

人物姿勢推定アルゴリズムをあれこれ試す

ラズパイで エッジAI



笹尾 幸良

第3回

複数人の関節点の検出が可能な YOLOv8,
3次元の姿勢推定が可能な MediaPipe



図1 YOLOv8による複数人の姿勢推定結果

今回はYOLOとMediaPipeを使った人物の姿勢推定を行います。YOLOは前回のPoseNet/MoveNetを使った場合と同じ2次元の姿勢推定ですが、複数人の姿勢推定が可能です(図1)。また、MediaPipeは3次元の姿勢推定も可能です(図2)。

1 YOLOで2次元の人物姿勢推定

● YOLOv8は姿勢推定にも対応

YOLOには複数のバージョンがあり、開発者やライセンス/深層学習フレームワークもバージョン間で異なります。Ultralytics⁽¹⁾はYOLOv5とYOLOv8をリリースしてきました。YOLOといえばオブジェクト検出で有名ですが、YOLOv8は他のタスクにも対応しており、姿勢推定もYOLOv8n-pose、YOLOv8s-poseのようにモデルの大きさに応じて複数のモデルが用意されています。

今回は最も小さいモデルであるYOLOv8n-poseをラズベリー・パイで動作させます。YOLOv8姿勢推定の関節点の数は、前回のPoseNet/MoveNetと同じ17点ですが、前回と違い複数人の関節点の検出が可能です。

● ステップ①…モデルを量子化する

YOLOv8の公式⁽²⁾が配布するモデルは、PyTorch形式で量子化されていません。姿勢推定は処理量が多

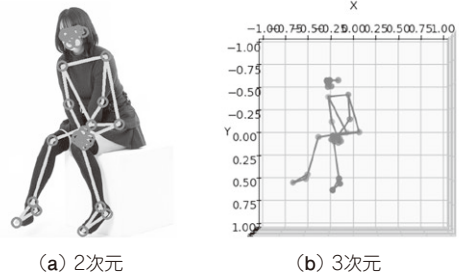


図2 MediaPipeによる姿勢推定の結果(関節33点検出)

いので、軽量化するために8ビット整数のtflite形式ファイルに変換します。

なお、モデル変換はラズベリー・パイ上で行う必要はありません。PC上で作業を行って、変換したファイルをラズベリー・パイに配置します。変換済みのモデル・ファイルをpose/models下に置いていますので、この作業はスキップしていただいても構いません。

▶公式の手順には問題がある

公式の手順としては、YOLOv8のパッケージであるultralyticsをインストールした後、次のコマンドで指定したモデルを8ビット量子化し、tflite形式で出力します。

```
$ pip3 install ultralytics
$ yolo export model='yolov8n-pose.pt'
format='tflite' int8=True
```

ところが、この手順で変換されたモデルには問題があります。モデルの出力データとして、物体の位置情報や信頼度など複数種類のデータがありますが、これら全て結合して1つのテンソル・データとして出力します。つまり、データの値域が各種別のデータで大きく異なるため、8ビット量子化すると1つのテンソル・データの1つの量子化パラメータではデータの多くを表現できず、結果として正常な出力を得ることができなくなります。

▶正しい手順

そこで、元のリポジトリをforkして開発された、ultralytics_yolov8⁽³⁾を用います。結合されていた出