

カメラから抽出した車両特徴量に対する 選択的利用方式の評価

大鐘 崇史^{1,a)} 渡辺 陽介² 高田 広章^{1,2}

概要：詳細な地図の上に標識や信号情報、車両や歩行者の動的情報を載せたダイナミックマップが提案されており、その活用場面の一つとして、自動運転車と人運転車が混在する状況が挙げられている。人間のドライバーが運転中であってもダイナミックマップからの情報を有効に活用するために、道路上にあるカメラから得られる視覚的な情報を元にして、ドライバーに的確に伝わる言語情報に変換した上で音声案内をするシステムを構築した。

提案システムでは、複数の車両が認識され、それらが同じ色や特徴をもつ車両の場合、それ以外の特徴を最小限の言語情報で伝える必要がある。そこで、カメラから得られる画像データの車両特徴量の画像認識を行い、ドライバーがどの車両かを特定でき、なおかつ指示を伝えるための必要十分な車両特徴量を選択するアルゴリズムを開発した。そして、実車両に対して画像認識を行い、人運転車両に対して適切な言語情報で音声案内を行う実験と評価を行った。

キーワード：画像認識、物体検出、カメラ、合流調停、音声案内、自動運転と人運転の混在、車両特徴量、ダイナミックマップ

An Evaluation of the Selection Method for Vehicle Characteristics Obtained from Cameras

1. はじめに

近年、自動運転や安全運転支援の研究が進んでおり、カメラやセンサーなどの情報から車両周辺の他車両、歩行者、交通標識などを認識した情報を通信で共有して、協調型運転支援や自動運転制御を行う研究が行われている [9][10]。

この協調型運転支援の1つとして、合流調停が挙げられる [2]。合流調停とは、ある交差点において複数台の車両が存在する時に、どの車を先に進行させるか、さらにその通過速度や走行車線などを制御するものである。将来、自動運転が社会で一般的に使われるようになった際に、自動運転車両と人運転車両は混在していくと予想され、混在交通環境下では、人運転の車両に対しても合流調停の情報が提供される。

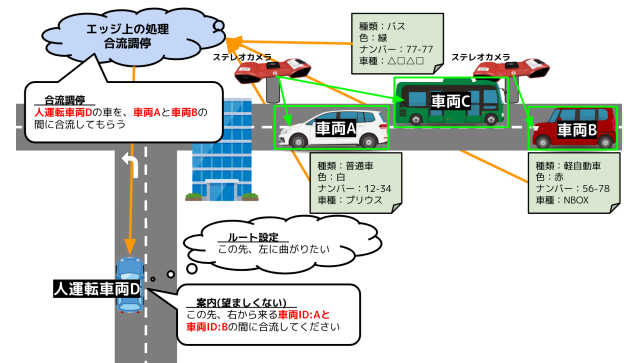


図 1 自動運転車両と人運転車両の混在状況における合流調停

ここで、図 1 のように、交差点において合流調停を行う場合を考える。(本研究では、都市レベルでの車両制御を行うクラウドサーバーではなく、交差点等の局所的な車両制御を行うエッジサーバーでの合流調停を想定している。) 幹線道路に自動運転車両が走っていて、支線道路から人運転車両が幹線道路に合流する。適切な合流調停のアルゴリズムが使われれば、合流する順序等を定めることはできる

¹ 名古屋大学大学院情報学研究科
Graduate School of Informatics, Nagoya University
² 名古屋大学未来社会創造機構
Institutes of Innovation for Future Society, Nagoya University
^{a)} ganechi@ertl.jp

が、人運転車両は幹線道路を走行している車両のどこに合流すべきかを運転中のドライバーへ伝達する必要がある。そこで、幹線道路に複数台設置されているカメラから幹線道路を走行している複数の車両を物体認識し視覚的特徴を抽出する。エッジサーバ上で、車両の情報から合流調停を行い、音声案内で人運転車両に合流調停の指示を伝える。

この指示においては、以下の4つのポイントがある。

- (1) 「どの車の後ろについて交差点に進入すべきか」など、進むべき道を他の車を目印にして伝える
- (2) 目印となる車を識別するために、車体の色や車種などの車両の視覚的情報を用いて伝える
- (3) 指示を画面上に表示すると運転の際に危険なため、音声案内が望ましい
- (4) 指示の目印となる対象車両を簡潔でわかりやすく伝えるのに最適な視覚的特徴のタグの組み合わせを選択する

また、同じ色や特徴をもつ複数の車両が連なって合流地点へ近づいていた場合、重複していない特徴の中から最小限の情報で伝える必要がある。例えば、2台の赤い車が幹線道路を走行している時に、どちらかの車両の後ろに合流するように伝えたいとすると、「赤い車の後ろに」という案内ではどちらの車を目印にするのか人運転車両から判断ができない。そこで、車体の色だけではなく「赤いバスの後ろに」といった案内や他に同じ車種がいなければ「バスの後ろに」という案内だけで車両を特定できるようにする。

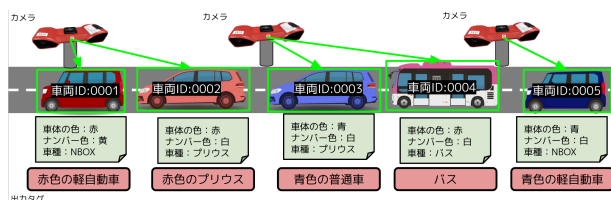


図2 車両特徴量の選択的利用

本研究では、カメラを用いて道路上の車の画像の物体認識を行い、車両の視覚的特徴を抽出することを目的としている。これまでに、ドライバーへ合流調停の指示を出す目印となる車両を識別するための、必要な視覚的特徴を選択的に利用するシステムを開発してきた[1]が、実環境での評価実験は行われていなかった。本論文では、実際に車両とカメラを用意し、レーン側PCとエッジ側PCで車両特徴量の抽出と選択的利用の処理を行って音声案内の出力までを行い、その評価を行った。

2. 関連研究

近年、ニューラルネットワークを用いた物体検出技術が進化しており、この技術を応用した車両特徴量の抽出をする研究が行われている。Raysonら[3]は、本研究で用いている物体検出ツールのYOLO[8]を用いて、自動ナンバー

プレート認識を行った。自動ナンバープレート認識では、ナンバープレート認識、文字の分割、文字の認識の3段階で行われており、より現実的な場面での認識をリアルタイムに処理するためにYOLOが使われた。しかし、この研究ではナンバープレートを認識してその文字情報を出力するのみを目的としているため、本研究と目的が異なっている。また、田中ら[4]は、ニューラルネットワークを用いた自動車の3次元モデルによってセダン・ミニバン・SUVといった自動車の車種の特徴量の抽出・分類を行っている。車両を360度でモデリングし、そのデータをもとに学習、者型認識、特徴部位の可視化を行った。この研究では車種の特徴量抽出という部分を目的としているため、本研究の目的の中の一部にすぎないため本研究とは目的が異なっている。また、岩崎ら[5]は、YOLOv2を用いた交差点内の交通量と車両走行軌跡の自動計測を行うアルゴリズムの構築と実験を行った。限定的な状況ではあるが、交通量と車両走行軌跡の自動計測を行えることを確認している。また、フレームごとで同一車両を認識して追跡を行っている。あくまで同一車両を追跡して交通量を自動計測することを目的としているため本研究とは目的が異なっている。

将来自動運転が実用化する中で、自動運転車両と人運転車両が混在する環境が考えられる。このような混在環境における運転支援に関する研究が行われている。宮崎ら[6]は、混在環境でT字路における各車の譲り合いを円滑にすることを考えた。交通シミュレーションによって自動運転車両の占有率が高まるほど渋滞長や待ち時間が低減することが判明しており、路車間通信だけでなく自動運転車両による車車間通信や自動運転車両による車車間距離の調節によって交通流を円滑にできると提案されている。この研究では、通信装置を設置していない人運転車両の認識のため、交差点に通信基地局を設置していることを想定しており、本研究と共通する部分がある。ただ、この研究では人運転車両への伝達方法が課題とされており、本研究がその解決策の1つとなると考えられる。矢島ら[7]は、自動運転車両と人運転車両による譲り合いを9つのパターンに分類し、それぞれのパターンで譲り合い状況導出のフローチャートが提案されている。この譲り合いは車車間通信による解決が考えられており、識別IDや譲り合いパターン番号などをメッセージで送信し譲り合う。車体の識別には、ID、メーカー、車種、車体カラーの4つの情報を全て伝達しており、全く同じメーカー、車種、車体カラーの車が並んでいる場合には譲り合いを中止するという判断がとられている。この研究では、車両特徴量の選択的利用という面で、メーカー・車種・車体カラーに注目して分類を行っているという点が共通している。しかし、車車間通信を想定していて本研究の路車間通信とはアプローチが異なる点、車両特徴量を選んで伝えるということを行っていない点で本研究とは異なる。

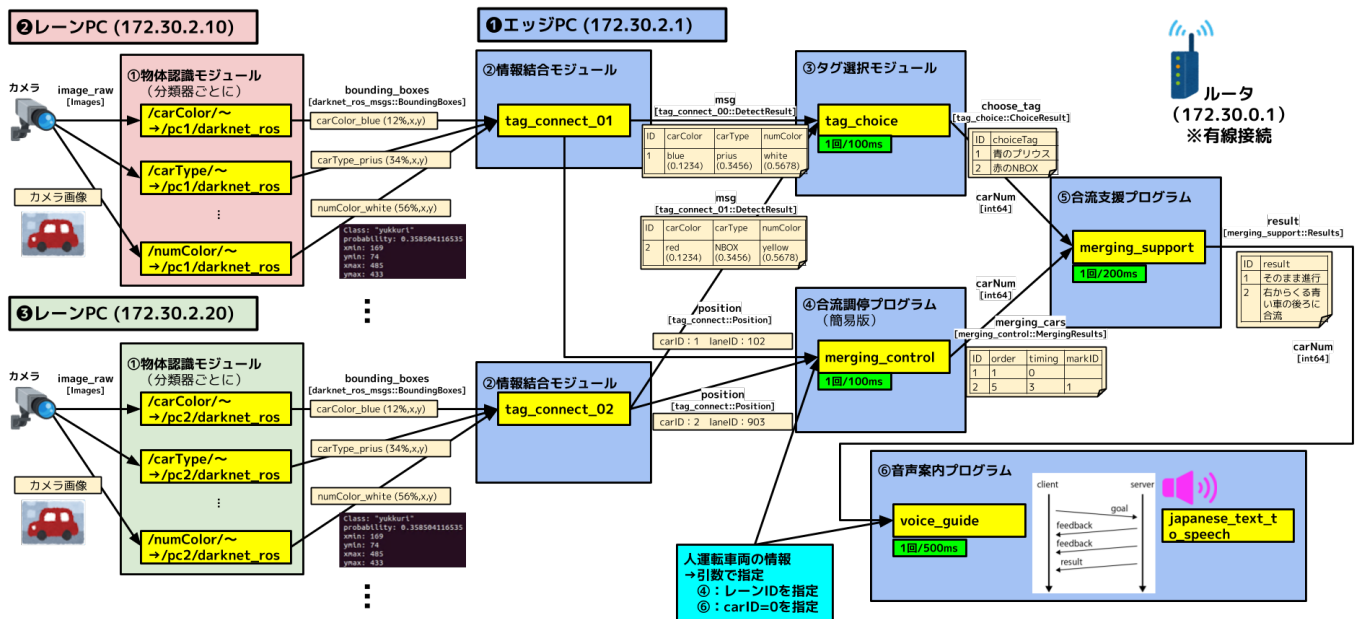


図 3 システムの全体図

3. 車両特徴量の抽出と選択的利用システム

本章では、カメラを用いて道路上の車両の視覚的特徴を抽出し、人運転の車両に最適な案内をする提案システム [1] について概要を紹介する。提案システムの全体像を、図 4 に示す。また、各モジュールをつなげたシステム全体図を図 3 に示す。

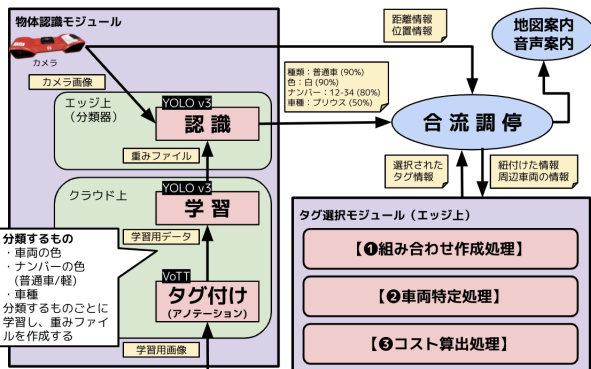


図 4 システムの全体像とアプローチ

3.1 ROS について

本研究では、システム全体を ROS を用いてシステムを構築した。ROS[11] とは、Robot Operating System の略で、Publish/Subscribe 型の通信が行える。各モジュール (ノード) から Topic にデータを配信 (Publish) し、他のモジュールは自由に Topic から情報を購読 (Subscribe) することができる。

本研究のシステム全体を ROS を用いて実装し物体検出や車両特徴量のタグの管理などを行っている。

3.2 カメラを用いた車両特徴量の抽出

車両の視覚的特徴を、カメラから得られた画像を物体検出し分類するモジュールの実装について記述する。

3.2.1 YOLO v3 について

YOLO[8] は、2016 年に Remon らによって提案された物体検出ツールである。人間が画像を見た時に、どこにどの物体があるのかを正確に把握できるように、計算機上で物体検出を行えるものである。YOLO v3[12] は、YOLO の 3 代目のプログラムであり、精度が大きく改善している。YOLO v3 は、他の物体検出のシステムに比べて高速でありリアルタイム性が高く誤検出が少ないなどの利点がある。また、YOLO v3 は自前の学習データを入力することで、学習を行い分類器を作成することができる。

今回の研究では、リアルタイム性と自前のデータを用いることができる点を考慮し、YOLO v3 を用いて実装を行った。

3.2.2 データの収集、学習、認識

データ収集においては、評価実験に用いる車両をカメラで撮影し、画像データを収集した。その後、VoTT[13] というタグ付けツールを用いて 1 枚ずつ画像のタグ付け (画像内の物体の範囲と分類を入力) を行った。最後に、そのデータを用いて YOLO v3 で学習を行い、認識に用いる重みファイルを得た。

今回、学習した分類器は表 1 の 3 種類である。

表 1 今回作成した分類器

分類器	分類する視覚的特徴のタグ		
車体の色	白色	黒色	赤色
車種	アクア	プレミオ	ピクシススペース
ナンバー色	白ナンバー	黄ナンバー	

3.3 車両特徴量の選択的利用

分類器から抽出した複数車両の車両特徴量のタグを人運転車両のドライバーに伝達する際に、最適なタグの組み合わせを出力するアルゴリズムについて説明する。今回は、抽出した車体の色・車種・ナンバーの色のタグ情報とそれぞれの信頼度の情報を利用した。

3.3.1 タグ選択におけるポイント

合流調停などのシステムから別の車両を目印にして案内をする際、物体認識によって得られたその車両の視覚的特徴をそのまま伝えることは、ドライバーにとっての負担になり、なおかつ伝達時間がかかってしまう。そこで、抽出した車両特徴量の中から厳選して必要な情報だけをドライバーに伝えるアルゴリズムとなっている。

本研究では、厳選した必要な情報の定義を以下のように定めた

- 目印となる車両を一意に特定できる
- 伝達する情報量（伝達に必要な文字数が少ない）
- 情報の信頼度が高い

3.3.2 タグ選択アルゴリズム

タグ選択アルゴリズムでは、図5の3つの処理が行われている。

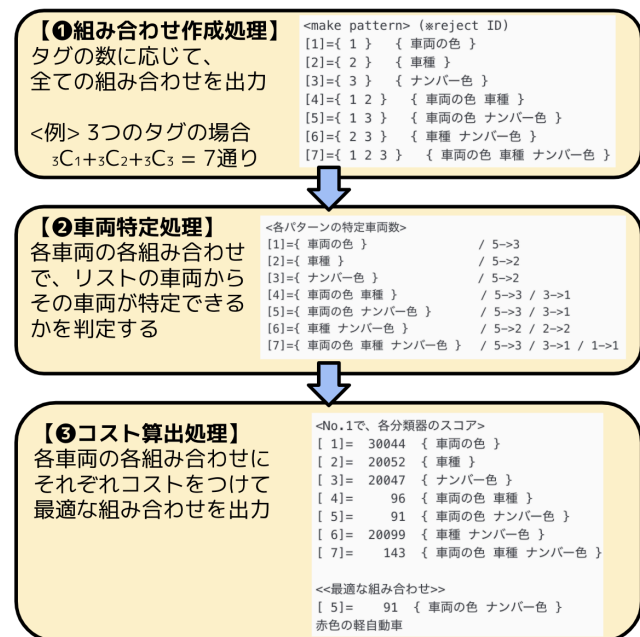


図5 タグ選択アルゴリズムの処理の流れ

組み合わせ作成処理では、車両特徴量のタグの全ての組み合わせに対して探索を行う。現時点の実装ではタグの種類が限られていることから、分類器の全ての組み合わせを作成する。ここでは、分類器の順番を考慮せずにタグの組み合わせを扱う。

例えば車体の色、車種、ナンバーの全ての組み合わせは以下の通りである。

- (1) 車体の色
- (2) 車種
- (3) ナンバー色
- (4) 車体の色・車種
- (5) 車体の色・ナンバー色
- (6) 車種・ナンバー色
- (7) 車体の色・車種・ナンバー色

次に、車両特定処理では、組み合わせ作成処理で出力されたそれぞれのタグの組み合わせにおいて、複数車両の中から1台を特定できるかを計算する。1台ずつ順番にその車両がそれぞれのタグの組み合わせで特定できるかどうかを算出する。各組み合わせにおいて、タグの組み合わせが同じ車両がある場合には、その車両は特定できないということになる。

最後に、コスト算出処理では、最適なタグの組み合わせを出力するために、それぞれのタグの組み合わせに対してコストを算出する。コストの算出においては、特定度、情報量、信頼度の3つの点数の合計で算出している。このコストが最小であるタグの組み合わせが、最適なタグの組み合わせとなる。

特定度 1台に特定できる場合はコスト0、特定できなかった場合、台数に応じてコストを高く設定する。

情報量 タグの内容（青色の、軽自動車など）の文字数をそのまま伝達に必要な情報量としてそのままコストと設定する。複数のタグを使用した組み合わせについては、その文字数を合計してコストとする。コストが少ない方が、少ない文字数で伝達できる組み合わせとなる。

信頼度 物体認識の際に各タグに付けられる信頼度を用いる。信頼度が高いほどコストが低くなるように設定する。複数のタグを使用した組み合わせについては、それぞれのタグの信頼度の指数関数の値を合計したものを信頼度のコストとする。

上記の特定度・情報量・信頼度の3つの点から、各車両の各組み合わせにおけるコストを、以下のように定義づけた。

$$f_{cost} = Ax + \sum_{\text{組み合わせに含まれるタグ}} By + \sum_{\text{組み合わせに含まれるタグ}} e^{-Cz}$$

特定できた台数を n 台において、 x, y, z は以下のように定義する。

$$x = \begin{cases} 0 & (n = 1) \\ n & (otherwise) \end{cases}$$

$$y = (\text{各タグの文字数})$$

$$z = (\text{各タグの信頼度})$$

実験では、 A, B, C の変数を以下のようにおいた。

$$A = 10000, B = 1, C = 1$$

4. 実世界での評価

実世界の車両をカメラで撮影し、リアルタイムに提案プログラムの処理を行い、評価を行った。

4.1 評価実験内容

大学構内で複数台の車両を異なる組み合わせで走行させ、レーンごとに設置されたカメラと PC によって物体検出の処理を行い、エッジ PC で簡易的な合流調停とタグ選択、音声案内の処理を行った。

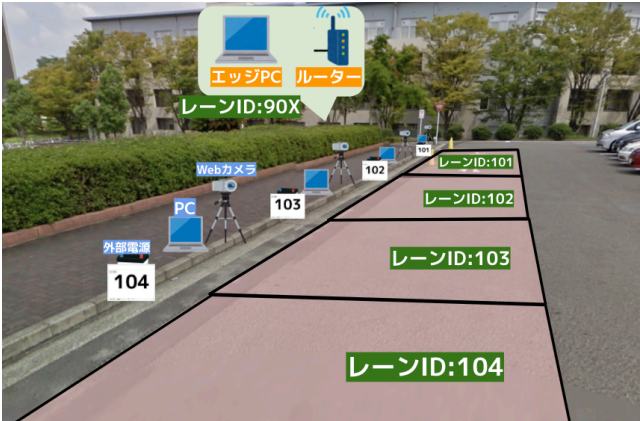


図 6 実験場所

4.1.1 実験機材

実験に使用した車両は、表 3 の通りである。

表 2 今回使用した車両

	車体の色	車種	ナンバー色
1	白	アクア	白ナンバー
2	黒	アクア	白ナンバー
3	赤	プレミオ	白ナンバー
4	白	ピクシススペース	黄ナンバー

実験に使用した機材は、以下の通りである。また、各 PC で動作させたモジュールは図 4 に示す通りである。

- エッジ PC 1 台
- レーン PC (GPU 付き) 4 台
- Web カメラ, 三脚 4 台
- ポータブル電源 4 台

4.1.2 実験手順

本線道路に車 1 台分をレーンとして、レーン ID を割り振った (図 6)。そのレーンごとに Web カメラとレーン PC、電源を設置した。レーン PC では、カメラの画像を取得して、物体認識モジュールを動かし、ROS の Topic レーン情報と車両特徴量のタグを配信する。

さらに、タグ選択から音声案内の処理を行うエッジ PC を用意した。エッジ PC とレーン PC は LAN ケーブルで接続し、各レーンからの情報をまとめて処理を行い、図 7

のように音声案内を行った。

そして、図 8 のように 3 台の車両を組み合わせて走行した場合を抽出して、評価を行った。車両の組み合わせは、表 3 の通りである。

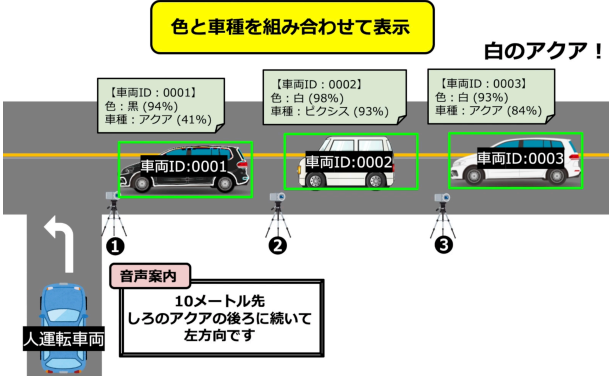


図 7 音声案内の例



図 8 実験風景

表 3 実験車両の組み合わせ

No.	1 台目	2 台目	3 台目
1.	赤プレミオ	ピクシス	白アクア
2.	赤プレミオ	黒アクア	白アクア
3.	赤プレミオ	黒アクア	白アクア
4.	ピクシス	白アクア	赤プレミオ
5.	白アクア	赤プレミオ	ピクシス
6.	白アクア	黒アクア	ピクシス
7.	黒アクア	白アクア	ピクシス
8.	黒アクア	白アクア	ピクシス
9.	黒アクア	白アクア	赤プレミオ
10.	ピクシス	黒アクア	白アクア
11.	ピクシス	黒アクア	白アクア
12.	黒アクア	ピクシス	白アクア
13.	黒アクア	ピクシス	白アクア
14.	ピクシス	白アクア	黒アクア
15.	ピクシス	白アクア	黒アクア

表 4 音声案内出力結果

No.	音声案内出力結果	判定
1	10m 先, アクア の後ろに続いて, 左方向です	○
2	10m 先, の後ろに続いて, 左方向です	× (a)
3	10m 先, の後ろに続いて, 左方向です	× (b)
4	10m 先, あかの車の後ろに続いて, 左方向です	○
5	10m 先, 軽自動車 の後ろに続いて, 左方向です	○
6	10m 先, 軽自動車 の後ろに続いて, 左方向です	× (c)
7	10m 先, 軽自動車 の後ろに続いて, 左方向です	△ (d)
8	10m 先, 軽自動車 の後ろに続いて, 左方向です	○
9	10m 先, あかの車 の後ろに続いて, 左方向です	○
10	10m 先, しろのアクア の後ろに続いて, 左方向です	○
11	10m 先, しろのアクア の後ろに続いて, 左方向です	○
12	10m 先, しろのアクア の後ろに続いて, 左方向です	× (e)
13	10m 先, しろのアクア の後ろに続いて, 左方向です	○
14	10m 先, くの車 の後ろに続いて, 左方向です	△ (f)
15	10m 先, くの車 の後ろに続いて, 左方向です	○

※判定の列において, ○は想定通りの音声案内ができたもの, △は最初異なる認識をしたがその後正しい認識をして想定通りの音声案内をしたもの, ×は想定とは異なる音声案内をしたものである。

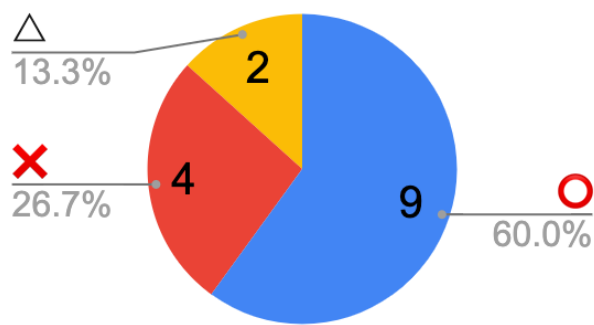


図 9 実験結果

4.2 実験結果

4.2.1 音声案内出力結果

音声案内出力結果は, 表 4, 図 9 の通りである。60 % の場合において, 想定通りの結果を出力していた。

また, 想定通りの結果となっていない場所について原因分析を行い, その結果を表 5, 図 10 に示す。

表 5 ミスとなった原因分析

ミス	ミスの原因分析	ミスの分類
× (a)	黒色を白色と誤認識	色誤認識
× (b)	ピクシスをアクアと誤認識	車種誤認識
× (c)	黒色を赤色と誤認識	色誤認識
△ (d)	最初に色誤認識, その後正しい認識	色誤認識
× (e)	カメラの向きによって, 誤認識	色誤認識
△ (f)	最初に色誤認識, その後正しい認識	色誤認識

4.2.2 実行時間結果

車両特徴量の抽出の部分とタグの選択的利用の部分について, 実行時間の評価を行った。評価対象は, 前節の実験結果のうち結果が成功しているものを対象としている。

車種誤認識
16.7%

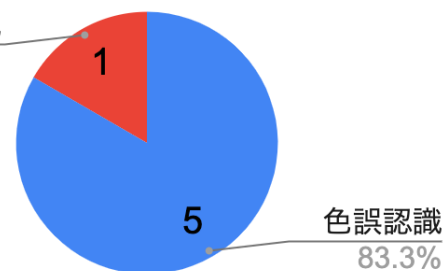


図 10 ミスの分類

実験を記録したデータを rqtbag というツールを用いて分析し, 各部分の実行時間を評価した。各部分での実行時間を表 6 に示す。車両特徴量の抽出の部分の実行時間 (表 6 の「物認 T」列) は, どれか 1 つの分類器が認識を始めてから車両特徴量の情報が揃って配信するまでの経過時間である。タグの選択的利用の部分の実行時間 (表 6 の「タグ T」列) は, 車両特徴量の情報が揃って配信されてから, タグ選択を行い音声案内を出力するまでの経過時間である。

車両特徴量の抽出における平均実行時間は, 5.70 秒, タグの選択的利用における平均実行時間は, 0.29 秒であった。

表 6 各モジュールの実行時間 (結果成功時 単位:秒)

ID	物認 T	タグ T	物認 T	タグ T	物認 T	タグ T
1	4.774	0.136	10.067	0.203	4.963	0.203
4	5.316	0.25	6.435	0.349	5.374	0.349
5	4.73	0.253	6.325	0.253	4.882	0.352
8	5.709	0.303	8.088	0.303	4.351	0.204
9	3.61	0.4	7.194	0.3	5.248	0.4
10	3.393	0.322	9.298	0.222	6.233	0.322
11	4.797	0.331	4.289	0.231	6.937	0.331
13	4.365	0.25	4.657	0.35	4.814	0.249
15	4.39	0.354	5.775	0.554	7.838	0.054

4.3 考察

4.3.1 車両特徴量の抽出についての評価

4.2.1 節の結果からもわかる通り, 物体認識モジュールは, 6 割の場合において全ての分類器が正しい認識をしていた。

しかし, 日光の当たり具合によって, 黒色の車が白色と誤判定とされた場合があった。カメラでは, 反射した色で認識してしまうが, 人間の目にとってもその車が黒色であると判断しにくいと考えられる。この場合, 車体の色の信頼度が小さくなると考えられるので, 他の情報を用いて伝達を行うことができると考えられる。

また, 現時点の実装では, レーンが空であることを判定する手段がなく, 手動で車両がレーンにいるかどうかの判定を行っていた。これは, YOLOv3 が物体を検知した段階で Topic を配信しているので, 何も物体検出がないこと

を Topic として配信したり、他のレーンと連携することによって、同一車両を連続的に管理する仕組みが必要であると考えられる。

4.3.2 タグ選択アルゴリズムについての評価

4.2.1 節の結果からもわかる通り、検出した車両の組み合わせによって、タグが変化して最適なタグを用いて案内をしていることが確認できた。

今回の実装では案内の文字数を減らすことを重視した重み付けとなっていたため、車種がわかりにくい場合でも色より車種の情報を伝達してしまうといったことが発生した。ドライバーが識別しやすい色などの情報を優先的に選ぶようなアルゴリズムが必要であると考えられる。

4.3.3 提案システム全体の評価

4.2.1 節の結果からもわかる通り、物体認識モジュールとタグ選択モジュールを結合できており、音声案内まで6割以上の場合で想定通りの結果を出力することができていた。

音声案内が失敗した事例の中には、認識途中で案内をしてしまう場合があった。これは、タグの最適な選択が終わらないまま、合流調停の処理が先行し、正しくない案内をしてしまったと考えられる。タグ選択と合流調停が完了した段階で、適切なタイミングで人運転の車に音声案内をする必要があると考えられる。

4.3.4 実行時間の評価

4.2.2 節の結果からもわかる通り、車両特徴量の抽出における平均実行時間は、5.70 秒、タグの選択的利用における平均実行時間は、0.29 秒であった。

車両特徴量の抽出においては、車種の分類器の識別に時間がかかり、車両の統合された車両特徴量を出力するのに時間がかかった結果、平均実行時間が長くなってしまったと考えられる。車種の分類器のしきい値を下げたり、車種の情報なしでもタグ選択を行えるようにすることによって、実行時間を削減できると考えられる。

一方、タグの選択的利用の実行はかなり高速であった。この部分については、実世界で用いてもリアルタイム性が保証できると考えられる。

5. おわりに

本稿では、カメラを用いて車両の視覚的特徴量を抽出し、その結果を選択的に利用する手法の評価を行った。提案手法により、人運転車両に対しても自動運転車両と協調して走行できるような音声案内をすることができた。

今後は、システム全体をより実世界で実現できるものに近づけていくために、物体認識や音声案内の改良や、外部の合流調停システムと連携できるようにしていく予定である。

謝辞 本研究は、JST, OPERA, JPMJOP1612, の支援を受けたものである。

参考文献

- [1] 大鐘 崇史, 渡辺 陽介, 高田 広章ステレオカメラを用いた車両特徴量の抽出と選択的利用 https://ipsj.ixsq.nii.ac.jp/ej/?action=repository_uri&item_id=205441&file_id=1&file_no=1(2020)
- [2] 渡辺 陽介, 竹内 栄二郎, 高田 広章, 二宮 芳樹交通マネジメントに向けたダイナミックマップアーキテクチャの研究; システム/制御/情報 Vol.60, No.11, pp. 488-494. https://www.jstage.jst.go.jp/article/isciesci/60/11/60_488/_pdf(2016)
- [3] Rayson Laroca, Evair Severo, Luiz A. Zanlorensi, Luiz S. Oliveira, Gabriel Resende Goncalves †, William Robson Schwartz and David Menotti: *A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector* <https://arxiv.org/pdf/1802.09567.pdf> (2018).
- [4] 田中 俊太郎, 原田 利宣, 小野 謙二: 畳み込みニューラルネットワークを用いた自動車の三次元モデルにおける各車種の特徴抽出と視覚化 https://www.jstage.jst.go.jp/article/jjske/18/1/18_TJSKE-D-18-00039/_pdf (2019).
- [5] 岩崎 洋一郎, 竹原 洋志, 宮田 俊彦, 倉本 俊昌, 北島 俊孝, 瀬戸口 恵: 物体検出アルゴリズム YOLO を用いた交通量と車両走行軌跡の自動計測 https://www.jstage.jst.go.jp/article/jceek/2018/0/2018_307/_pdf (2018).
- [6] 宮崎 千展, 松山 聖路, 齋藤 正史, 清原 良三: 人運転車両と自動運転車両の混在環境における, ドライバ支援方式: https://ipsj.ixsq.nii.ac.jp/ej/?action=pages_view_main&active_action=repository_view_main_item_detail&item_id=177288&item_no=1&page_id=13&block_id=8 (2016).
- [7] 矢島 颯人, 高見 一正: 自動運転車と手動運転車混在時の進路交差のための車車間通信プロトコルと試作評価: https://ipsj.ixsq.nii.ac.jp/ej/?action=pages_view_main&active_action=repository_view_main_item_detail&item_id=190249&item_no=1&page_id=13&block_id=8
- [8] Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: CVPR. https://pjreddie.com/media/files/papers/yolo_1.pdf (2016)
- [9] ダイナミックマップ <https://www.nces.i.nagoya-u.ac.jp/ddm2/index.html>
- [10] 安全運転支援システム (DSSS) <http://www.utms.or.jp/japanese/system/dsss.html>
- [11] ROS <http://wiki.ros.org/ja>
- [12] Darknet YOLOv3 <https://pjreddie.com/darknet/yolo/>
- [13] Microsoft VoTT <https://github.com/microsoft/VoTT>