Kecilin Technical Test

Faruq Fathin Azzaim

1. Model selection: YOLOv8
   I am using YOLOv8 for high FPS detection and good accuracy model. YOLOV8 from Ultralytics is easy to train, and inference. The data i trained on is from BeyBlade video from YouTube. The labelling process is divide into two steps:
   - First: Label manual all extracted frames using LabelImg, and then train YOLOv8 model using labelled images.
   - Second: Use the trained model on new images to generate labels, review and correct the new labels using LabelImg. Repeat these steps until we achieve our desired quantity of data.

   yolov8m.pt is used for transfer learning.

2. Model accuracy:

```
Validating runs/detect/train23/weights/best.pt...
Ultralytics YOLOv8.2.4 🚀 Python-3.9.19 torch-1.12.0+cu102 CUDA:0 (Quadro RTX 5000, 16125MiB)
Model summary (fused): 168 layers, 3006233 parameters, 0 gradients, 8.1 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%
                   all         33         98      0.859      0.898      0.919      0.796
              beyblade         33         59      0.934      0.953      0.982      0.896
                  stop         33         28      0.763      0.922      0.915      0.792
                  hand         33         11       0.88      0.818      0.859      0.701
Speed: 4.2ms preprocess, 1.8ms inference, 0.0ms loss, 0.8ms postprocess per image
```

3. Logic behind program:

```
if key == ord("s"):
    initBB = cv2.selectROI("Frame", frame, fromCenter=False,
        showCrosshair=True)
    init_time=True
```

   To draw area, press s and click-drag mouse

```
def check_stop_beyblade(count_stop_beyblade, bbboxes):
    beyblade1 = bbboxes[0]
    beyblade2 = bbboxes[1]
    class_beyblade1 = beyblade1.cls
    class_beyblade2 = beyblade2.cls
    if count_stop_beyblade > 10:
        if class_beyblade1 == 0:
            bb = beyblade1.xyxy[0]
            res = ['winner', (int(bb[0]),int(bb[1]), int(bb[2]),int(bb[3]))]
        else:
            bb = beyblade2.xyxy[0]
            res = ['winner', (int(bb[0]),int(bb[1]), int(bb[2]),int(bb[3]))]
    else:
        if class_beyblade1==1 or class_beyblade2==1:
            num = count_stop_beyblade+1
        else:
            num=0
        res= ['continue', num]
    return res
```

   To know any beyblade stopped, we detect any stop class from YOLOv8, and count how many stop class is appear in a frame, if stop class appear more than 10 frame then the match is over

```python
def check_outside_beyblade(area, bbboxes):
    beyblade1 = bbboxes[0]
    beyblade2 = bbboxes[1]
    if is_inside(beyblade1.xyxy[0], area) and is_inside(beyblade2.xyxy[0], area):
        return ['continue', 0]
    else:
        if is_inside(beyblade1.xyxy[0], area)==is_inside(beyblade2.xyxy[0], area):
            return ['draw', 0]
        if is_inside(beyblade1.xyxy[0], area):
            bb = beyblade1.xyxy[0]
            return ['winner', (int(bb[0]),int(bb[1]), int(bb[2]),int(bb[3]))]
        else:
            bb = beyblade2.xyxy[0]
            return ['winner', (int(bb[0]),int(bb[1]), int(bb[2]),int(bb[3]))]
```

To know if any beyblade is outside from arena, we check if center of bounding boxes from detected object is outside from Area