研　究　ノ　ー　ト

自動運転車いすの障害物検出機能を実現するための コンピュータビジョン技術に関する研究と実装

大内　誠[1]、岡　正彦[1]、柴田理瑛[1]
[1]東北福祉大学

要旨

本報告では、電動車いすの安全性を向上させることを目的に、クルマと同じような安全運転支援システムを実現するための要素技術として、最新のコンピュータビジョンについて調査を行った。その上で、その応用である物体検出機能を実装し、機械学習を行った。その結果、全体のmAP（mean Average Precision）は0.526と低い結果となった。その理由として学習に用いた画像の枚数が不足していたことが考えられるため、今後は枚数と種類を増やす必要がある。次に、単眼カメラによる距離推定機能を実装した。その結果、2m～4mの距離ではほぼ正確に距離推定できたが、それ以外の距離では精度が悪かった。今後は、安全性を確保するためにステレオカメラやRiDAR等のセンサ利用を検討する必要がある。最後に、開発した電動車いすの走行実験を行った。その結果、前方に障害物を検知すると衝突せずに安全に停止することができた一方、処理が追いつかずに衝突してしまうこともあった。

キーワード：自動運転、ディープラーニング、物体検出、コンピュータビジョン、深度測定

1．研究の背景と目的

1．1　セーフティ・サポートカーについて

近年発売されているクルマのほとんどには安全運転支援システムが搭載されており、交通事故の防止や軽減化に寄与していると言われている。このようなシステムを搭載したクルマは、通称サポカー（セーフティ・サポートカーの略）と呼ばれている。サポカーの特徴として「衝突する前に自動でブレーキをかける」「踏み間違え時の急発進を抑制する」「車線からはみ出さないようにハンドル操作のアシストをする（警報音を鳴らす）」「自動で前照灯を切り替える」などの機能を有しており、経済産業省では、これら4つの機能を搭載したクルマに「サポカーS（ワイド）」という愛称をつけて、自動車メーカー等と共に普及に取り組んでいる[1)]。

サポカーが、実際に交通事故抑止に寄与したかどうかに関して興味のあるところである。2020年に警察庁が公開した「高齢運転者交通事故防止対策に関する調査研究調査結果[2)]」によると、2017～2018年に登録・届け出されたクルマにおいて、10万台当たりの第1当事者人身事故件数は、全車両が509件であったのに対してサポカーS（ワイド）対象車が297件とサポカーの方が約42％ほど低かった（Table 1）。つまり、安全運転支援システムが搭載されたサポカーは、交通事故を起こしにくいという結果になったのである。なお、ここで言う全車両とは、普通車と軽乗用車のことであり、サポカーS（ワイド）対象車を含んでいる。

Table 1．登録台数10万台当たりの第１当事者事故件数比較，［2］より引用

	第１当事者 人身事故件数	登録台数	登録台数10万台当たりの第１当事故件数 （１当事故件数×10万／登録台数）
全車両	524,281	102,962,091	509.20
サポカーＳ（ワイド）	2,500	841,202	297.19

１．２　サポカーを実現するためのコンピュータビジョン技術

ここで、サポカーＳ（ワイド）を実現するための要素技術について着目してみる。

①「衝突する前に自動でブレーキをかける」

クルマの前方にいる歩行者・クルマ・自転車・バイク等を検出し、衝突の可能性がある場合には、運転者に対して警報を鳴らす。さらに衝突の可能性が高い場合には、自動で自車（自分が運転するクルマ）のブレーキを作動させる。

②「踏み間違え時の急発進を抑制する」

クルマのすぐ前方や後方に建物やクルマ等の物体が存在するとき、停止、もしくは低速走行状態にあるクルマのアクセルを踏み込んで急発進させようとすると、エンジンの出力を自動的に抑えてスピードが出ないようにする。

③「車線からはみ出さないようにハンドル操作のアシストをする（警報音を鳴らす）」

クルマが走行車線からはみ出しそうになった場合や、はみ出してしまった場合に、運転者に対して警報を鳴らしたり、車線から逸脱しないようにハンドル操作をアシストしたりする。

④「自動で前照灯を切り替える」

対向車の有無を検出して前照灯を自動でハイビームにしたり、ロービームにしたりする。

上記の４つの機能を実現するためにはいずれも、クルマのフロントガラスやリアガラス付近にカメラやレーダを設置し、常に周囲の状況を検出・判断する必要がある。このようにクルマに「目」の役割を持たせるためのAI（Artificial Intelligence：人工知能）技術のことを「コンピュータビジョン（computer vision）」と呼ぶ。コンピュータビジョンについては、第3章で詳しく述べる。

１．３　電動車いすの安全性について

超高齢化社会が進展する中、電動車いすの需要が急激に高まっている。中でも運転免許返納者の増加に伴い、電動車いすの販売台数が増加し、2021年には国内で約1万9千台も販売されている[3)]。また、下肢に障がいのある人にとっても、電動車いすは自立のための重要なアイテムとなっている。一方、警察庁から発表された資料[4)]によると、走行中に段差で動けなくなったり、障害物に衝突してしまったり、側溝に転落してしまったりというような事故が年間100件以上発生しており、毎年数名の死傷者も出ている（Table ２）。

Table 2．当事者別電動車いすの交通事故死傷者数の推移，［4］より引用
※第１当事者：過失（違反）がより重いか又は過失（違反）が同程度の場合にあっては、被害がより小さい方の当事者、第２当事者：過失（違反）がより軽いか又は過失（違反）が同程度の場合にあっては、被害がより大きい方の当事者

		2012年	2013年	2014年	2015年	2016年	合計
死者数	第１当事者	2	1	0	0	0	3
	第２当事者	5	4	6	6	9	30
	計	7	5	6	6	9	33
負傷者数	第１当事者	5	7	0	3	2	17
	第２当事者	201	179	175	167	142	864
	計	206	186	175	170	144	881

電動車いすの場合、制限速度が時速6kmとなっているため、クルマほど大きな事故になることは少ないが、歩道上を走行するため人と衝突したり接触したりする事故が発生しやすい。このような事故を防ぐためにも、クルマと同じように安全運転支援システムの開発と搭載が急務となっている。

１．４　研究の目的

前述した通り、電動車いすの需要が急速に高まっている中、衝突事故や転落事故が散見されるようになってきた。中でも高齢ドライバーの運転免許返納率が徐々に高まっていることもあり、今後ますます電動車いすの需要が高まる可能性があり、事故の予防が急務となっている。

そこで、我々は、市販の電動車いすにカメラとAIを搭載し、クルマのサポカーと同じような安全・安心機能を有する電動車いすを開発することにした。将来は、目的地を入力するだけで、屋外や屋内を問わず自動走行し、目的地までナビゲーションしてくれるような高度なAIの開発を目指している。

本論文では、その第一段階として、人間が操作する車いすが障害物等に衝突することを未然に防ぐためのコンピュータビジョンと電動車いす制御機能を試作し、その性能を評価する。

１．５　過去の研究事例

溝端らは、普段電動車いすを利用している高齢者約1,000名に対して利用に関するアンケートを実施した[5]。その結果、常に歩道を通行すると答えた割合が7割程度であったが、歩道を通らないことがある、または、常に車道を通行すると答えた割合が3割程度であった。歩道を通らないことがある、または、常に車道を通行すると回答した人にその理由を聞いたところ、「歩道が狭い（63％）」「歩車道境界の段差（51％）」「歩道に障害物がある（47％）」と回答した割合が高く、他に「歩行者が多く通りにくい」「自転車がこわい」「電動車いすは車と同じだから」という意見もあった。

佐藤らは、社会的な問題となっている電動車いすの安全を技術的側面から支援することを目的に「障害物・段差危険回避技術」「段差踏破技術」「直進走行技術」「対人衝突回避技術」「対人協調走行技術」等の研究を行い、それぞれ試作機を開発した[6]。中でも障害物・段差危険回避技術開発においては、段差の検出にレーザレンジファインダを用い、5cm以上の下り段差と上り段差を検出し、車いすが段差を回避できるようになった。また、障害物の検出に関しては、全方向ステレオカメラを使用することにより、1秒間に30回の頻度で車いす周囲の環境を三次元的に捉えることができ、時速6kmで前進している車いすにおいて、約5.6cm進むごとに1回の頻度で障害物を検出することができたと報告している。

敷島らは、クルマに搭載した単眼カメラで、動いている障害物を検出するための機械学習システムを提案した[7]。このシステムは、RGB画像と三次元地図から生成した深度マップを入力して，Semantic

SegmentationとDepth Completionを同時に行なうMulti-Taskネットワークとなっている（Fig. 1）。開発したMulti-Taskネットワークを用いて実験を行ったところ、三次元地図上に存在しない動的障害物の検出精度が、既存手法と比べて、IoU（Intersection over Union：物体検出する際にAIの予測と正解がどれくらい重なっているかを表す指標）が1.4points向上することが分かった。

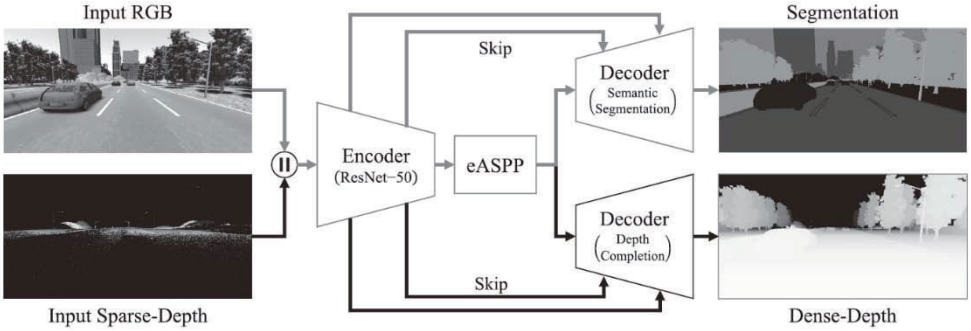

Fig. 1．　Semantic SegmentationとDepth Completionを行うMulti-Taskネットワーク［7］より引用

久留米大学では、高齢者が自由に外出できることを目指して、サイバー（AIとIoT）とリアル（介護，モビリティ）を融合したSociety 5.0に基づく「先進高齢者MaaS（Mobility as a Service）」を実現するために「福祉インテリジェントモビリティサービス」を提唱し、パーソナルモビリティ（電動車いすなど）の自動運転技術の開発に取り組んでいる[8]。この自動運転システムは、2D-RiDAR、Intel RealSense、ビーコン、ソナーなどの複数のセンサを組み合わせることにより、限られたエリアに限っては数センチメートルの自己位置推定精度を実現した。また、屋外においては、前述のセンサ類に加え準天頂衛星システム「みちびき」も利用して自動運転を実現している。ただし、環境によっては数十センチメートルレベルの自己位置推定誤差が生じたり、段差や大きな穴を認識できなかったりなどの課題も残されている。

2．研究の方法

まず始めに、人間が操作する車いすが障害物等に衝突することを未然に防ぐために必要なコンピュータビジョン技術について俯瞰する。次に、さまざまなコンピュータビジョン技術を応用した「物体検出」ならびに「深度測定（距離測定）」のアルゴリズムについて述べる。得られた知見を元にAIを開発し、市販の電動車いすに実装する。最後に開発物の性能評価実験を行う。

3．コンピュータビジョン

3．1　代表的なコンピュータビジョン技術

コンピュータビジョンというワードを最初に用いたのはマサチューセッツ工科大学のMarvin Minskyである[9]。彼は、コンピュータにカメラを接続して、カメラによって映し出されたものが何であるかをコンピュータに判断させようと試みたのである。その試みは容易ではなく、その後、多くの研究者たちがさまざまな手法を使ってコンピュータビジョンを実現しようとした。代表的な研究成果をいくつか挙げてみよう。

まず、Violaらが提案したHaar-like特徴を用いたカスケード顔分類機がある[10]。Haar-like特徴とは、顔を構成するパーツの明暗差パターンのことで、Fig. 2のように暗いエリアと明るいエリアが隣接するいくつかのパターンで構成される。鼻のエリアの場合、鼻そのものは突き出ているので明るく、鼻の左側や右側は鼻そのものよりも暗くなる傾向がある。一方、目のエリアの場合、眼窩の周囲は暗く、下まぶたの周囲は明るくなる傾向がある。同時に、左右の眼窩の周囲は暗く、左目と右目の間は明るくなる傾向がある

（Fig. 3）。顔全体を検出するためには、パターンの大きさや画像中の位置を変えながら、明暗差が類似する場所を探索し、評価するための分類機が必要になる。この分類機は、機械学習によって数千～数万種類作成され、これらの分類機が多数決を取ることによって顔全体を検出する。これらの複数の分類機をカスケード分類機と呼ぶ。

なお、Haar-like特徴だけでは顔検出（画像中に顔があるかどうかを判別する）はできるものの、顔識別（だれの顔かを特定する）は容易ではないため後述するニューラルネットワークを利用する。

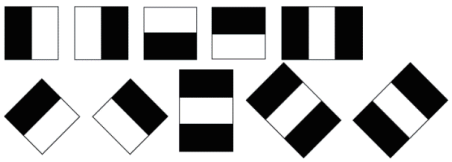

Fig. 2．Haar-likeパターンの例

Fig. 3．顔の画像から目のエリアを特定した様子，［10］より引用

次に、人の全身画像やクルマなどを検出することができるHOG（Histogram of Oriented Gradients）について述べる。前述したHaar-like特徴を用いたカスケード分類機では、正面を向いた顔の検出精度は比較的高い。なぜならば、人の顔の場合、目や鼻や口近辺の明暗パターンは概ね類似しているからだ。一方、人の体全体を含むような画像の場合、腕や足を上げたり、しゃがんだり、姿勢が変わると身体のパーツごとの明暗パターンや明暗差が大きく変化してしまい、Haar-likeパターンを適用することが困難になる。そこで、Dalalらは、明暗差を特徴量とするのではなく、局所的な画像の勾配（傾き）を求めてヒストグラム化し、正規化したものをHOG特徴量とするアルゴリズムを提案した[11]。これにより画像の部分ごとの輪郭がわかり、これらを連結すると画像に写っている物体の大まかな形状が分かる。Fig. 4は、女性の右目の、向かって左側ならびにその上側に帽子の縁（縁の角度はおおよそ45度程度であることが見てとれる）が写っており、この縁の勾配と強度をヒストグラム化している[12]。ヒストグラムの横軸にある１，２，３…９は勾配の角度を表しており、それぞれ０度、20度、40度…180度となっている。左上とその直下のヒストグラムは両者共横軸の３番目に大きな値が表れているため、その局所画像に含まれる輪郭が40度近辺にあることを表している。実際の例として、著者自身の画像からHOG特徴量を求めて画像の輪郭を求めたものをFig. 5に示す。

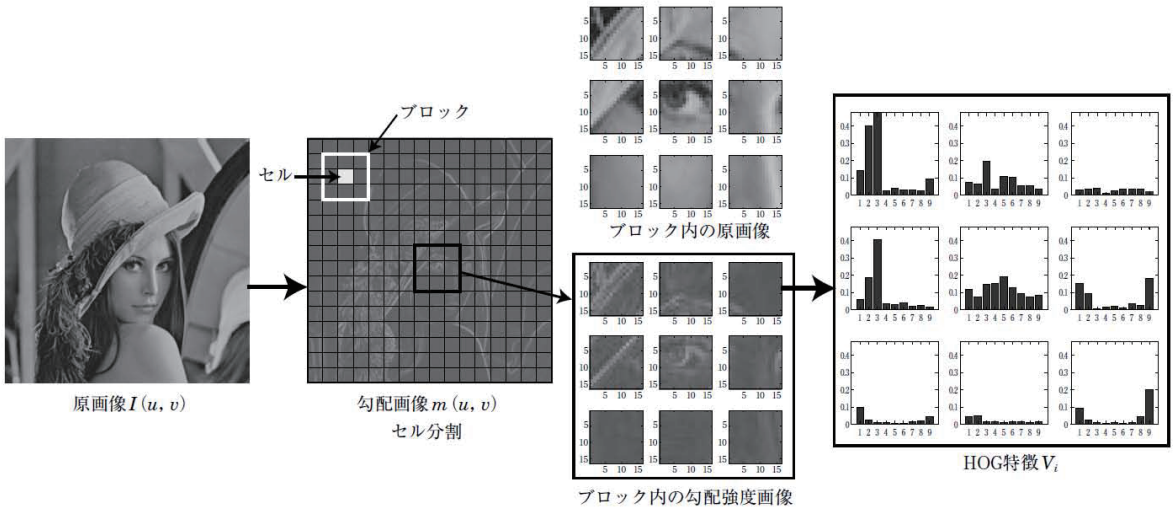

Fig. 4．HOG特徴量の計算手順，［12］より引用

Fig. 5．原画像（左）とHOG特徴量を用いて画像の輪郭を抽出した例

次に、コンピュータビジョンの研究として忘れてはならないものに深層学習（ディープラーニング、または、ディープニューラルネットワークとも言う）とConvolutional Neural Network（CNN）がある。

深層学習のベースになっているものをニューラルネットワークと言う。ニューラルネットワークは、人間の脳にある神経細胞、すなわちニューロンとその回路をコンピュータ上のソフトウェアによって模したものである。

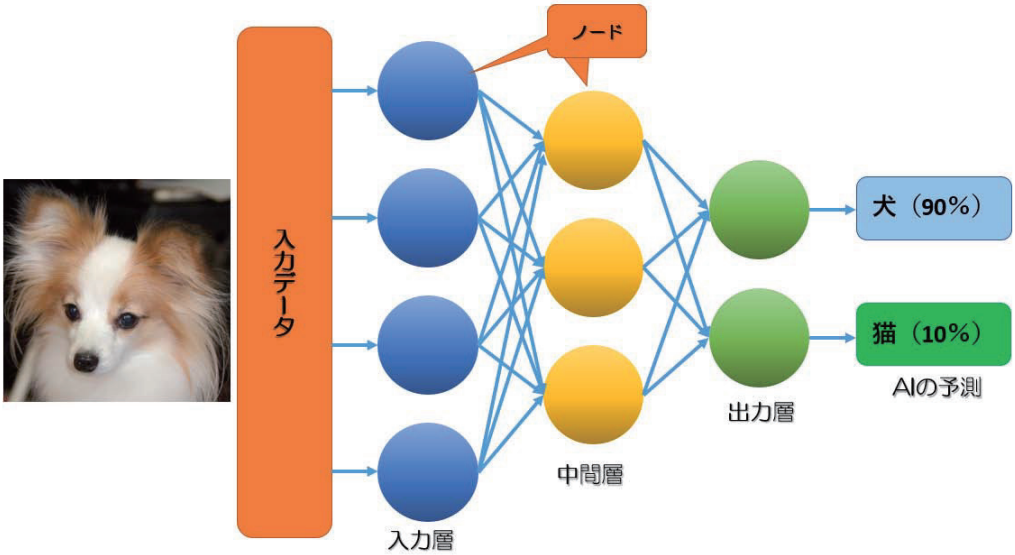

Fig. 6．ニューラルネットワークの概念図

Fig. 6 は、ニューラルネットワークの概念図である。図中のノードは、ニューロンを表す。ノードとノードは線によって連結されている。この線は、人間の神経細胞のシナプス（synapse）に当たり、データを左側（上流という）から右側（下流という）に流す役割を持っている。ノードは入力層、中間層（隠れ層とも言う）、出力層から構成され、出力層から予測結果が出力される。Fig. 6 の場合、犬の画像をニューラルネットワークに入力し、最終的に犬か猫かの予測結果を確率で出力する様子を表している。

ところで、犬の画像はどのようにしてニューラルネットワークに入力するのだろうか。画像は、コンピュータ内に取り込まれるとピクセル（点）の集まりで表現される。モノクロ画像の場合、各ピクセルは輝度の異なる灰色で表現（８ビット表現の場合、256段階）され、各ピクセルの輝度がそれぞれ別々のノードに入力される。一方、カラー画像の場合は、各ピクセルを光の三原色である赤・緑・青（RGB）に分解し、その輝度がノードに入力される。

ここで、ノードの役割について詳しく述べる。Fig. 7 は、2つの上流ノードからのデータが下流のノードに流れる様子を示している。上流ノード内のデータを下流に流すとき、線（シナプス）上にある「重みw」の値がデータに乗算される。すなわち、変数wの値はノード内のデータの重要度を表している。すべての上流ノードのデータには、それぞれ別々の重みwの値が乗算され、下流のノードで合算される。その後、さらにバイアスbの値が加算される。バイアスbは、後段の活性化関数の式を調整して、データの出力値（つまりニューロンの発火量）を調整している。活性化関数は、次のノードへ流すデータ量を決める非線形関数である。代表的な活性化関数には、ReLU（Rectified Linear Unit）関数とシグモイド（sigmoid）関数があり、どちらも非線形であるためにニューラルネットワークが複雑な関数を表現できるようになり、さまざまな回帰問題や分類が可能となった。以上の流れを式（1-1）に、ReLU関数を式（1-2）に表す。

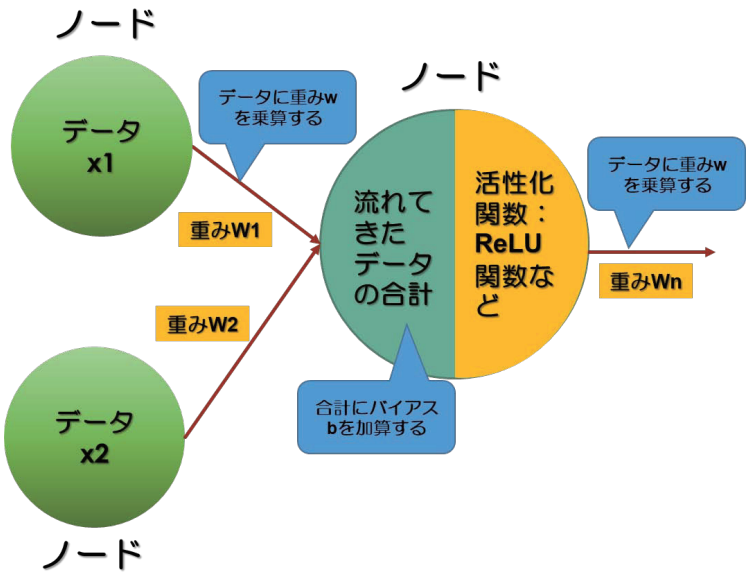

Fig. 7．ニューラルネットワークの基本構

$$\mathrm{a} = \mathrm{h}(x_1 w_1 + x_2 w_2 + b) \tag{1-1}$$

$$\mathrm{h}(y) = \begin{cases} y\ (y \geq 0) \\ 0\ (y < 0) \end{cases} \tag{1-2}$$

ここで、再度重みwの役割について考えてみる。前述した犬の画像をニューラルネットワークが犬と認識するためには、犬の特徴をいくつか探し出す必要がある。例えば、口吻、耳、牙、鼻、体躯などである。ニューラルネットワークは、これらの特徴を学習によって検出できるようになり、口吻のデータを流すと発火するエリア、耳のデータを流すと発火するエリア…のように脳の神経回路網と同様にエリアごとに役割分担をして認識する。あるエリアにあるノードが発火するということは、その近辺にあるノードに大きな値のデータが入っており、かつ、重みwの値も大きくなければならない。このとき、入力されるデータの値は、ニューラルネットワーク側では制御できないが、重みwの値はニューラルネットワーク側で制御できる。つまり、口吻の特徴を捉えるエリアには大きな重みwの値が格納されてお

り、実際に口吻のデータがそのエリアを通過すると、そのエリアのノードが発火するようになる。この重みｗの初期値は乱数で決められるが、沢山の犬の画像をニューラルネットワークに入力することにより、重みｗの値は更新される。この重みｗを更新して、犬の各特徴を捉えることができるようにすることを「学習」または「機械学習」と言う。画像認識するためのニューラルネットワークに学習させるためには、大量の画像と正解データを表すラベルデータが必要になる。このような学習を「教師あり学習」と呼ぶ。実際の教師あり学習では、ニューラルネットワークの予測値とラベル（正解データ）を比較し、その差（二乗和誤差やクロスエントロピー誤差などがあり、その総称を損失関数という）をゼロに近づけるように重みｗの値を更新する。これを勾配降下法という。勾配降下法は、その名のとおり、損失関数の勾配（傾き）を偏微分によって求め、誤差が小さくなるように重みｗの値を更新する。この時、損失関数で求めた誤差を合成関数の連鎖率によって上流の各層に戻しながら勾配を求める方式を「誤差逆伝播法（Backpropagation）」と言う。同様にしてバイアスｂも更新する。旧式のAIでは、検出する特徴を人間がプログラミングしていたが、最新のニューラルネットワークでは、自動で学習できる。「機械学習」と呼ばれるようになった所以（ゆえん）である。

次に、ディープラーニングついて述べる。前述のニューラルネットワーク内にある中間層（隠れ層）を２層以上の多層にしたものをディープラーニングという（Fig. 8）。

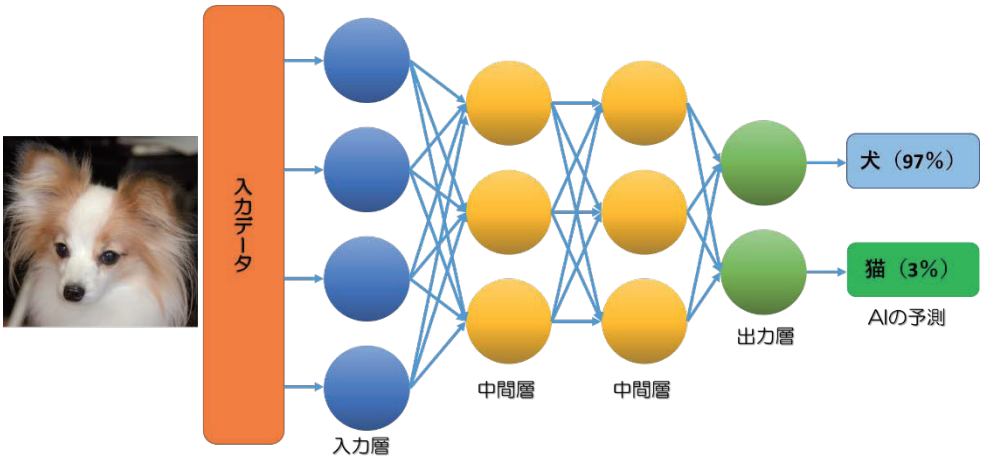

Fig. 8．ディープラーニング（深層学習）の概念図

これにより、色を判定する層、形を判定する層、…などのようにデータに含まれる特徴を各層で自動的に学習・予測できるようになった。また、層を増やしたことでパラメータ（重みｗやバイアスｂの数）が増えて表現力が増し、複雑な関数を表現できるようになった。現代のAIの性能が飛躍的に向上したのは、このディープラーニングのお陰と言っても過言ではない。なお、中間層の数を増やしすぎると計算コストやメモリー量が膨大になり、実用性が低下する。そのため、中間層を何層にするかや1層のノードの数をいくつにするかなどの「ハイパーパラメータ（人間が決めるパラメータ）」の決定には試行錯誤と経験が必要になる。

ところで、ニューラルネットワークに画像データを入力する際には、各ピクセルの輝度値を入力層の各ノードに入力すると述べた。例えば、横32ピクセル、縦32ピクセルの場合、1024個の直列に並んだノードに入力することなる。この時、2次元の画像を1次元の入力層に入力することにより、画像内の位置情報が失われてしまう。例えるなら、人間の顔にある2つの目や鼻や口などが一直線に並んでいるようなものだ。そこで、Lecunら[13]は、2次元画像の位置関係を保ったままニューラルネットワークに入力する

「畳み込みニューラルネットワーク（Convolutional Neural Network：以下、CNNと略す）」を考案した。Fig. 9にCNNの概念図を示す。原画像はピクセルごとに輝度値に変換され、２次元のまま「畳み込み層（Convolution Layer）」と「プーリング層（Pooling Layer）」に入力される。「畳み込み層」と「プーリング層」は複数回登場し、画像の特徴を抽出する。その後、1次元の全結合層（通常のニューラルネットワーク）に引き渡され、抽出した特徴を組み合わせることによってカテゴリ分類や予測を行う。

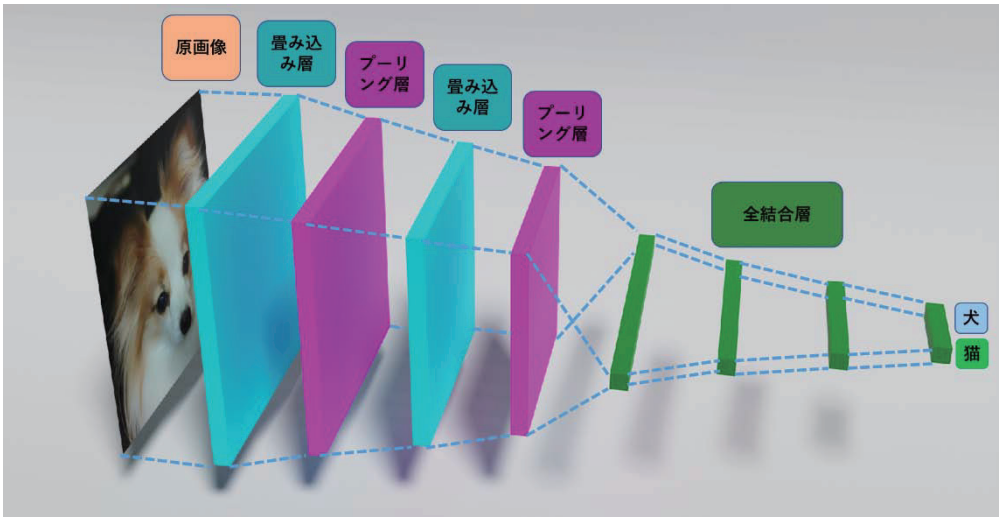

Fig. 9．畳み込みニューラルネットワークの概念図

畳み込み（convolution）とはどのような処理か、具体的な計算例を示して説明する（Fig.10）。前述したように画像データは2次元の表（行列、テンソルとも言う）の形で入力される。数値は各ピクセルの輝度を表している（カラー画像の場合は、赤緑青ごとの輝度が入力される3枚の表がある）。表の左上にある横3マス×縦3マスの黄色い枠のデータに対して、別の表にあるフィルタ、またはカーネルと呼ばれる横3マス×縦3マスの赤い枠のデータを、それぞれ同じ位置のマス内で乗算した後、すべての乗算結果を合算（これを「積和演算」と言う）し、その値を畳み込み結果を格納する別の表に代入する。続いて、黄色い枠を1マス（ピクセル）分、右にスライド（スライドするマスの数とストライドと言う）した後、上記と同様に乗算し、その総計を結果の表に代入する。黄色い枠が右端まで移動したら1マス分下にスライドし、左端に移動してから上記と同様の計算を行い、黄色い枠が右下に移動するまで同様の計算を行ったら畳み込みが完了する。

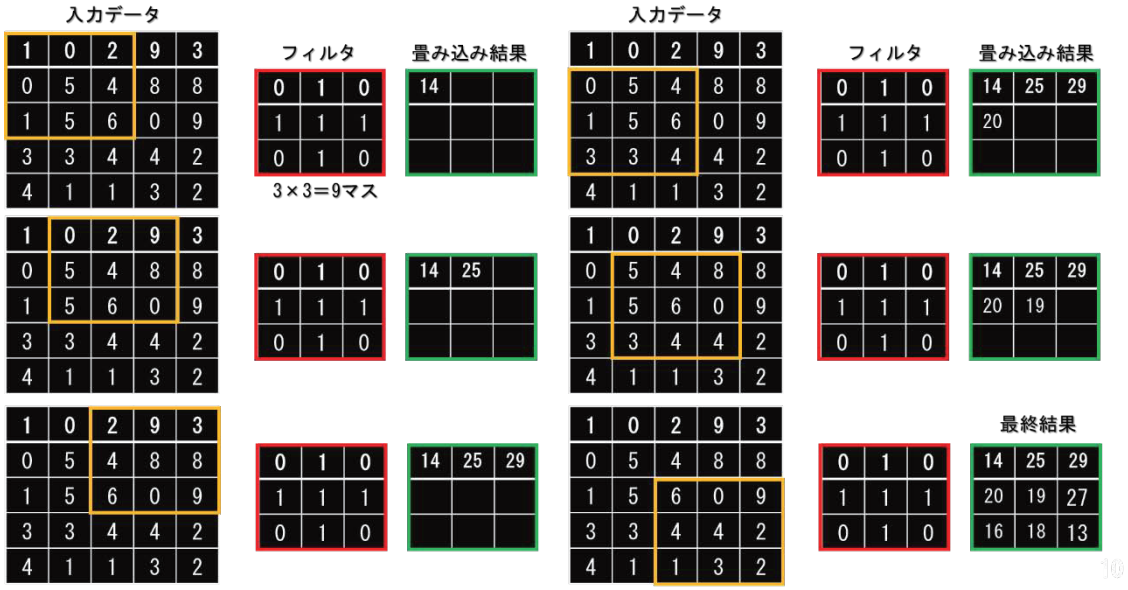

Fig.10. 畳み込みの計算例

この計算により、画像データの特徴を際立たせるのである。ではなぜ畳み込みで特徴が際立つのだろうか。実はフィルタは1種類ではなく32種類や64種類用意する。Fig.10のフィルタの場合、1が十字の形に配置され、それ以外はゼロになっている。そのため、入力された画像の黄色い枠の中にフィルタと同じくらいのサイズで輝度値が高い十字模様が存在すると積和演算の値は大きくなるが、十字模様が存在しなければ値は小さくなる。これは、ベクトルの内積を求めることと等価である。つまり、さまざまな形のフィルタを用意すれば、入力された画像内にフィルタと類似する形状が存在するかどうかが分かる。そのため、畳み込みの結果を格納する表はフィルタの数分だけ必要になる。

次に、プーリング層について具体例を示して解説する（Fig.11）。プーリングのアルゴリズムは単純で、横２マス×縦２マス（緑色の枠）の４マスの中で最大の値を、プーリングの結果の表に代入する。これをマックス・プーリング（max pooling）という。最大値を取り出すことにより画像の特徴を際立たせることができ、かつ、画像サイズを小さくできる。なお、平均値を代入するアベレージ・プーリング（average pooling）もある。

Fig.11. マックス・プーリングの計算例

３．２　物体検出とは何か

コンピュータビジョンによって、車いす前方に存在する障害物を検出・回避するためには、物体のカテゴリ（種類：人、自転車、看板、クルマなど）のみならず、大きさ、位置、物体までの距離をリアルタイムに検出する必要がある。これらの中で距離を除くカテゴリ、大きさ、位置などを特定することを「物体検出」と言う。これまで物体検出を行うためのアルゴリズムは種々考案されている。例えば、R-CNN[14]（Region Convolutional Neural Network）、YOLO[15]（You Only Look Once）、SSD[16]（Single Shot MultiBox Detector）などである。また、画像の全ピクセルにラベル情報を埋め込み、ピクセル単位に写っている物体の分類を行う技法を「セマンティックセグメンテーション（Semantic Segmentation）」と言い、クルマの自動運転や医療分野での利用が期待されているが、全ピクセルについてディープラーニングするため学習にも推定にも大きなマシンパワーが必要になる。

では、ここで、ニューラルネットワークによる画像分類をベースにした物体検出技法の代表的なアルゴリズムについて解説する。

まず始めに、物体検出を行うAIのほとんどは「教師有り学習」である。教師有り学習とは、機械学習する際に、あらかじめ正解データ（ground truthとも言う）を用意し、ニューラルネットワークの出力層から出力された予測値と正解データを比較して、ほぼ一致するようになるまで重みwとバイアスbの値を勾配降下法で更新する手法である。そのため、物体検出においても、教師データが必要になる。物体検出の目的は、画像中の物体のカテゴリ（種類）、大きさ、位置を見つけ出すことであるため、教師データにもそれらが含まれる必要があり、それらを用意する工程を「アノテーション」と言う。以下にアノテーションツール「LabelImg」を用いた例を示す。

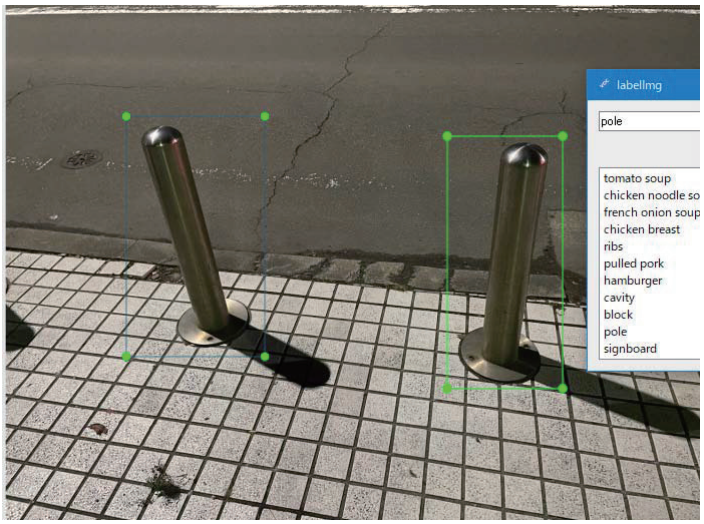

Fig.12. アノテーションツール「LabelImg」を用いてアノテーションを実行している様子（２つのポールをバウンディングボックスで囲み、カテゴリを「pole」としている）

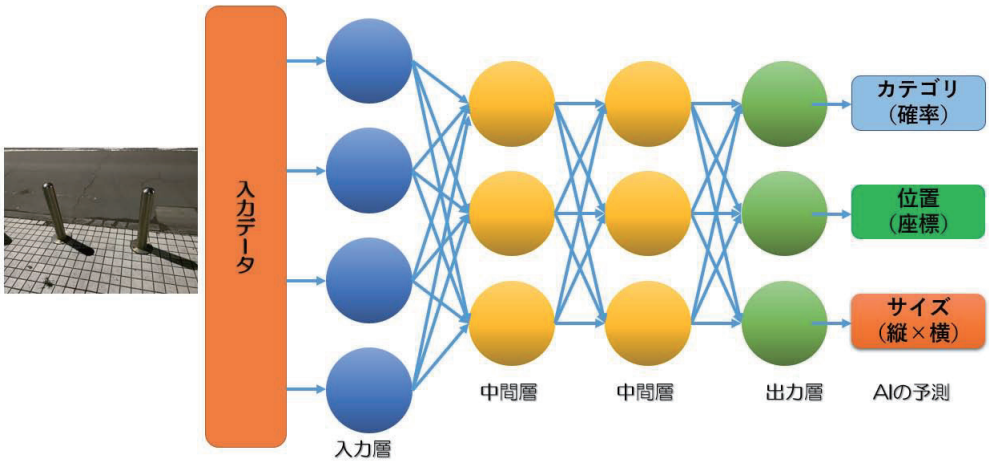

Fig.13.　物体検出ニューラルネットワークの概念図

次に、物体検出のアルゴリズムを説明する。Fig.13のように、入力層に入力するのは画像データである。画像データの各ピクセルは、RGBごとの輝度値に変換され入力層に２次元のまま入力される。ニューラルネットワーク内では、前述した畳み込みが行われ、最終的に出力されるのは、画像内に写っている物体のカテゴリ、位置座標、およびサイズである。

最初に検出するのは、各物体の中心位置である（Fig.14）。この例では、Fig.12のようにポールが２本立っており、畳み込みによって物体を検出した後、その中心位置を図のように（Pole１とPole２）表示する。出力層Ａには、物体カテゴリ数分の２次元確率マップが用意される。このマップは、例えば横８マス×縦８マスの格子（グリッド）で構成され、それぞれのマス内のデータは、物体の中心位置が含まれる確率を表している。つまり、横８マス×縦８マスのグリッドの場合、１枚の画像内に存在する最大64個の物体を検出できることになる。出力層Ａを含むニューラルネットワークでは、物体の中心位置を含むマスに高い確率が出力されるように機械学習するわけである。

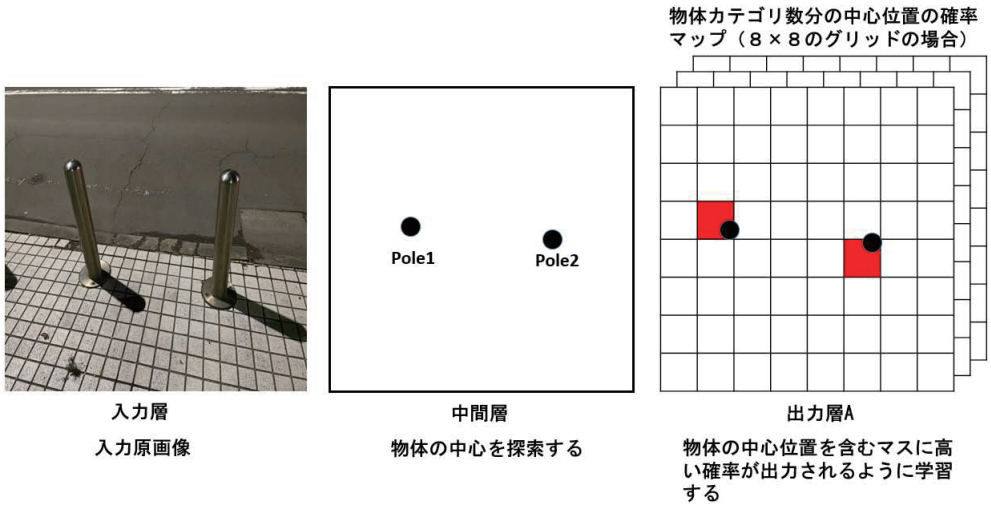

Fig.14.　検出した物体カテゴリとおおよその中心位置を出力する出力層Ａ

このままでは、物体のおおよその位置しか分からないため、次に正確な中心座標を求める必要がある。Fig.15は、検出した物体の中心位置が、そのマスのどこにあるか座標値で表す手順を示している。まず、

Poleの中心が入っているマスの左上の座標を０、右下の座標を１とする。つまり、マス内の横座標と縦座標を０～１に正規化した数値で表現することにより、検出した各物体の正確な中心座標を求めることができる。出力層Ｂを含むニューラルネットワークでは、アノテーションで示した物体の中心座標とAIが推定した物体の中心座標の差を最小化するように機械学習するわけである。

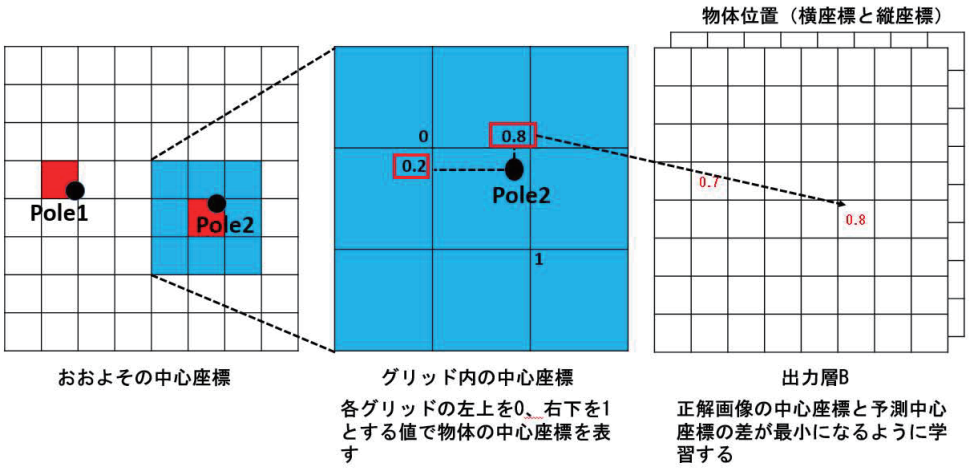

Fig.15. 検出した物体の正確な中心座標を出力する出力層Ｂ

最後は、検出物体のサイズ（幅と高さ）を求める（Fig.16）。前述した中心座標と同様に、アノテーション時に物体の幅と高さを指定しており、これが教師データとなる。AIは、その予測値とアノテーションの値の差が最小になるように学習する。なお、サイズは、原画像のサイズを１としたときの相対値で表現する。

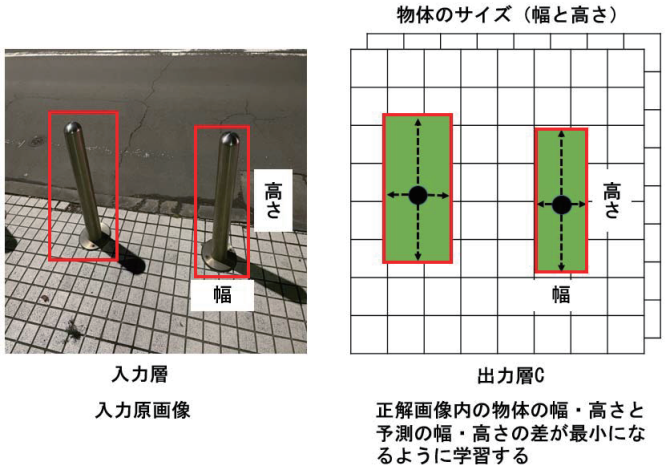

Fig.16. 検出した物体のサイズ（幅と高さ）を出力する出力層Ｃ

３．３．深度推定とは何か

物体検出により車いす前方に存在する障害物のカテゴリ、位置（距離を含まない）、サイズが判別できるようになった。次は車いすから障害物までの距離を推定する必要がある。これを「深度推定」と言う。

深度推定は、さまざまな技法が考案されているので、簡単に紹介する。

まず、超音波センサやミリ波レーダを用いて深度推定するものがある。超音波は、１秒間に約340m進むので音波を発射し、障害物に反射して戻ってくるまでの時間を計測することにより距離を測定できる。

しかし、超音波の場合、障害物の形状や素材によっては反射しにくく、測定不能となる場合がある。

一方、ミリ波レーダは、周波数帯30～300GHzの非常に高い周波数の電波を用い、照射してから障害物に電波が反射して戻ってくるまでの時間から距離を測定する。ミリ波レーダは、夜間や天候の悪い日でも安定して距離を測れる反面、小さい物体や段ボール等の検知能力が低下する。

ステレオカメラは、人間の目と同様に２つのカメラを用いて「カメラ間距離」「焦点距離」「視差」から距離を計算する。２つのカメラの光軸を正確に平行にキャリブレーションする必要があるが、それがうまくいけば精度の高い深度測定が可能である。

RiDAR（Light Detection And Ranging）は、レーザ光を周囲に照射し、その反射光を捉えて再構築することにより、距離はもちろんのこと、周囲に存在する物体の形状も正確に捉えることができる。これまでのRiDARは、レーザ光を照射する部分をモータで360°回転させることによりあらゆる角度の障害物を捉えることができたが、最近のRiDARはSolid State方式やMEMS（Micro Electro Mechanical Systems）方式などの、モータを使わないでも360°レーザ光を照射できるタイプが登場し、故障率が下がった。

最後に、単眼カメラとニューラルネットワークを用いた深度推定について説明する。ステレオカメラと違って単眼カメラはカメラを１台しか使わない。人間の場合、片目を閉じて周囲を見ると、距離感が著しく減少するが、単眼カメラでも同様のことが起こる。では、どのようにして深度を推定するのだろうか。この手法では、教師あり学習によって画像と奥行きの対応関係を直接学習する。したがって、学習段階でニューラルネットワークに入力するのは、大量の画像データとそれに対応する距離データであり、画像の撮影時に並行して距離データも収集する必要がある。そのため、先に述べたRiDAR（全方位型）が使われることが多い。学習が完了すると、文字通り単眼カメラで深度推定ができるが、ステレオカメラ、RiDAR、ミリ波レーダ等に比べると距離推定精度が落ちる。しかしながら、カメラが１台で済むため安価でシステムがシンプルになるというメリットがある。

４．障害物検出機能を有する電動車いすの実装

４．１　開発物の概要

本研究で開発する電動車いすには、次のような機能を実装する。

①車いす前方に存在する障害物を検出する。

②障害物のカテゴリ（種類）、位置、サイズを特定する。

③障害物までの距離を推定する。

④車いすが障害物に衝突しそうなときは自動で停止する（車いすの運転は人間が行う）。

システムの構成図をFig.17に示す。使用した電動車いすはWHILL Model CR、カメラはLogicool C270n、PCはPanasonic Let's Note CF-SZ6（CPU：Intel Core i7600U，RAM：16GB）である。

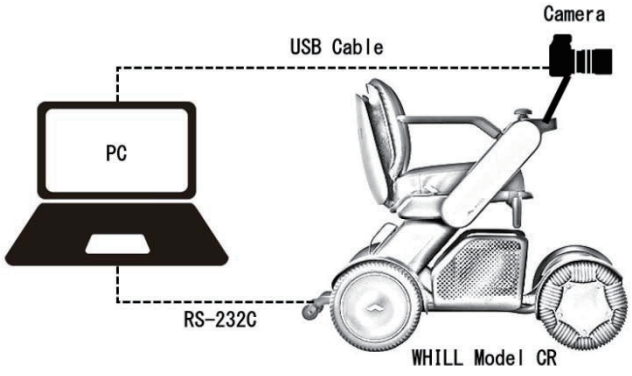

Fig.17．システム構成図

開発に使用したソフトウェアをTable 3に示す。今回の実装では、物体検出アルゴリズムで定評のある「YOLOv5」を採用することにした。また、距離推定（深度推定）については、ステレオカメラ方式、RiDAR方式、単眼カメラと深層学習を利用する方式について検討した結果、物体検出で使用する単眼カメラをそのまま利用でき、コスト的にも安価な「単眼カメラと深層学習を利用する方式」を暫定的に採用することにした。アルゴリズムについては、「MiDAS[17]」と「Dense Depth[18]」について検討した結果、10mまでの絶対距離が測定可能な「Dense Depth」を採用することにした。

Table 3．開発に使用したソフトウェア

カテゴリ	開発用ソフトウェア名
OS	Windows10（22H2）
プログラミング言語	Python Ver.3.8
統合開発環境	Anaconda3（64bit）
物体検出アルゴリズム	YOLOv5
距離推定アルゴリズム	Dense Depth
電動車いす制御ソフト	pywhill（WHILL Model CR SDK for Python）
アノテーションソフト	LabelImg（Ver.1.8.1）

Fig.18にプログラム・ブロックダイヤグラムを示す。プログラムは、大きく「物体検出と深度推定プロセス」ならびに「車いす制御プロセス」から構成される。物体検出のルーチンも深度推定のルーチンもCPUに対する負荷が極めて大きいため、Pythonのmultiprocessingライブラリを用いて、2つのプロセスを別々のCPUコアに割り振って負荷分散を図った。

物体検出部では、カメラから送られてきた画像の中から車いす前方の物体を検出すると、検出した物体のバウンディングボックスを深度推定部に引き渡す。深度推定部では、各バウンディングボックス内の最も小さい距離（車いすに近い）を計算し､バウンディングボックスが車いすの進行方向に存在する場合は､そのボックスの座標と距離を車いす制御プロセスに引き渡す。

車いす制御部では、車いすの進行方向で、かつ、障害物までの距離が4m以内になったら、車いすの速度を徐々に落とし、2mに達した時点で停止するように制御している。なお、車いす制御用コールバック関数は、車いすに搭載されたジョイスティック、モータ、バッテリーなどの状態情報をリアルタイムに受け取る関数である。

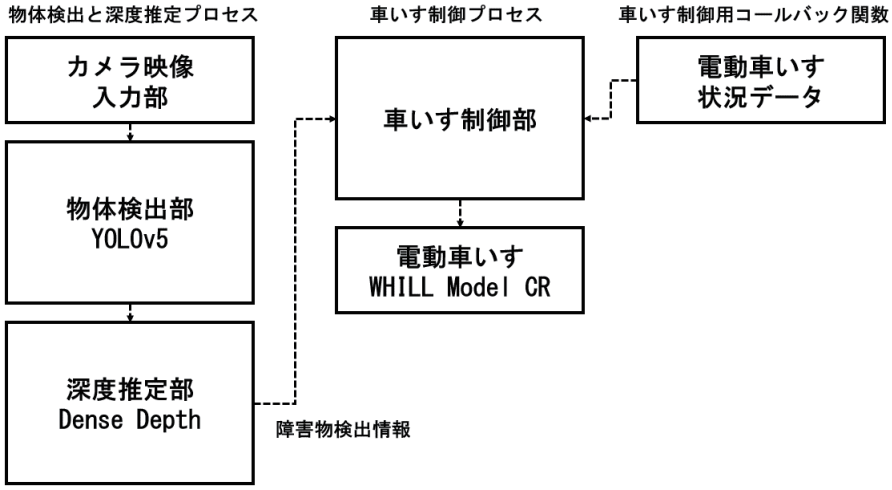

Fig.18. プログラム・ブロックダイヤグラム

４．２　機械学習工程

物体検出を行うYOLOv5を使用する場合、検出させたい画像を事前に収集し、機械学習させる必要がある。本研究では、電動車いすが通行する歩道上にある代表的な障害物として、「ブロック」「ポール」「看板」の写真をそれぞれ200枚ずつ、合計600枚撮影した（fig.19）。その後、フリーのアノテーションツールである「LabelImg」を使ってアノテーションを実施した。次に、YOLOv5の学習プログラムを用いて機械学習させた。その際のハイパーパラメータは、バッチサイズが20、エポック数（学習回数）が300回である。

Fig.19. 機械学習に使用したブロック、ポール、看板の写真の例

以下に機械学習の結果を示す。Fig.20の横軸は学習回数、縦軸は損失関数の値、つまり教師用データ（アノテーションされたデータ）とニューラルネットワークの推測の差を表している。左側の図（train/box_loss）は、教師用データのバウンディングボックスとニューラルネットワークが推測したバウンディングボックスの回帰誤差の推移を表している。簡単に言うと、学習用データのバウンディングボックスと推測されたバウンディングボックスがどれくらい合致しているかを表しており、差が小さいほどよい。次に、ニューラルネットワークが物体検出用グリッドの各セル内に物体が存在するかどうか表す指標を「確信度」といい、中央の図（train/obj_loss）は、教師用データとの差の推移を表している。右側の図（train/cls_loss）は、推測された物体のカテゴリ（種類）と教師用データとの差の推移を表している。いずれも学習が進むにしたがって差が小さくなっているため、学習が良い方向に進んでいることが分かる。

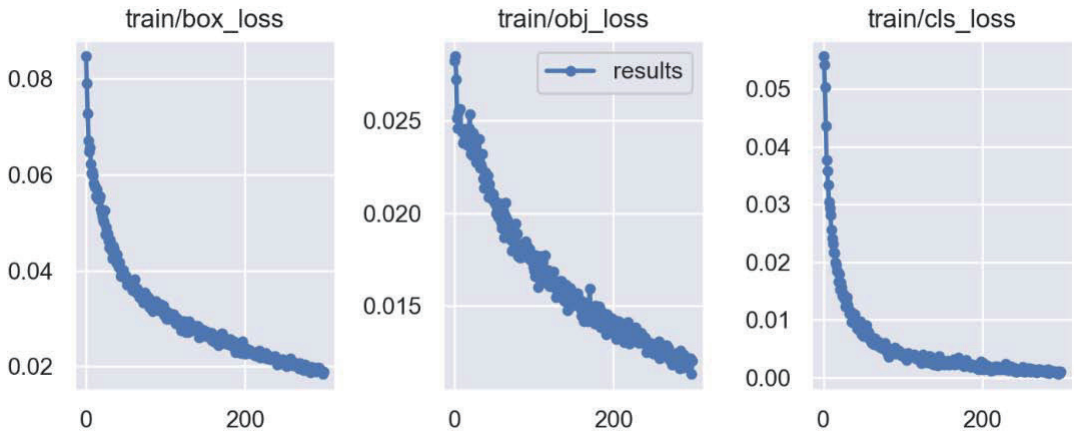

Fig.20. 機械学習時の損失関数値の推移

Fig.21は、適合率（precision）と再現率（recall）を表しており、いずれも学習が進むに従って値が1.0に近づいていることから検出力が高く、間違いを犯しにくい物体検出AIに近づいたことを示している。

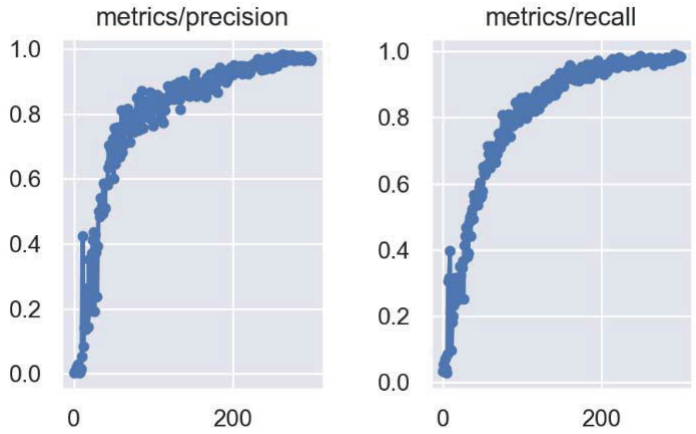

Fig.21. 適合率と再現率の推移

Fig.22は、mAP（mean Average Precision：平均適合率の平均）の推移を表している。mAPは、次のようにして求める。

①IoU（Intersection over Union）を求める。IoUは、ニューラルネットワークが検出した物体のバウンディングボックスと正解のバウンディングボックスがどれだけ近いかを表す指標である。Fig.23のようにArea of Intersection（二つのバウンディングボックスが交差した部分の面積）をArea of Union（二つのバウンディングボックスを重ね合わせた全体の面積）で除して求める。IoUが1の場合、二つのバウンディングボックスは完全に一致しており、0の場合、全く一致していないことを表す。通常、IoUが0.5以上であれば検出したバウンディングボックスは正しかった（これをTrue Positiveと言う）と判断し、0.5未満であれば正しくなかった（これをFalse Positiveと言う）と判断する）。

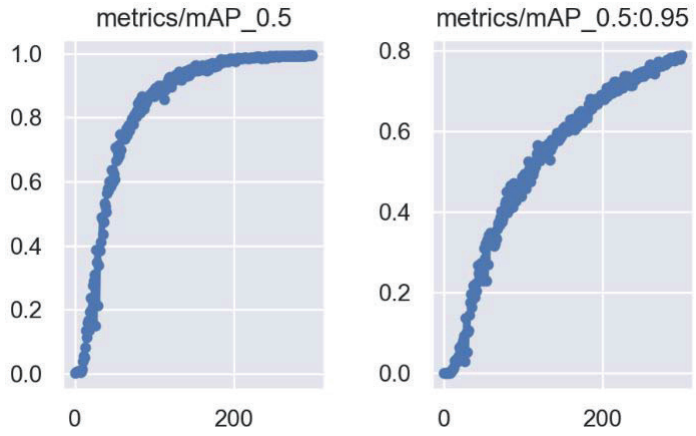

Fig.22. mAP（mean Average Precision：平均適合率の平均）の推移

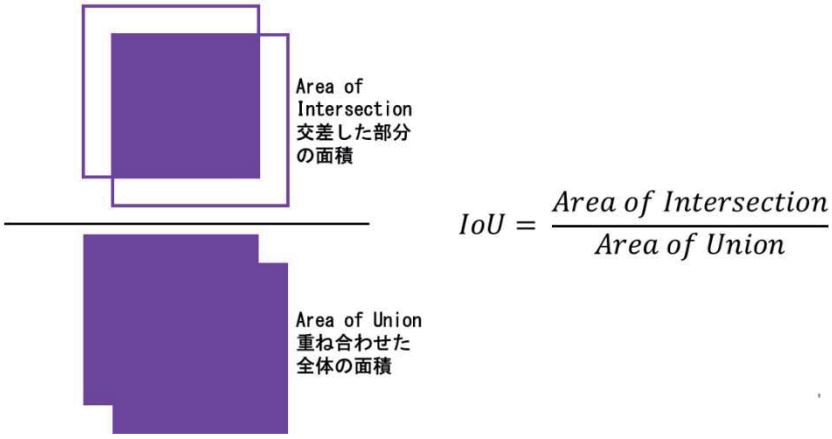

Fig.23. IoU（Intersection over Union）の求め方

②該当するバウンディングボックス内にあると予想される物体のクラス（種類）とクラス確率の最大値を抽出する。例えば、猫である確率が0.8、犬である確率が0.2の場合、クラス確率は0.8になる。

③物体のクラスごとの信頼度スコアを求める。信頼度スコアは、IoUとクラス確率の積である。

④信頼度スコアの高い順にバウンディングボックスの「正解・不正解」「適合率」を列挙する。

⑤正解となっている適合率の平均を求める。この値が、AP（Average Precision：平均適合率）である。計算例を以下に示す（Table4）。

Table 4．（例）猫と予測されたバウンディングボックスの検出結果
（この例では、ニューラルネットワークが予想した猫の数は5だが、正解は3ということを表している）

順位	信頼度スコア	正解・不正解	適合率
1位	97	**正解**	**1/1=1**
2位	93	**正解**	**2/2=1**
3位	90	不正解	2/3=0.667
4位	89	**正解**	**3/4=0.75**
5位	85	不正解	3/5=0.6

$$AP = \frac{1 + 1 + 0.75}{3} = 0.917$$

このようにして平均適合率APを求めるわけだが、なぜ順位を付けるのだろうか。これは、正解数が同じであっても信頼度スコアが大きく上位のほうに正解があるほどAPの値を大きくするためである。

このようにして各カテゴリ（例えば、猫、犬、鳥など）ごとのAPを求め、その平均を取ったものがmAP（mean Average Precision：平均適合率の平均）である。Fig.22のmAP_0.5はIoUを0.5に設定した場合の結果を表し、mAP_0.5：0.95はIoUを0.5から0.9まで0.05きざみでmAPを求め、さらにその平均を取ったものである。いずれも学習が進むに従ってmAPが1.0に近づいているため、学習時の精度は比較的良いことが分かる。

なお、深度推定を行うDense Depthについては、すでに開発者によって機械学習が実施され、学習モデルとソースコードが公開されているため、著者らによる機械学習は実施せず、公開されている学習モデルを使用した。ただし、Dense Depthの出力は、深度画像（距離を輝度で表現したもの）であるため、画像中の各ピクセルの輝度を絶対距離に変換するプログラムを著者らが開発した。

5．実験

5．1　物体検出部の性能評価実験

①実験の目的

前章で示した通り、機械学習時における本ニューラルネットワークの精度は悪くない。今度は実際に画像データを入力し、物体検出部のプログラムがどれだけ正確に障害物（ブロック、ポール、看板）を検出でいるかどうかを検証する。

②実験の手続き

機械学習時に使用した画像とは別に、ブロックの静止画像108枚、ポール53枚、看板171枚を撮影し、前章で機械学習させたニューラルネットワークに入力し、認識性能を確認する。

③実験結果

Table 5に、適合率、再現率、mAP（IoU=0.5）、mAP（IoU=0.5～0.95）を示す。このようにポールと看板の性能はまずまずだが、ブロックの性能はすべての指標においてあまり良くない。

Table 5．物体検出部の性能評価実験結果

	適合率	再現率	mAP（IoU=0.5）	mAP（IoU=0.5～0.95）
ブロック	0.434	0.195	0.240	0.087
ポール	0.600	0.819	0.681	0.368
看板	0.701	0.673	0.656	0.305
全体	0.578	0.559	0.526	0.254

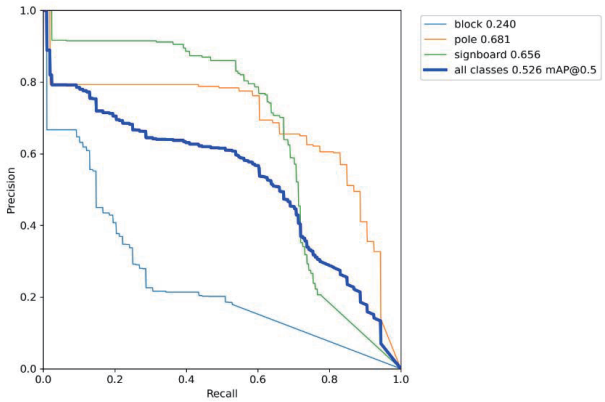

Fig.24．再現率・適合率曲線

Fig.24は再現率・適合率曲線である。曲線の下側の面積がmAP（平均適合率の平均）になるため、やはりブロックの検出性能が良くないことが分かる。Fig.25は検出した物体をバウンディングボックスで囲み出力した図である。（a）～（c）は概ね正しく検出しているが、（d）のブロックは全く認識していない。また、（e）では、ブロックは認識しているもののポールは認識しておらず、左側のブルック塀も認識していない。同様に（f）では、ポールを看板と誤認識している。

（a）（b）（c）

（d）（e）（f）

Fig.25. 検出結果をバウンディングボックスで表したもの

④考察

ポールと看板の検出精度は比較的高く、それに対してブロックの検出精度はあまり良くなかった。その理由として考えられるのは、ポールと看板は、地面に対して垂直に立っており、地面と区別しやすかったのに対して、ブロック（特に縁石）は、地面との区別がつきにくかったからだと推測する。これを解決するためには、縁石を撮影する際にさまざまな角度から行うことや、道路や歩道も学習させることにより区別させることが可能と考える。また、学習に使用した画像の枚数が600枚しかなかったことによる学習不足も要因となっているだろう。今後は、学習に使用する画像の枚数を大幅に増やすこと、また、ブロック、ポール、看板だけではなく、電動車いすが路上で遭遇するであろうさまざまな障害物についても学習させる必要がある。

5．2　深度推定実験

①実験の目的

単眼カメラとDense Depthにより推定した、カメラから障害物までの距離がどれくらい正確なのかを検証する。

②実験の手続き

PCにカメラ（Logicool社 C270n）を接続し、カメラからパイロン（三角コーン）までの距離を1m～10mまで0.5mおきに離していき、YOLOv5でパイロンを検出した後、パイロンを包含するバウンディングボックスをDense Depth側に引き渡す。Dense Depthでは、渡されてきたバウンディングボックス内の深度画像からもっとも短い距離（カメラからパイロンまでのもっとも近い距離）を探索し、絶対距離に

変換して画面に表示する。これを距離ごとに5回繰り返す。

測定は、東北福祉大学2001館4階の廊下で、晴天の日に実施した。

③実験結果

Fig.26に距離別の平均推定距離を示す。エラーバーは、標準誤差である。このように2m～4mでは推定した距離に誤差は少ないが、1mと1.5m、ならびに5m　以上で誤差が大きくなっている（グラフでは、4.5m、5.5m…9.5mは省略している）。

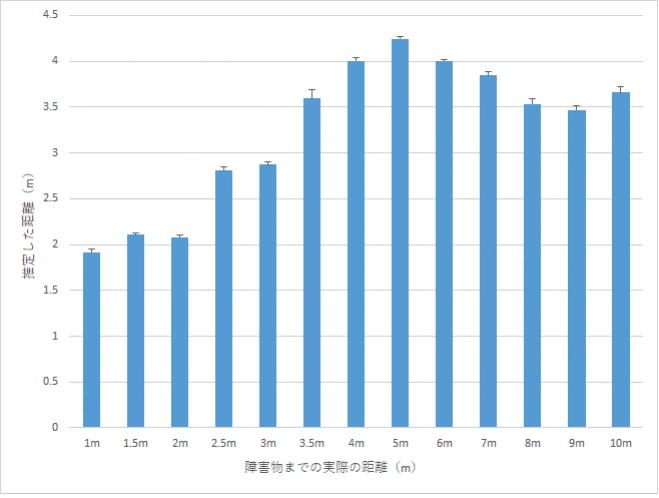

Fig.26.　距離推定実験の結果

④考察

Dense Depthでは、NYU Depth Dataset V2（以下NYU）と呼ばれる、室内の2次元画像に距離データを付加した約5,000枚の画像で機械学習している。NYUはマイクロソフト社のKinectV1を使って画像と深度データを同時に記録している。KinectV1は、基本モードで0.8m～4mまでの深度測定が可能だが、拡張モードにすると10mまでの深度測定が可能である。ただし、拡張モードでの深度精度は、あまりよくないため、4mを超える距離で誤差が検出されたのはそのためだと思われる。

今後は、深層学習と単眼カメラによる深度推定の研究が進み、精度が向上するまでの間、ステレオカメラ方式やRiDARによる距離測定方式を利用することを検討したい。

5．3　電動車いす走行実験

①実験の目的

本研究では、コンピュータビジョン技術を応用した物体検出と距離推定のアプリケーション・ソフトウェアを開発し、電動車いすに実装した。本実験では、電動車いすの走行中、前方に障害物を検出した際に衝突せずに停止できるかを検証する。また、開発に関して、今後の課題を抽出する。

②実験の手続き

東北福祉大学2001館4階の廊下に「立て看板」を設置し、立て看板の前方5mの位置からまっすぐ立て看板目がけて電動車いすを直進させ、立て看板に衝突せずに停止できるかを検証する。

電動車いすの運転は、実験者が電動車いすに乗ってジョイスティックで行う。運転速度は約3km/hとする。なお、事前にアプリケーションソフトウェアの設定で、障害物から2m以内に車いすが入ったら、車いすが自動停止するように設定し、障害物から停止した電動車いすまでの距離を計測する。計測は10回行う。

③実験結果

実験結果をTable 6に示す。このように安定しない結果となった。衝突も10回中5回も発生した。

Table 6．電動車いすの衝突実験結果（停止した車いすから障害物までの距離）　（cm）

	1回目	2回目	3回目	4回目	5回目	6回目	7回目	8回目	9回目	10回目
距離	102	衝突	165	310	衝突	衝突	178	衝突	285	衝突

④考察

このような結果となった原因を考察してみる。まず、最大の原因は、障害物の検知から距離測定までのレスポンスタイムが3秒前後もかかったため、車いす制御部に距離データが届いて停止命令を出すまでに時間がかかってしまい、オーバーランして衝突することになってしまったと思われる。また、廊下のタイルをブロックと誤認識したり、壁を看板と誤認識することもあり、物体検出そのものが不安定だったために距離データに影響を与えたものと思われる。加えて、前述の距離推定実験でも述べたとおり、2m未満や4mを超えるような距離推定には大幅な誤差を含んでいるも影響を与えていると思われる。

これらを解決する方法としては以下のものが考えられる。

（1）物体検出や深度推定のアルゴリズムを見直し、軽量化を図る。

（2）深度推定に関しては、ステレオカメラ方式やRiDAR方式に変更する。

（3）高速なハードウェア（CPUやグラフィックボードなど）を導入する。

（4）物体検出の精度を上げるために機械学習に利用する画像の枚数を大幅に増やす。また、種類も増やす。

6．まとめ

本研究の到達目標は、電動車いすを安全安心、かつ快適に自動走行させることである。本報告では、その第一弾として、電動車いすの安全性を向上させることを目的に、クルマと同じような安全運転支援システムを開発・実装することにした。そのための要素技術として、最新のコンピュータビジョンのメカニズムについて調査した上で、その応用である物体検出機能を、YOLOv5をベースにして実装し、機械学習を行った。その結果、ポールのmAPは0.681、看板のmAPは0.656であったのに対して、ブロックのmAPは0.240と著しく低い結果となった。その理由として学習に用いた画像の枚数が不足していたことが考えられるため、今後は枚数と種類を増やす必要がある。また、単眼カメラによる距離推定機能を、DenseDepthをベースにして実装した。その結果、2m～4mの距離ではほぼ正確に距離推定できたが、2m未満や4mを超える距離では精度が悪かった。単眼カメラによる距離推定は、コスト的にもシステム的にも魅力があるが、安全性を確保するためにはステレオカメラやRiDAR等のセンサ利用を検討する必要がある。最後に、開発した電動車いすの走行実験を行った。その結果、衝突せずに停止できることもあったが、停止位置が安定しなかったり、オーバーランして衝突してしまうこともあった。今後は、各機能の精度を上げつつ、自動走行機能の開発に着手したい。

謝辞

本研究は、東北福祉大学感性福祉研究所において、文部科学省の研究施設運営支援の助成により実施した。また、本研究を推進するに際して、コニカミノルタ株式会社の上田隆司氏、東北福祉大学情報福祉マネジメント学科4年の村上宙君、丸山奏君、庄子命成君にご支援いただいた。

WHILL and the WHILL logo are registered trade marks of WHILL, Inc., used with permission.

参考文献

1）「安全運転サポート車」の普及促進に向けた取組，経済産業省，https://www.meti.go.jp/policy/mono_info_service/mono/automobile/sapoca/sapoca.html, 2017

2）「高齢運転者交通事故防止対策に関する調査研究調査結果」，警察庁，pp.47，2020

3）「電動車いす国内出荷台数（電動車いす安全普及協会会員合計）」，電動車いす安全普及協会，https://www.den-ankyo.org/society/transition.html, 2023

4）「電動車いすの交通事故 最近の交通事故の実態」，警察庁，pp.10，2017

5）溝端光雄,北川博巳,「高齢者のモビリティ確保のための電動車いす利用に関する研究」，土木計画学研究講演集，2002

6）佐藤雄隆，松本治，後藤茂樹，本間敬子，加茂光広,「障害者自立支援機器等研究開発プロジェクト統括研究報告書」，産業技術総合研究所，pp.3-16，2010

7）敷島惇也，田崎豪,「単眼カメラと三次元地図を用いた動的障害物の検出と三次元復元」，計測自動制御学会論文集Vol.57 No.1，pp.37-46，2021

8）「対話型AI自動運転車いすを核とした福祉インテリジェントモビリティサービスの開発　―社会実装に向けた産学官連携体制について―」、久留米大学研究報告43号、pp.2-12，2021

9）Richard Szeliski, “Computer Vision: Algorithms and Applications”, Springer London, pp.11, 2011

10）P.Viola and M.Jones, “Rapid Object Detection using a Boosted Cascade of Simple Features”, CCVPR2001, 2001

11）P.Dalal and B.Triggs, “Histograms of Oriented Gradients for Human Detection”, International Conference on Computer Vision & Pattern Recognition, 2, pp.886-893, 2005

12）庄野　逸,「局所画像特徴量」，映像情報メディア学会誌Vol.67，No.3，pp.256～258，2013

13）LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D., “Backpropagation Applied to Handwritten Zip Code Recognition”. Neural Computation 1（4）: 541–551, 1998

14）R.Girshick, J.Donahue, T.Darrell and J.Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation”, CVPR, 2014

15）J.Redmon, S.Divvala, R.Girshick and A.Farhadi, “You Only Look Once : Unified, Real-Time Object Detection”, CVPR, 2016

16）W.Liu, D.Anguelov, D.Erhan, C.Szegedy, S.Reed, C.Fu1 and A.C.Berg, “SSD: Single Shot MultiBox Detect”, ECCV, pp21–37, 2016

17）R.Ranftl, K.Lasinger, D.Hafner, K.Schindler and V.Koltun, “Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022

18）I.Alhashim and P.Wonka, “High Quality Monocular Depth Estimation via Transfer Learning”, arXiv e-prints, 2018