CSE 641 Deep Learning

Assignment-1

Kesanupalli Srivatsava(MT18054)
Subhani Shaik(MT18117)
Palla Akhil Kumar(MT18130)
Group-15

Dataset: Our data set contains 13553 camera trapped animal images in the jungle. Ground Truth is in a JSON format which contains Annotations like bounding box coordinates, category-id, image-id. Etc.
Most of the images are taken in dark.
Few images have multiple objects while some of the images don't have any object in the image.
Few images are such that only a part of the animal is visible in the image like tale etc
Few images have a shadow of the animal, while some images have an animal in the bushes
There are a total of 16 classes in the given data set, we have reindexed the class categories to

1-opossum
2-rodent
3-raccoon
4-squirrel
5-bobcat
6-skunk
7-dog
8-coyote
9-rabbit
10-bird
11-cat
12-badger
13-empty
14-car
15-deer
16-fox

Base-Line Model:

- ➔ We have used YOLOv3 as our baseline model
- ➔ We have referred two repositories andy-yun's and other is eriklindernoren's repo
- ➔ The repository used the DarkNet implementation

- ➔ DarkNet contains 53 convo layers
- ➔ Each repo has many changes to be done according to our data set, we have adopted the best practices from both the repositories
- ➔ We have to give the ground truth in the form of a text file, where the first field in the file is the category of the image and the remaining fields are bounding boxes of the object
- ➔ The bounding box coordinates are scaled with respect to the height and width
- ➔ DarkNet was trained on COCO data set
- ➔ We have used pre-trained weights on coco data set and done transfer learning on our data set
- ➔ Images are scaled to 416*416 dimensions
- ➔ We have trained only the later layers of the network and ignored the beginning layers because the initial layers contain the features like edges etc
- ➔ We have trained for 30 epochs

Yolo Architecture:

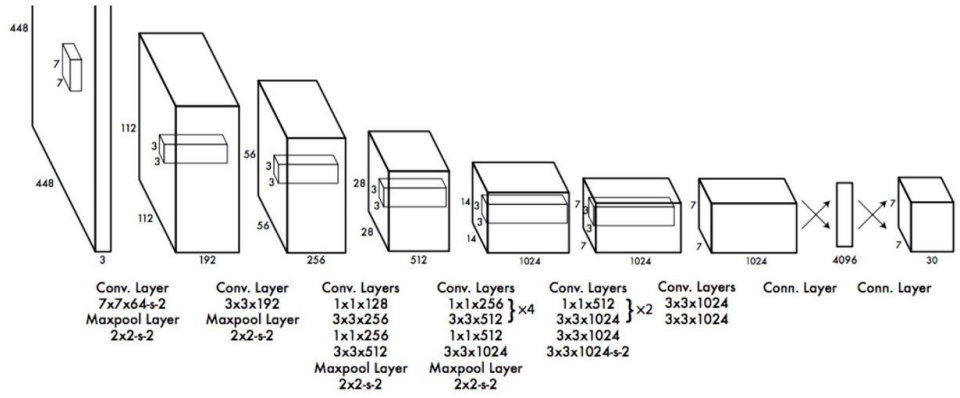
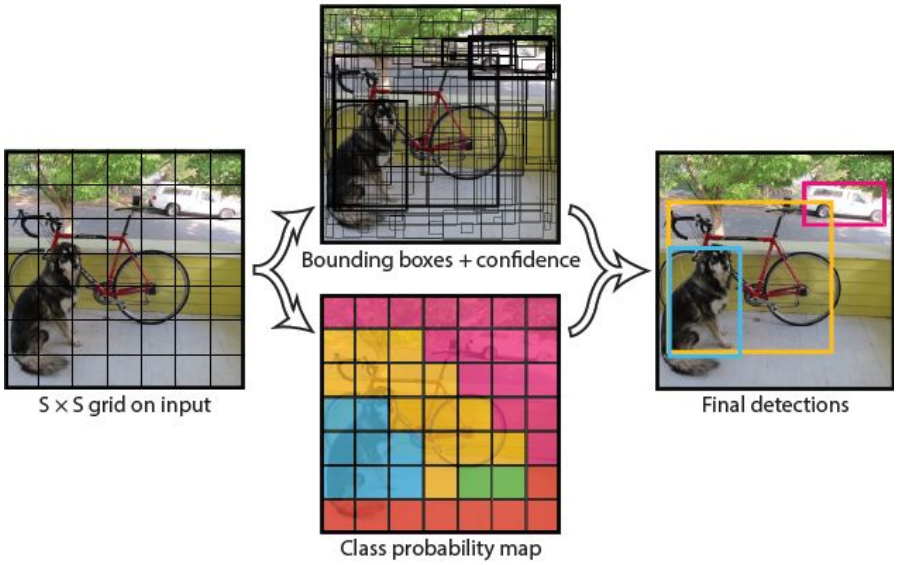
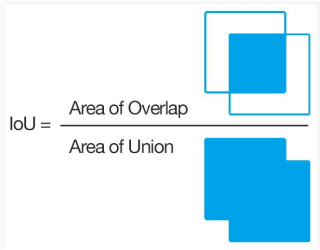


Figure 3: The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection.



Evaluation Metrics:

- IoU: Intersection over union: it is the ratio of “area of intersection of ground truth bounding box with the predicted bounding box” to “area of union of ground truth bounding box with predicted bounding box”



- We use this metric in the non max suppression to eliminate the bounding boxes which don't have an IoU metric less than a given threshold
- We have used the threshold of 0.4

Training:

- ★ We have trained our model using Tesla V100 NVIDIA GPU on google cloud platform

```
[srivatsava@srivatsava:~/new$ python train.py --data cfg/animals.data --config cfg/yolov3_.cfg --weights yolov3.weights -r
Training for (1,10000)
2019-02-08 16:36:47 [001] processed 4135 samples, lr 1.000000e-03
torch.Size([16, 256, 26, 26])
torch.Size([16, 512, 26, 26])
torch.Size([16, 128, 52, 52])
torch.Size([16, 256, 52, 52])
16: Layer(082) nGT  17, nRC   1, nRC75   0, nPP   0, loss: box 46.622, conf  3.864, class  5.626, total  56.112
16: Layer(094) nGT  17, nRC   7, nRC75   1, nPP 32448, loss: box  2.537, conf 1403.025, class 11.772, total 1417.334
16: Layer(106) nGT  17, nRC   3, nRC75   1, nPP 129792, loss: box  6.266, conf 8864.568, class 12.146, total 8882.980
torch.Size([16, 256, 26, 26])
torch.Size([16, 512, 26, 26])
torch.Size([16, 128, 52, 52])
torch.Size([16, 256, 52, 52])
```

• The model is trained using the above command
• The test images are placed in the “data/” folder, which will be predicted with the bounding box plotted around the image.
• To detect the image, the above command is issued to generate test images with bounding boxes around them.

Commands:
To train:
python train.py --data cfg/animals.data --config cfg/yolov3_.cfg --weights yolov3.weights -r

To detect:
python detect.py cfg/yolov3_.cfg yolov3.weights data/dog.jpg data/animals.names

To validate: python valid.py cfg/animals.data cfg/yolov3_.cfg yolov3.weights

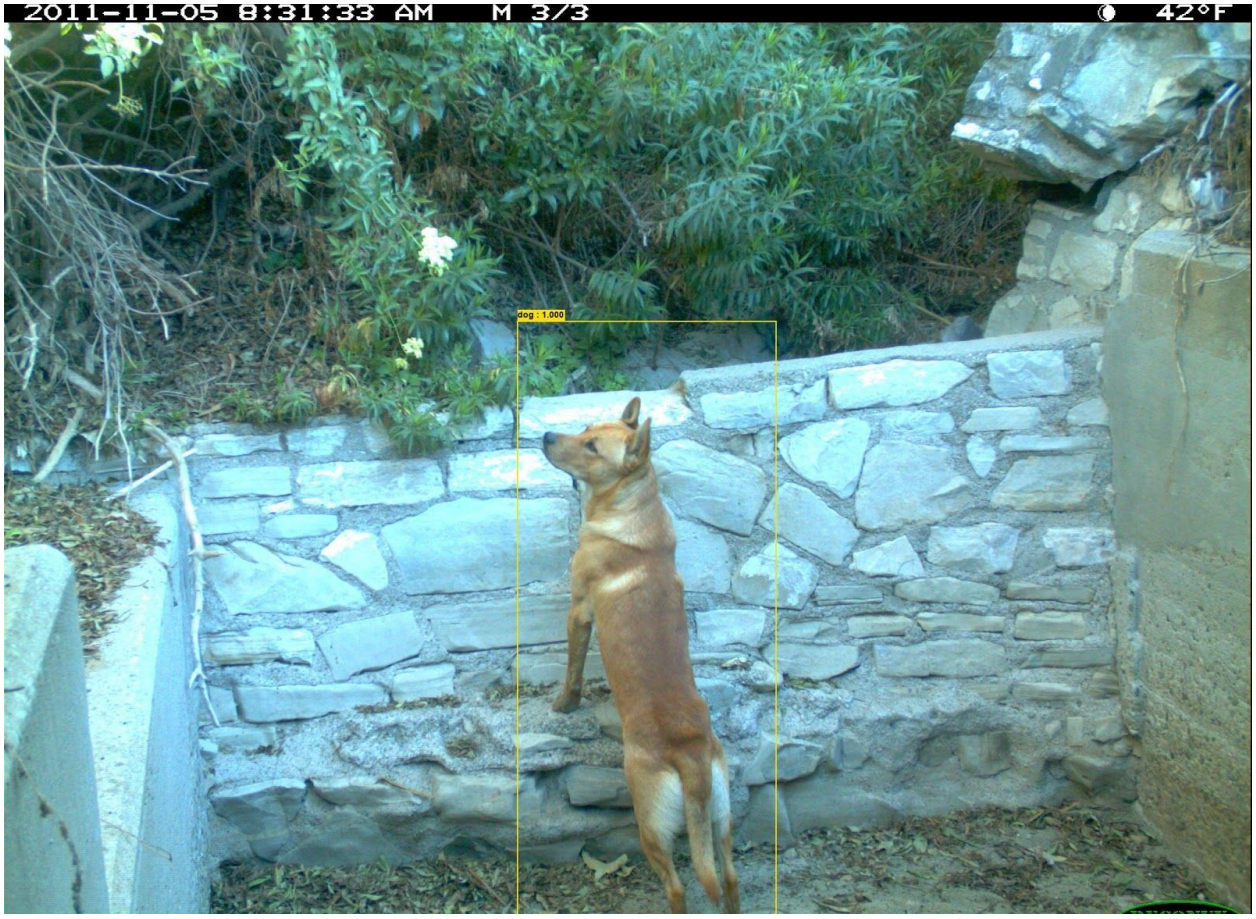
Sample outputs:


2013-08-12 4:30:32 PM M 2/3
95°F
bird : 0.507


2011-12-14 6:29:08 PM M 3/3 53°F
coyote : 0.424


2011-11-14 12:35:33 PM M 3/3
61°F
bobcat : 0.987


2011-07-16 3:54:18 AM M 1/3
57°F
bobcat : 1.000


2011-11-05 8:31:33 AM M 3/3 42°F
dog : 1.000


car : 1.000
car : 1.000
1/08/12 10:14 PM
Cuddeback

References:

- IoU:https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/
- Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, You Only Look Once: Unified, Real-Time Object Detection, ArXiv, May 2016, paper
- DarkNet