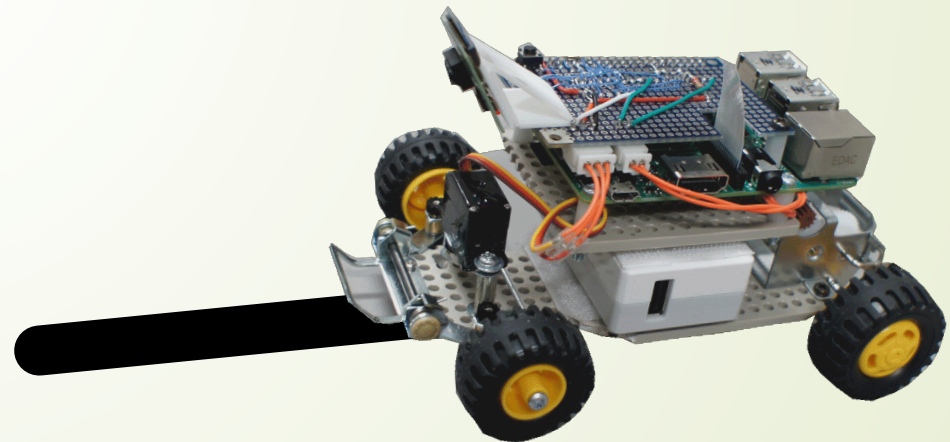
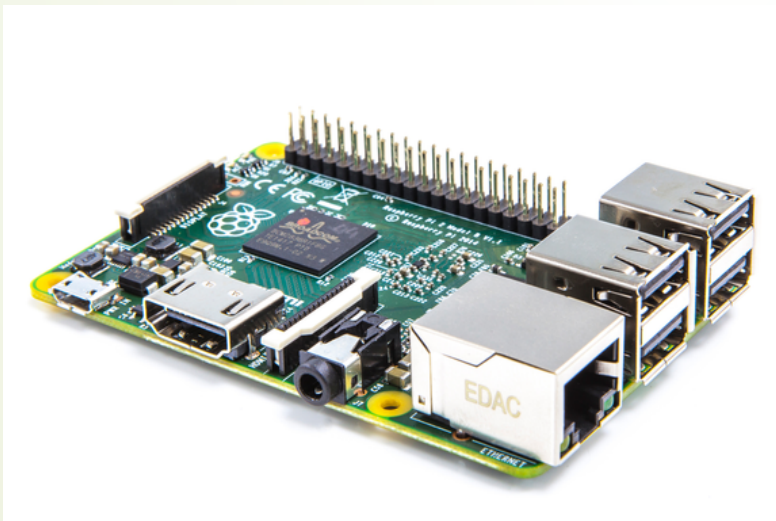


# Raspberry Piを用いた 模型自動車の制御

-車載カメラ画像の鳥瞰変換を用いた  
ステアリング制御-

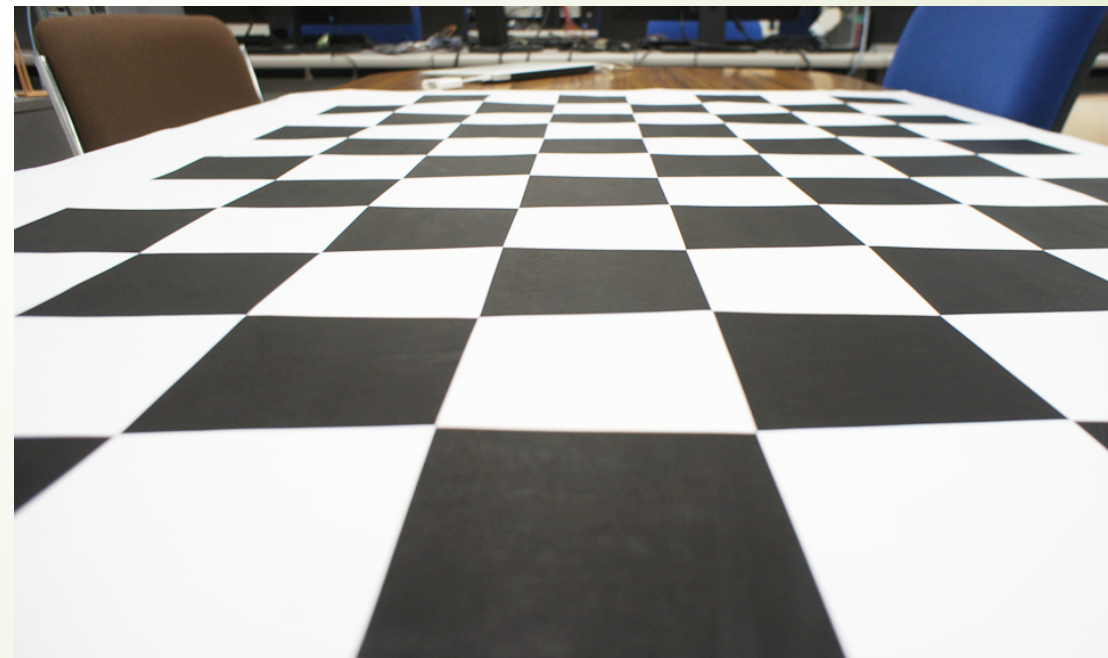
# 研究内容

- Raspberry Pi を搭載した模型自動車でカメラを使ったライトレースを行う。

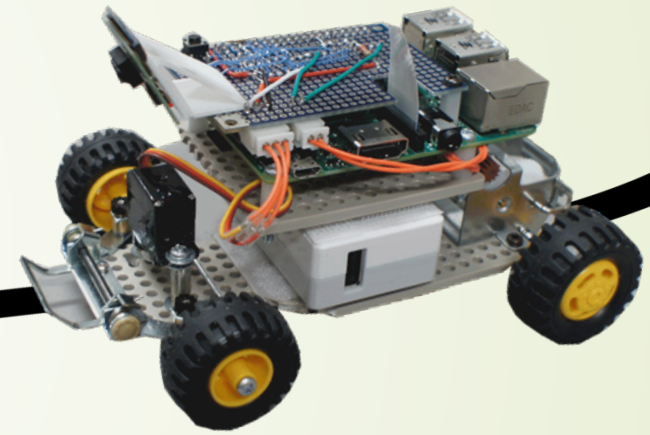


# カメラ画像の問題点

- 映る範囲が手前ほど狭く奥ほど広い
- 路面上の位置関係を把握しづらい



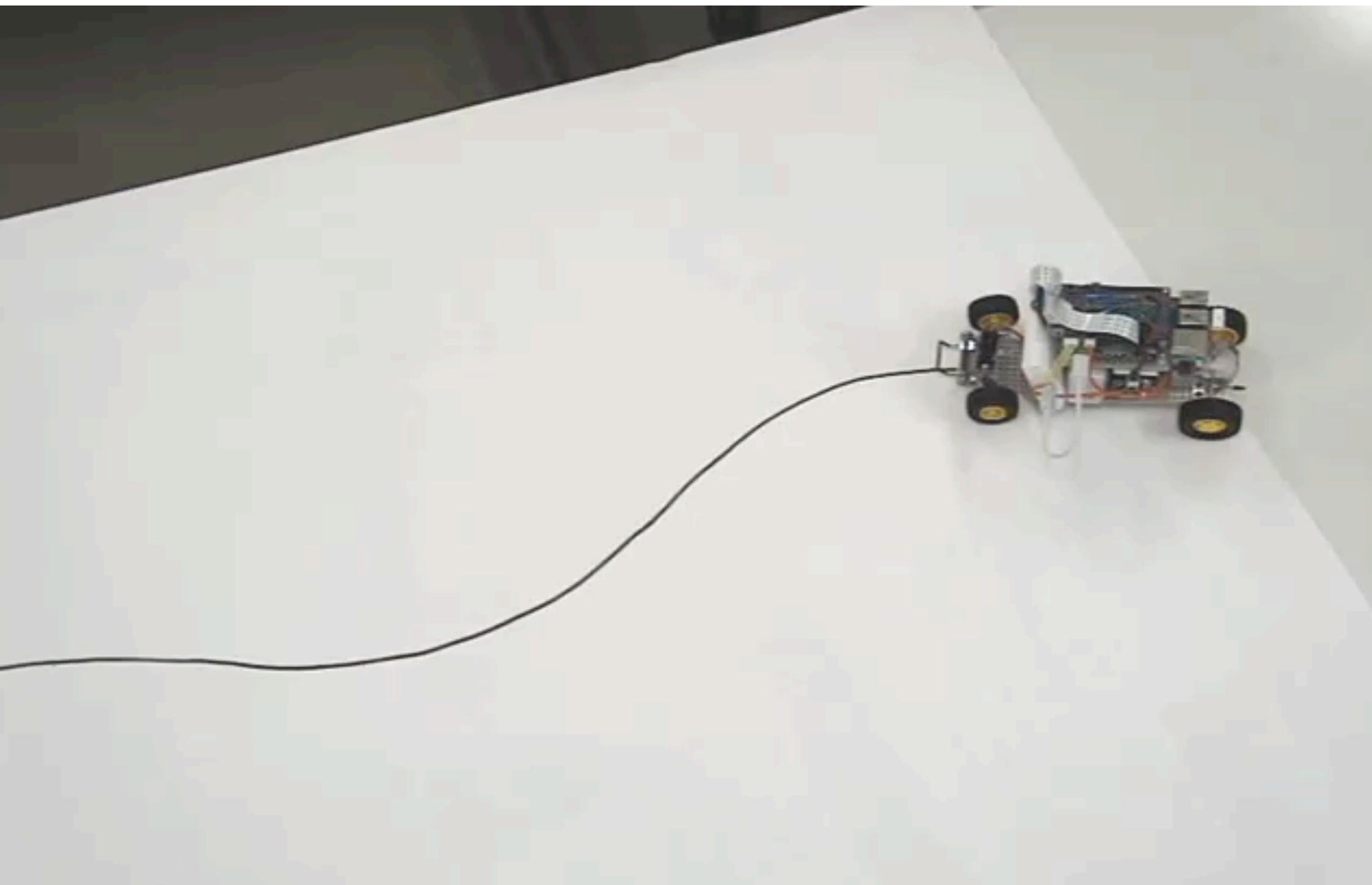
# ステアリング制御の問題点






# 研究内容

- カメラ画像と路面の位置関係を把握して車線の正確な位置を求めること
- 車線に応じて適切なステアリング制御を行うことでスムーズに走行すること



# 車線検出

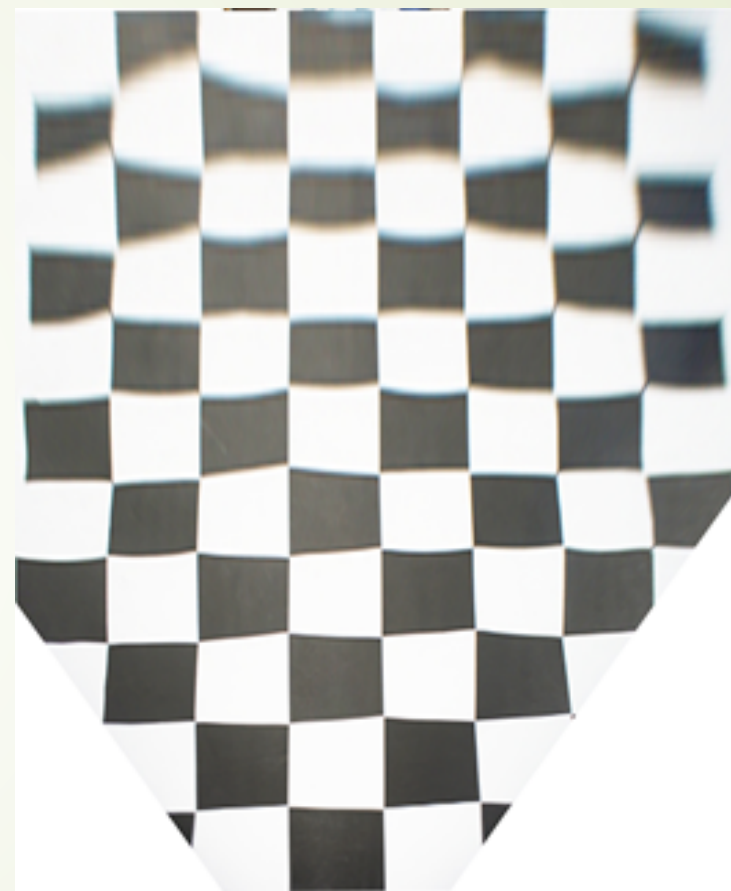
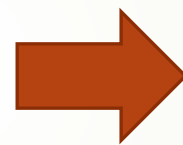
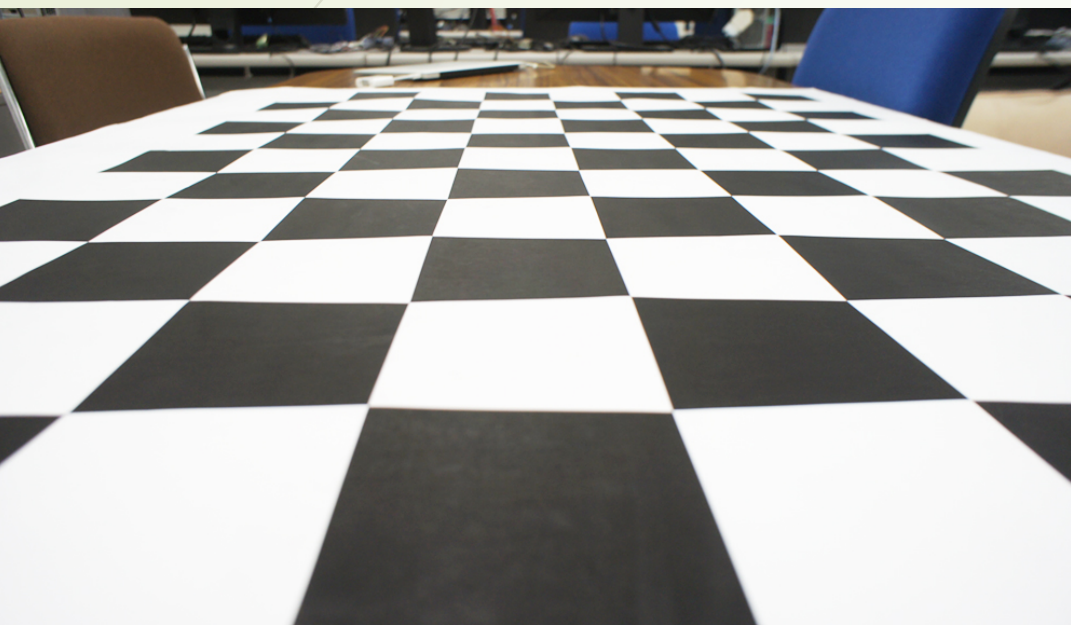




# カメラ画像の問題 提案手法



# 鳥瞰変換



※この画像はイメージで

# 鳥瞰変換

## ■ 同次座標系による座標変換

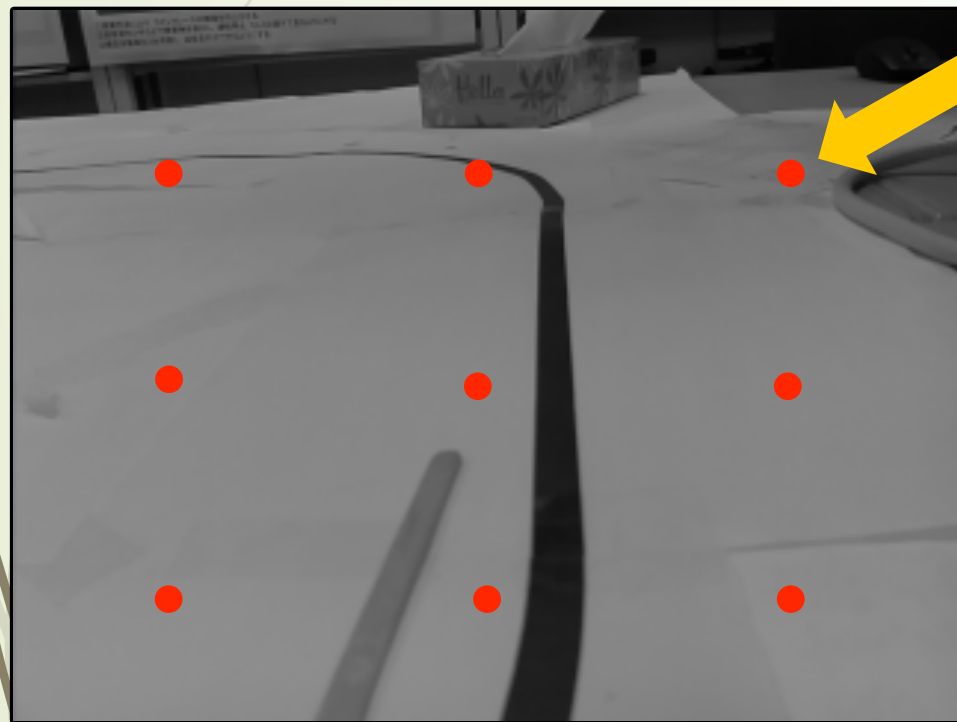
$$\bullet \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

■  $x, y$  : カメラ画像上の座標(pixel)

■  $X, Y$  : 車体を原点とした2次元座標(mm)

# 鳥瞰変換

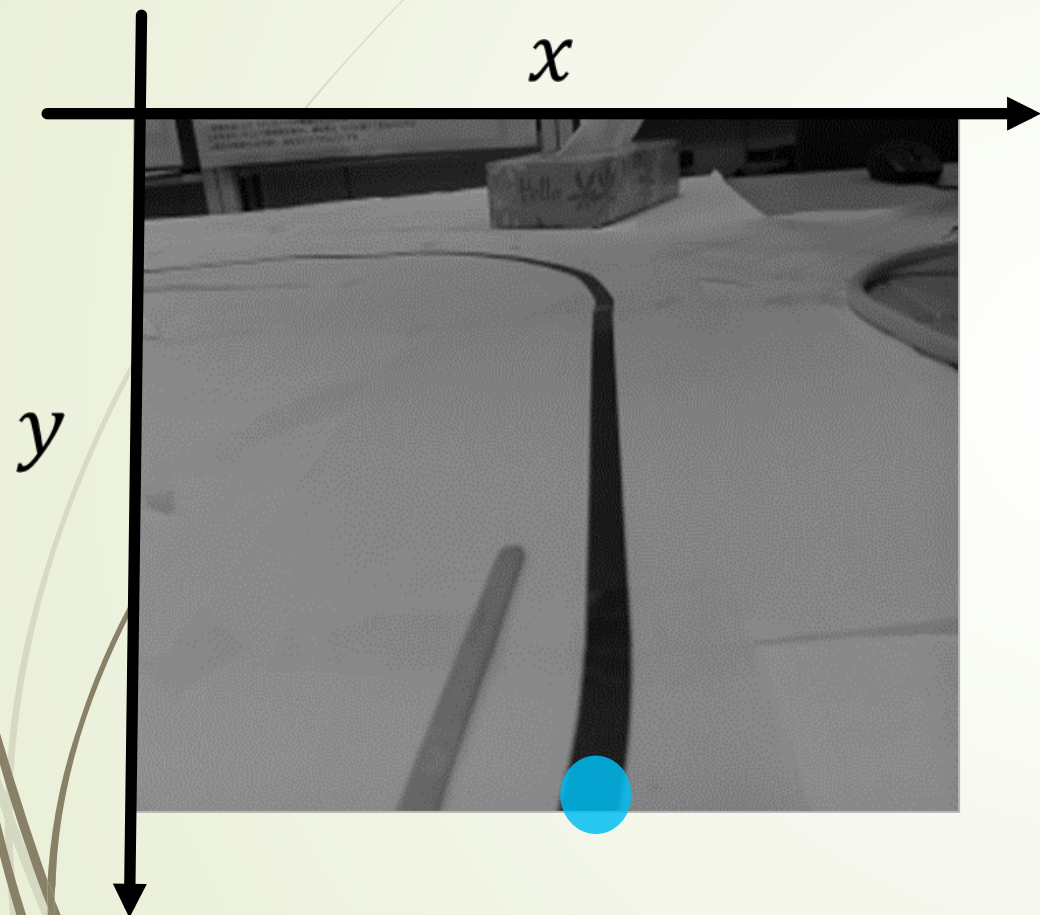
カメラ画像



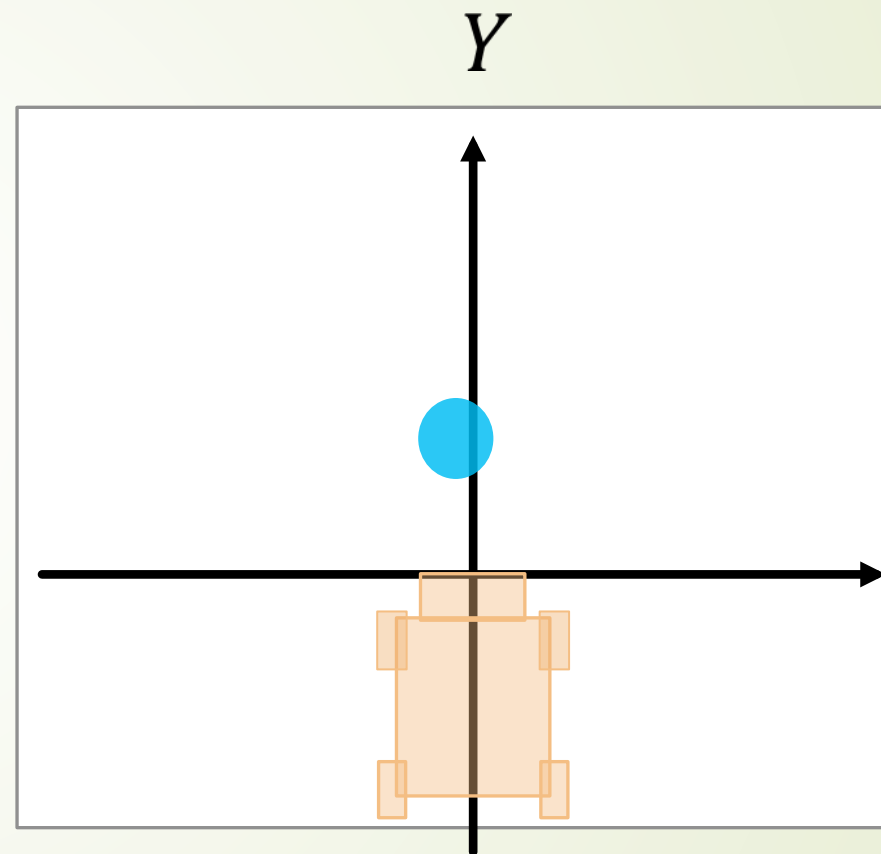
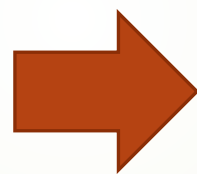
$$(x, y) = (300_{\text{pixel}}, 70_{\text{pixel}})$$
$$(X, Y) = (200_{\text{mm}}, 409_{\text{mm}})$$

$$\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \begin{pmatrix} 4.79 & -0.142 & -781 \\ -0.536 & -4.19 & 1909 \\ -0.0002 & 0.0369 & 1 \end{pmatrix}$$

# 鳥瞰変換



$$(x, y) = (150_{\text{pixel}}, 235_{\text{pixel}})$$

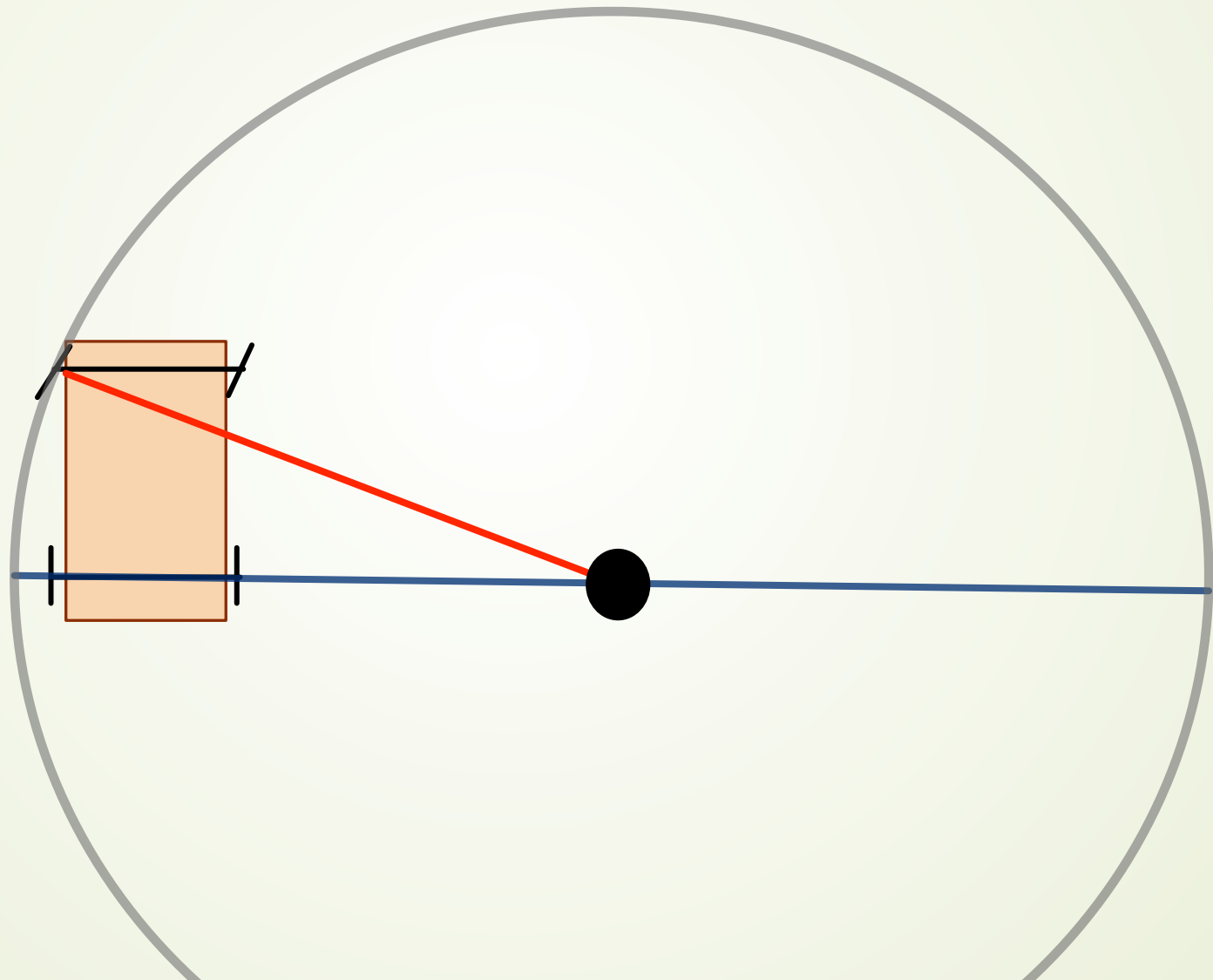


$$(X, Y) = (-3.701_{\text{mm}}, 95.725_{\text{mm}})$$



# ステアリング制御の問題 提案手法

# 車の進み方



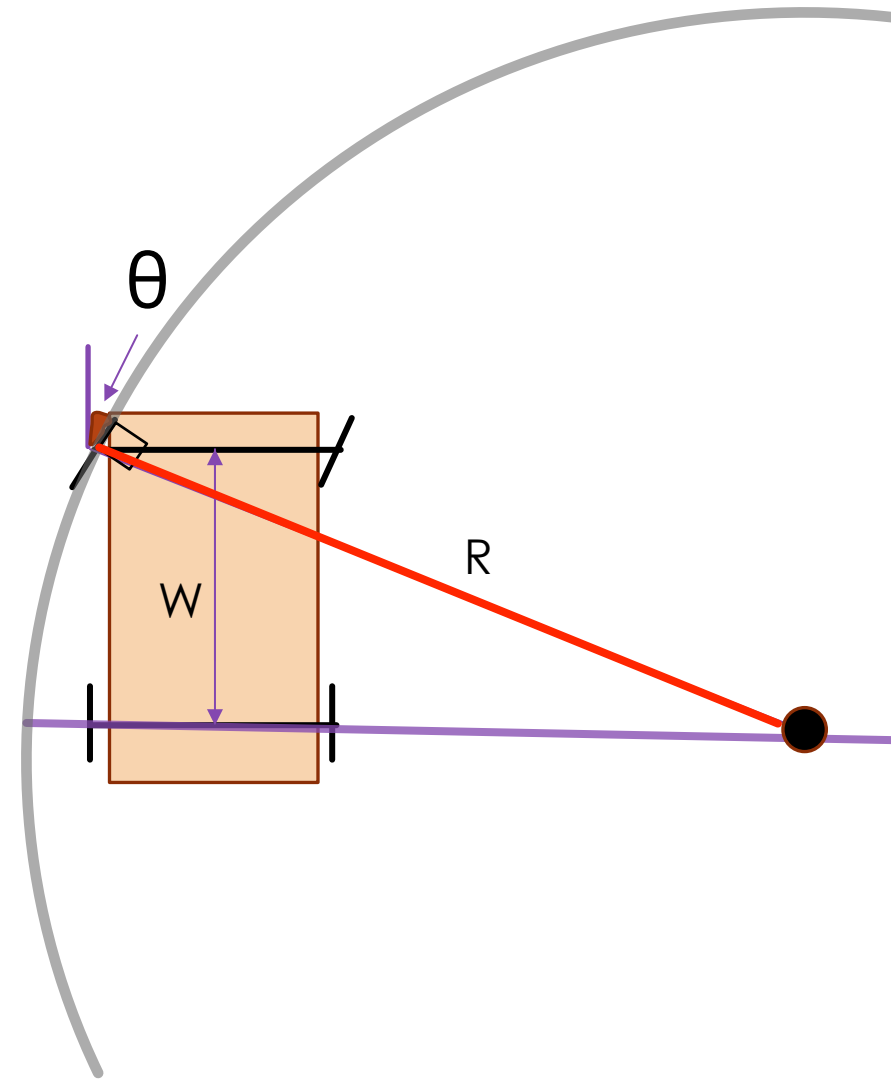
# ステアリング制御

## ➡ 回転半径

$$R = w / \sin(\theta)$$

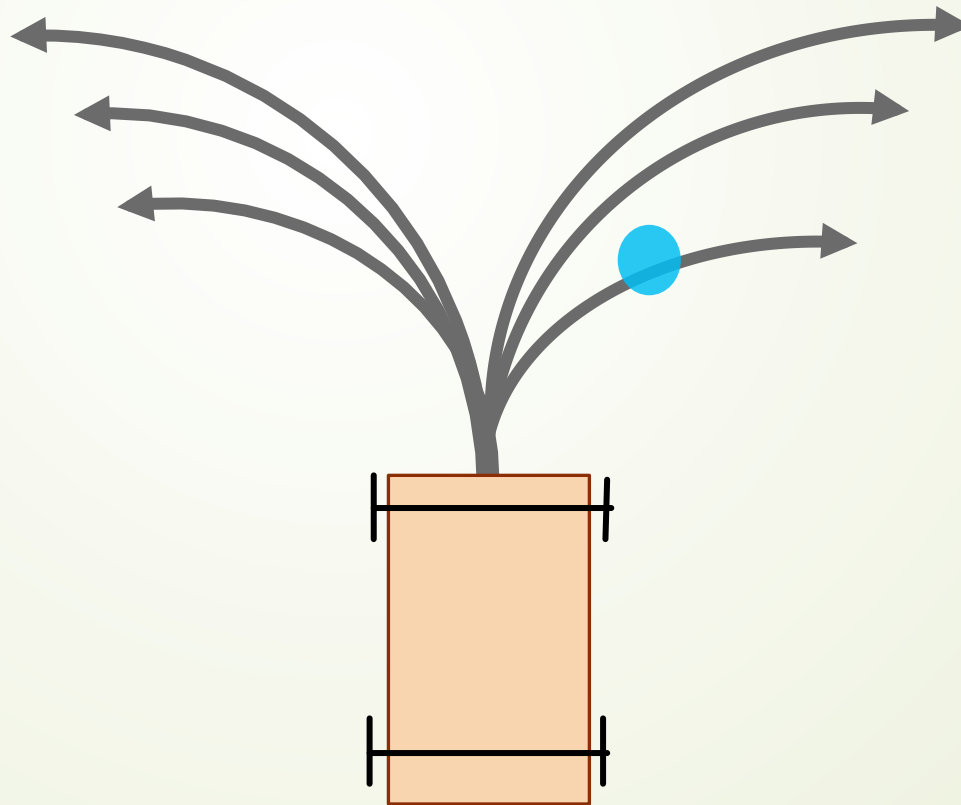
- $w$  : ホイールベース
- $\theta$  : 外側前輪の切れ角

## ➡ タイヤの角度は左右に 7段階ずつ制御可能



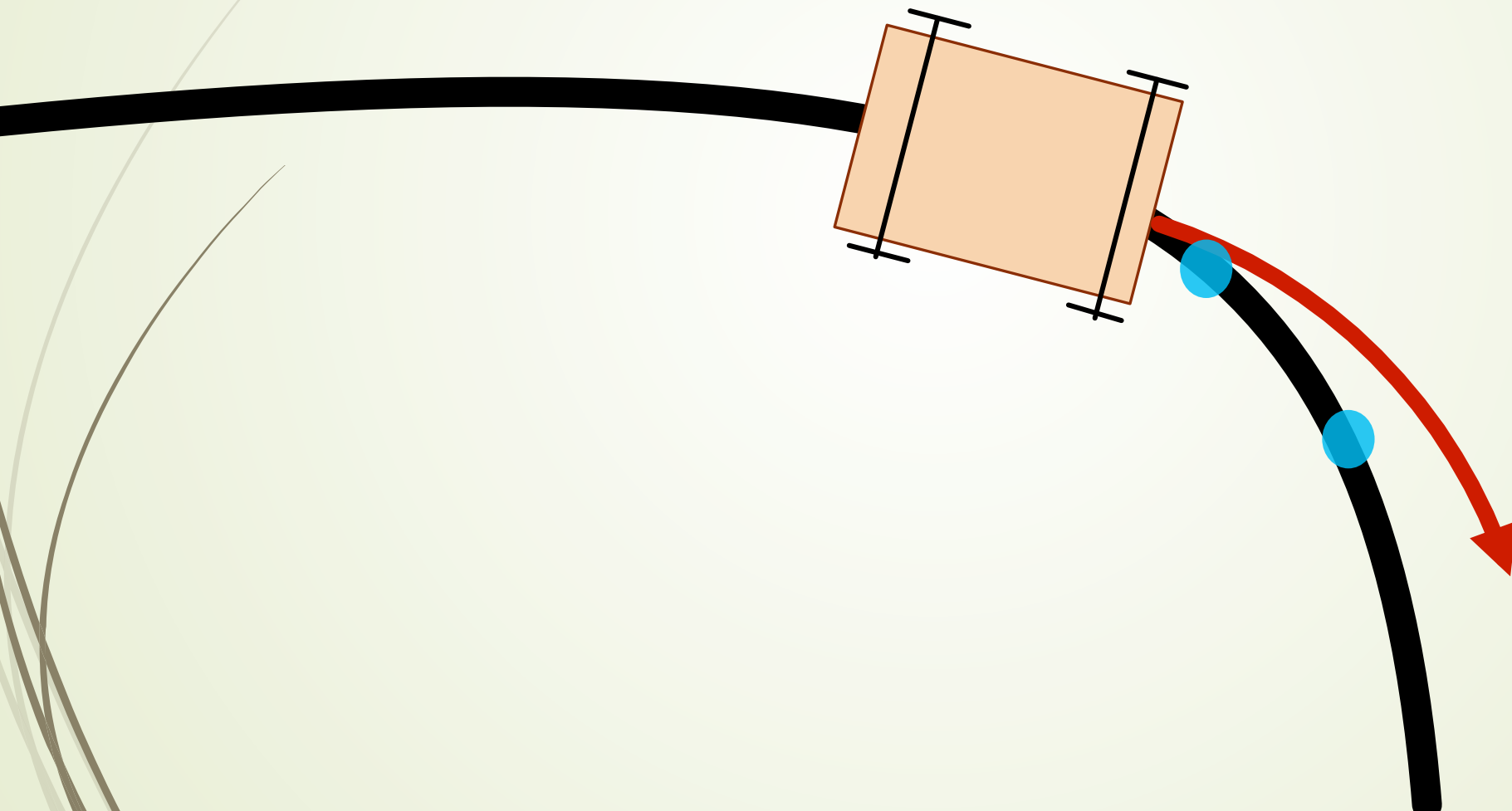
# ステアリング制御

- 全経路と目標点を比較し最も近い経路を採用





# 曲がりきれない場合



# まとめ

- 鳥瞰変換を使うことで、車線の位置を正確に把握しその上をスムーズに走行できた
- 車線検出の精度が低いためノイズが多い路面を走行できない