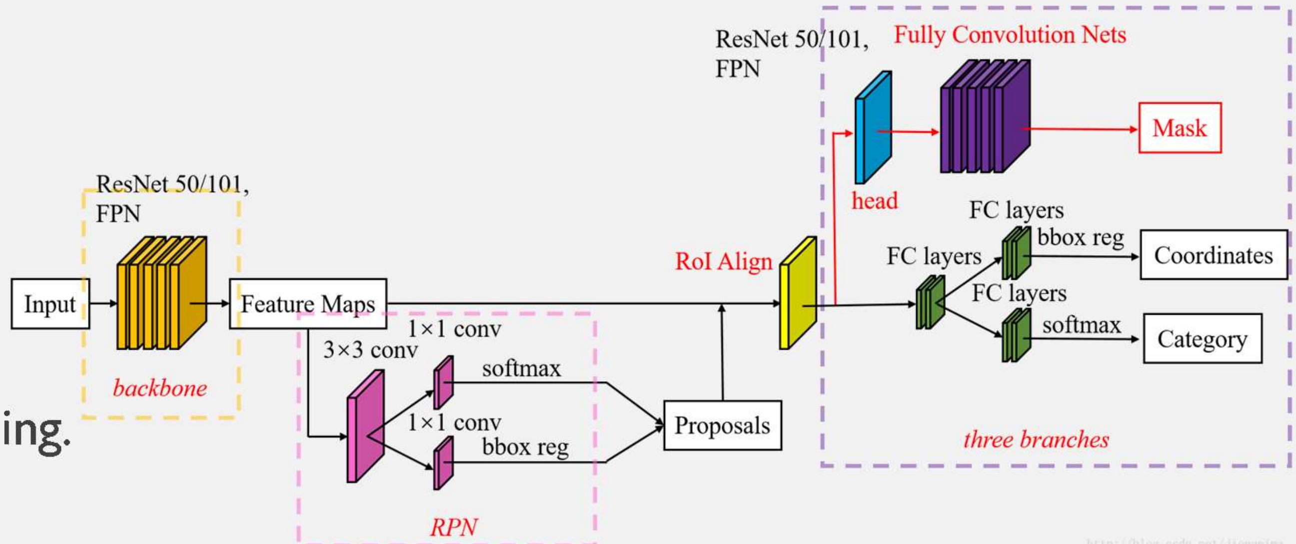
Classifier and bounder adjust

Classifier: see what exactly is this ROI (Human, car, flower)

bounder adjust: also use SmmothL1Loss to adjust non-background ROI bounders.

MODEL

Mask R-CNN



1. ROI polling -> ROI Align:
Better align with the original ROI region compared to polling.

2. Mask branch
FCN SoftmaxLoss -> average binary cross-entropy loss of K Mask predicts

MODEL

Mask R-CNN

Great result with great cost.

	backbone	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
MNC [7]	ResNet-101-C4	24.6	44.3	24.8	4.7	25.9	43.6
FCIS [20] +OHEM	ResNet-101-C5-dilated	29.2	49.5	-	7.1	31.3	50.0
FCIS+++ [20] +OHEM	ResNet-101-C5-dilated	33.6	54.5	-	-	-	-
Mask R-CNN	ResNet-101-C4	33.1	54.9	34.8	12.1	35.6	51.1
Mask R-CNN	ResNet-101-FPN	35.7	58.0	37.8	15.5	38.1	52.4
Mask R-CNN	ResNeXt-101-FPN	**37.1**	**60.0**	**39.4**	**16.9**	**39.9**	**53.5**

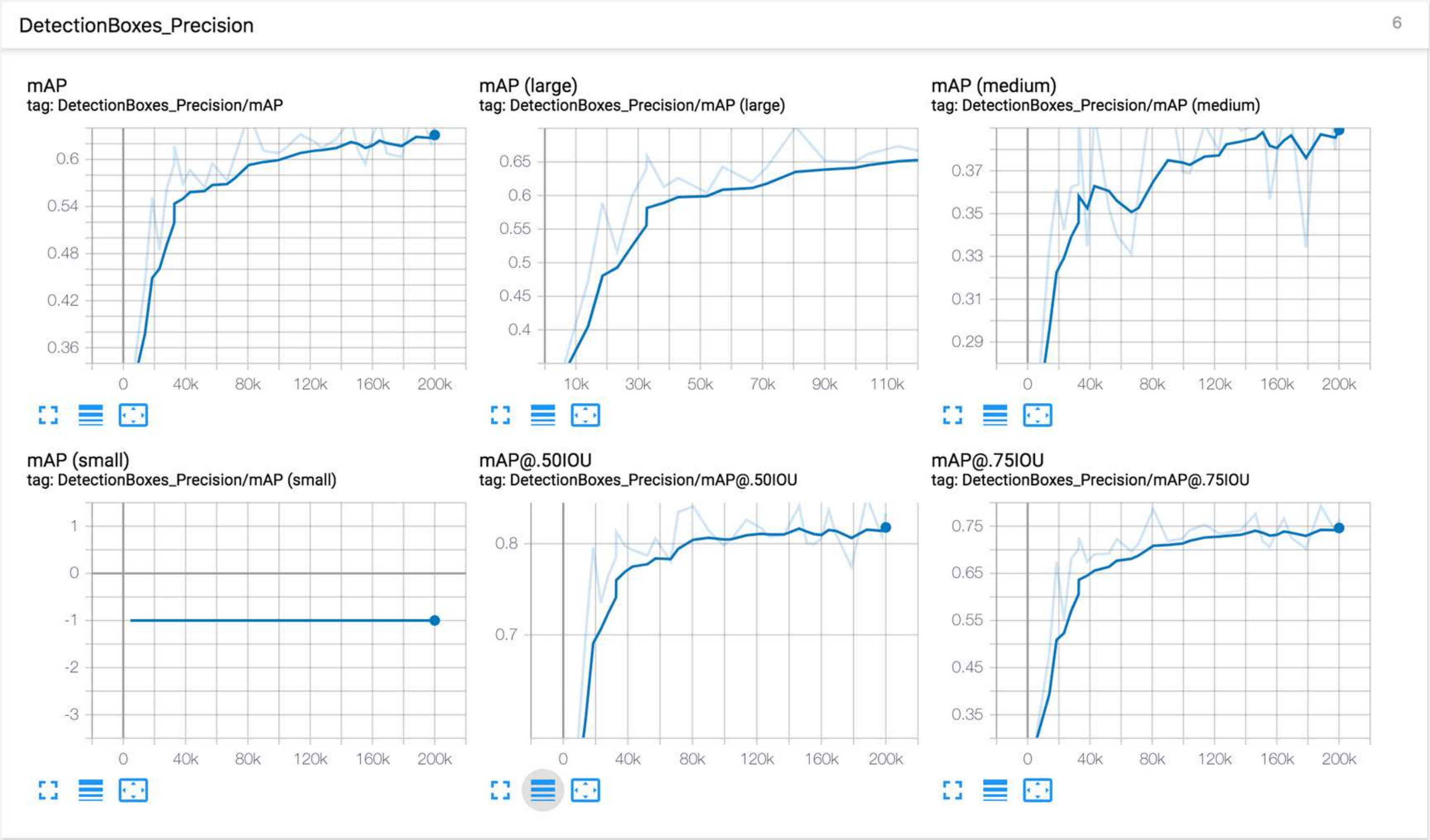
RESULT

Training Accuracy &
Model Size

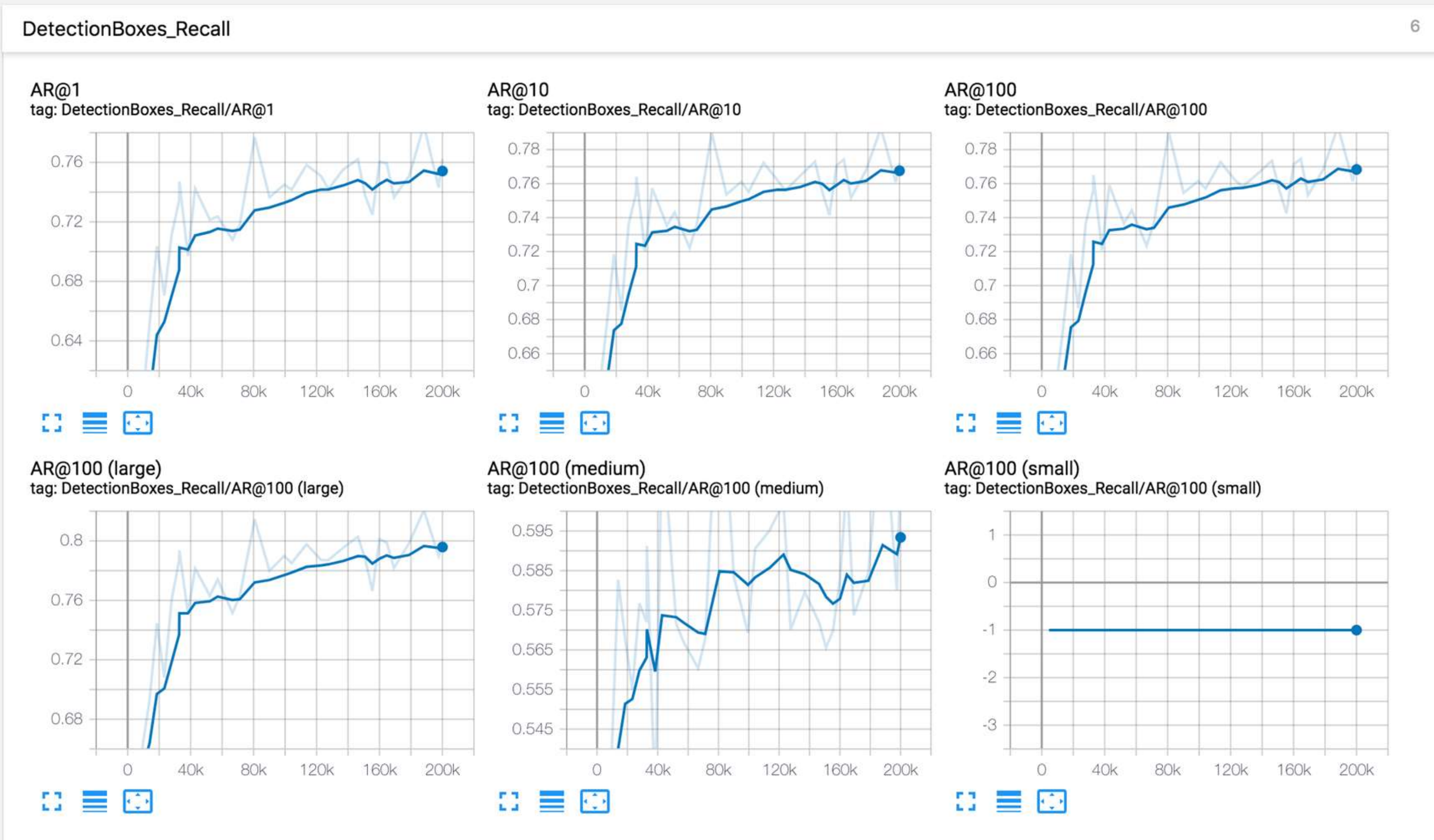
80% Training
10% Validation
10% Testing

Inception v3	**Accuracy: 0.98 Size: 88M**
MobileNet 100, 224	**Accuracy: 0.94 Size: 10M**
MobileNet 050, 224	**Accuracy: 0.92 Size: 3M**
MobileNet 050, 128	**Accuracy: 0.91 Size: 3M**
MobileNet 035, 224	**Accuracy: 0.94 Size: 2M**

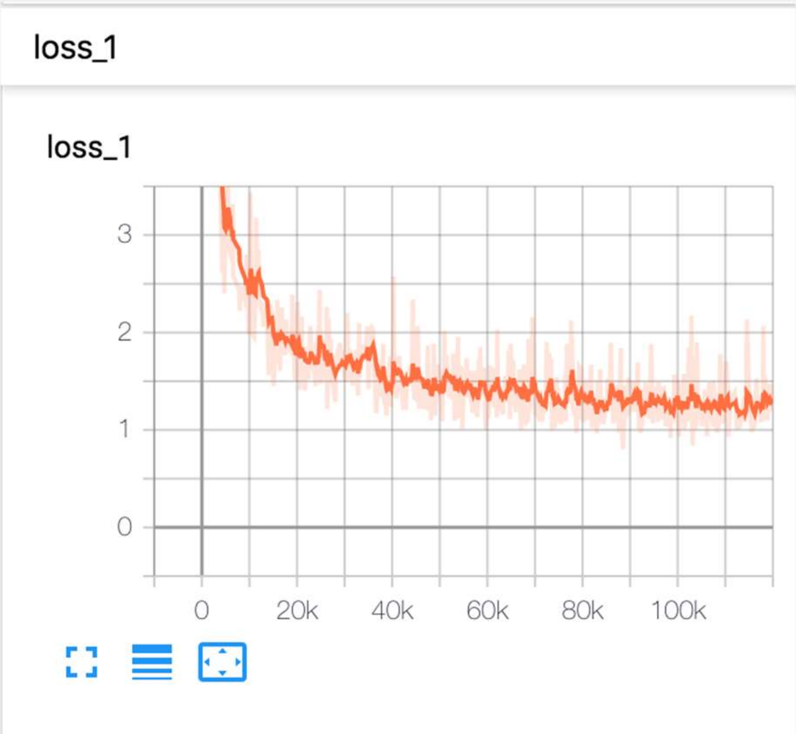
RESULT


DetectionBoxes_Precision
6
mAP
tag: DetectionBoxes_Precision/mAP
mAP (large)
tag: DetectionBoxes_Precision/mAP (large)
mAP (medium)
tag: DetectionBoxes_Precision/mAP (medium)
mAP (small)
tag: DetectionBoxes_Precision/mAP (small)
mAP@.50IOU
tag: DetectionBoxes_Precision/mAP@.50IOU
mAP@.75IOU
tag: DetectionBoxes_Precision/mAP@.75IOU

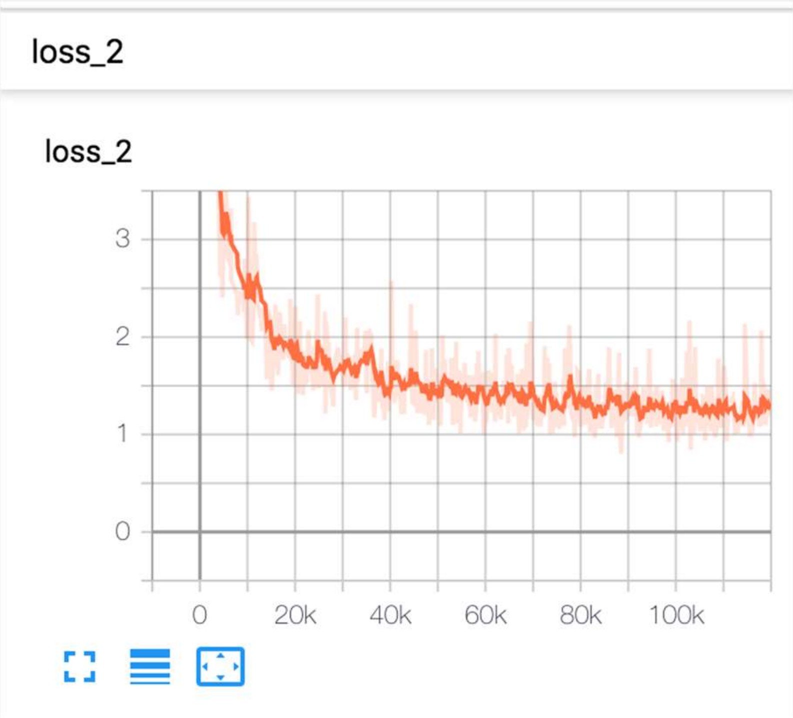
RESULT


DetectionBoxes_Recall
6
AR@1
tag: DetectionBoxes_Recall/AR@1
AR@10
tag: DetectionBoxes_Recall/AR@10
AR@100
tag: DetectionBoxes_Recall/AR@100
AR@100 (large)
tag: DetectionBoxes_Recall/AR@100 (large)
AR@100 (medium)
tag: DetectionBoxes_Recall/AR@100 (medium)
AR@100 (small)
tag: DetectionBoxes_Recall/AR@100 (small)
0
40k
80k
120k
160k
200k

RESULT



loss_1
loss_1
3
2
1
0
0
20k
40k
60k
80k
100k



loss_2
loss_2
3
2
1
0
0
20k
40k
60k
80k
100k

DEMO - CLASSIFICATION

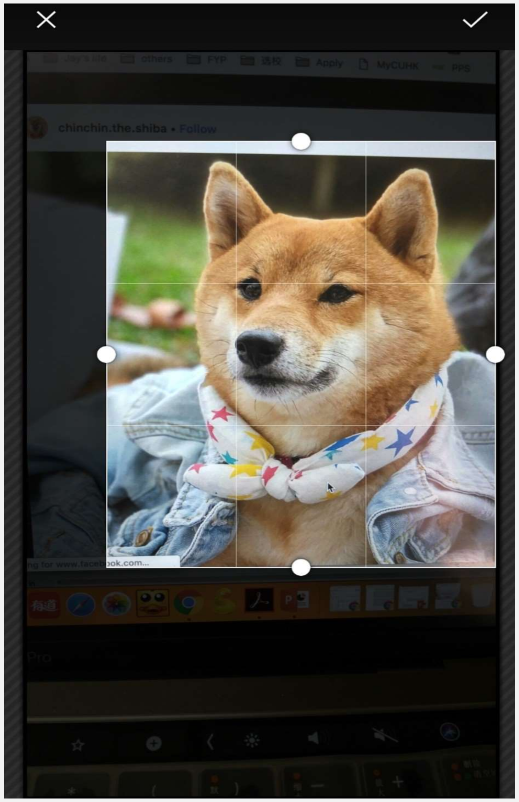
Application

1. Take/Choose photo with Inception model
2. Real-time Classify with Inception model

DEMO - CLASSIFICATION

Take/Choose photo with Inception model

1. Take & Crop a photo in APP

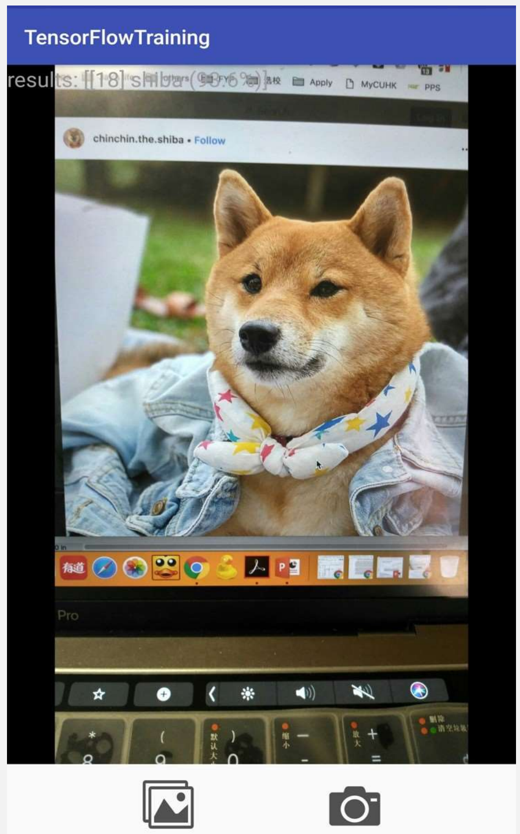


DEMO - CLASSIFICATION

Take/Choose photo with Inception model

1. Take a photo in APP

2. Choose a photo from album: Crop & Not Crop

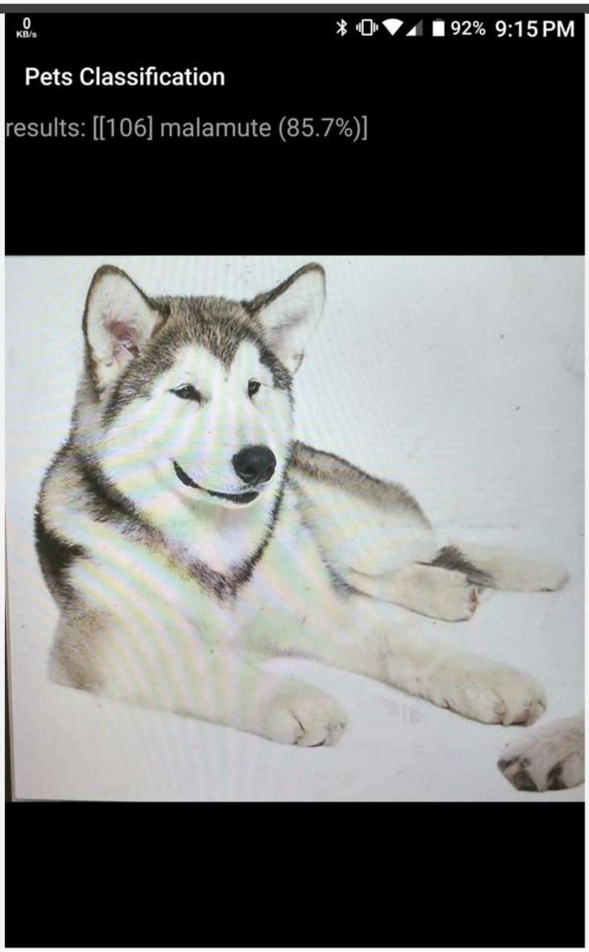


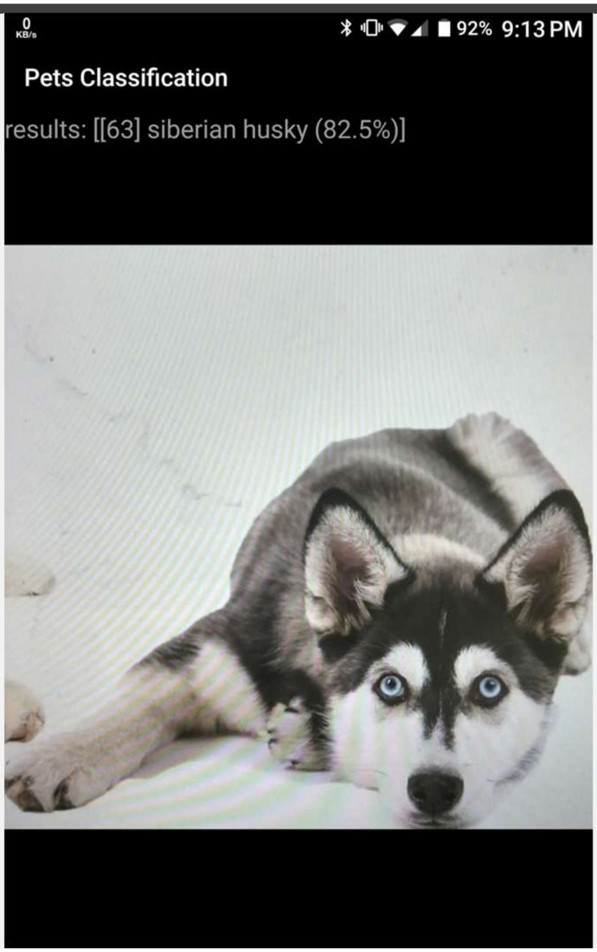
	SIBERIAN HUSKY	ALASKAN MALAMUTE
Picture		
Origin	Siberia	Alaska
Size	51 -60 cm	58 - 71 cm
Weight	16 -34 kg	39 - 57 kg
Function	To carry a light load at moderate speed over great distances	To carry a heavy load
Eyes	Blue or Brown	Only Brown
Ears	Set high on the head	Set wide apart on the head
Tail	Fox brush carried in a sickle	A waving plume
Personality Traits	Highly Active & Vocal	Laid Back
	Friendly towards other dogs	Gender aggresive towards dogs of the same sex
	No loyality to one person - they love everyone & everything	Family orientated - Babysat the Mahlemut children in the tribe

DEMO - CLASSIFICATION

3. Similar Species (Husky vs Malamute, High Quality)

DEMO - CLASSIFICATION

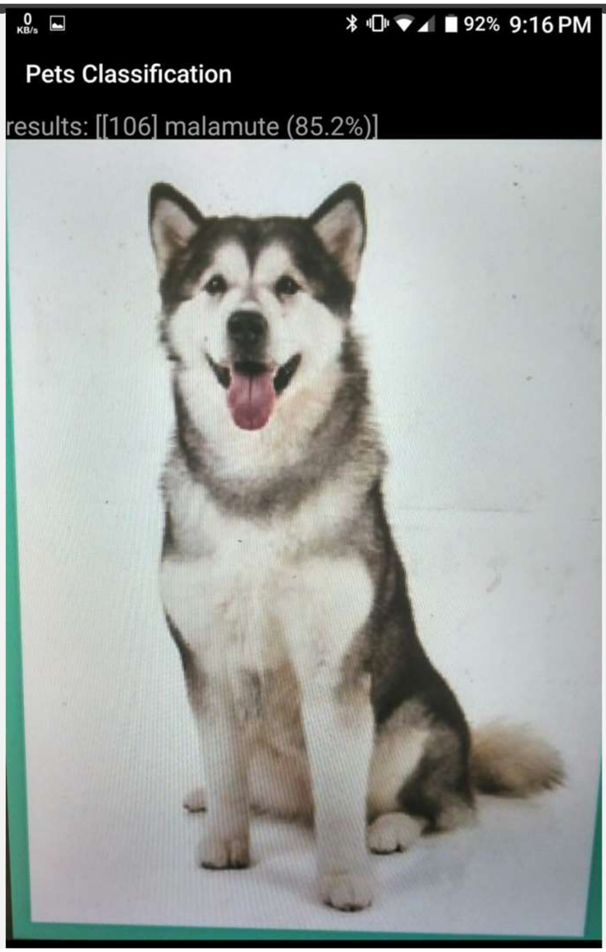




DEMO - CLASSIFICATION

3. Similar Species (Husky vs Malamute, Normal Quality)

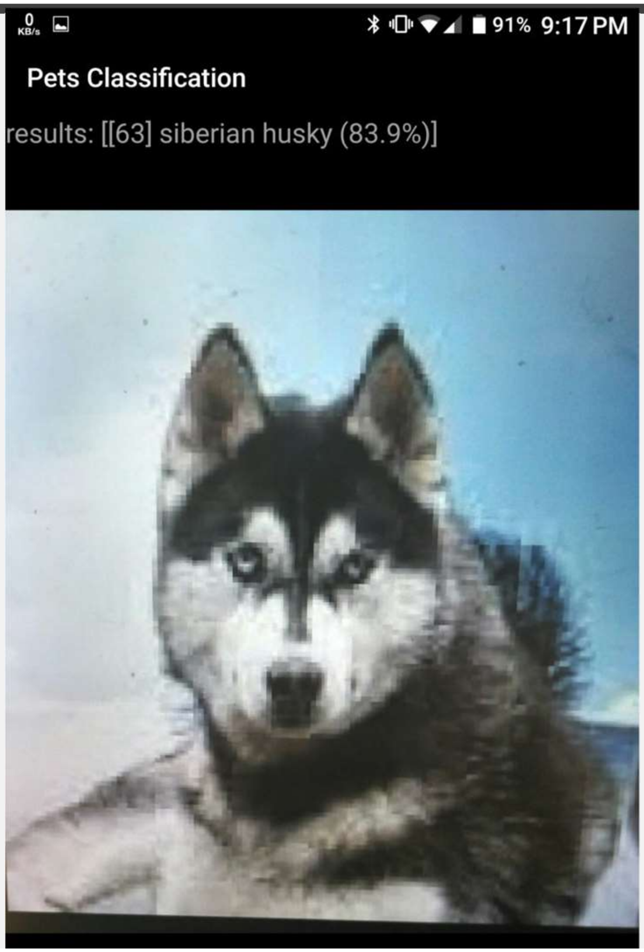
DEMO - CLASSIFICATION

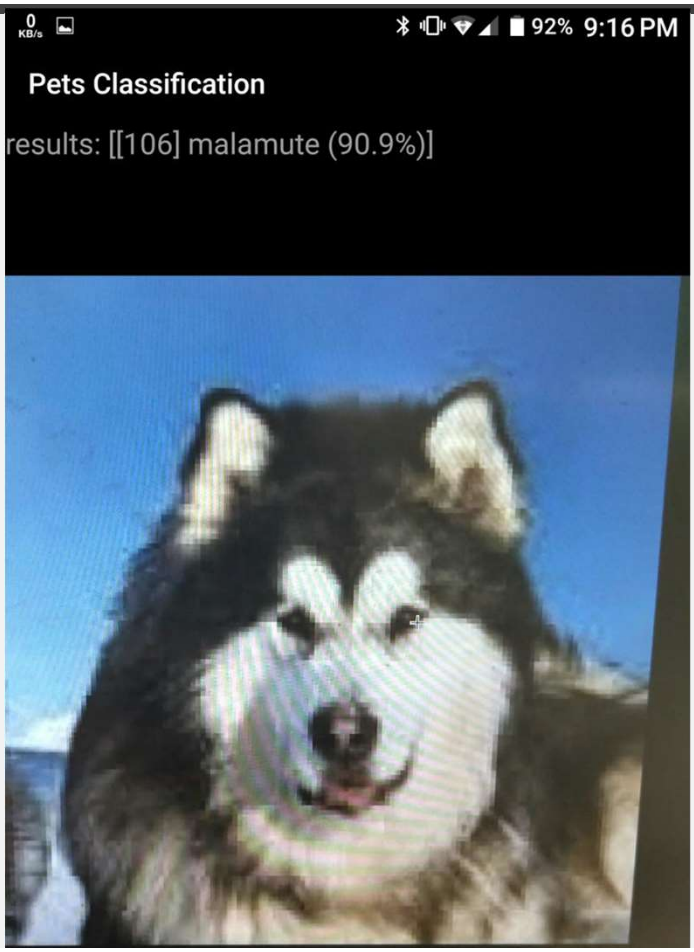


DEMO - CLASSIFICATION

3. Similar Species (Husky vs Malamute, Low Quality)

DEMO - CLASSIFICATION

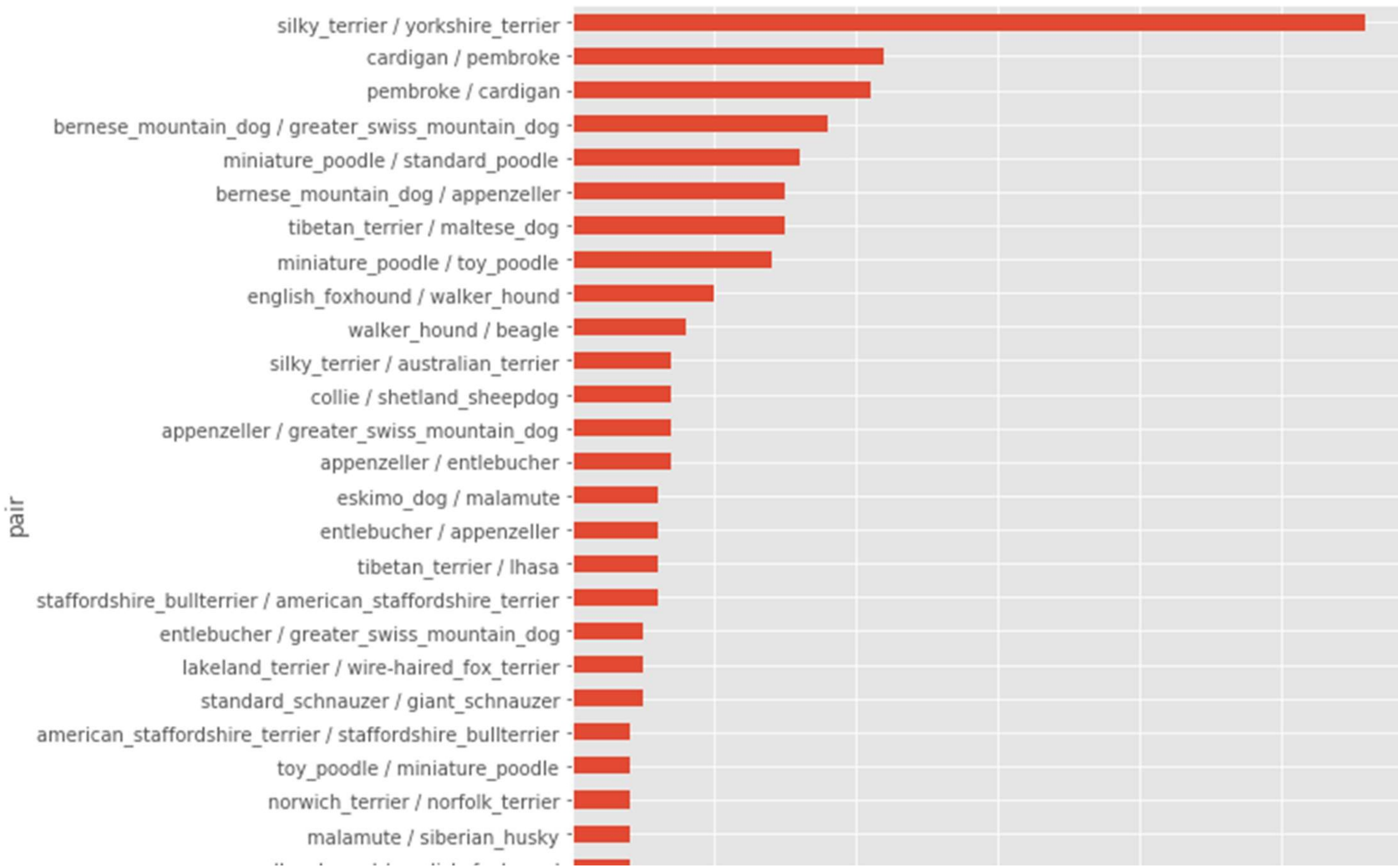




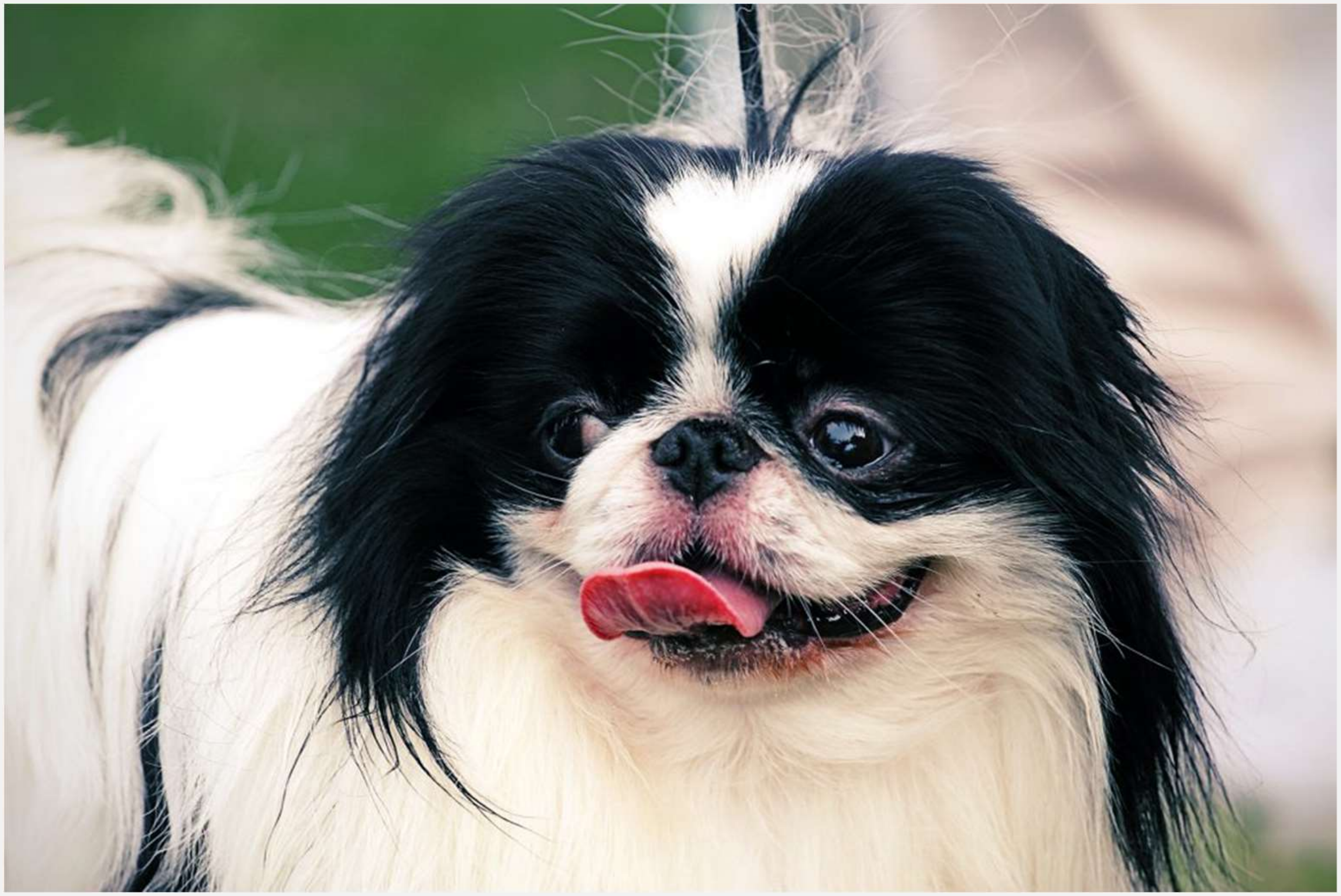
DEMO - CLASSIFICATION

4. Other Similar Species (Cairn vs Affenpinscher vs Schnauzer)



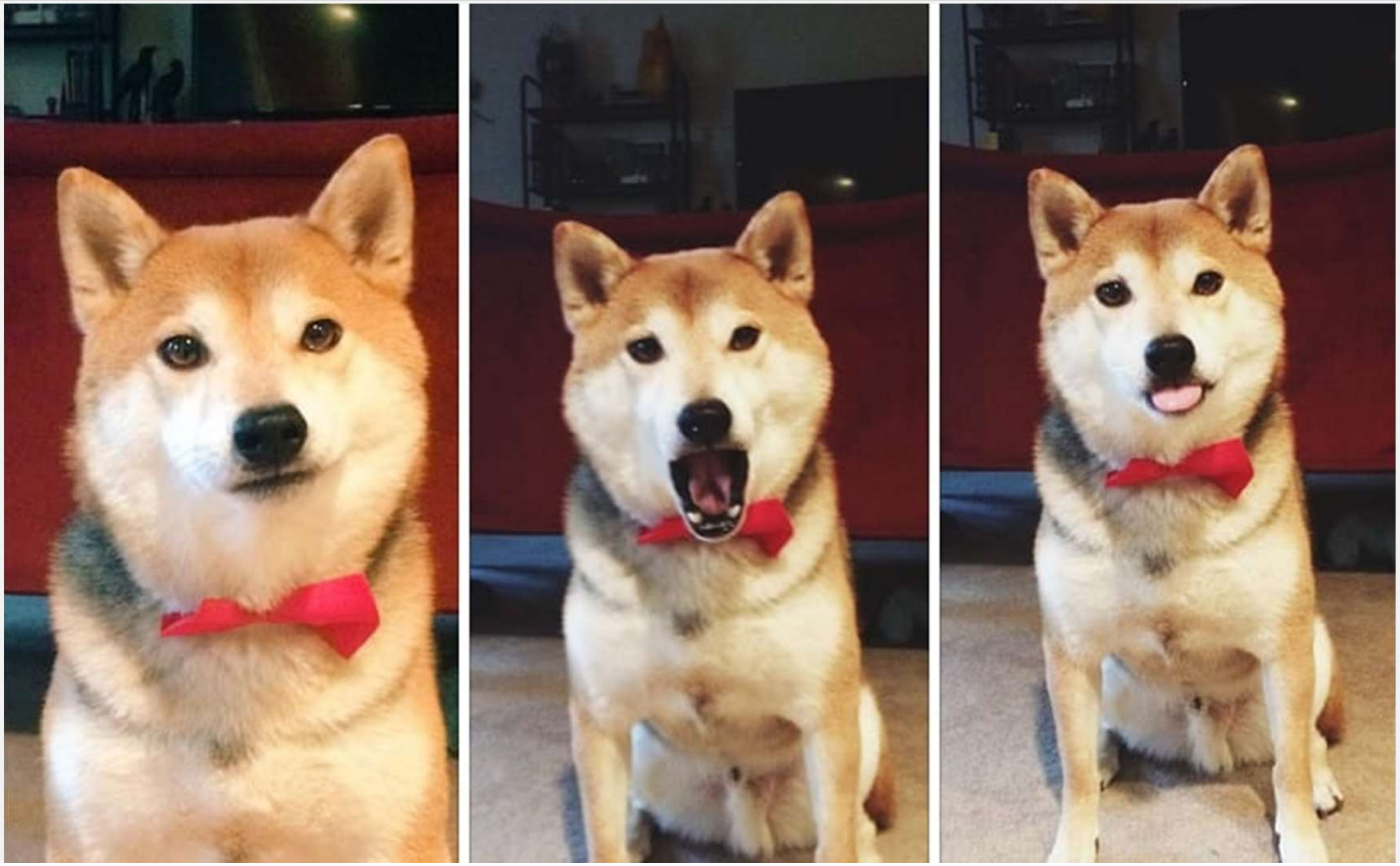

pair
silky_terrier / yorkshire_terrier
cardigan / pembroke
pembroke / cardigan
bernese_mountain_dog / greater_swiss_mountain_dog
miniature_poodle / standard_poodle
bernese_mountain_dog / appenzeller
tibetan_terrier / maltese_dog
miniature_poodle / toy_poodle
english_foxhound / walker_hound
walker_hound / beagle
silky_terrier / australian_terrier
collie / shetland_sheepdog
appenzeller / greater_swiss_mountain_dog
appenzeller / entlebucher
eskimo_dog / malamute
entlebucher / appenzeller
tibetan_terrier / lhasa
staffordshire_bullterrier / american_staffordshire_terrier
entlebucher / greater_swiss_mountain_dog
lakeland_terrier / wire-haired_fox_terrier
standard_schnauzer / giant_schnauzer
american_staffordshire_terrier / staffordshire_bullterrier
toy_poodle / miniature_poodle
norwich_terrier / norfolk_terrier
malamute / siberian_husky

DEMO – DOG FACE DETECTION



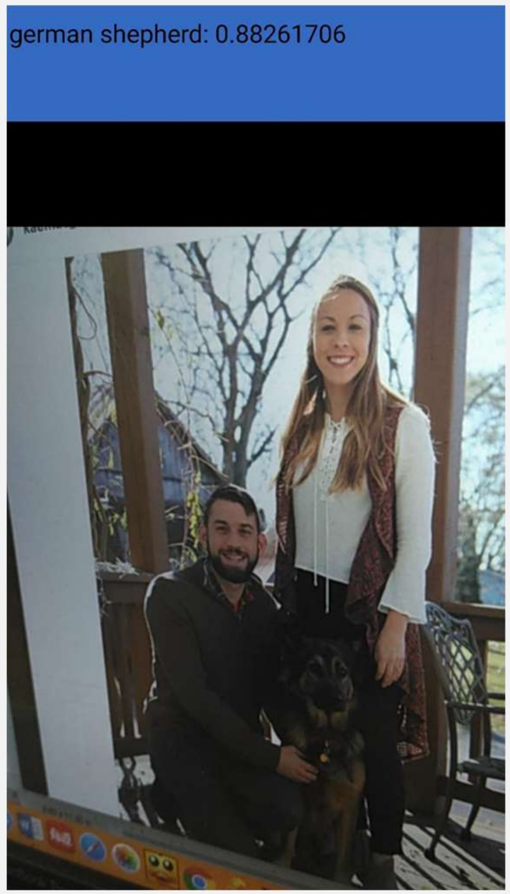
DEMO – DOG FACE DETECTION

DEMO – DOG FACE DETECTION

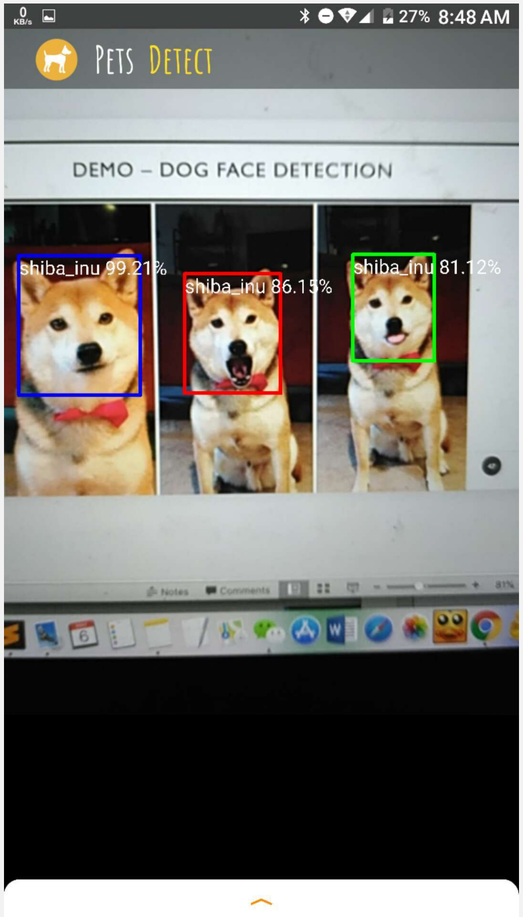


DEMO – DOG FACE DETECTION

SMALL UI IMPROVE



SMALL UI IMPROVE

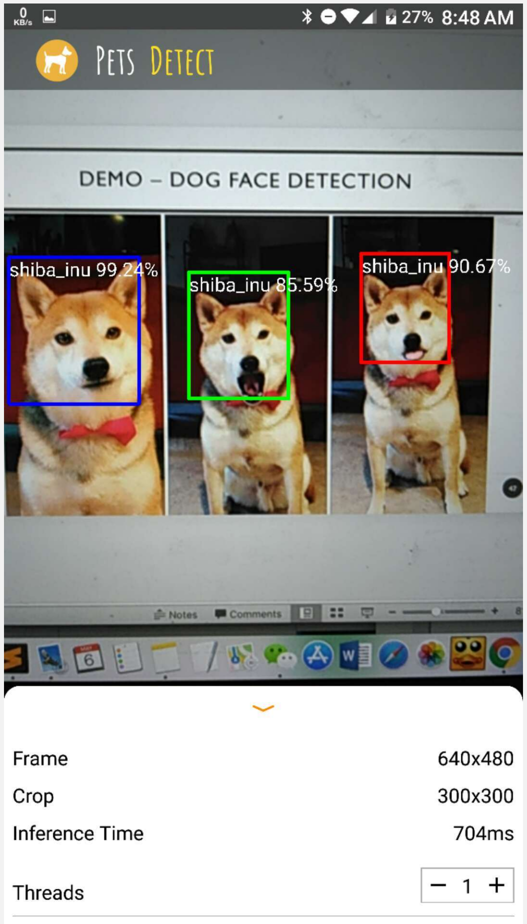


Top bar – transparent

Change icon, logo

Bottom – hidden menu

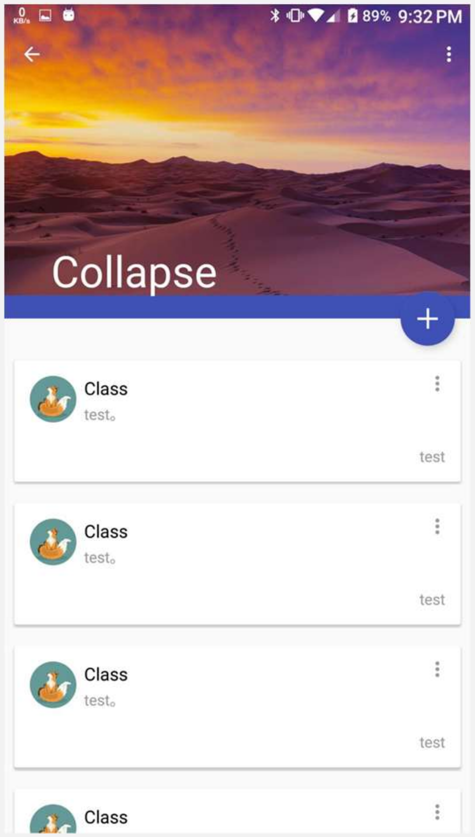
SMALL UI IMPROVE

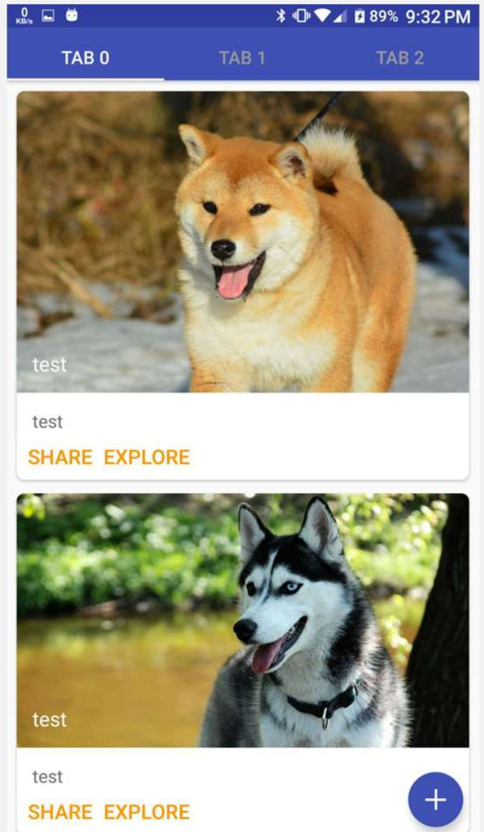


Swipe up -> hidden menu will show

Current frame size
Crop size
Inference Time
Thread Num

UI EXPLORE





The names & numbers are somehow packed and protected in this version.

Better not to change the library function or nested structure.

CONCLUSION

Improvements:

For classification:
More images for each class probably would be better.
UI could be improved.

For detection:
More models should be trained with gpu, especially faster one.

Stanford dataset would be better.

CONCLUSION

Thank you!