readme.pdf

Yan Trindade's Solution:

The Solution:

1. Pick a video from gallery (videos must be on real device (I sent trough airdrop from Mac to iPhone)

2. Video is processed frame to frame

3. On each frame lets rotate the image before processing to avoid boundingBox coordinate issues.

4. Using CoreMLProcessor we use CoreML and Vision to load the YOLOv3FP16 model.

5. Each observed person object convert the boundingBox to iOS Coordinates

6. Evaluate the distance to center of view, if its less than arbitraty value choosen its green otherwise red

7. Update the Rectangle on screen

Issues during the coding challenge:

1. Tried to play a video using a VideoPlayer loading the VideoAsset while processing the video frame to frame and updating the retangle overlaying the videoPlayer. Stucked into an issue trying to sync that.

2. Tried to process all frames, store them, create a new video using AVAssetWriter then playing into videoPlayer. Stucked into overflow memory / buffer handling issues.

3. Final solution that worked process and update ui frame to frame. Instead of playing the video and processing, update everyFrame as Image on screen, that got a smoothly UX and worked prefectly!

Choose the best model for the job and explain why

Performing tests on both models on the Iphone 14 was promising, all of the compute units were using the Neural Engine. Looking at model size, the full model had 256 compute units against 86 for the Tiny version.

Looking at older device models, I had a spare iPhone 11 to test. Since the iPhone 11 Neural Engine is not as powerful as the Iphone 14, I was seeing ~60% of the compute units using the GPU instead of the NE.

There is a trade off between accuracy, speed and app total size due to full model is a large asset.

On the video 2, with lower resolution Tiny model takes more time to identify the person walking from the border to center but they are basically the same.

On video 1 and 3 I see no difference between the models.

On all videos there is a short difference about the person size what made me assume tiny algorithm focus more on head and upper part of body while full model recognizes almost full body perfectly.

Considering all information and results observed on 3 asset videos, I would choose the Tiny version because it works fine even for lower resolutions. Maybe if we need more accreted information the full model may be consider in future.