Circle Detection using YOLOv5

This project involves developing an object detection system using YOLOv5 to identify and locate circles in images. The task includes creating a synthetic dataset of images containing circles of varying sizes, colors, and backgrounds, and training a YOLOv5 model on this dataset. The final output is a trained model capable of detecting circles under various conditions, along with performance metrics such as precision, recall, and mean Average Precision (mAP). All steps, including dataset creation, model training, and evaluation, are documented in a Google Colab notebook.

Setup Instructions

#1. Open the Google Colab Notebook

-Download the circle_detection.ipynb file.

-Open Google Colab in your browser.

-Upload the circle_detection.ipynb file to Google Colab.

#2. Clone the YOLOv5 Repository and Install Dependencies

Run the first cell in the notebook to clone the YOLOv5 repository and install the necessary Python packages:

```
!git clone https://github.com/ultralytics/yolov5
%cd yolov5
!pip install -r requirements.txt
```

This will set up the environment by downloading YOLOv5 and installing dependencies like PyTorch and OpenCV.

#3. Generate the Synthetic Dataset

Run the dataset generation cell to create a synthetic dataset of images containing circles of varying sizes, colors, and backgrounds:

```
# Generate the dataset
generate_dataset(image_count=1000, image_size=(256, 256))
```

This will create a dataset of 1000 images with corresponding labels.

#4. Prepare the Dataset for YOLOv5

Next, prepare the dataset for training by organizing it into training and validation sets:

```
#Prepare the YOLO dataset
prepare_yolo_dataset()
```

#5.Create the data.yaml Configuration File

Create the data.yaml file that YOLOv5 uses to locate your dataset and define the class names:

```
# Create the data.yaml file
data_yaml_content = """
train: ../yolo_circle_dataset/images/train
val: ../yolo_circle_dataset/images/val

nc: 1
names: ['circle']
"""
with open('/content/yolov5/data.yaml', 'w') as f:
f.write(data_yaml_content)
```

Train the YOLOv5 Model

Run the following cell to start training the YOLOv5 model on your generated dataset:

```
# Train the YOLO model
!python train.py --img 256 --batch 16 --epochs 30 --data data.yaml --cfg models/yolov5s.yaml
```

Evaluate the Model

```
# Evaluate the model
!python val.py --weights runs/train/circle_detection/weights/best.pt --data data.yaml --img
```

This will print the model's performance metrics, including precision, recall, and mAP.

Running the Notebook

1.Follow the setup instructions above to set up the environment and generate the dataset.

2.Run each cell sequentially, making sure to wait for each step to complete before moving on to the next.

3.After training, evaluate the model and review the performance metrics.

4.Save your trained model and notebook as needed.