Custom Object Detection using YOLO

Name : Raj Jaiswal
Github : https://github.com/Jaiswalraj2908/Jaiswalraj2908

Important 🚨

Weights files of YOLOv4 Model trained on Custom Dataset :

https://drive.google.com/drive/folders/17KI5A7rVd_QwC0neW8vdqMtKgbyPMEUy?usp=sharing

Output of Model :

1. Images :
   https://github.com/Jaiswalraj2908/Jaiswalraj2908/tree/main/results_of_models/images
2. Videos :
   https://github.com/Jaiswalraj2908/Jaiswalraj2908/tree/main/results_of_models/videos

Contents :

1. Data Collection.

We will look at 5 such ways of collecting data for training your custom model that solves your problem.

- Publicly available open labelled datasets.
  eg : ImageNet , COCO , Google's Open Image etc.
- **Scraping the Web (which i used for this Project)**
- Taking Photographs.
- Data Augmentation.

- Data Generation (Synthetic Data by GANs)

2. Data Annotation.

Here is a list of tools that you can use for annotating images:

1. MakeSense.AI **(which i used for this Project)**
2. LabelImg
3. VGG image annotator
4. LabelMe
5. Scalable
6. RectLabel

3. Download required python files and Create dataset hierarchy.

- Download Yolo cfg file and edit it according it to your own needs.
  Edit :

  Training

  batch=1

  subdivisions=16

  max_batches = 6000 (class = 1,2,3 if classes > 3 then each class x 2000)

  policy=steps

  steps=4800,5400 (steps is 80% and 90% max_batches)

  filters = 18 (no. of class + 5) x 3

  classes=1

- Download create_train.py and create_test.py

- Create this hierarchy in your local system.

```
.
└── ./dataset /
    ├── ./dataset /train/
    │   └── ./dataset /train/obj
    └── ./dataset /validation/
        └── ./dataset /validation/test
```

- Create yolov4 folder in Google drive.

```
.
└── ./google drive/
    ├── ./google drive/yolov4/
    │   ├── ./google drive/yolov4/obj.zip
    │   ├── ./google drive/yolov4/test.zip
    │   ├── ./google drive/yolov4/create_train.py
    │   ├── ./google drive/yolov4/create_test.py
    │   ├── ./google drive/yolov4/yolo_cfg
    │   └── ./google drive/yolov4/backup/
    │       └── ./google drive/yolov4/backup/yolo_weights
    └── ./google drive/darknet/
        ├── ./google drive/darknet/cfg/
        │   └── ./google drive/darknet/cfg/yolo_cfg
        └── ./google drive/darknet/data/
            ├── ./google drive/darknet/data/obj.data
            └── ./google drive/darknet/data/obj.names
```

4. Training YOLO Weights.

Note : Following code are use to train YOLO on your custom dataset .
The most important thing here is your directory hierarchy which i have shown above. Your aim should be to provide correct path in to the code and Conncet you Collab with GPU (Provided by Google for free).

```
# 1. Mount your google drive to collab
from google.colab import drive
drive.mount('/content/drive/')

# 2. Download darknet folder
!git clone https://github.com/AlexeyAB/darknet
```

```
# 3. This is customizable according to your need 0 : Default | 1 : Active
%cd darknet
!sed -i 's/OPENCV=0/OPENCV=1/' Makefile
!sed -i 's/GPU=0/GPU=1/' Makefile
!sed -i 's/CUDNN=0/CUDNN=1/' Makefile
!sed -i 's/CUDNN_HALF=0/CUDNN_HALF=1/' Makefile

!make

# 4. To read your train and validation data with labels (YOLO format)
!cp /content/drive/MyDrive/yolov4/obj.zip ../
!cp /content/drive/MyDrive/yolov4/test.zip ../

!unzip ../obj.zip -d data/
!unzip ../test.zip -d data/

# 5. Create prebuild custom configuration file (Make changes according to your datas
!cp cfg/yolov4-custom.cfg /content/drive/MyDrive/yolov4/yolov4-monkey-custom.cfg
!cp /content/drive/MyDrive/yolov4/yolov4-monkey-custom.cfg ./cfg

# 6. Create this data and names files according to your class
!cp /content/drive/MyDrive/yolov4/obj.data ./data
!cp /content/drive/MyDrive/yolov4/obj.names ./data

# 7. This Python files will arrange your test and train data to feed network
!cp /content/drive/MyDrive/yolov4/create_train.py ./
!cp /content/drive/MyDrive/yolov4/create_test.py ./

!python create_train.py
!python create_test.py

!wget https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v3_optimal/

# 8. Training of your model to get weights on your custom data
!./darknet detector train /content/darknet/data/obj.data /content/darknet/cfg/yolov4

# 9. Testing your model weights on inference data
!./darknet detector test data/obj.data cfg/yolov4-custom-monkey.cfg /content/drive/M
```

5. References

- Data Annotation : https://towardsdatascience.com/image-data-labelling-and-annotation-everything-you-need-to-know-86ede6c684b1

- YOLO Training :
  https://youtu.be/70mq_HbFkfo

- Heroku and Docker Deployment
  https://youtu.be/Gs15V79cauo

- Object Detection Theory
  Deep learning using Vision Systems by Mohamed Elgendy (chapter 7)

6. Future Improvements to develop model further

1. We can take this detection model further to track and use it to predict behaviour (predicting its actions) and deploy this moeel to Edge devices .
2. We can also make an andriod map which can detect different species of monkey .

Different Methods to improve :

1. ***Gather More Data***
2. ***Scraping the Web (which i used for this Project)***
3. ***Image Preprocessing and Augmentation***
4. ***Image Input Resolution Size***
5. ***When to Use Pretrained Weights***
6. ***Choosing a Model Size and Architecture***
7. ***Picking Up From a Previous Training Run***
8. ***Choosing the Best Model after Training***
9. ***Track Your Model Evaluations***

Reference : https://www.youtube.com/watch?v=GUw6at85un8