

深層学習を用いた物体検出

2018SC090 田中淳平 2018SC092 寺本佳弘

2018SC107 芳野茂樹

指導教員：中島明

1 はじめに

自動運転は近年、著しい発展を遂げており、実用化への期待が高まっている。自動運転化が進むことによって、運転する人のストレスの軽減や生産性の向上が見込まれる。また、社会的な貢献として、交通渋滞の緩和や交通事故の防止がある。

自動車を運転する際には、周囲の環境を認識し、起こりうる状況を予測して、必要な操作を行わなければならない。自動運転でも同様のことがいえ、周囲の環境認識予測がとも重要となっている。自動運転車に環境認識させる方法として深層学習を用いたものがある。本研究では、カメラを用いて信号機を認識をすることを目指している。

2 物体認識における深層学習の概要

本章では物体認識の学習にあたって必要となる深層学習について示す。[1]

2.1 深層学習

従来の機械学習では別々に行われていた特徴抽出と分類ルールを同時に進めることができる。精度の良いモデルを構築することができ、大量の学習データを扱えるデータ環境の充実、自由度が高く複雑なニューラルネットワークを簡単に実装できるフレームワークや計算能力の充実、精度よく安定に動作するための各層の構造・学習方法の発展など、数多くの要素が重なり、精度のよいモデルを構築できるようになる。

2.2 ニューラルネットワークの基本構造

次に深層学習に用いられているニューラルネットワークの基本構造について説明する。ニューラルネットワークは、脳の働きを模倣するモデルとして提案されたもので、ノード（結節、人工ニューロン）が集まった層と呼ばれる構造を複数持っている。また、基本的な構造は入力層、出力層、隠れ層に分けることができる。

2.3 ニューラルネットワークの学習方法

ある入力データ（ベクトル） \mathbf{x} 、および正解データ（ベクトル） \mathbf{y} について関数 f のモデルとしてニューラルネットワークを用い、そのパラメータ Θ を、 N 組の学習データ

$$D = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$$

から決定する問題である。ニューラルネットワークを学習する基本的な要素として、次の3つが挙げられる。

2.3.1 損失関数

一般に機械学習で用いられる損失関数を2種類紹介する。

(1) 2乗損失

$$\mathcal{L}(D, \Theta) = \frac{1}{2} \cdot \frac{1}{N} \sum_{n=1}^N \|\mathbf{y}_n - \hat{\mathbf{y}}\|^2$$

(2) 交差エントロピー

$$\mathcal{L}(D, \Theta) = -\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K y_{n,k} \log \hat{y}_{n,k} \quad *k = \text{カテゴリ数}$$

2.3.2 確率的勾配降下法

損失関数の極小値を逐次的に探索する手法の1つである。

$$W_{t+1}^{(l)} = W_t^{(l)} - \frac{\alpha}{B} \sum_{b=1}^B \frac{\partial \mathcal{L}(\mathbf{x}_b, \mathbf{y}_b, \Theta_t)}{\partial W_t^{(l)}} \\ *B = \text{ミニバッチのデータ数} (1 \leq B \leq N)$$

2.3.3 誤差逆伝播法

以下に示す式は誤差逆伝播法を理解する上で重要な式となっている。この式は入れ子の形になっていることから前の計算結果を使うことができ、勾配を求める計算量を減らすことができる利点がある。

$$\frac{\partial \mathcal{L}}{\partial W^{(l)}} = h^{(l-1)} \left(\frac{\partial \mathcal{L}}{\partial z^{(l)}} \right)^T \\ \frac{\partial \mathcal{L}}{\partial z^{(l)}} = \frac{\partial h^{(l)}}{\partial z^{(l)}} \frac{\partial h^{(l+1)}}{\partial h^{(l)}} \frac{\partial h^{(l+2)}}{\partial h^{(l+1)}} \cdots \frac{\partial h^{(L-1)}}{\partial h^{(L-2)}} \frac{\partial h^{(L)}}{\partial h^{(L-1)}} \frac{\partial \mathcal{L}}{\partial h^{(L)}}$$

3 YOLOv5 を用いた物体検出

以上に示した、深層学習を用いた認識手法としてYOLOがある。本章ではYOLOの概要について示す。[2]

YOLOを用いた物体検出アルゴリズム YOLO 複数のバージョンがあり、YOLOは2015年6月に論文が発表された。物体検出し画像や動画をコンピューターに読み込ませることによって画像の中に写っている物体が何であるのか、またその物体である確率（確率値）を出力できる。上記のことを可能としているのは深層学習でニューラルネットワークを用いているからである。

また、YOLOの最新のものとしてYOLOv5がある。特徴として、データセットをYOLOv5で簡単に学習できることにより、自分自身で検出したいオブジェクトを簡単に設定することができる。

今回の研究では信号機の模型で自動運転における信号機の認識をYOLOv5を用いて行った。[4]

3.1 YOLOv5 を動作させた環境

YOLOv5 を動作させた環境は以下のようである。

使用 PC スペック

OS : Ubuntu18.04.6LTS

CPU : 11th Gen Intel(R) Core(TM) i7-11700F @ 2.50GHz

メモリ : 32GB

GPU : NVIDIA Corporation Device 2504

python 環境

python : 3.9

使用したカメラ

iPhone 11

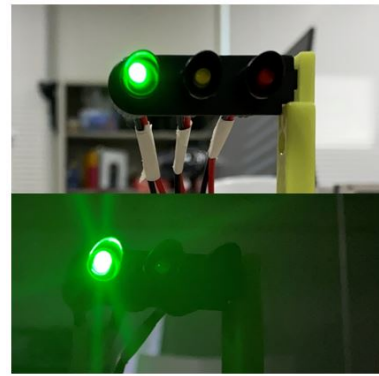


図1 上:明るい 下:暗い

3.2 信号機認識の精度向上

本章では信号機認識の精度向上において、背景の色、明暗、距離について場合分けを行い、YOLOv5 の精度の評価や比較を行う。また、YOLOv5 の評価指標として Precision, Recall, Average Precision (AP), Mean Average Precision (mAP) が用いられる [3]。本章でも以上の評価指標を用いて、評価や比較を行う。また、今回は認識結果を定量的に判断するために、確率値を用いる。一般的に認識結果の確率値が 0.9 以上であれば認識ができているといえる。

4 YOLOv5 を用いた信号機認識

4.1 明るさを変えた場合での信号機の認識

明るい場合と暗い場合の 2 パターンで写真を撮り、学習・認識を行った。この場合の '明るい' は写真を撮った部屋の電気を付け外からの日光を入れ、'暗い' 場合は部屋の電気を消し外からの日光を遮断することで区別した。図 1 に明るい場合と暗い場合で学習させた画像の例を示す。

以下に学習設定を示す。

学習データの枚数:430

検証データの枚数:170

バッチサイズ:16

エポック回数:300

学習率:0.01

4.1.1 目的

実際の自動運転を行う場合は常に昼のような明るく信号機が見やすい状況下にあるわけではなく、当然夜のように暗く信号機が見にくい場合もあるので、明るさを変えて学習・認識を行うべきだと判断した。

4.1.2 結果

学習結果

図 2 に明るさを変えた場合を教師データとした、YOLOv5 の学習結果を示す。図 2 で損失は 0 に収束し、Precision, Recall と mAP は 1 に収束している。結果としては望ましい結果となっている。

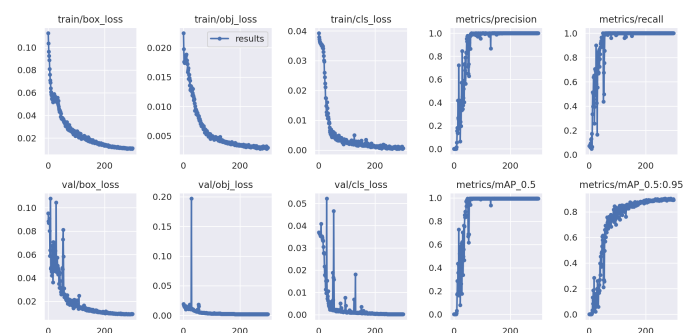


図 2 教師データが明るい場合での学習結果

認識結果認識時間:

一枚目 : 0.008[s]

二枚目以降 : 0.004[s]

図 3 に実際に認識した結果、表 1、表 2 に認識結果を示す。

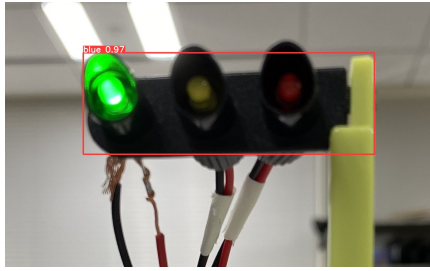


図3 教師データが明るい場合での信号機の認識結果 (カメラとの距離 10[cm])

表1 教師データが明るい場合での信号機の認識結果

距離 [cm]	青	黄	赤
10	0.98	0.95	0.67
20	0.96	0.98	0.92
30	0.92	0.97	0.92
40	0.96	0.97	0.87
50	0.91	0.94	0.65
60	0.93	0.93	0.64
70	0.71	0.75	0.54(黄)
80	0.43	0.58	0
90	0	0	0
100	0	0	0

表2 教師データが暗い場合での信号機の認識結果

距離 [cm]	青	黄	赤
10	0.89	0.88	0.75
20	0.95	0.95	0.88
30	0.89	0.93	0.82
40	0.86	0.85	0.60
50	0.90	0.88	0.62
60	0.72	0.84	0.48
70	0.75	0.80	0.48
80	0.74	0.78	0.56
90	0.43	0.70	0.34
100	0.36	0	0.33

4.1.3 評価

明るい場合は青色が 10~60[cm] まで確率値が 0.9 を超えているが、暗い場合はすべての距離において認識できなかった。黄色についても明るい場合では 10~60[cm] まで確率値が 0.9 以上であったが、暗い場合ではどの距離でも確率値を 0.9 を超えることはなかった。また明るい場合は赤色の確率値が 20~30[cm] まで 0.9 以上であった。黄色の場合と同様に赤色でも暗い場合はどの距離においても確率値が 0.9 を超えることはなかった。

これらの結果から総じて明るい場合よりも暗い場合の方が認識精度は悪かったといえる。信号機の色のみで判断しているのであれば、周りが暗い時は色の判別はしやすいはずであるが、認識の精度は下がった。では何故精度が落ちたのかと考えた時、周りが暗くなると信号機の外枠が確認

しづらくなってしまうので、認識するには信号機の色だけでは無く信号機の形を認識し、光っている場所も考慮して色を判別しているのではないかと考えた。

そこで、実際に信号機の色のみで判断しているかどうかを判別するために次にバウンディングボックスの大きさを変えて認識を行った。

4.2 バウンディングボックスの大きさを変えた場合での信号機の認識

バウンディングボックスの大きさを大, 中, 小の 3 種類に分けて学習・認識を行う。バウンディングボックスの大きさ設定は次のようにした。また、全てにおいて、実験機との距離 10[cm] であり、背景は白という条件の下行った。

1. 大：信号機の外枠よりも大きめにする
2. 中：信号機の外枠に沿うようにする
3. 小：信号機の色の部分のみを正方形の形にする

以下に学習設定を示す。

学習データの枚数:386

検証データの枚数:154

バッチサイズ:16

エポック回数:300

学習率:0.01

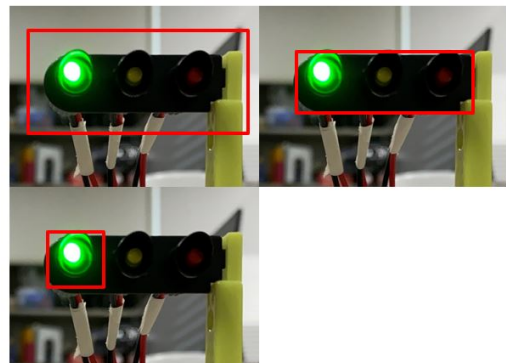


図4 左上:バウンディングボックス大, 右上:バウンディングボックス中, 左下:バウンディングボックス小

4.2.1 目的

先ほどの明るさを変えた場合の結果から、信号機の外枠や光っている場所も含めて認識しているか判断するためにバウンディングボックスの大きさを変えることで、バウンディングボックス内のみで認識しているかどうかを確認する。また、学習させる範囲を変えることで収束するまでの時間が変化するのを知りためである。

4.2.2 結果

学習結果

教師データがバウンディングボックス大きさ大, 中, 小での学習結果としては損失は 0 に収束し, Precision, Recall と mAP は 1 に収束しているため結果としては望ましく

なっている。

認識結果認識時間:0.018[s]

教師データと同様の条件 (カメラとの距離 10[cm] であり、背景は白という条件でバウンディングボックスの大きさを変えた) で認識実験を行った際の確率値の結果を表 3 に示す。

表 3 バウンディングボックスの大きさを変えた場合での信号機の認識結果

色 \ 大きさ	青	黄	赤
大	0.9	0.85	0.71
中	0.98	0.95	0.98
小	0.53	0.62	0.45

4.2.3 評価

結果の表 3 から分かるようにバウンディングボックスの大きさが中いわゆる信号機の外枠に沿うようにした際に最も確率値が高く、次に信号機の外枠よりも大きめにした場合、最後に信号機の色部分を正方形の形にした場合である。信号機の光っている色の部分すなわち、青、黄、赤信号の形は同じであり、光っている色のみが違うという点から、YOLOv5 における認識は色による識別よりも、形によって識別の方が重要であるということが分かる。バウンディングボックスの大きさを変えた場合では、色よりも形が重要であると述べたが、今回の結果から、色も重要であることが分かる。

4.3 背景の色を変えた場合での信号機の認識

背景の色が白い場合と黒い場合の 2 パターンで写真を撮り、学習・認識を行った。この場合の‘白’は信号機の背景に白い画用紙を入れて写真を撮り、‘黒’は信号機の背景に黒い画用紙を入れて写真を撮ることで区別した。図?? に背景の色を白、黒での画像を示す。以下に学習設定を示す。

学習データの枚数:430

検証データの枚数:170

バッチサイズ:16

エポック回数:300

学習率:0.01

4.3.1 目的

先ほどの暗い場合での認識を行っている際に、もともと認識できていた信号機の後ろに人が映り込むとその信号機を認識しなくなった事から認識する際には背景が関係しているのではないかと思ったからである。

また、教師データを作成する際に、矩形にぴったりと収まらない場合は少なくはなく、少々背景が矩形に含まれてしまう。その結果、背景の状況や色によってどのような認識精度の違いや背景によってどのような影響が出るのか、またどの程度影響するのか評価する。また、どの程度影響

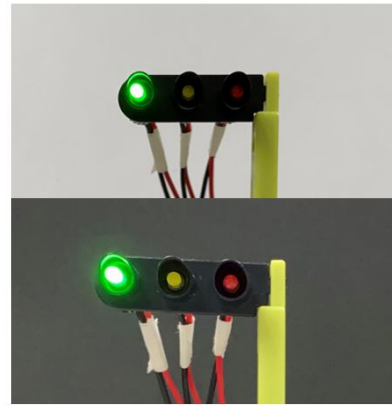


図 5 上:白 下:黒

するのか考察する。

4.3.2 結果

学習結果

背景の色を変えた場合を教師データとした、YOLOv5 の学習結果として損失は 0 に収束し、Precision, Recall と mAP は 1 に収束しているため結果としては望ましくなっている。

認識結果認識時間:0.018[s]

表 4, 表 5 では、学習結果として得られた重みを用いて信号機認識を行った際の結果を示す。

表 4 教師データの背景が白の場合での信号機の認識結果

距離 [cm]	青	黄	赤
10	0.91	0.94	0.84
20	0.93	0.52	0.69
30	0.88	0.41	0.63
40	0.90	0.26	0.45
50	0.91	0.28	0.35
60	0.90	0.41	0.33
70	0.87	0	0
80	0.84	0	0
90	0.78	0.30(赤)	0.25
100	0.75	0	0

表 5 教師データの背景が黒の場合での信号機の認識結果

距離 [cm]	青	黄	赤
10	0.96	0.93	0.98
20	0.66	0.62	0.90
30	0.59	0.52	0.55
40	0.35	0.32	0.67
50	0.48	0	0.67
60	0.27	0.34	0.57
70	0	0.30	0.65
80	0	0	0.82
90	0	0.30	0.70
100	0.66	0.26	0.69

4.3.3 評価

背景白の場合、黒の場合ともに距離 10[cm] で精度が最大となった。背景白の場合は青が 60[cm] まで離れても 0.90 と安定した。しかし、黄と赤は 60[cm] で 0.4 前後、それ以降は 0 と LED の色によって大きな差が生じていた。一方、背景黒の場合では赤が 100[cm] まで認識でき、青と黄は認識ができない距離がいくつか見られる。背景を変えることで、mAP0.8 以上から 0 になることもあった。この結果から、背景の色の変化は、各 LED の認識精度に大きな影響を及ぼすと考えられる。

4.4 距離を変えた場合での信号機の認識

実際の自動運転を行うことを考えたときに自動運転車が信号を認識する時はほとんどの場合で車は動いているため、距離は常に動いている状態での認識をする必要がある。そこで今回は距離を変えた場合での認識を行う。

実験機との距離 10[cm], 40[cm], 70[cm] の場合において認識精度の評価、比較を行う。以下に学習設定を示す。

学習データの枚数:215

検証データの枚数:85

バッチサイズ:16

エポック回数:300

学習率:0.01

4.4.1 目的

最初に説明した明暗の認識をしている際に、信号機を動かしカメラからの距離を変化させることで認識精度に差が出たことから認識する際のカメラと信号機との距離によって認識精度に違いが出ると考えたからである。

また、実際の自動車は動きながらドライバーは信号機の認識を行い適切な判断を行う必要があります。自動運転でも同様のことが必要となる。カメラで信号機を認識する際に、距離によって認識精度にどのような影響が出るのか評価考察を行う。また、認識の精度に差が出るのではないかと考えたからである。

4.4.2 結果

学習結果

実験機との距離 10[cm], 40[cm], 70[cm] を教師データとした、YOLOv5 の学習結果としては損失は 0 に収束し、Precision, Recall と mAP は 1 に収束しているため結果としては望ましい結果となっている。

認識結果認識時間:

一枚目: 0.007[s]

二枚目以降: 0.004[s]

表 6, 表 7, 表 8 では、学習結果として得られた、重みを用いて信号機認識を行った際の結果を示す。

表 6 教師データがカメラとの距離 10[cm] の場合での信号機の認識結果

距離 [cm]	青	黄	赤
10	0.97	0.98	0.97
20	0.93	0.95	0.92
30	0.79	0.73	0.43
40	0.51	0	0
50	0.60	0	0
60	0	0	0
70	0	0	0
80	0	0	0
90	0	0	0
100	0	0	0

表 7 教師データがカメラとの距離 40[cm] の場合での信号機の認識結果

距離 [cm]	青	黄	赤
10	0	0	0
20	0	0.28	0.38
30	0.86	0.90	0.90
40	0.79	0.87	0.82
50	0.54	0.86	0.92
60	0	0.89	0.90
70	0	0.81	0.84
80	0	0.66	0.77
90	0	0.64	0.74
100	0	0.26	0.48

表 8 教師データがカメラとの距離 70[cm] の場合での信号機の認識結果

距離 [cm]	青	黄	赤
10	0	0	0
20	0	0	0
30	0	0	0
40	0	0.56	0.54
50	0.42	0.93	0.89
60	0.36	0.94	0.95
70	0.38	0.89	0.93
80	0.30	0.93	0.91
90	0	0.94	0.84
100	0.35	0.85	0.81

4.4.3 評価

表 6, 表 7, 表 8 の結果から固定した距離周辺でなければ精度が落ちてしまう。教師データとしたものは先ほど示した 3 つの距離であり、その距離であれば高い確率値となるが、それ以外での信号機は教師データ学習されていないために認識する際に精度が悪くなってしまったと考える。

次に距離ごとでの比較を行う。教師データが 10[cm] の場合では、距離 10[cm] の場合では、確率値は 0.97~0.98 であり、同様に 40[cm] の場合では 0.79 0.87, 70[cm] では 0.38~0.93 となっており、カメラとの距離が離れるほど

認識の精度が落ちることがわかる。また 40[cm] では認識した際に 30[cm] の場合で一番確率値が高く、70[cm] では認識した際に 50[cm], 60[cm] の場合で一番確率値が高くなった。このように教師データと同じ距離で認識した場合でも、離れている物体を認識することは難しいということが分かる。この結果からより多くの教師データが必要であるのではないかと考える。

4.5 全ての場合分けを合わせた際の信号機の認識

最後に場合分けを行った教師データをまとめて YOLOv5 に学習をさせて認識を行った。

以下に学習の設定について示す。

学習データの枚数:2572

検証データの枚数:1028

バッチサイズ:16

エポック回数:300

学習率:0.01

4.5.1 目的

自動車を運転する際には様々な状況が考えられる。そのために一つの重みを用いて認識を行うことが必要となる。上記で示したような場合分けをしたものを合わせることによって、認識の精度にどのような影響が出るのかを評価考察する。

4.5.2 結果

学習結果

図?に全ての場合分けを合わせたものを教師データとした、YOLOv5 の学習結果を示す。図??で損失は 0 に収束し、Precision, Recall と mAP は 1 に収束している。結果としては望ましい結果となっている。

認識結果認識時間:0.004[s]

表 9, 表 10 に認識結果を示す。

表 9 明るい場所での信号機の認識結果

距離 [cm]	青	黄	赤
10	0.98	0.97	0.89
20	0.95	0.96	0.81
30	0.92	0.94	0.81
40	0.95	0.95	0.82
50	0.93	0.95	0.92
60	0.95	0.97	0.87
70	0.90	0.89	0.83
80	0.87	0.94	0.69
90	0.78	0.75	0.74
100	0.41	0.64	0.34

表 10 暗い場所での信号機の認識結果

距離 [cm]	青	黄	赤
10	0.96	0.96	0.97
20	0.96	0.95	0.92
30	0.94	0.95	0.91
40	0.92	0.96	0.89
50	0.92	0.95	0.88
60	0.90	0.80	0.90
70	0.94	0.80	0.91
80	0.88	0.92	0.92
90	0.72	0.90	0.88
100	0.81	0.84	0.73

4.5.3 評価

すべての場合を教師データにして行った場合の認識結果から、10~100[cm] どの距離においても認識することができており、確率値が 0.9 を超えている範囲もどの場合よりも広く、また、認識精度が低くなりがちだった暗い場合では、認識した際の確率値が大きく低下することもなく比較的安定した精度で認識を行えていることがわかる。このことから、一番精度がよいと判断できる。

5 おわりに

本研究では物体検出の基本となる深層学習や、ニューラルネットワークについて学習し、実際に最新の物体検出の手法 YOLOv5 を用いて物体検出を行った。また、これまで PC に接続されているカメラからの映像の信号機を認識してきたが、今後実験機を用いたリアルタイム認識を行っていく。

参考文献

- [1] 日本ロボット学会 監修, 香月 理絵 編著, 荒井 幸代 etc 共著『自動運転技術入門』. オーム社, 東京, 2021.
- [2] 三谷 大暁:『YOLO とは? 物体検出の手法を大きく進化させた YOLO を徹底解説』. AI 研究所, 2021-06-21, <https://ai-kenkyujo.com/yolo/>, (参照 2022-01-09).
- [3] Jonathan HUI『mAP (mean Average Precision) for Object Detection』. 2018-03-07, <https://jonathanhui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173> (参照 2022-01-09).
- [4] 株式会社アルファシステムズ:『YOLOv5 を使った物体検出』. アルファテックブログ, 2021-08-20, https://www.alpha.co.jp/blog/202108_02, (参照 2022-01-09).