深層強化学習を用いた自動駐車のシミュレーション

米元聡[1]　鶴山侑生[1]　新田龍一[1]

概要：深層強化学習を用いた運転行動学習の 1 つに自動駐車の学習がある．自動駐車とは，予め検出したターゲットとなる駐車スペースに車両を誘導するタスクである．本稿では，ターゲット位置，車両の速度などの観測情報をもとにアクセル，ブレーキ，ステアリング操作の行動を深層強化学習により決定する手法を提案する．提案手法の有効性を検証するために、自動駐車学習のシミュレーションを行った．実験の結果，ターゲットとなる駐車スペースに安定して誘導できることを確認した．また，車両の向きを考慮する改善により，学習性能が向上することを確認した．

キーワード：自動運転シミュレーション，深層強化学習，自動駐車シミュレーション

Simulation of Autonomous Parking Based on Deep Reinforcement Learning

SATOSHI YONEMOTO[†1]　YUMA TSURUYAMA[†1]　RYUICHI NITTA[†1]

Abstract: This paper presents a simulation of autonomous parking method based on deep reinforcement learning. In this simulation, a parking slot is used as a reaching target for autonomous parking. In this method, the vehicle agent observes the target position, the vehicle velocity, and the direction from the agent to the target. The agent selects the actions defined by the continuous values such as throttle, brake, and steering. The simulation results show that the proposed approach improved the learning performance considering the direction of the vehicle.

Keywords: Self-Driving Simulation, Deep Reinforcement Learning and Autonomous Parking Simulation

1. はじめに

近年，自動運転技術が進展し，実際の車両に自動運転機能が搭載されるようになってきている．実機を用いた運用実験，シミュレーション環境を用いた検証により自動運転の性能も急激に向上している．実機による自動運転を実現するためには，多様なセンサの情報，受信した道路情報などを統合することで，自己位置およびその周辺の道路状況をリアルタイムに把握し，安定な走行を行わせることが課題となる．

実機により安定な走行を実現することも重要であるが，自動運転の問題を一部切り出し，シミュレーションを行うことでアルゴリズムや手法の性能向上を図っていくことも重要となる．また，汎用の問題として取り扱うことで，自動運転以外の分野への手法の応用といった利点も生まれる．

自動運転技術は人工知能(AI)技術の進展とともに発展してきている．特に，観測情報をもとに行動を決定するというルールを学習する強化学習を用いたアプローチが盛んに研究されている．その中でも深層強化学習は，ゲーム画面をもとにゲームを攻略する AI から始まり，カメラ画像をもとに運転行動を学習する自動運転の研究にも応用されている．例えば，「End-to-End Learning」というアプローチでは，道路画像を入力，車の行動を出力とする形で畳み込みニューラルネットワークの学習を行う．

この他，先進的な自動運転に関連する研究には，Waymo 社の模倣学習（Google 社の Self-Driving Car Project）[1]，Apple 社の低リスク運転行動方策の学習[2]などがあり，自動車メーカー以外からのプロジェクトも進行している点が興味深い．

関連して，自動運転に利用可能な走行シミュレータが数多く提案されている[3][4][5]．例えば，CARLA[3]では，最新の CG 技術を用いて，天候や街並みなどをリアリスティックに表現した道路シーンの生成が可能であり，公道を走る自動運転車の学習を想定した本格的なシミュレータとなっている．走行シミュレーションに関するコンテストが頻繁に開催されており，自動運転エンジニアに同じシミュレーション環境を提供することで，開発手法の公正な性能比較ができるようになっている．

本研究では，この深層強化学習のアプローチにもとづき，自動運転行動学習のための走行シミュレータの開発を進めている．先行研究[6][7]では，鳥瞰画像およびフロントカメラ画像からの運転行動学習を実現している．

1 九州産業大学
Kyusyu Sangyo Univ.

運転行動学習で取り扱われるシーンや行動タスクの種類はいくつか存在するが，この中の 1 つである自動駐車についても検討した．自動駐車の問題は，シーンを限定した場合の自動運転に相当する．自動駐車問題とは，駐車シーンにおいて，車を停止状態から指定した（あるいは検出した）駐車スペース(parking slot)に車両を誘導する問題のことである．

2. 走行シミュレータの概要

2.1 深層強化学習による運転行動学習

運転行動学習では，以下のような問題が取り扱われる．

(1) 道路コースの安定走行の実現

定められた道路コースを安定に走行するためのステアリング操作の獲得が基本となる．通常，白線を観測に用いる．

(2) 交通参加者の影響回避

セマンティック・セグメンテーションの情報を用いて道路内に存在する他車，歩行者などの影響を回避できるよう走行軌道を予測する．

(3) 交差点の右左折・追い越し・合流

他車との関係を把握し，安全に車線変更や合流ができるように行動する．

(4) 駐車スペースへの誘導

限定シーンにおける自動運転行動学習となる．通常の自動運転とは異なるゴールの設定が必要になる．

(1)(2)(3)については既に多くの研究例が報告されている．本研究では，(1)について提案しているが(2.2 節)，今回，Unity による実装について述べる(2.3 節)．本稿では，(4)の自動駐車問題への適用を中心に述べる(3,4 章)．

2.2 運転行動学習のための走行シミュレータ

先行研究[6][7]において運転行動学習のための走行シミュレータを開発した（図 1）．運転行動学習のための走行シミュレータでは以下の機能を実現している．

(1) 曲線モデルを用いたランダムコースの生成

学習用に 2 本の白線からなる道路コースを生成する．

(2) 車両モデル

車両モデルとして Kinematics モデルを採用する．

(3) 観測画像の生成

鳥瞰画像，フロントカメラ画像の生成に対応している．シミュレータ上のコースは 3 次元であるため，フロントカメラを想定する場合には透視投影像を生成する．

(4) 走行レーンの推定

標準的な白線検出手法を採用し，実際の道路画像にも適用できることを確認している．具体的には，白線エッジの検出，Warp 変換による鳥瞰画像生成，レーン曲線モデルによる走行レーンの推定を行うことができる[8]．これにより，推定したレーンモデルの投影画像を生成し，学習時の観測画像として利用できる．

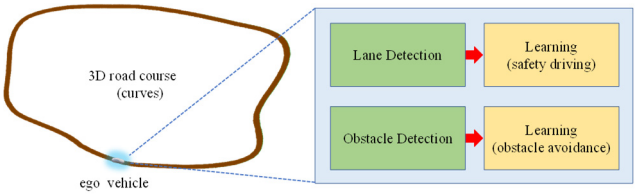


図 1 自動運転行動学習の概要

Figure 1 Our Framework for Self-driving Car Learning.

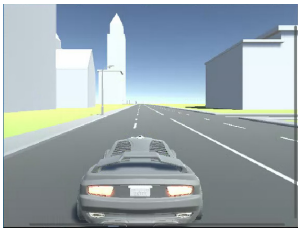
2.3 Unity による実装

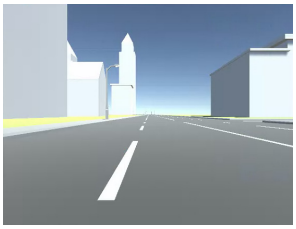
本研究では，Unity 上で動作する走行シミュレータも追加実装している．Unity を利用するという制約が生じるものの，今後，学習アルゴリズム，観測用センサが充実し，自動運転シミュレータ開発への様々な対応が行われると予想している．現在のところ，Visual Observation 機能の利用（Unity によるフロントカメラ画像の生成），Unity で生成した汎用の道路コースの導入，Unity ML-Agents の深層強化学習アルゴリズムを用いた運転行動学習が可能であるため，本シミュレータに統合した．

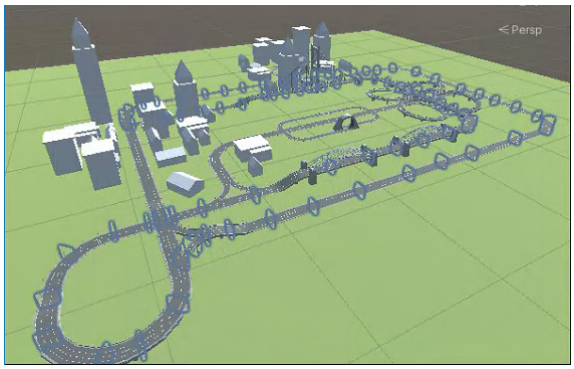
楕円形の周回コースのように簡易なコースの場合，道路の分岐は必要ないが，都市型のコースの場合は道路の分岐が複数存在するため，走行時の道標となる waypoint 情報を設定する必要がある．そこで，今回は waypoint 情報の代わりに，コース上に「チェックポイント」を配置し，これらのチェックポイントを通ることで報酬が得られるよう工夫した．

図 2(a)に学習後の自動運転の様子，図 2(c)に，コースに配置したチェックポイントを示す．道路を囲うように配置したリング形状のものがチェックポイントである．なお，道路コースとして，Unity Asset Store で公開されている「Low Poly Road Pack」を用いている．図に示すように，高架橋が設定されているため，高低差のある道路コースになっている．図 2(b)に，学習時に観測情報として用いるフロントカメラ画像の例を示している．この画像は，図 2(a)の車両のフロントカメラから見たシーンを表している．2.2 で述べた走行シミュレータと異なり，建物や道路の縁石などのオブジェクトが画像内に現れるものとなっている．ただし，実際の道路画像に比べ，道路以外の物体は限定されており，走行条件のよい環境での学習用画像となっている．

車両モデルは，Unity で一般に用いられている「Wheel Collider」をベースとした制御プログラムにより実現している．深層強化学習アルゴリズムには Unity ML-Agents の Proximal Policy Optimization(PPO)を用いる．観測情報は Visual Observation 機能により生成されるフロントカメラ画像とした．報酬はチェックポイントを正しい順路で通ると増加するよう設定している．コース外の縁石に衝突した場合，あるいは想定したコースを 1 周した場合にエピソード終了となる．400 万ステップ以上の学習で，全てのチェックポイントを達成できるようになることを確認している．


(a) 自動運転の様子


(b) フロントカメラ画像



(c) 設定したチェックポイント

図 2 Unity を用いた自動運転行動学習

Figure 2　Self-driving Car Learning Using Unity Toolkit.

3.　自動駐車への適用

3.1　自動駐車問題

現在，実際の車両に，指定した駐車スペースに自動で駐車する機能が搭載されつつある．フロント，バック，左右ミラー下の 4 点にカメラを配置し，車両本体を中心とした鳥瞰画像をリアルタイムに生成することでターゲットとなる駐車スペースを特定し，車両を誘導することができるようになっている．誘導エリア内の障害物は，カメラあるいは超音波センサなどで感知できるようになっている．また，駐車スペースの 2 本の平行な白線を手がかりとして車両を駐車スペースに誘導する手法も存在する．

自動駐車を目的として，生成した鳥瞰画像から駐車スペースであるターゲットの位置を検出する手法の開発も行われている[9][10]．ターゲットとなる駐車スペースは，白線の検出により求めることができる．図 3 に，シミュレーションで生成した鳥瞰画像に対し，検出した駐車スペースの例を示す．この例では，3 本の赤線が車両に近い駐車スペースであることを表している．

一方，自動駐車を運転行動の 1 つとして捉え，自動運転と同様に行動学習により実現する試みも存在する[11][12]．本研究でも，深層強化学習により車エージェントを駐車スペースに誘導する行動タスクの学習を試みる．

今回，仮想環境下の自動駐車のシミュレーションのため，以下の環境を想定する．

・ 車エージェントを基準としてターゲットの位置，姿勢を把握できる．
・ カメラ画像を直接の観測情報として利用しない．
・ 駐車スペース内外の障害物との接触を判定できる．

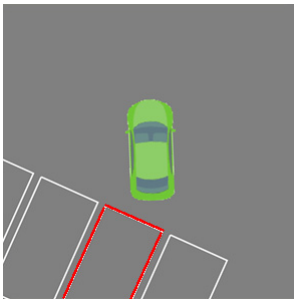


図 3 鳥瞰画像解析による駐車スペースの検出例

Figure 3 Parking Slot Detection from Bird's Eye View Images.

3.2　車エージェントの定義

本研究で基本となる車エージェントの行動，状態，報酬，エピソード終了条件を以下のように定義する（なお，後述の実験において，この定義を一部拡張することがある）．

(1) 行動

アクセル，ブレーキ，ステアリング操作とする．

(2) 観測

ターゲットの位置，車両の速度，ターゲットまでの距離とする．

(3) 報酬

ターゲットに到達したら+1，範囲外に移動したら-10 を与える．

(4) エピソード終了条件

ターゲットに到達, 範囲外に移動, 最大ステップ数を達成, のいずれかでエピソード終了となる．

図 4 に本研究で想定する駐車シーンを示す．駐車シーンは，図の車エージェントの位置を中心とした正方形の範囲とし，車エージェントがこの範囲外に出るとエピソード終了となる．図でターゲットとなる駐車スペースを青色で示している．車エージェントの初期位置，姿勢は，図に示すように，シーン中央で駐車スペースに垂直な向き（左右の向きはランダムに切り替える）とする．これは，実機の場合，ターゲットとなる駐車スペースの白線などが観測できる環境下で行動を開始することに基づいている．また，ターゲットとなる駐車スペースの位置がランダムに変更されることから，初期位置を中央に固定しても特に影響はないと考える．

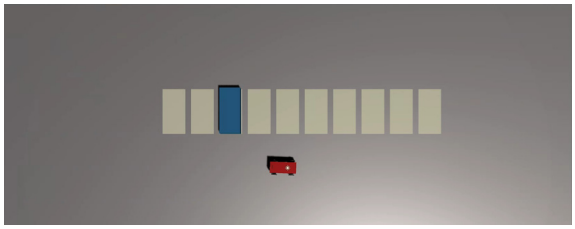


図 4 想定する駐車シーン

Figure 4　Definition of Parking Scene.

3.3 ML-Agents を用いた学習

本研究では，自動駐車のシミュレーションを，Unity の仮想環境により実現する．車両モデルには，Unity の Standard Assets で公開されている「Wheel Collider」をベースに実装し，「Rigid Body」を持つ車両オブジェクトとして定義する．アクセル，ブレーキ，ステアリングに関する連続値のみで車両オブジェクトを操作できる．

深層強化学習のアルゴリズムには，強化学習の公開プラットフォームである OpenAI Stable Baselines にも実装されている Proximal Policy Optimization(PPO)を用いる[13]．PPO は方策勾配法の 1 つで，方策オン型のアルゴリズムとなっており，行動が離散値，連続値のどちらの場合にも対応できる．今回，後述の実験では Stable Baselines の実装ではなく，ML-Agents で提供されている PPO の実装を用いている．このため，自動駐車のための学習アルゴリズムの改良は行っていない．PPO の学習に必要なハイパーパラメータについては実験に基づき調整した．

基本となる観測情報は，車両オブジェクトから見た，ターゲットの位置(tx, ty, tz)，車両の速度(vx, vy, vz)，ターゲットまでの距離(dx, dy, dz)の計 3 個の 3 次元ベクトルとなる．行動はステアリング，アクセル，ブレーキの計 3 個の連続値で表現する．PPO では，車エージェントの行動を連続値として決定できるため，車両モデルにこれらの決定した連続値を直接与えることができる．

4. 実験結果

4.1 駐車シーンの表現

実験において想定する駐車スペース全体のシーンは図 4 に示した通りである．基本となる駐車シーンでは，ターゲットの候補となる駐車スペースを横に 10 台分並べて表現する．なお，今回の実験において駐車スペースの向きを固定しているが，車エージェントは学習時にシーン内を自由に走行できるため学習性能に影響は出ない．エピソード開始時にターゲットの駐車スペースをランダムに選択する．

学習に要する最大ステップ数は 500 万とした．なお，1 エピソードあたりの最大ステップ数は 1,500 とした．学習には GPU を搭載する計算機を用いた．学習結果として，平均累積報酬，エピソード長を計測し，学習性能を判断する．具体的には，平均累積報酬が 1.0 に近づくこと，エピソード長が一定の長さに収束することで判断する．

4.2 駐車スペースへの誘導実験

まず，駐車スペースを固定せず，任意の地点として表現し，任意の位置，姿勢，速度からターゲットに向かう基本タスク[12]について学習した．学習結果である平均累積報酬の推移を図 5 のグラフに示す．青色がこのタスクの学習結果に該当する．なお，Unity ML-Agents では，Tensorboard[14]により平均累積報酬やエピソード長を取得できる．

次に，実際の駐車シーンに近づけるため，ターゲットを駐車スペースごとに区切って表現し，同様の学習を行った．駐車スペースにはいくつかパターンがあるが，今回はターゲット位置，姿勢の解析結果を利用しないので矩形でわかりやすく表現している．自動駐車は，停止状態からスタートすることを考慮して，エピソード開始時に車エージェントを速度 0 でシーン中央に設定するように変更した．ターゲットは駐車スペースの中心位置としているため，当然ながら，自由な向きから進入する形でターゲットに到達することになる．学習結果を図 5 のグラフに赤色で示している．駐車スペースを限定し，停止状態からスタートするという変更により学習の進行がやや遅れたが，100 万ステップを超えたあたりから急激に上昇している．学習時のエピソード長は，ほぼ同じ値に収束し，ほとんど差はなかった．駐車シーン下方にさらに駐車スペースを 10 台分並べて 2 列構成に拡張した場合も同様の学習済モデルでタスクを遂行できることを確認した．

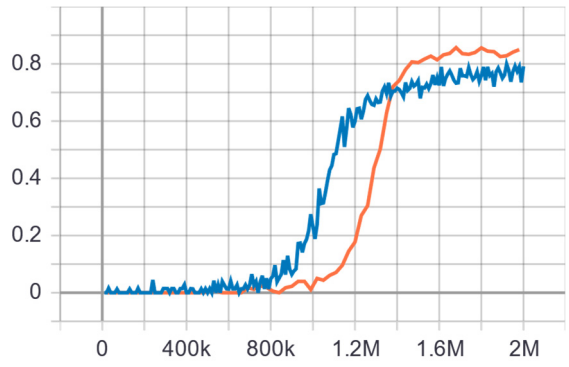


図 5　平均累積報酬の推移
Figure 5　Cumulative Reward .

4.3 車両の向きの考慮

次に，駐車スペースに進入する際の進入制限や車両の向きを観測することを検討した．具体的には以下の(1)(2)の順に工夫を行った．

(1) 駐車スペースへの進入制限

まず，ターゲットとなる駐車スペースの両サイドに障害物（壁）を設置することで進入制限した．実機の場合は，障害物センサにより判定できるため，今回のシミュレーションではその代替措置となる．障害物は判定の厳しい条件とならないように駐車スペースからやや間隔をあけて設定した．これにより，駐車スペースを大きくはみ出したり，両サイドから進入したりする場合はタスク失敗となる．この障害物については直接観測情報にも学習時の報酬にも反映されず，衝突時にエピソード終了となることで行動学習に反映されることになる．なお，今回の実験では，前進，後進のどちらでも進入できるように車両の前後の向きに制限を設けていない．

(2) 車両の向きの観測

次に，ターゲットである駐車スペースの向きを鳥瞰画像の解析により把握できると仮定し，駐車スペースへの進入の向きを観測情報に追加する工夫を行った．図 3 の検出例のように，駐車スペースの両サイドの平行線分が特定できればこのような仮定は可能となる．具体的には，新たな観測情報として，駐車スペースに対する車両の向き（3 次元ベクトル）を追加し，合計 4 個のベクトルからなる観測空間に変更した．

学習結果は，図 6 のグラフの通りとなった．赤色が(1)のみ，青色が(1)(2)の両方を反映させた場合の平均累積報酬である．(1)の進入制限だけでは，平均累積報酬は伸び悩んでいるが，(2)の車両の向きに関する観測情報を追加した場合は，200 万ステップを超えたあたりから伸び始め，平均累積報酬が 1.0 に近づく結果となった．学習時のエピソード長は，(1)のみの場合が 70，(1)(2)の場合が 30 に収束したことから，この工夫によりタスク完了までの車両の行動も洗練されていることがわかった．

学習済モデルを用いて 4 エピソードの間，それぞれ走行させた結果を図 7 に示す．図 7(a)は(1)の進入制限のみ適用した場合の走行結果，図 7(b)はさらに(2)の観測情報を追加した場合の走行結果である．それぞれ，中央下の初期位置，姿勢からターゲットとなる駐車スペース①～④までの 4 エピソード分の車両（の外接矩形）の軌跡を重ねたものを表している．エピソード終了時に車両は初期位置へ戻り，ターゲットはそれぞれ①～④の順に切り替わったことを表している．図 7(a)のエピソード③では，横側から進入したためにタスク失敗となっていることがわかる．エピソード①②はタスク成功であるが，駐車スペースの端に近い形での進入となっている．図 7(b)はいずれのエピソードもタスク成功となっているものの，車両の向きはやや斜めに進入していることがわかる．向き情報を観測しただけではターゲットに平行になるわけではないため，対策が必要である．

4.4 ターゲット接近時の減速

4.3 で想定した車エージェントの設定では，ターゲットとなる駐車スペースに停止することをタスク完了の条件としていない．そこで，車両がターゲットとなる駐車スペースに到達する際に減速するよう報酬の改良を行った．具体的には，ターゲット到達で+1 としていた報酬について，ターゲットへの接近時に速度が 0 に近づくと報酬が高くなるように変更を加えた．これは，速度の大きさを評価項目に追加することで実現できる．ただし，この計算方法では，1 ステップあたりの報酬が 1.0 を超える計算となるため，平均累積報酬が 1.0 を超えることなり，改善が必要である．4.3 の設定から，この報酬に変更して学習を行った．学習の結果，斜めに進入して停止しているが，ターゲット付近で停止状態になることを確認できた．図 8 に 1 エピソードの間の走行結果を示す．図 8(a)に車両の軌跡，図 8(b)に速度の大きさの推移を示す．図 8(a)に(1)の障害物（壁）を表示している．この実験では，3 つの障害物を設置している．切り返しの行動のため，分布は 2 つ山型となっている．また，ターゲット付近で一定の間，速度が 0 付近を保っているのが確認できる．

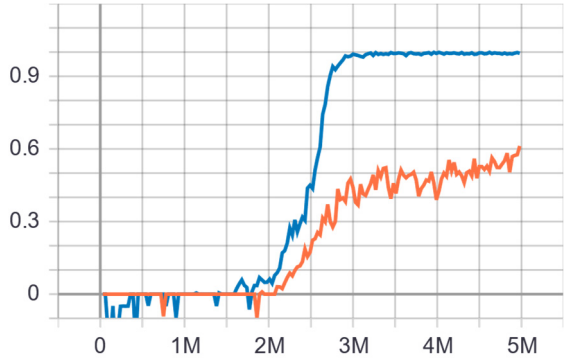
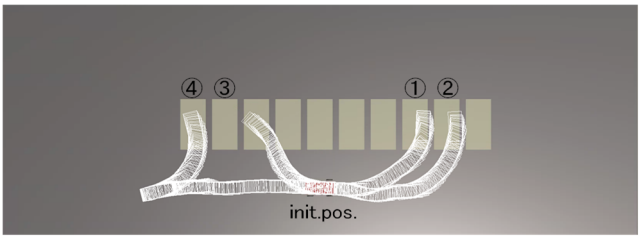
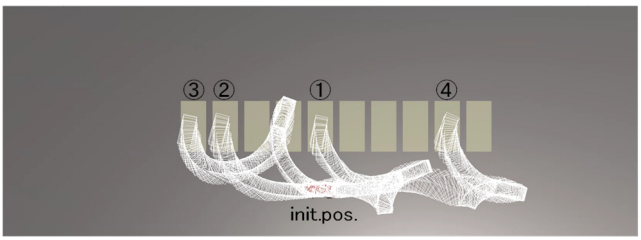


図 6　車両の向きを考慮した場合の平均累積報酬の推移
Figure 6　Cumulative Reward Considering the Direction of Vehicle.

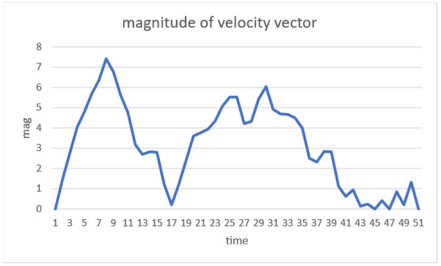


(a)　進入制限のみ適用



(b)　車両の向きを観測に追加
図 7　走行結果の比較（4 エピソード分）
Figure 7　Vehicle Paths during 4 episodes.

(a)　走行結果



(b)　車両速度の大きさの推移
図 8　走行結果と車両速度の大きさの推移
Figure 7　Vehicle path and the magnitude of the velocity.

5. おわりに

本稿では，深層強化学習を用いた運転行動学習の 1 つである自動駐車問題のシミュレーションを中心に述べた．具体的には，深層強化学習アルゴリズムの 1 つである PPO を用いてターゲットとなる駐車スペースに車両を誘導するタスクを学習する手法について提案した．ターゲット位置，車両の速度などの観測情報をもとにして，駐車に必要となる，アクセル，ブレーキ，ステアリング操作の行動を学習することができた．また，車両の向きを考慮するために，駐車スペースの両サイドに障害物を設定し，車両の向きを観測情報として追加する改良を行った．学習実験により，10 台分の駐車スペースからなる駐車シーンについて，車両を誘導できることを確認した．次に，車両の向きを考慮する工夫により，車両が駐車スペースに安定して進入できることを確認した．これは，エピソード長が減少し，平均累積報酬が 1.0 に収束したという結果に基づいている．学習済モデルを用いて走行結果を確認した結果，途中で切り返して駐車スペースに進入する行動も確認され，ターゲット位置からタスク達成のために必要な行動を計画できていることがわかる．ターゲット到達時の減速についても検討を行い，停止行動をとることを確認できた．ただし，やや斜めに進入してタスク完了となっており車両の進入行動に改善の余地がある．

今後の課題としては，縦列駐車への対応，鳥瞰画像解析による駐車スペースの検出およびその適用が挙げられる．

参考文献

[1] M. Bansal, et al. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst, arXiv:1812.03079, 2018.
[2] Y.C. Tang, et al. Worst Cases Policy Gradients, arXiv:1911.03618, 2019.
[3] Dosovitskiy, Alexey, et al. CARLA: An open urban driving simulator. In: Conference on robot learning, PMLR, pp.1-16, 2017.
[4] Shah, Shital, et al. Airsim: High-fidelity visual and physical simulation for autonomous vehicles, In: Field and service robotics. Springer, Cham, pp.621-635, 2018.
[5] Udacity Self-Driving Car Nanodegree Project, Finding lane lines, https://in.udacity.com/nanodegree .
[6] 米元聡，國武佑哉，道路俯瞰画像を用いた車エージェントの運転行動学習，火の国シンポジウム, 2020 年.
[7] 米元聡，菅河凌太，深層強化学習を用いた運転行動学習のための走行シミュレータの検討，火の国シンポジウム, 2021 年.
[8] Luca Venturi ,Krishtof Korda, Hands-On Vision and Behavior for Self-Driving Cars, Packt Press, 2020.
[9] Jae Kyu Suhr and Ho Gi Jung, End-to-end trainable one-stage parking slot detection integrating global and local information, IEEE Transactions on Intelligent Transportation Systems, 2021.
[10] A. Zinelli et al., A deep-learning approach for parking slot detection on surround-view images. In: 2019 IEEE intelligent vehicles symposium, pp.683-688, 2019.
[11] Edouard Leurent, An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env.
[12] Learning to Drive with Unity ML-Agents https://auro.ai/blog/2020/05/learning-to-drive.
[13] OpenAI https://github.com/openai/baselines.
[14] Tensorboard https://www.tensorflow.org/tensorboard.