

画像認識とディープラーニングを用いた自立走行 AI ロボットの考察

名古屋工学院専門学校,産業技術研究科1年,松永南斗

1. 目的 カメラ画像によるリアルタイムのコーストレース、および障害物回避を可能とする自律走行車の制作に関わる基礎的な技術を考察する。

2. JetRacer の概要 JetRacer は、ディープラーニングと AI を用いたリアルタイムでの画像認識を指向した NVIDIA 社 Jetson Nano を組み込んだ WaveShare 社が販売している AI ロボットカーである。(図-1)



図-1 JetRacer の外観と主要機構

JetRacer に搭載されている Jetson Nano は、GNU/Linux ディストリビューションである Ubuntu をインストールし、その中に Jupyter notebook サーバとオブジェクト指向言語である Python、およびその実行環境を備えています。したがって、開発環境と実行環境を本体そのものが備えており、Jetson Nano に搭載されている GPU (Graphics Processing Unit : 画像処理装置)、機械学習および AI の推論エンジンを利用したリアルタイムの画像分類と物体検出を可能にしています。

3. ディープラーニングによる画像認識

JetRacer をディープラーニングを用いた AI 画像認識によってコース走行させるためには、認識したいもの (カメラから見たコースの形状) の大量の画像から特徴を見つける「学習フェーズ」(図-2 (a)) と、学習された特徴データ (学習モデル) を使ってカメラが捉えた画像から映っているもの

を推測する「推論フェーズ」に分かれます。(図-3 (b))

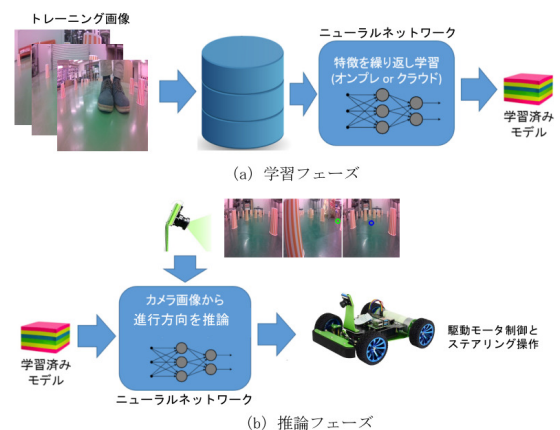


図-2 JetRacer の「学習・推論フェーズ」モデル

4. 自動走行の仕組み 自動走行には人間の目になり変わりコースの形状・車の位置などを認識し、それに応じた制御をする必要があります。

4.1 カメラ画像と教師データの作成 自動走行を実現するために、JetRacer では Web カメラのようなカメラ (単眼カメラと呼ばれる) からロボットカーが取るべきステアリング角を直接推定しています。Nvidia の提供する推論エンジンは、ディープラーニングを使って、「画像を入力→ステアリング角を出力」のアルゴリズムを用い、「コースを走行中の画像」と「その画像を撮影した時点での車のステアリング角」のペアのデータを大量に収集します (図-3)。

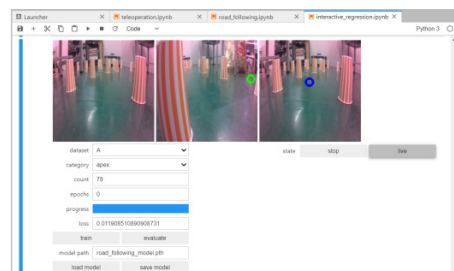


図-3 JetRacer の「学習フェーズ」画面

JetRacer は、教師データとなる各画像に付与した車が向かうべきターゲットピクセル座標 (x,y)

を使ってステアリング角を擬似的に作り出しています。学習フェーズによって得られた画像は「175_75_xx...xx.jpg」のようなファイル名で保存されます(図-4)。このファイル名先頭の2つの数字が画像を保存する際にクリックしたピクセル座標 x,y に対応しています。

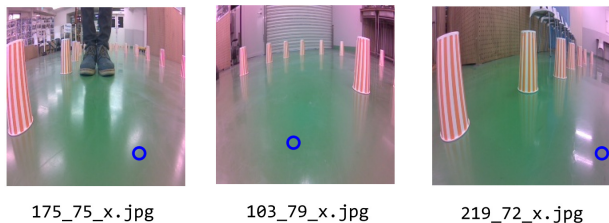


図.4 カメラ画像とターゲットピクセル座標の例

4.2 ネットワークのトレーニング ニューラルネットワークをトレーニングする際は、得られた画像とファイル名を読み込んで学習を行います。なお実際にネットワークから出力する x,y の値はピクセル座標 ($ann[x],ann[y]$) ではなく、式.1 および式.2 によって画像サイズ(幅:width、高さ:height)を元の値から割って得られる -1 から 1 の値です。この式から、ターゲットのピクセル座標 $ann[x]$ 、 $ann[y]$ がそれぞれ $-1 \sim 1$ に変換された x,y となります。

$$x = 2.0 * (ann[x] / width - 0.5) \quad \text{式.1}$$

$$y = 2.0 * (ann[y] / height - 0.5) \quad \text{式.2}$$

4.3 ステアリング角の決定 JetRacer ではニューラルネットワークが予測したピクセル座標 ($-1 \sim 1$ に正規化)の x の値をそのままステアリング角とみなします。(図.5)

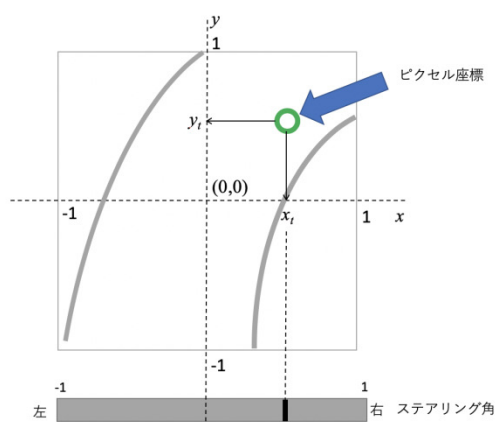


図.5 ピクセル座標とステアリング角の関係
 x の値をそのままステアリング角とみなすことで、カメラが車体進行方向に対してまっすぐを取

り付けられているとするならば、画像の真ん中を境にマークの位置に応じて左右のステアリング値に直接変換することで、走行制御のアルゴリズムがシンプルになります。

ピクセル座標 y の値は標準の JetRacer では使用していませんが、奥行き方向に関連する情報を持っている可能性があり、視界の先をどれだけ見渡せるかで速度制御への応用も考えられます。

5. トレーニングと自立走行の検証 紙コップをコースパイロンに見立て、パイロン間隔約 $20 \sim 30\text{cm}$ 、コース幅は約 $10\text{cm} \sim 50\text{cm}$ 、全長約 30m の滑らかな曲線コースを設け、教師データの作成、ニューラルネットワークのトレーニング、および自立走行の検証を行った。最終的に滑らかなコース走行と障害物の回避を行うために、画像数は 581 枚、トレーニングのエポック数(繰り返し学習の回数)は 10 回を要した。



図.6 コップパイロンコース自立走行の様子

6. 謝辞 機体の組み立てやロボットのトレーニングにあたっては、名古屋工学院専門学校の方のアドバイスがあつて進めることができました。また、先輩方が残して下さったメモやデータが大変役立ちました。

Minato Matsunaga

E-Mail: minatomatsunaga@gmail.com

〈キーワード〉 AI ロボット、ディープラーニング、自立走行

[参考文献]

[1]Karol Zieba(NVIDIA Corporation), End to End Learning for Self-Driving Cars,2016