

レーン別渋滞検知技術の提案と フィールド実験への適用評価

森 皓平^{1,a)} 横畑 夕貴¹ 林 亜紀¹ 秦 崇洋¹ 尾花 和昭¹

受付日 2021年4月5日, 採録日 2021年10月8日

概要: 本論文では、コネクティッドカーから収集可能な映像データと車速や位置情報等の走行データ情報からレーン単位の渋滞車列を検知し、先頭/末尾車両を特定することで交通流最適化に貢献するレーン別渋滞検知技術の提案と、そのアプリケーション構築および性能評価について述べる。近年、自動運転やナビゲーション高度化を目的とした地図データの高精度化や交通規制・渋滞情報・信号情報といった様々な情報を重ねたダイナミックマップの構築が進められている。これらの実現にあたり我々は、商業施設の駐車場への入庫待ち・観光地付近の路駐車両群等、レーン単位の渋滞車列を検知し情報を提供することの必要性と実現上の課題を認識した。本論文で提案する技術はコネクティッドカーの映像データから周辺車両を追い抜いた台数やすれ違った台数やその密度を求め、車速を考慮して評価することでレーン単位の渋滞車列を検知する。さらにレーン単位の渋滞車列が続く区間を推定することで先頭/末尾車両を特定し、その位置関係から渋滞長を合わせて算出する。我々は本技術を用いた評価環境を構築し、東京都港区台場にて実施された走行実験の映像を用いた精度評価で 89.7% の検知精度を達成した。本論文では評価結果を通じて提案技術の有用性を示す。

キーワード: レーン別渋滞, コネクティッドカー, 交通流最適化, 物体検知

Traffic Jam Detection in Each Lane and Its Evaluation in Field Trials

KOUHEI MORI^{1,a)} YUKI YOKOHATA¹ AKI HAYASHI¹ TAKAHIRO HATA¹ KAZUAKI OBANA¹

Received: April 5, 2021, Accepted: October 8, 2021

Abstract: In this paper, we propose technology to detect traffic jams in each lane on a road and describe the construction of a prototype application and its performance evaluation. This technology detects traffic jams in each lane on a road from vehicle-status data and video data collected from connected cars. It detects traffic jams by determining the number and direction of passing vehicles in the vicinity from the video data in accordance with the speed of the car. It also identifies the positions of the first and last vehicles and calculates the length of the traffic jam. We built evaluation software using this technology and used video data captured in a driving experiment conducted in Tokyo to evaluate its accuracy. The evaluation results demonstrate the effectiveness of the proposed technology. We believe that this technology will make driving assistance more valuable and contribute to optimizing the flows of vehicles in smart cities.

Keywords: traffic jam detection in each lane, connected car, traffic flow optimization, object detection

1. はじめに

近年、自動運転やナビゲーション高度化を目的とした地図データの高精度化や、交通規制や渋滞情報・信号情報といった様々なデータを重ねたダイナミックマップ [1] の構

¹ NTT スマートデータサイエンスセンター
NTT Smart Data Science Center, Minato, Tokyo 108-0023,
Japan

a) kouhei.mori.eh@hco.ntt.co.jp

築が進められている。これらの情報は、完備されれば道路交通上の状態を把握することができ、交通流を妨げる要因の特定や交通流を最適化する様々なサービスの実現が期待できるが、現実にはこれらの情報を網羅的に計測する手段の設置、その情報を処理し自動的に走行する車両の普及、それらを実際に走行させる法整備等、様々な課題を抱えている。たとえば網羅的に計測する手段としてコネクティッドカーの活用が想定されるが、2018年末時点の世界自動車保有台数14億3,318万台[3]に対するコネクティッドカーの新車販売台数は2035年時点で9,420万台[2]となっており普及には時間を要する。また官民ITS構想・ロードマップ2020[4]では、2030年の将来像においても従来型の移動手段を想定している。このように、ダイナミックマップの活用はすぐには進まない想定せざるを得ないのが現実である。

こういった将来への取り組みの一方で、これまでも従来から交通流を妨げる事象についての研究は進められている。ミリ波レーダによる交通量の検知[6]やトラフィックカウンタ[7]等、定点における通過車両の流量等を把握することができるが、これらは事前に渋滞が想定される個所にセンサを設置する必要があり網羅性に欠ける。あるいは、スマートフォン等、運転者の端末を利用し車両速度等から渋滞を検知する方法[8]が提案されている。しかしながら、これらの方法は道路リンクの混雑を可視化することはできても、走行車線レベルの分解能を備えていないため、特定の車線にのみ発生する車列を認識することが難しい。現実の交通においては商業施設の駐車場の入場待ち車列や、高速道路の降り口の車列、交差点の右左折レーンの車列等、特定の車線にのみ先頭を見通せない車列が発生することがあり、土地勘のないドライバーは自分の目的地へ到達するためにこれらの車列に並ぶ必要があるのか判断することができず、不必要にもかかわらず車列に並んでしまう場合や、本来は並ぶべき車列を通りすぎることによって目的地へ遠回りさせられる場合がある。こうした特定車線の車列（以下、レーン別渋滞と呼称）の発生や、その位置を検出することができれば、運転者の目的地に応じて並ぶべき車列と並ぶ必要のない車列を判断し、レーン別渋滞を回避しつつ目的地へ案内するレーンレベルナビゲーション等、これまでにない高度な運転支援を提供することができる。

以上の考察をふまえて、単一のコネクティッドカーが得た映像データとその車速から他の車両を追い抜いた台数やその密度を求めることでレーン別渋滞を検知する技術を提案する。さらに先頭/末尾車両を特定し、その位置をナビゲーションアプリや地図情報サービスに反映することでドライバーに適したナビゲーションの提供と交通流最適化を実現する。本論文では、2章で関連する研究やサービスについて触れ、3章で提案手法とアプリケーション実装、4章で精度評価と課題、5章でまとめについて述べる。

2. 関連研究

2.1 既存サービスと渋滞の定義

日本道路交通情報センター（JARTIC）の「道路交通情報NOW!」[5]では、一般道路では時速10km以下で走行している状態を渋滞・時速20km以下で走行している状態を混雑と定義し、渋滞長500m以上のものの情報を進行方向単位で提供している。渋滞および混雑における車両の密度、レーン別渋滞に関する定義について述べた既存サービスや文献は見つけられなかった。

2.2 設置型機器を用いた渋滞検知

各地点を通過した車両の台数カウントや速度を計測するミリ波レーダ[6]等のセンサやトラフィックカウンタ[7]等のカメラを用いた渋滞検知手法が存在する。これらは交差点や主要幹線道路上に機器を設置することでレーン単位の交通量計測や逆走等の突発事象検知が行えるが、設置された場所以外の情報は収集されず網羅性に課題がある。本論文では、コネクティッドカーにより取得した映像データを用いて全国の道路に対し網羅的に検知を行う点でこれらの手法とは異なる。

2.3 位置情報を用いた渋滞検知

車載機器やドライバーの保有するスマートフォンの位置情報変化から各地の車両速度や渋滞発生を網羅的に検知する手法が存在する。渡部らは位置情報変化から車両速度を求め一定値以下の値を記録した区間を渋滞として把握する手法[8]を提案している。さらに寺田らは携帯電話ネットワークの運用データから作成されるモバイル空間統計（リアルタイム人口統計）と天候やイベント開催等の情報から渋滞を予測する手法[9]を提案している。いずれも前述の設置型機器を用いた手法と比較して網羅性が高いが、現在普及している位置情報計測手段単体の精度では幅員3.0m~3.5m程度のレーン単位の情報粒度を得ることは難しい[10]。加えていずれの手法もGPS等の位置情報変化を計測する機器を搭載していない、またはエンジンを切った状態の位置情報を収集できない車両の影響による渋滞の検知漏れが発生する。本論文では、こうした車両の影響を受けず、また普及率が低い状況でも一部のコネクティッドカーから取得可能な情報を用いて検知を行う点でこれらの手法とは異なる。

2.4 周辺環境のセンシングによる渋滞検知

車載機器による周辺環境のセンシングは近年関心を集めており、単眼カメラやステレオカメラを用いたビジョンベースのものからレーダやLiDAR（light detection and ranging）を用いたものまで広く研究され自動運転等に活用されている[11]。機器を搭載した1台のコネクティッド

カーの走行経路上をセンシングできるため網羅性が高く、前述のようにコネクティッドカーが完全に普及していなくても活用が見込める。特にビジョンベースのセンシングはカメラが安価かつ小型化していることに加えて、GPU等ハードウェアの進化により映像内の車両の検出から追跡・行動予測までリアルタイム性を追求できるようになってきている [12]。

進藤らは車載カメラで得た映像から車両を検知しオペティカルフローを求め、観測車両のプロープ情報と組み合わせることで対向車の車速を推定することで渋滞を検知する手法 [13] を提案している。しかし著者らも述べているように片側複数車線道路において異なる車線に存在する対向車のフロー長が異なること、中央分離帯等の構造物から発生するフローとの分類に課題がある。松田らは車載カメラ映像から車両を検知し、その画像上の座標情報とその時系列変化、観測車両のスピードや加速度の情報を学習させ路上駐車を判定する手法 [14] を提案している。本手法により複数の路上駐車車両を判定することでレーン別渋滞の一例を検知することは可能と考えられる。しかしレーン別渋滞は路肩に限定されず、いずれの車線上にも存在しうるためすべては検知できない。また車線や路肩の幅員は道路によって差がある [15] ため対象車両について算出したフロー長や画像上の座標は変化する。加えて観測車両に搭載するカメラの機種や搭載位置の差異による影響も考慮する必要がある。本論文では、車線や路肩の幅員に関する道路ごとの差異や観測車両に搭載するカメラの機種や搭載位置の差異に影響されない、観測車両が他の車両を追い抜いたという事象からレーン別渋滞を検知する点でこれらの研究とは異なる。

3. 提案手法

3.1 定義

一般道路における渋滞の定義を参考に時速 10km 以下かつレーン単位の車列をレーン別渋滞と定義し、より詳細に把握できるように検討した結果、既存サービスの 1/10 以下の渋滞長である 42m 以上のレーン別渋滞を検知の対象とした。また本論文で使用する用語を定義する。

- 観測車両：レーン別渋滞の横の車線や対向車線を走行する車両。車載カメラから映像を、OBD (On-Board Diagnostics) から車速と位置情報を取得する。
- 他車両：観測車両が搭載した車載カメラで得た映像に映った観測車両以外の車両。観測車両の周囲を通過した車両、観測車両が追い抜いた車両、対向車線ですれ違った車両、敷地内車両が存在する。

3.2 処理概要

本節ではレーン別渋滞検知技術の処理概要について述べる。図 1 にレーン別渋滞に遭遇した際の道路状況イメージ

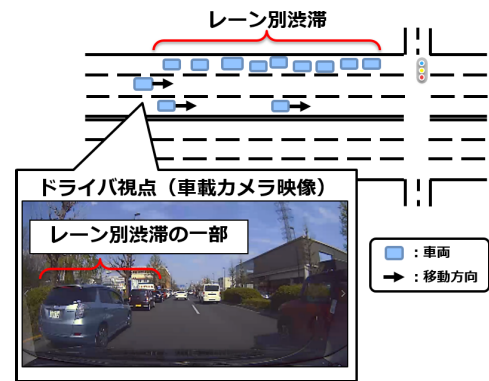


図 1 レーン別渋滞遭遇時のドライバ視点
Fig. 1 Driver's view (near the lane with traffic jam).

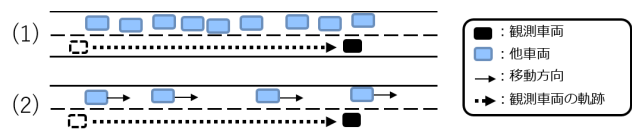


図 2 観測車両と他車両の追い抜き台数イメージ
Fig. 2 Image of overtake.

と車載カメラ映像を例示する。このとき観測車両は片側 3 車線道路の第 2 レーンを走行し、レーン別渋滞は第 1 レーンに発生している。観測車両からは車載カメラ映像と車速と位置情報を取得する。また観測車両の加減速に対応できるように、各情報は時刻同期がとれているものとする。映像が 10 fps の場合は 0.1 秒単位の同期が望ましいが、車速・位置情報が急変動することは稀と考え 1.0 秒程度の同期でも性能への影響は軽微と考え許容範囲を広げてもよい。

次に車載カメラ映像から、観測車両が追い抜いた車両と対向車線ですれ違った車両の台数を単位時間ごとにカウントし、観測車両の車速と、評価期間ごとの台数やそこから換算した他車両の密度からレーン別渋滞を検知する。レーン別渋滞を検知した場合は単位時間だけ評価期間をずらして再度評価するという処理を繰り返してレーン別渋滞が続く区間を推定し先頭車両と末尾車両を特定する。なお単位時間は複数の車両を追い抜いた/すれ違った際の間隔が読みとれないほど粗い粒度 (単位時間 60 秒に対して追い抜いた台数が 2 台等) にならないよう注意する。

レーン別渋滞の検知と区間の推定について図 2 を用いて補足する。観測車両が第 1 レーンに発生したレーン別渋滞の横を通過する場合、図 2(1) に示すように時速 10km 以下かつ車間距離の詰まった状態の他車両を追い抜くため評価期間内で追い抜いた台数は多く高密度になる。対してレーン別渋滞が発生していない場合、図 2(2) に示すように評価期間で追い抜いた台数は少なく低密度になる。したがって観測車両の速度に応じた閾値を定め、閾値以上の台数を追い抜いた場合や高密度である場合にレーン別渋滞を検知したと推定する。こうした処理を単位時間だけ評価期間をずらして繰り返し実施することでレーン別渋滞と推定

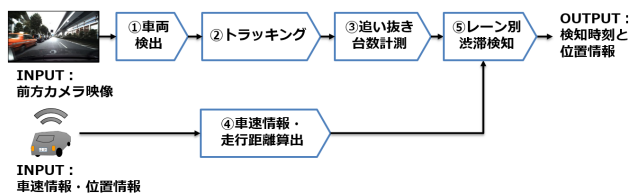


図 3 処理フロー

Fig. 3 Outline of traffic jam detection by lane.

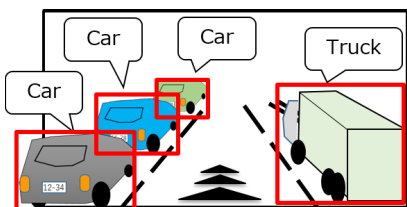


図 4 フレーム画像から車両を検知 (Car/Truck/Bus)

Fig. 4 Detect vehicles in each frame (car/truck/bus).

される区間を推定し、区間内で最初に追い抜いた他車両を末尾車両、最後に追い抜いた車両を先頭車両とする。なお観測車両はレーン別渋滞と同じ道路上の別車線を走行しているが、位置情報に差異がある。道路幅員や車線数の情報を取得し、レーン別渋滞を画像の左右どちらに検知したか、対向車線の他車両とすれ違ったものかを考慮し位置情報をずらすことで正確な区間を求めることも可能だが、本論文では簡易的に観測車両の位置情報をレーン別渋滞の区間として扱った。

3.3 アプリケーション実装

本節ではアプリケーション実装したレーン別渋滞検知技術の処理内容について述べる。処理フローを図 3 に示す。前方カメラで観測車両の前方を撮影した映像を取得し処理①～③を順に実施する。並行して車速情報・位置情報を取得し処理④を実施する。最後に③と④の出力結果を用いて処理⑤を実施する。

① 車両検知

取得した映像をフレームごとの画像に分割し、物体検知技術 YOLOv3 [16] を用いて各フレーム画像に映りこんだ他車両を検知する。学習モデルは文献 [17] に記載された URL から yolov3.weights ファイルを取得して用い、信頼スコアの閾値は 0.5 以上として、全フレーム画像に対する出力を得る。検知の結果は矩形（以降、検知矩形）で表現され、対象の車種に応じて Car/Truck/Bus の 3 種にラベル付けされる（図 4）。

② トラッキング

観測車両と他車両の相対位置は変化するため、他車両のフレーム画像間の移動を追跡しナンバリングを行う。まずは各フレーム画像で他車両に適当なナンバーを割り当てる。次に評価する現フレーム N の検出矩形 X とその f フレーム前（フレーム $N-f$ ）の検出矩形 Y の一致度を、2つの集

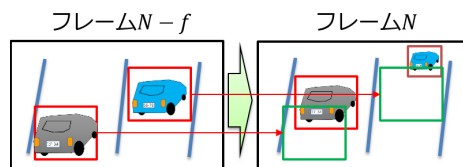


図 5 フレーム画像間の他車両トラッキング

Fig. 5 Track vehicle movement.

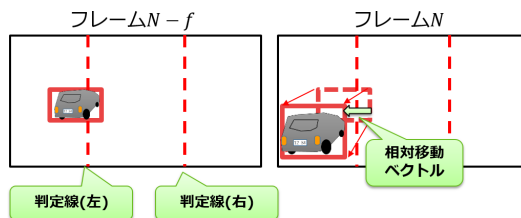


図 6 判定線による追い抜き台数のカウント

Fig. 6 Count when the vehicle crosses the decision line.

合の類似度を測る指標 Intersection over Union (IoU) [18]

$$IoU = \frac{|X \cap Y|}{|X \cup Y|}, \quad (1)$$

を用いて評価する。このとき IoU が最も大きい、つまり最も共通する画素の多い検知矩形どうしを同一の他車両によるものと見なしナンバーの再割り当てを行う（図 5）。この処理を繰り返し実施することで他車両の移動をトラッキングし軌跡を求める。なお車両検出においてラベル付けを誤る可能性を考慮し（Car を Truck と誤認する等）本処理内においてラベルはいずれも同一と見なした。またフレーム間で検出矩形が重なることを前提としているため、観測車両が高速で走行する場合は fps を上げて撮影する必要があり、逆に低速走行時は処理時間短縮のため fps を下げるか間引く処理を追加する。

③ 追い抜き台数計測

フレーム画像間で共通する判定線を設定する（図 6 左）。次にトラッキングで求めた他車両の軌跡についてベクトルで表現する。観測車両の前方を撮影した映像のため道路消失点は画像横軸中央付近にあるとし、画像左半分に映った他車両の場合は検知矩形の右端座標、右半分に映った他車両の場合は検知矩形の左端座標について、前フレーム $N-f$ から現フレーム N への軌跡を相対移動ベクトルとして算出する（図 6 右）。この相対移動ベクトルのうち道路消失点から画像外側に向けたものが判定線と交差した際に観測車両が他車両を追い抜いたとカウントする。なお対向車線の他車両とすれ違う場合との判別については後述する後処理にて対応した。

④ 車速情報、走行距離算出

取得した車速情報から観測車両の移動距離を算出する。

⑤ レーン別渋滞検知

3.2 節で述べた処理概要では、観測車両が追い抜いた車両の台数や密度と、追い抜いた際の観測車両の車速からレー

表 1 他車両の占有距離

Table 1 Vehicle sizes and distances between vehicles.

	Car	Truck	Bus
車体長	4.8m	12.0m	12.0m
車間距離	5.0m(Car1台分程度)		

ン別渋滞を検知しその区間を推定すると述べた。現状では評価期間ごとの追い抜いた車両の台数に関する閾値が明確でないことから、追い抜いた車両の密度（車両密度）を用いた。前述のとおり渋滞の定義における車両密度に関する記述は見つけられなかったが、少なくとも車間距離は通常どおり走行している場合と比較して短く詰まった状態にあるものとする。

他車両を追い抜いた台数から、隣接レーンにおいて車両が占める距離の閾値（以下、占有距離閾値 L_θ と呼称）を追い抜いた他車両 i の全長 l_i と車両 i の前方車間距離 τ_i として次のとおり求める。

$$L_\theta = \sum_i (l_i + \tau_i). \quad (2)$$

ここで、実際には様々な全長の車両が存在するが、簡単のため①で述べた学習モデルにおいて判別する3クラス Car/Truck/Bus の代表値を用いることとする。Car の全長 l_{car} は普通自動車の条件（全長 4.7m・全幅 1.7m・全高 2.0m のいずれか1つでも上回る車両）を元に、 $l_{car} = 4.8m$ とした。Truck の全長 l_{truck} は、中型・大型トラックの条件の1つ（全長 12.0m 以下）から $l_{truck} = 12.0m$ 、Bus も同じく $l_{bus} = 12.0m$ とした。車間距離 τ は、“車両が詰まった状態”ではあるが急ブレーキが間に合う距離が妥当と考え、クラスによらず $\tau = 5.0m$ とした。なお渋滞の定義の上限である時速 10km（秒速 2.778m）時の停止距離（空送距離 + 制動距離）[19] は

$$\text{空走距離} : \text{秒速 } 2.778 \text{ m} \times 0.75 \text{ 秒} = 2.08 \text{ m}$$

$$\text{制動距離} : (10 \times 10) \div (254 \times 0.7) \approx 0.56 \text{ m}$$

の合計で 2.64m となり余裕をもってブレーキが間に合うが、混雑の定義の上限である時速 20km では 6.42m となり前車両に衝突する値となる。反応時間は一般的な値とされる 0.75 秒、摩擦係数は乾いた路面を想定し 0.7 とした。

以上をまとめると、表 1 のとおりになる。このとき、追い抜いた車両のクラス (Car/Truck/Bus) ごとに x, y, z とすると、

$$L_\theta = xl_{car} + yl_{truck} + zl_{bus} + (x + y + z)\tau, \quad (3)$$

この占有距離が観測車両の走行距離 D よりも長い場合

$$L_\theta \geq D, \quad (4)$$

は車間距離が定めた閾値よりも狭い状態、つまりレーン別

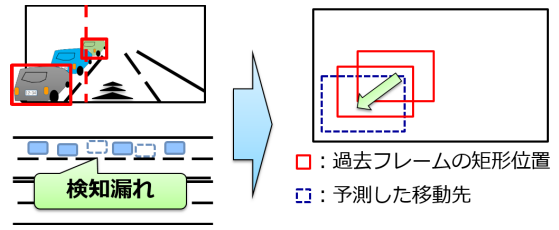


図 7 検知漏れの影響と補完処理

Fig. 7 Complementary missing detection.

渋滞の状態であると判断する。

以降は上述の処理を単位時間ごとにずらして繰り返し実施することでレーン別渋滞の区間を推定し、区間内で最初に追い抜いた他車両を末尾車両、最後に追い抜いた車両を先頭車両とする。

3.4 アプリケーション実装におけるその他工夫点

本節では 3.3 節で述べたアプリケーション実装において精度向上のため実施した工夫点について述べる。

(1) 車両検知における検知漏れフレーム画像の補完

物体検知技術を用いて他車両を検知する過程において映り込みや逆光に起因した検知漏れが発生し、トラッキング処理時に各他車両の移動軌跡が途切れる事象が発生した。この事象が追い抜き台数計測における相対移動ベクトルと判定線との交差判定時に発生した場合は追い抜き台数が少なくカウントされる (図 7 左)。

そこで各他車両の移動軌跡が過去の数フレーム画像で求めた軌跡の延長線上にあると仮定し、基本的に他車両は線形移動することからカルマンフィルタ [20] を用いて移動先を予測し補完する処理 (図 7 右) を追加した。なお補完を実施するか否かは前術の IoU を用いて共通する画素の多い検知矩形が現フレーム N にあるかどうかを評価し判断する。なお実装ではフレーム $N + 2t$ を補完する上限としたが、検知漏れが少ない物体検知技術を用いる場合は上限を下げるか処理そのものを省略する。

(2) トラッキング対象とする他車両の限定

取得した映像には敷地内車両が撮影されることがあり、追い抜いた台数としてカウントすることでレーン別渋滞と誤検知することがある。

そこでトラッキングの対象とする他車両を限定するため関心領域を設定する。映像から道路消失点を算出し、道路消失点と各フレーム画像の左右端をつなぐ境界線を設ける。境界線より外側の検知矩形が敷地内車両のものと判断し処理フローから除外する (図 8)。道路消失点は確率的ハフ変換を用いた直線検出と、検出した直線の厳選と交点算出により求めた。同様に映像内の小さい検知矩形は遠方の他車両によるものと判断し計算コストも考慮して除外した。

(3) 追い抜き台数計測の過剰カウント防止

物体検知では各フレーム画像内に映った他車両よりも広

い、または狭い検知矩形が得られることがある。この誤差により観測車両が同一の他車両を複数回追い抜いたと判定し車両密度を誤認することが想定される。他車両1台を追い抜く過程で過剰カウントする例を図9に示す。そこでトラッキングで他車両に割り当てられたナンバーを用いて、同一車両の相対移動ベクトルが判定線と連続して交差した場合に過剰カウントと見なし補正する処理を追加した。

(4) レーン別渋滞(対向車線)の識別

対向車線の他車両とすれ違う場合を判別するため、映像の右側にレーン別渋滞を検知した場合の後処理として他車両に対しフロント(車両前部)/リア(車両後部)判定処理を実施した。検知矩形より若干広い領域を画像として抽出し(図10上)、フロント/リアを検出する(図10下)。観測車両に対してフロントを向けた車両が多い場合は「対向車線の他車両とすれ違った」と判別し先頭車両と末尾車両を入れ替える処理を実施する。

フロント/リア判定処理は Vehicle-Front-Rear-Detection-

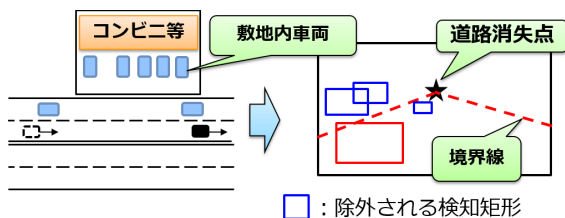


図8 処理対象とする他車両の限定

Fig. 8 Limit the number of bounding boxes.

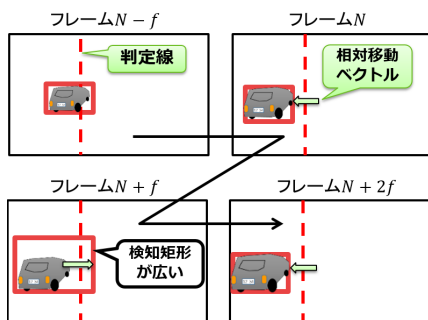


図9 過剰カウント

Fig. 9 Pattern of overcount.

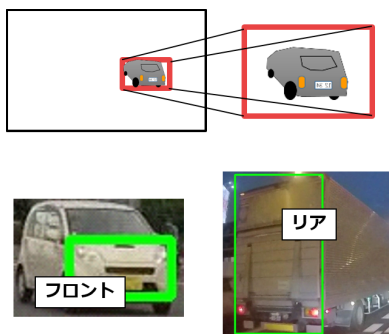


図10 フロント/リア判定処理

Fig. 10 Front/Rear detection.

for-License-Plate-Detection-Enhancement [21] をベースとした。学習モデルは文献 [21] に記載された静止状態の車両を撮影した画像 Cars Dataset proposed in 3D Object Representations for Fine-Grained Categorization とその Front/Rear アノテーションデータにデータセットを追加して学習した。信頼スコアの閾値は 0.5 以上とした。追加データセットは GoPro HERO5, Pioneer VREC-DZ500, Anker Roav Dashcam C2, CELLSTAR TZ-D001 の複数のカメラを異なる車両に搭載し、複数県の一般道路・高速道路で時間帯を問わず走行し撮影した映像を用いた。映像から Car/Truck/Bus を切り出した画像を用意し、Front 1,721 枚, Rear 1,631 枚の画像に対するアノテーションデータを作成した。

4. 精度評価と課題

4.1 データの取得条件と概要

東京都港区台場地区を 2020 年 10 月に走行した実験車両 CROWN 2.5L HYBRID RS と LEXUS LS500h を観測車両として、前方カメラ映像と位置情報を取得しレーン別渋滞検知の精度評価を実施した。精度評価の対象とするレーン別渋滞は自然発生したものとし、両車両は午前 10 時~12 時, 15 時~16 時の間に個別に走行した。走行ルートは図 11 に示す赤実線と青点線の 2 種で、事前調査により高確率でルート上にレーン別渋滞が存在すると判明している。両ルートの制限速度は時速 50km とされている。前方カメラはともに車内前方上部に設置した産業用カメラ DFK33UX265 にレンズ LM5NC1L を使用し、1,920 × 1,080 pixel の解像度で撮影して 60 秒の動画ファイルとして保存する。各車両の映像に対する①の処理では誤検出や検知漏れといった偏りはなかった。なお 2 台の観測車両は別の時間に独立して走行していることから検出したレーン別渋滞は異なり検出精度の比較はできない。

位置情報は GPS を用いて緯度・経度を 1 秒間隔で取得した。速度情報は取得していなかったため位置情報の変化から換算し代用した。また GPS の取得間隔が 1 秒である

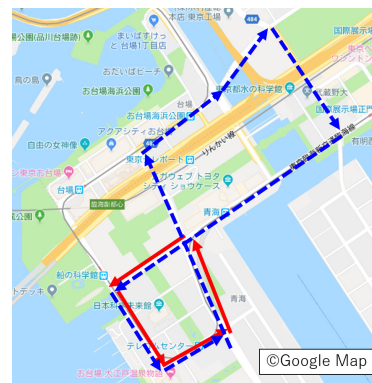


図11 走行ルート

Fig. 11 Traveling track.

表 2 レーン別渋滞の検知精度評価
Table 2 Confusion matrix of detection.

		アプリケーション出力	
		レーン別渋滞	非レーン別渋滞
目視	レーン別渋滞	TP: 70	FN: 8
	非レーン別渋滞	FP: 8	TN: 12

こと、制限速度である時速 50 km で Car 1 台を追い抜くの要する時間は車間距離を考慮して約 0.7 秒であることから複数の車両を追い抜いた間隔が読みとれないことのないよう単位時間は 1 秒とした。動画のフレームレートは、式 (1) により他車両の検知矩形についてフレーム間の類似度を算出する際に、同一他車両の検知矩形が現フレーム N とその f フレーム前 (フレーム $N-f$) で重なり共通する画素が存在するよう、観測車両のフレームあたりの移動距離が他車両の車体長以下になる値を最低条件とした。実際は車体長が 3.4 m 以下の軽自動車が存在すること、両ルート of 制限速度が 50 km であることを考慮し算出すると 5 fps が下限値となるが、本論文では余裕を持たせ倍の 10 fps、 f は 1 とした。

4.2 レーン別渋滞の検知精度評価

レーン別渋滞が存在すると目視確認結果から判断した全動画 (48 動画・各 60 秒) を抽出し精度評価を実施した。なおレーン別渋滞は定常的に路上駐車車両が集まる地点、信号手前の右左折レーンのものが確認でき、複数のレーン別渋滞が映った動画も存在した。アプリケーションの動作環境および目視確認内容と精度評価の内容を以下に示す。

■ 動作環境

下記性能のサーバ上に Docker 環境を構築し利用した。車両検知とフロント/リア判定処理、カルマンフィルタでは GPU を使用しているが並列化しておらず 1 アプリケーション 1 GPU で処理した。

CPU : Intel(R) Xeon(R) E5-2697v4 18 コア 32 スレッド
メモリ : 256 GB

GPU : GeForce GTX 1080 Ti 11178MiB 9 台

■ 目視確認内容

式 (4) に従い、42 m 以上の渋滞長のレーン別渋滞が検出対象となるよう Car 3 台 + Truck 1 台等の他車両の組合せを連続して追い抜くシーンを目視で確認し抽出した。

■ 精度評価内容

目視確認とアプリケーション出力の結果を比較した混同行列を表 2 に示す。精度 Precision (TP/(TP+FP)) は 89.7% となった。FP となった主な要因は右左折中で他車両の限定に失敗したことによる「駐車場等の敷地内に停車している他車両の影響」が 3 件 (例: 図 12) だった。FN となった主な要因は「対向車線のレーン別渋滞について、ガードレール等の陰に他車両が隠れてしまい物体検知ミス

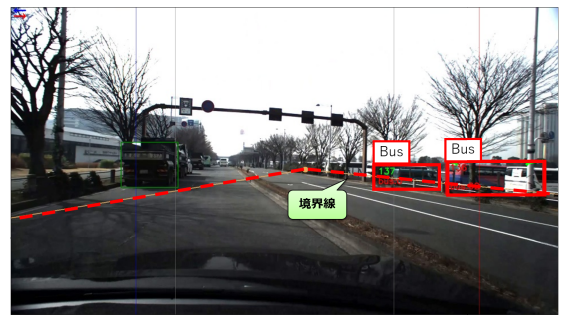


図 12 駐車場等の敷地内に停車している他車両の影響
Fig. 12 Failure to limit the number of bounding boxes.

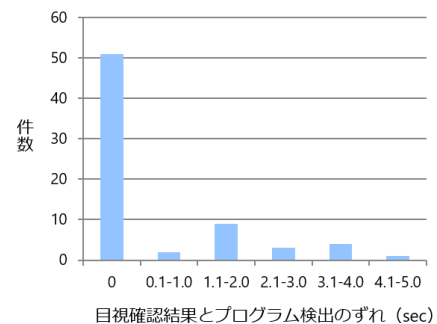


図 13 先頭車両の推定精度
Fig. 13 Error in accuracy of overtake the first vehicle.

が発生した」が 4 件、「目視確認ではレーン別渋滞と判断したが、車間距離が広く閾値を超えなかった」が 1 件だった。また FP/FN 双方に共通する要因として「物体検知漏れ」と「他車両の占有距離が固定値のため誤差が生じた」、「位置情報に誤差があり観測車両の移動距離が不正確」の影響が見られた。物体検知精度を向上させ検知漏れの割合を低減し、トレーラーやコンテナ車・小型トラックといった多種多様な車種に対応すること、移動距離を観測車両の OBD 等から速度情報を収集し用いることで改善が期待される。

4.3 先頭車両の推定精度評価

目視確認・アプリケーションの出力ともにレーン別渋滞だと判断した TP: 70 件のシーンについて先頭車両の推定精度を評価した。以下の 2 点を比較する。

- フレーム画像上に判定線を描画し、目視で先頭車両と判断した車両を追い抜いた (判定線を越えた) と目視で評価したフレーム
- アプリケーションが判断した先頭車両を追い抜いた (相対移動ベクトルが判定線と交差した) フレーム

精度評価の絶対値を取った結果を図 13 に示す。誤差の平均値が 0.57 秒、中央値が 0 秒、最大値が 4.1 秒となった。前述のとおり制限速度が時速 50 km の走行ルートでは Car 1 台を追い抜くの要する時間は車間距離を考慮して約 0.7 秒で、この値を許容範囲とすると平均値 0.57 秒は準拠した値と言える。誤差が前後 1.6 秒以上となった 14 件について

処理	処理時間
車両検知	51.555秒
検出漏れフレーム画像の補完	26.431秒
フロント/リア推定処理	10.740秒
レーン別渋滞判定等, その他処理	9.250秒
総処理時間	98.350秒

図 14 処理時間サンプル

Fig. 14 Sample of processing time.

て調査したところ「レーン別渋滞とは異なるレーンを走行していた他車両と混同した」が2件, 「位置情報に誤差があり, 観測車両の走行距離を誤認し処理を継続/中断」が9件, 「物体検知時の Car と Truck の誤認」が3件だった。

4.4 処理時間

アプリケーションの処理時間例を図 14 に示す。なお映像に映った車両の台数に応じて各処理時間が増減すること, 映像の右側にレーン別渋滞を検知した場合のみフロント/リア推定処理が行われることから, この条件を満たす動画10個の平均処理時間を計測した。平均 98.350 秒となっており, 4.2 節で述べたサーバを用いてもリアルタイムでは処理できていない。

既存技術を用いた車両検知とフロント/リア推定が平均 62.295 秒を占めており, 用いる映像の画質を下げる・より高速な技術を用いる等の改善案が考えられる。物体検知漏れフレーム画像の補完は車両検知の性能がより高い技術を用いることで処理時間の削減, あるいは処理の省略が可能になる。また物体検知結果のトラッキングが高精度に行えていれば映像が 10fps である必要はなく, 観測車両の車速に応じて fps を可変化する, 低 fps でもトラッキング可能なナンバープレート認識による処理に変更する等の改良により全体の処理時間削減が見込める。

4.5 結果考察と今後の課題について

アプリケーション実装とその精度評価結果に関する考察と今後の課題について示す。

- ◆ 単眼カメラを用いたビジョンベースのセンシングかつ, 精度評価の環境に合わせた追加学習等の調整を実施していないため, 日中帯かつ良天候で撮影した映像にもかかわらず物体検知漏れやラベル誤認が確認された。夜間や悪天候時の検知性能はさらなる低下が想定される。
- 活用する物体検知技術の見直しに限らず, 同じビジョンベースでも夜間や悪天候へも対応可能な HDR (High Dynamic Range) カメラや FIR (Far Infrared Rays) カメラの活用も検討する。
- ◆ 物体検知のラベルを Car/Truck/Bus の3種として扱い車名によらず全長の値と車間距離を一律にしており, 「他車両の占有距離が固定値のため誤差が生じた」

につながった。

- より細かなラベル付けが行えるよう学習モデルを作成するという解決策も想定されるが, 物体検知の結果に対し他車両の全長推定を実施することで占有距離のより正確な算出を可能にしたい。また車間距離も渋滞時の値を調査し実態に沿った値を用いることを検討する。
- ◆ 目視確認によりレーン別渋滞が存在しない判断した動画において, 対向車線を順調に走行する車両とのすれ違いをレーン別渋滞と誤検知したケースが確認された。
- 一定以上の相対速度で対向車線とすれ違う場合にもレーン別渋滞検知の判定式を満たすことから, トラッキングした車両の映像内における滞在時間によって補正する後処理の追加を検討する。

5. おわりに

単一のコネクティッドカーから収集可能な映像データと車速や位置情報からレーン別渋滞を検知し先頭/末尾車両を特定するレーン別渋滞検知技術を提案した。アプリケーションを実装し, 実験車両から得た前方カメラ映像と位置情報からレーン別渋滞検知の検知精度と先頭車両の推定精度を評価した。

今後の課題は, 物体検知漏れやラベル誤認の影響低減や他車両の全長推定によるより正確な占有距離の算出, 対向車線のレーン別渋滞誤認ケースへの対処である。またドライバーが自身の目的地に到達するためにレーン別渋滞を追い抜いてよいか最後尾に追従すべきか判断できるよう, 映像や位置情報を活用して要因を特定する手法も並行して検討する。

謝辞 本研究を進めるにあたり, 精度評価にご協力くださったトヨタ自動車株式会社の高橋克徳様, 澤井洋平様に深く感謝いたします。

参考文献

- [1] 小山 浩, 柴田泰秀: 自動走行におけるダイナミックマップ整備, システム制御情報学会解説記事, Vol.60, No.11, pp.463-468 (2016).
- [2] 株式会社富士経済 コネクテッドカー (つながる車) の世界市場を調査, 入手先 (<https://www.fuji-keizai.co.jp/file.html?dir=press&file=20061.pdf&nocache>) (参照 2021-07-19).
- [3] 一般社団法人 日本自動車工業会, 入手先 (<https://www.jama.or.jp/world/world/index.html>).
- [4] 官民 ITS 構想・ロードマップ 2020, 入手先 (https://www.kantei.go.jp/jp/singi/it2/kettei/pdf/20200715/2020_roadmap.pdf) (参照 2021-07-28).
- [5] JARTIC 道路交通情報 Now!!, 入手先 (<https://www.jartic.or.jp/s/attention/inquiry/#map.8>) (参照 2021-07-19).
- [6] ミリ波レーダーで交通量計測や突発事象を検知《停止・低速・渋滞・避走・逆走》, 入手先 (<https://www.c-nexco-htech.jp/detail/0048.php>) (参照 2021-03-11).
- [7] 日立画像トラフィックカウンタ「TRACOLRE」, 入手先 (<https://info.hitachi-ics.co.jp/product/gazoutraffics/>) (参照 2021-03-11).

- [8] 渡部大輔, 金澤文彦, 澤 純平: スマートフォンを活用したプローブ情報収集システムの開発, 国土交通省技術研究会論文集, pp.221–226 (2013).
- [9] 寺田雅之, 赤塚裕人, 永田智大, 仲西哲志: 東京湾アクアラインの渋滞を「AI 渋滞予知」で回避する—携帯電話ネットワークの運用データの統計処理に基づく未来予測の試み, NTT ドコモテクニカル・ジャーナル, Vol.27, No.2 (2019).
- [10] 小島祥子, 高橋 新, 二宮芳樹: 汎用 GPS と車載レーザーレーダを用いた高精度自車両位置推定, 情報処理学会論文誌, Vol.50, No.1, pp.64–74 (2009).
- [11] Marti, E., de Miguel, M.A., Garcia, F. and Perez, J.: A Review of Sensor Technologies for Perception in Automated Driving, *IEEE Intelligent Transportation Systems Magazine*, Vol.11, No.4, pp.94–108, DOI: 10.1109/MITS.2019.2907630 (2019).
- [12] Sivaraman, S. and Trivedi, M.M.: Looking at Vehicles on the Road: A Survey of Vision-Based Vehicle Detection, Tracking, and Behavior Analysis, *IEEE Trans. Intelligent Transportation Systems*, Vol.14, No.4, pp.1773–1795, DOI: 10.1109/TITS.2013.2266661 (2013).
- [13] 進藤 瞭, 白石 陽: 渋滞検知のためのオペティカルフローによる対向車速推定手法, 情報処理学会研究報告, 高度交通システムとスマートコミュニティ (ITS), Vol.2014-ITS-59, No.16, pp.1–9 (2014).
- [14] 松田明大ほか: ドライブレコーダ動画を用いた路上駐車判定手法の提案, マルチメディア, 分散協調とモバイルシンポジウム 2179 論文集, pp.1102–1110 (2020).
- [15] 国土交通省 車線幅員, 入手先 (https://www.mlit.go.jp/sogoseisaku/inter/keizai/gijyutu/pdf/road_design.j1_05_2.pdf) (参照 2021-07-19).
- [16] GitHub: darknet, available from (<https://github.com/pjreddie/darknet>) (accessed 2021-03-11).
- [17] YOLO, available from (<https://pjreddie.com/darknet/yolo/>) (accessed 2021-07-16).
- [18] 藤田広志, 上杉正人, 平原大助, 齋藤静司: 医療 AI とディープラーニングシリーズ Python による医用画像処理入門, オーム社 (2020).
- [19] 茄子川捷久, 宮下義孝, 汐川満則: 自動車の走行性能と試験法, 東京電機大学出版局, pp.46–48 (2008).
- [20] 松阪喜幸: デジタル画像処理 [改定新版], 公益財団法人画像情報教育振興協会 (CG-ARTS 協会) (2015).
- [21] GitHub: Vehicle-Front-Rear-Detection-for-License-Plate-Detection-Enhancement, available from (<https://github.com/hankerkuo/Vehicle-Front-Rear-Detection-for-License-Plate-Detection-Enhancement>) (accessed 2021-03-11).



横畑 夕貴

日本電信電話株式会社スマートデータサイエンスセンタ第二ドメインユニット主任研究員.



林 亜紀

日本電信電話株式会社スマートデータサイエンスセンタ第二ドメインユニット研究員.



秦 崇洋

日本電信電話株式会社スマートデータサイエンスセンタ第二ドメインユニット主任研究員.



尾花 和昭

日本電信電話株式会社スマートデータサイエンスセンタ第二ドメインユニット主幹研究員.



森 皓平 (正会員)

日本電信電話株式会社スマートデータサイエンスセンタ第二ドメインユニット研究主任.