



ARTIFICIAL INTELLIGENCE SERVICES





YOLO V3



VS



DATASET

**“KAGGLE
+
GOOGLE
+
BING”**



Image Detection: Fruits

Custom Training Steps

- Data collection of all possible scenarios from various resources like Kaggle + Google + Bing.
- Drop down the image dataset of 2 classes into separate specific folder containing the '/class_names'
- We are using **YOLO V3** architecture and specifically using the Github link of **AlexyAB darknet**.
- Then, Git Clone the repository using the command { **git clone** <https://github.com/AlexeyAB/darknet.git> }
 - Now, change the directory using command { **cd darknet/** }
 - Now, do the necessary changes in order to activate the GPU by making:
 - > **GPU = 1**
 - > **CUDA = 1**
 - > **CuDNN = 1**
- After making necessary changes in the make file, run the **make file**.
- Draw the bounding boxes around the images you want to detect.
- The annotation tool can be cloned through the link: { <https://github.com/tzutalin/labelImg> }
- After drawing the bounding boxes, you get the files in .xml format.
- Now, convert the .xml format to yolo format, i.e., .txt format and save in the dataset folder.



Image Detection: Fruits

Custom Training Steps

- Now, create 4 folders:
 - **train.txt** : GIVE THE LIST OF TRAINING IMAGES
 - **test.txt** : GIVE THE LIST OF TESTING IMAGES
 - **obj.names** : GIVE THE NAMES OF THE CLASSES WHICH WE ARE TRAINING
 - **obj.data** : WHICH CONTAINS THE FOLLOWING DETAILS
 - > **classes**= {{NUM_OF_CLASSES}}
 - > **train** = {{PATH TO TRAIN.TXT}}
 - > **test** = {{PATH TO TEST.TXT}}
 - > **names** = {{PATH TO OBJ.NAMES}}
 - > **backup** = backup/
- Now, change the parameters in the configuration file, i.e., **yoloV3.cfg** :
 - Line **3**, set batch=24, this means we will be using 24 images for every training step
 - Line **4**, set subdivisions=8, the batch will be divided by 8 to decrease GPU VRAM requirements
 - Line **603**, set filters=(classes + 5)*3
 - Line **610**, set classes= x, the number of categories we want to detect
 - Line **689**, set filters=(classes + 5)*3
 - Line **696**, set classes=2, the number of categories we want to detect



Image Detection: Fruits

Custom Training Steps

- Line **776**, `filters=(classes + 5)*3`
- Line **783**, set `classes=x`, the number of categories we want to detect
- Finally, run the training using the command: `{ ./darknet detector train obj.data yoloV3.cfg darknet53.conv.74 }`
- Stop the training with accordance with the minimal average loss.



Image Detection: Fruits

Custom Testing Steps

- Now, you can check the prediction using the command: `{ ./darknet detector test obj.data cfg/yolov3.cfg yolov3.weights }`
- Then, **Enter Image Path:** will appear where you must specify your test image path.

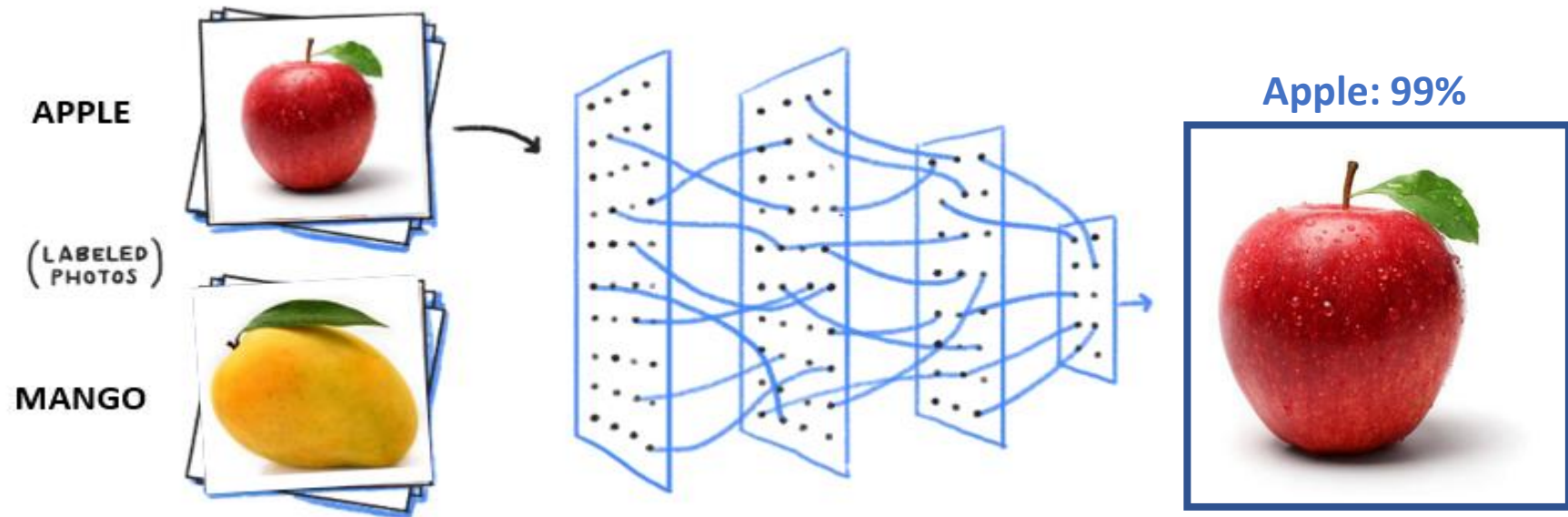




Image Detection: Fruits

RESULTS

- ✓ Random image of Apple and Mango is taken from Google or Bing and predicted score is tested.



Sample_1.jpg



Sample_2.jpg

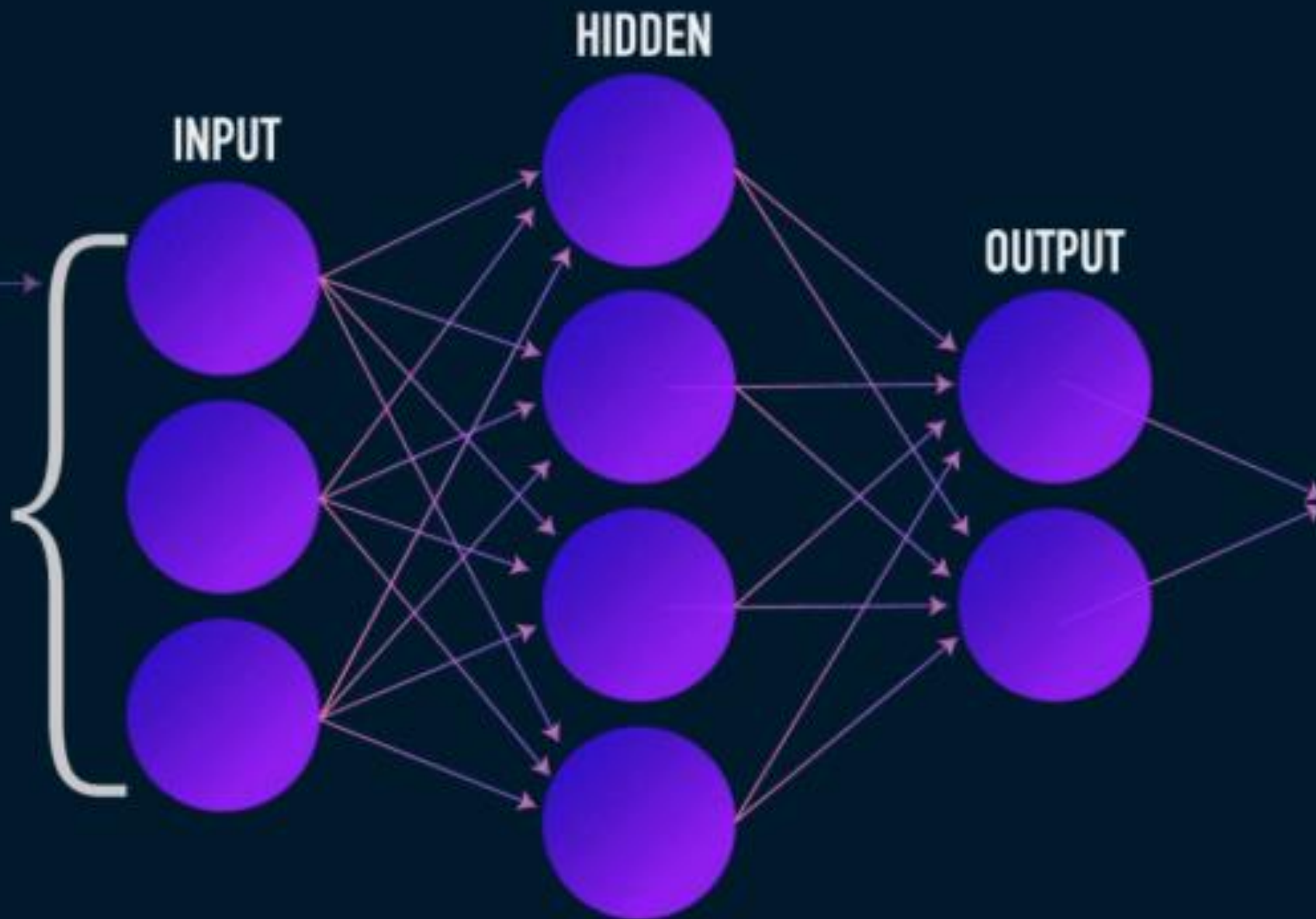


Sample_3.jpg



Sample_4.jpg

Test samples taken from Google & Bing



Apple: 99%

A photograph of a single red apple with a small green leaf attached to its stem. The apple is centered within a blue rectangular border.