

自動制御へのカメラ認識による 障害物回避制御の結合に関する研究

A Study on the Combination of Obstacle Avoidance Control
by Camera Recognition to Automated Control

1731053 道田 峻佑

研究背景



昨年度までの研究: Autowareによる経路追従自動運転が可能

- 経路上に駐車車両があると走行不可



- 本研究: カメラ認識による障害物回避の結合
 - ➡ 障害物回避に用いるセンサはWebカメラのみ



目的

- 従来のAutoware自動制御へのカメラ認識による障害物回避制御の結合
- カメラ認識精度や認識速度の検証
- カメラ認識のみでの障害物回避の達成

カメラ認識に用いた実験ツール

表1. Jetson 仕様一覧

機種	NVIDIA Jetson AGX Xavier 開発者キット
OS	Ubuntu 18.04 LTS
CPU	64-bit 8-core NVIDIA Carmel CPU
メモリ	32GB
GPU	512-core Volta GPU with Tensor Cores
主なソフトウェア	DeepStream 5.0
	Cuda 10.2
	cuDNN 8.0
	TensorRT 7.1.3
	Gstreamer 1.0
追加したソフトウェア (JetPackに含まれない)	Chrony
	ROS Merodic

- 開発に必要なツールが一式提供 (JetPack 4.4)
- 画像認識に使用するCNN(畳み込みニューラルネットワーク)
➡ JetPack4.4付属のResNet10
- 使用した学習済みモデルはJetPack4.4付属
➡ 「人」「車」「自転車」「交通標識」学習済みモデル



図1. Jetson AGX Xavier 外観



図2. Webカメラ

カメラのキャリブレーション及び距離推定方法

- 距離推定方法:遠近法
- 各位置の対象物をJetsonで認識させる
 - ➡ カメラから3m, 5m, 10m, 20m離れた車線中央, 車線左側, 車線右側
 - ➡ 画像内位置の距離ごとの平均値(ピクセル)を記録
- 距離推定使用点:図4の緑印の画像内位置(ピクセル)
- 記録点以外の認識対象物は線形補間によって算出

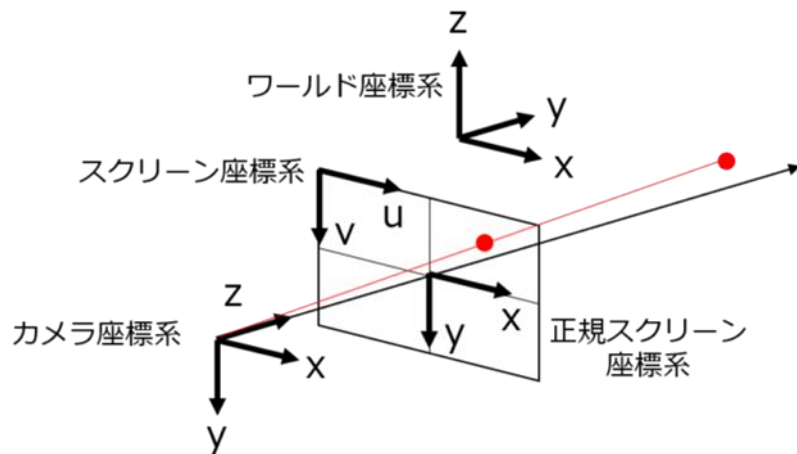


図3. 座標系の関係

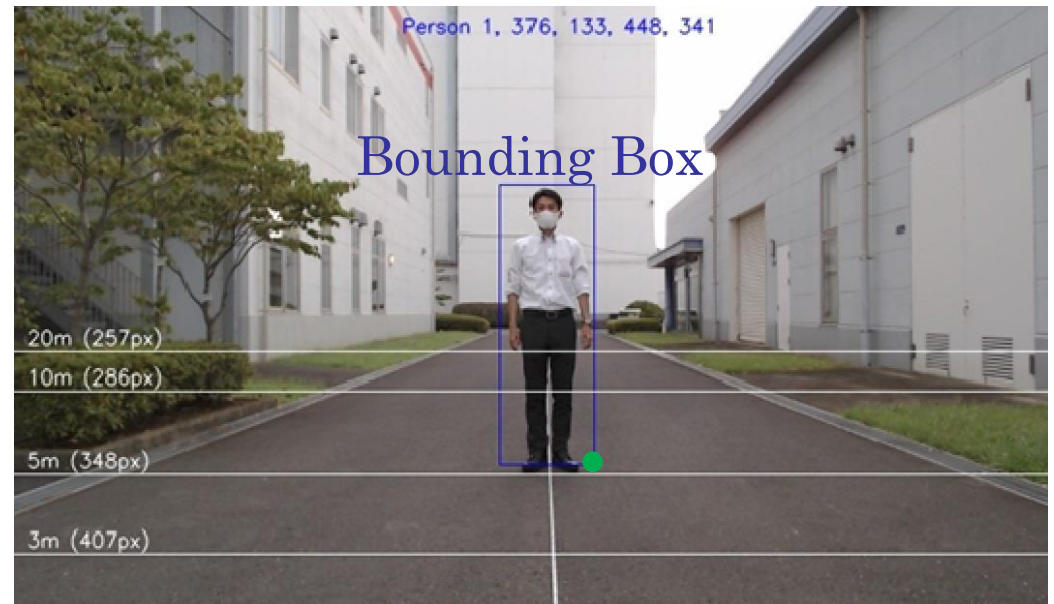
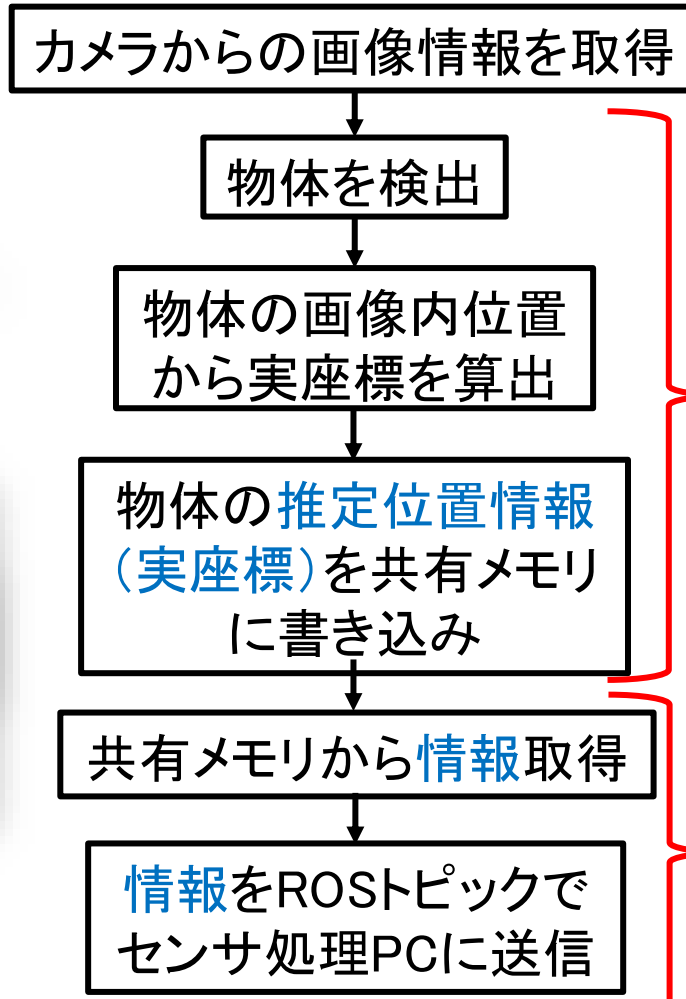


図4. キャリブレーションの様子(5m車線中央)

図3引用元: Developers.IO, 画像の座標を空間の座標に変換する, <https://dev.classmethod.jp/articles/convert-coords-screen-to-space/>

システム構成(カメラ認識)

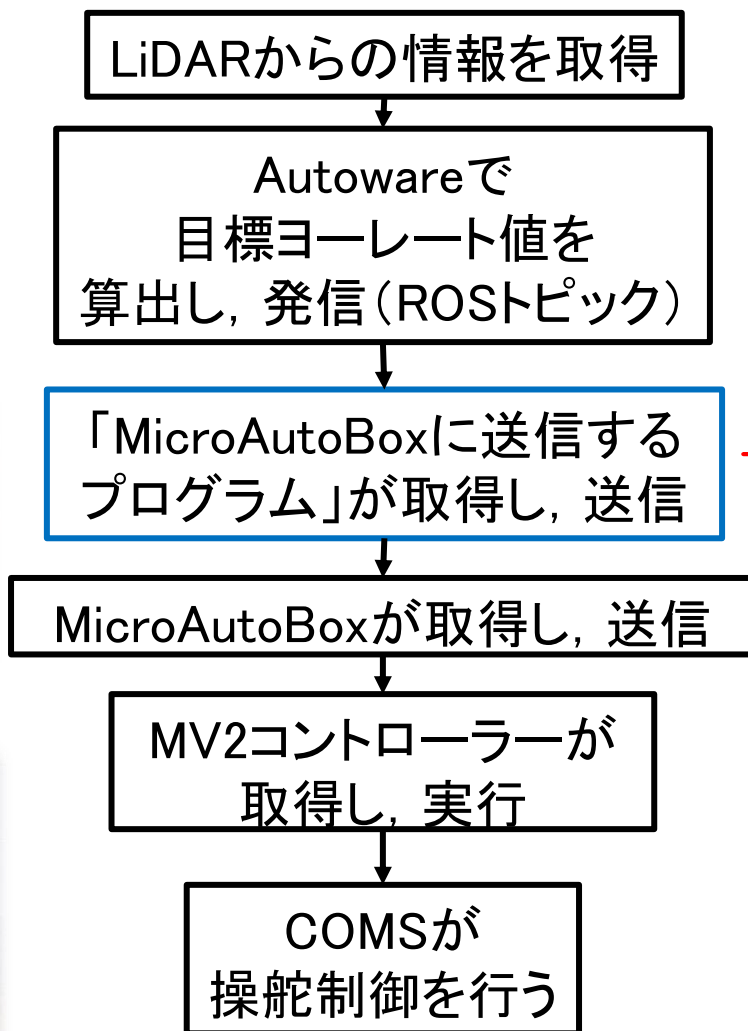


Jetpack付属の画像認識アプリケーションに機能追加

ROSノードを新規作成

図5. カメラ認識フローチャート

システム構成(ステアリング制御)



ここに障害物回避機能を追加した
• 目標ヨーレート値を変更し、
回避制御を行う。

図6. ステアリング制御フローチャート

回避可能判定式 $L_{wh} \geq L_a + C$

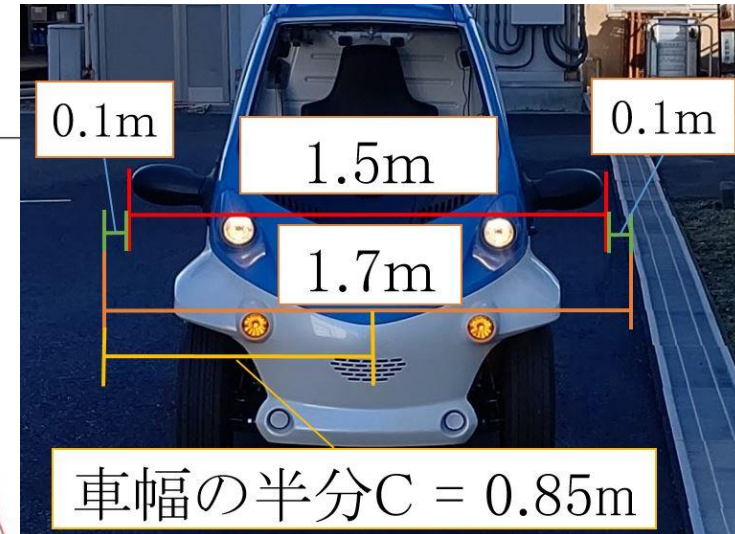
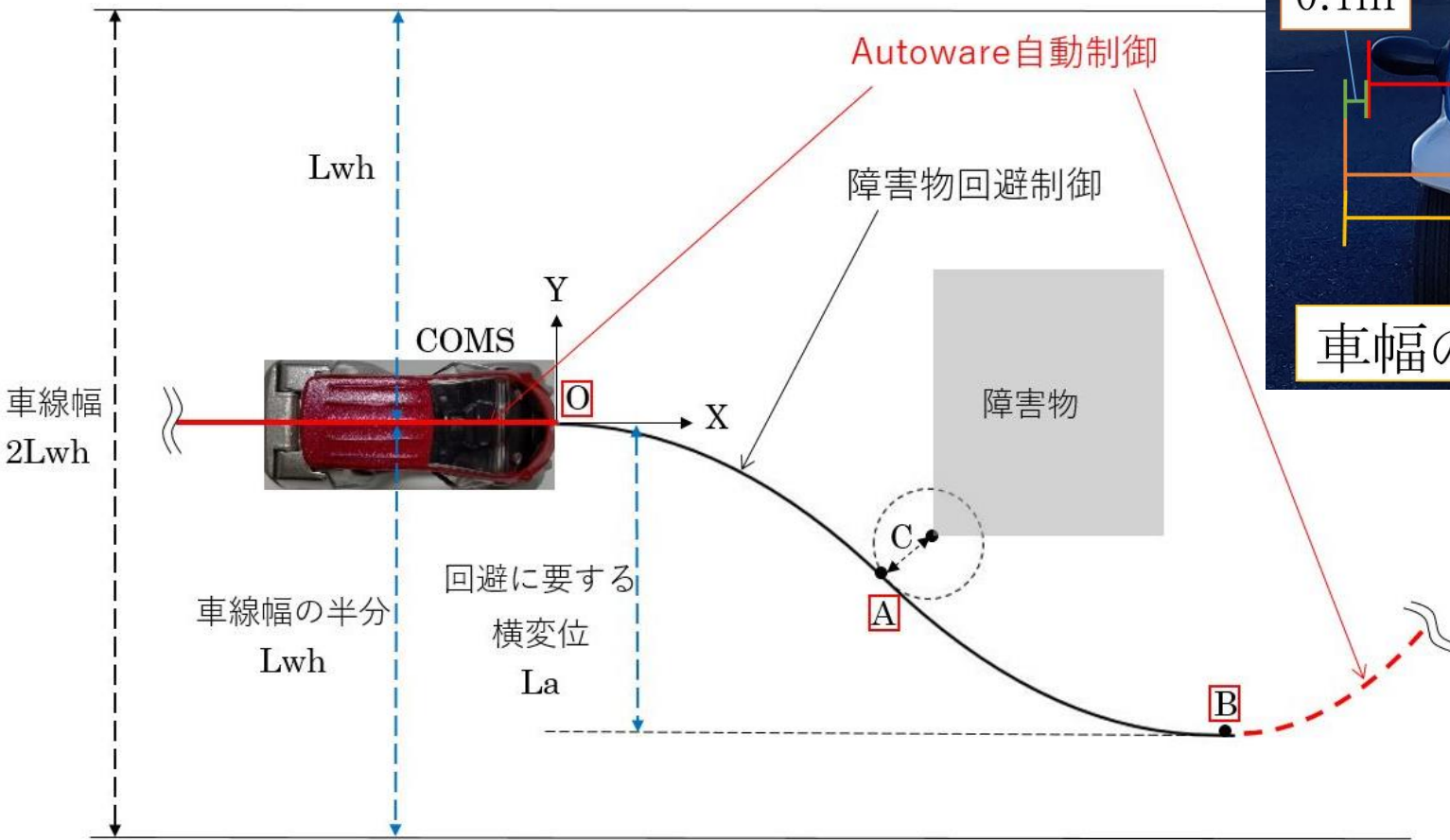
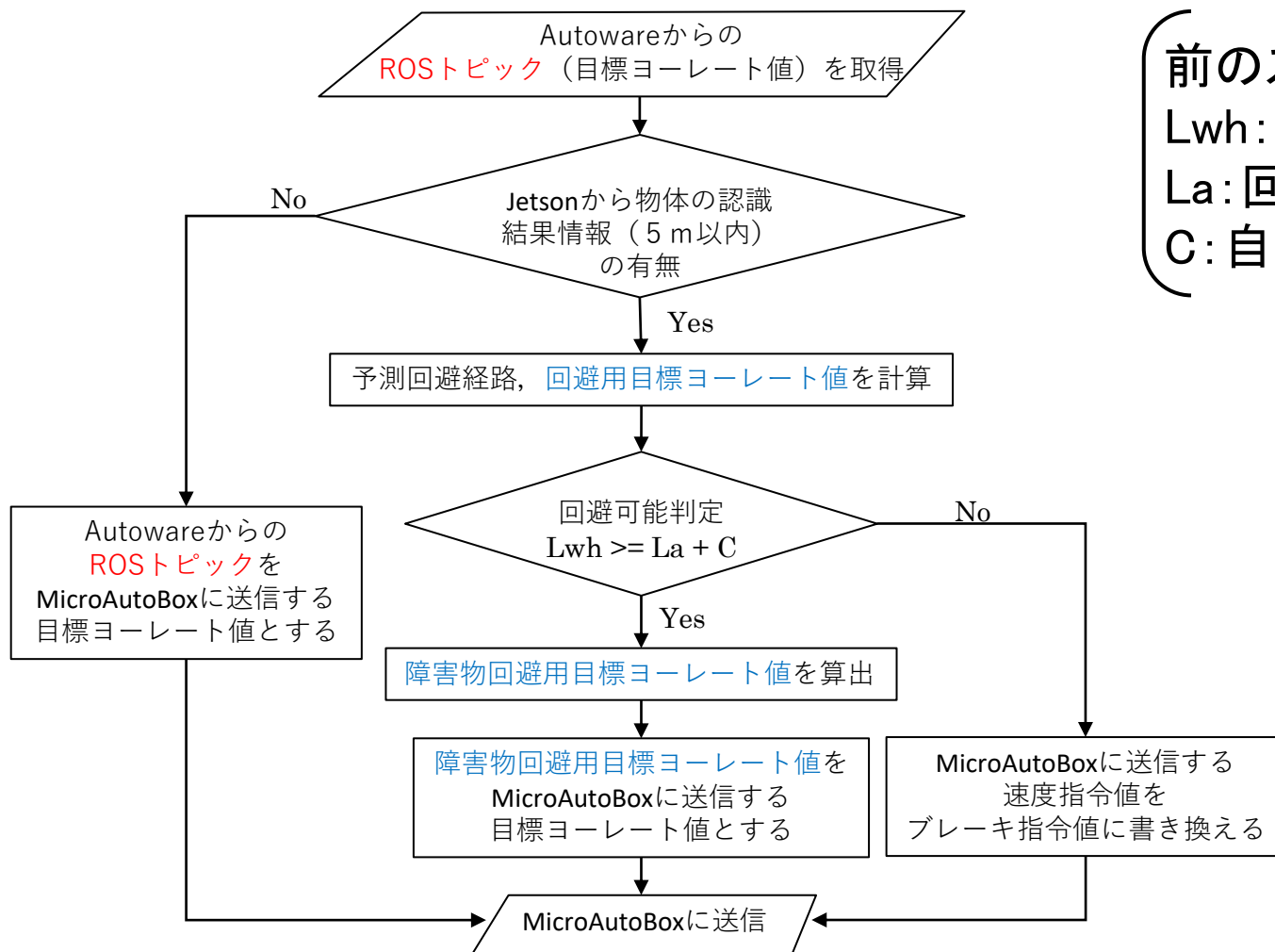


図8. Cの説明

図7. 障害物回避制御フローチャート

障害物回避制御

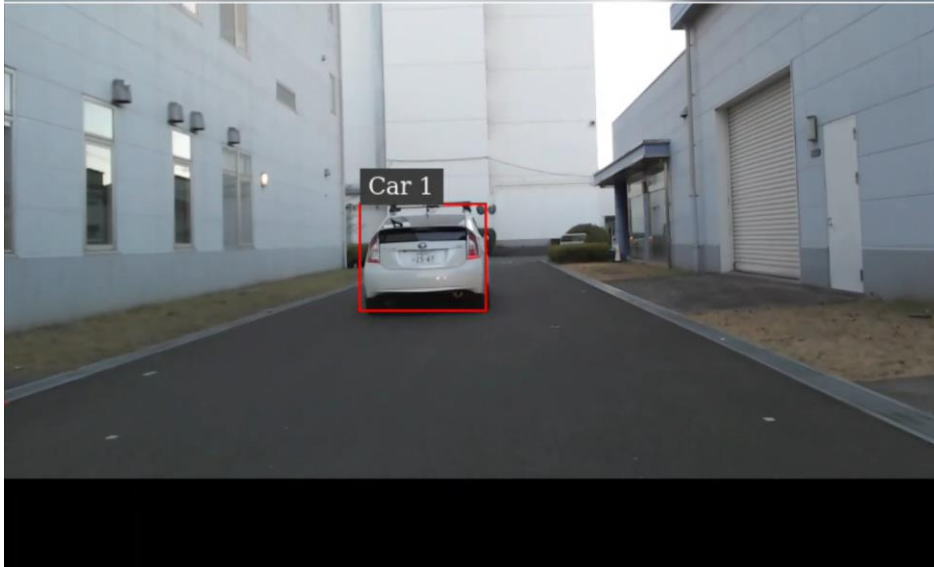


前のスライドで説明
Lwh: 車線幅の半分
La: 回避に要する横変位
C: 自車の車幅の半分と余裕距離

図8. 障害物回避制御フローチャート

障害物回避の様子

Autoware自動制御



カメラ認識映像



三人称視点映像

[実験結果] カメラ認識精度及び認識速度



図9. 認識対象物の幅の実寸値

- 障害物から5m地点での認識結果
 - 距離 約4.97m
 - ➡ 誤差 約0.03m
 - 幅 約1.55m
 - ➡ 誤差 約0.04m
- 認識速度
 - 平均値30FPS



回避制御を行える
認識精度及び認識速度

[実験結果] 走行軌跡

- OA区間: ほぼ逸脱なく走行できた
- AB区間: 目標経路と走行軌跡の逸脱幅が約0.3m
 - ➡ 指令速度3km/hを下回ったため
 - ➡ 実際の速度における最適な目標ヨーレート値より大きな値がCOMSに指示される状態
 - ➡ 目標経路より内回りとなった

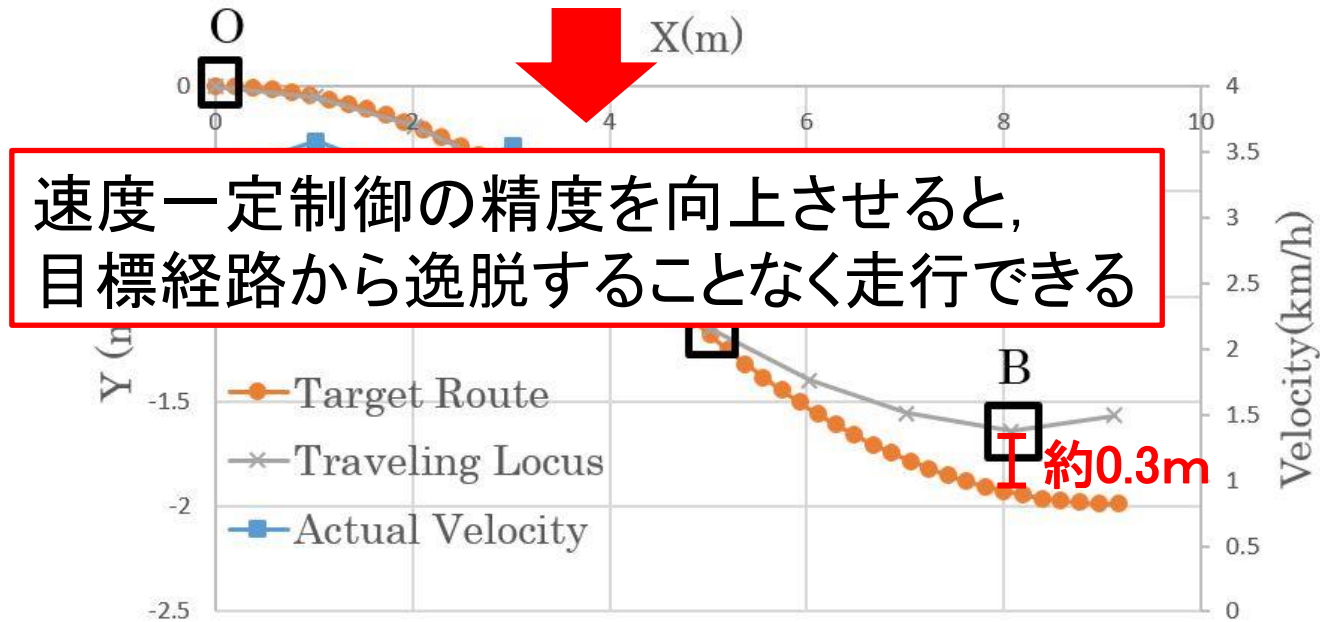


図10. 走行軌跡と速度の記録

まとめ

- 従来のAutoware自動制御へのカメラ認識による障害物回避制御の結合ができた.
- カメラからの画像情報で認識対象物の寸法や位置を推定できた.

今後の課題

- カメラ認識精度の向上
(坂道走行時, ピッチ変動時の対象物までの推定距離補正等)
- 車速制御精度の向上
- 障害物回避プログラムの改良