

LiDARと物体検出を用いた 駅ホームにおける視覚障害者向け 危険回避誘導スマホアプリの開発

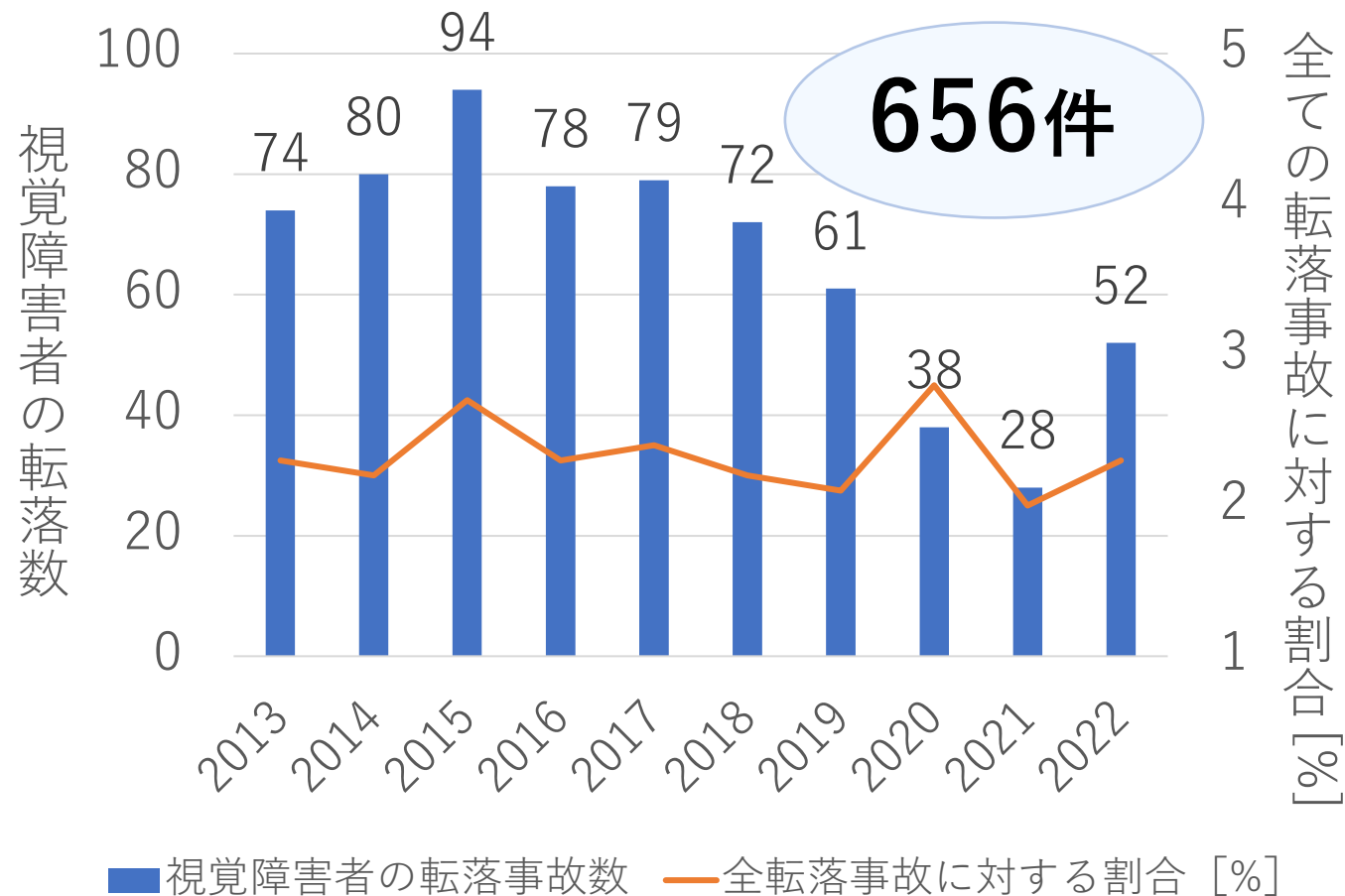
島川研究室

AE2 9番 白井 達也

背景

視覚障害者が駅ホームで転落する事故が繰り返し起きている

視覚障害者の転落事故数と割合



視覚障害者のデータ

全転落事故の割合

2.31%

約40人に1人

日本の人口との比率

0.25%

約400人に1人

10倍 近くの転落リスク

駅ホームにおける視覚障害者へのアンケート

日本の視覚障害者の人口比

約**400**人に**1**人

全転落事故における視覚障害者の割合

約**40**人に**1**人

視覚障害者へのアンケート調査

アンケート	回答数	転落数	割合
日盲連（2011）	252	92	36.5%
毎日新聞（2017）	222	70	31.5%
論文（2021）	297	108	36.4%
国土交通省（2021）	303	109	36.0%

3人 に **1人**

転落経験あり



駅ホームにおける視覚障害者へのアンケート

転落の原因として多く挙げられていたもの

ホーム端に気付かなかった

例) 定位の喪失、点字ブロックを見失う など

何らかの原因で方向感覚を失った

例) 人や障害物と衝突・回避行動、点字ブロックを見失う など

列車の位置を勘違いして踏み外した

例) 列車が向かい側に来ていると勘違い、列車の停止位置を誤認

駅ホームにおいて危険な状況

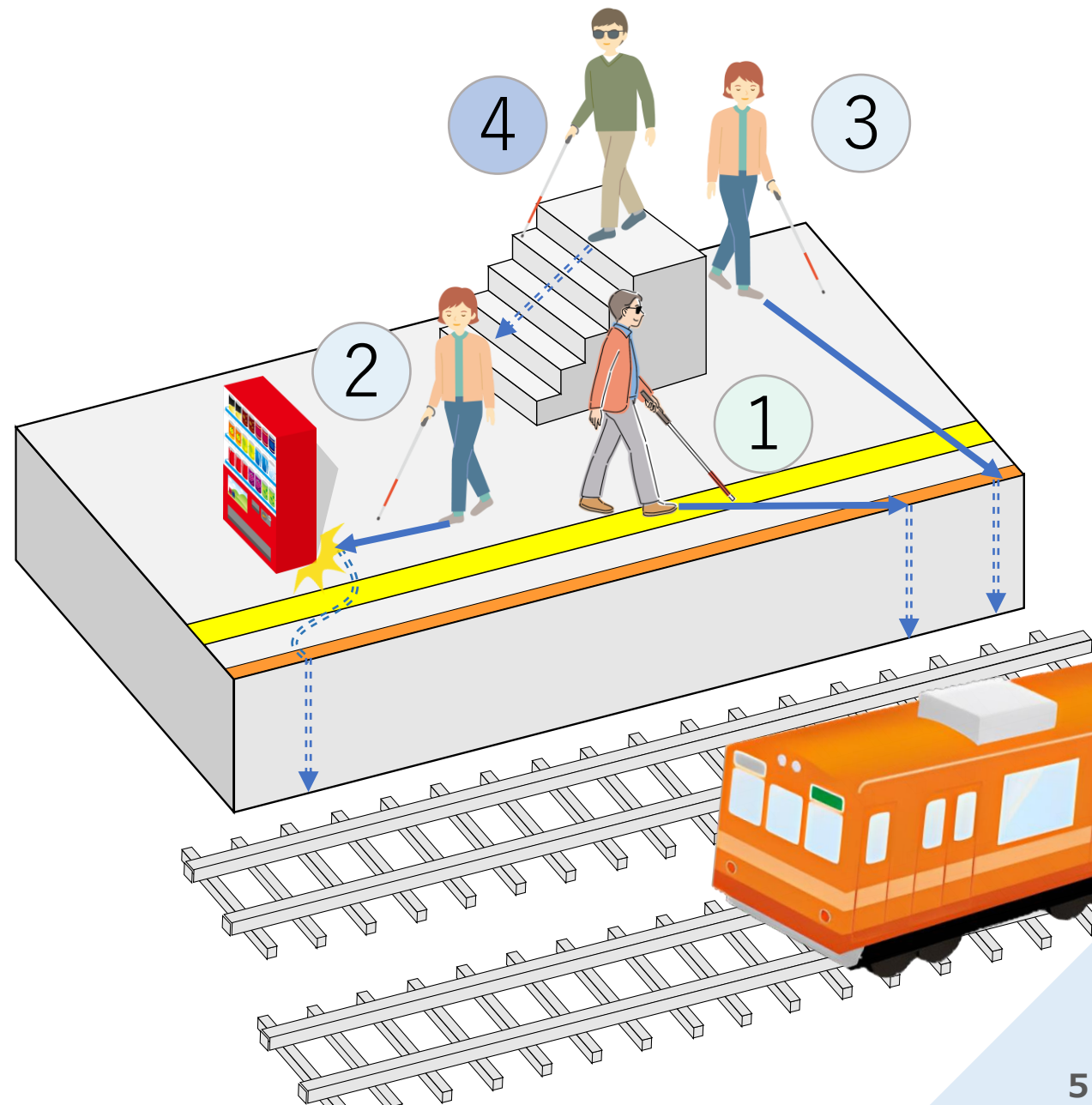
本研究で定義する危険な状況

① ホーム端に気付いていない

② 障害物との衝突・回避

③ 電車の位置の勘違い

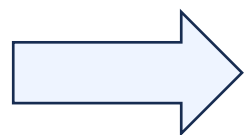
④ 階段からの転落



先行研究事例

駅ホームを使った研究

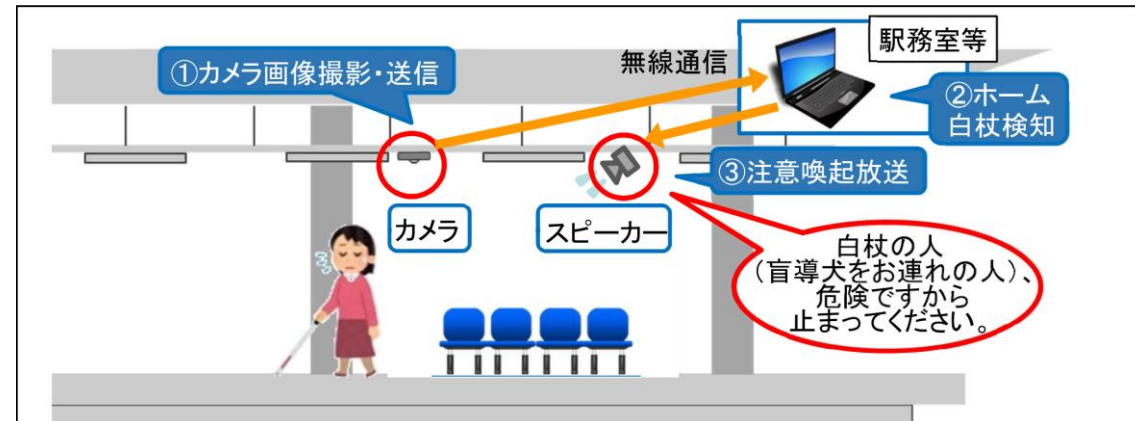
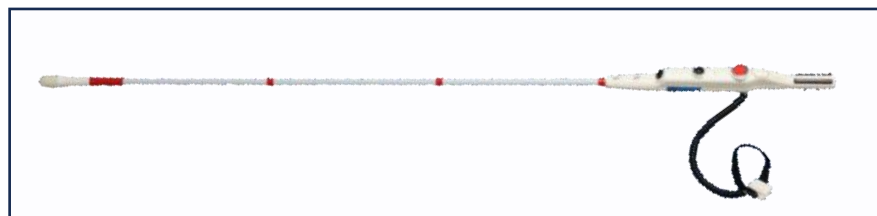
- 駅ホームの監視カメラによる警告
- QRコードによる誘導
- (ホームドア) → 設置率 11%以下



すべての駅に対応させるには
時間 と **コスト** が掛かる

支援道具を使った研究

- スマート白杖
- RGB画像や深度画像を使った危険回避アプリ



本研究の目的

転落の原因

- ホーム端を認識できなかった
- 何らかの要因で方向感覚を失った
- 電車の位置の勘違いをした

転落防止のために必要なこと

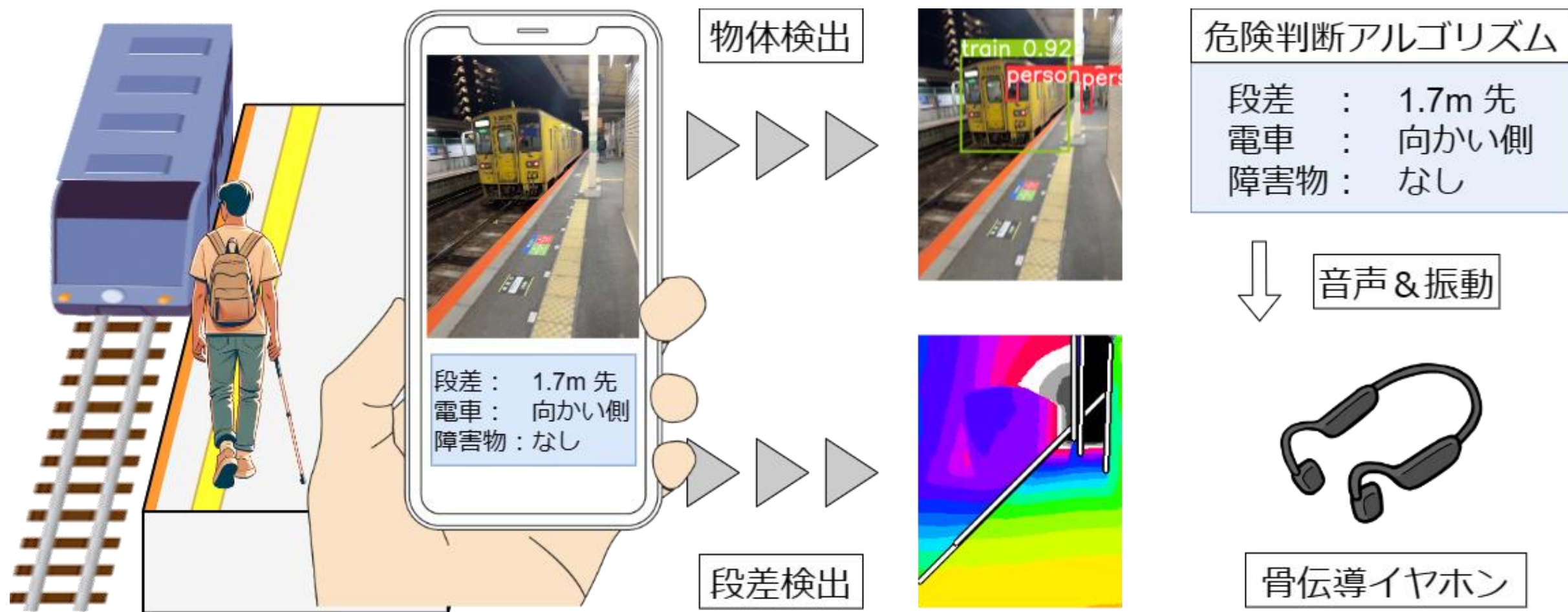
視覚障害者自身が
周辺の状況を把握し
危険を回避することが重要

視覚障害者が
周辺の状況を把握することを
手助けする手法が必要

研究の目的

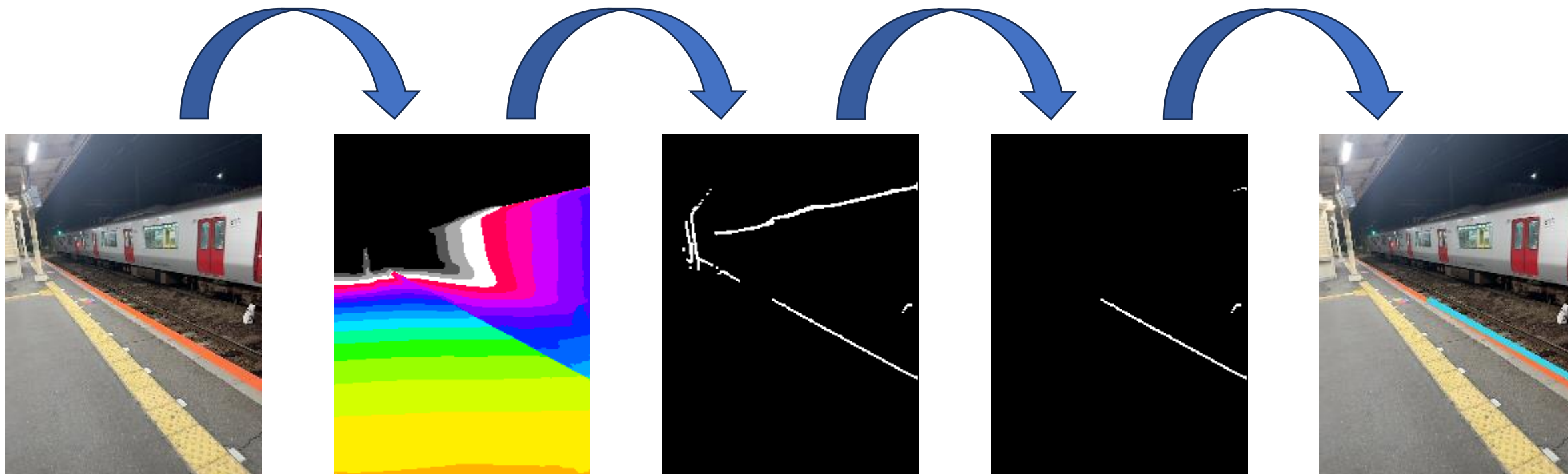
駅ホームにおいて視覚障害者が安全に歩行できるように支援する

アプリのシステム全体像



LiDARによるホーム端検出

ホーム端(線分) 検出までの流れ



線分(ホーム端)検出

OpenCV HoughLinesP 関数

閾値 : 65 最短の長さ : 25 同一直線としてみなす空白距離 : 20

障害物検出

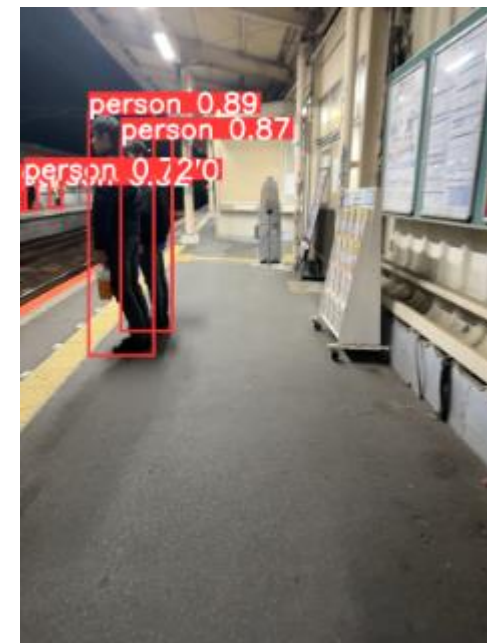
機械学習による障害物検出

機械学習フレームワーク CoreML

- デバイス上で処理 ⇨ ネットワーク不要
- Apple社のデバイスで最適化 ⇨ 高速処理

物体検出アルゴリズム YOLO

- 高速処理 ⇨ リアルタイム処理向き
- 複数の物体検出が可能



危険の通知

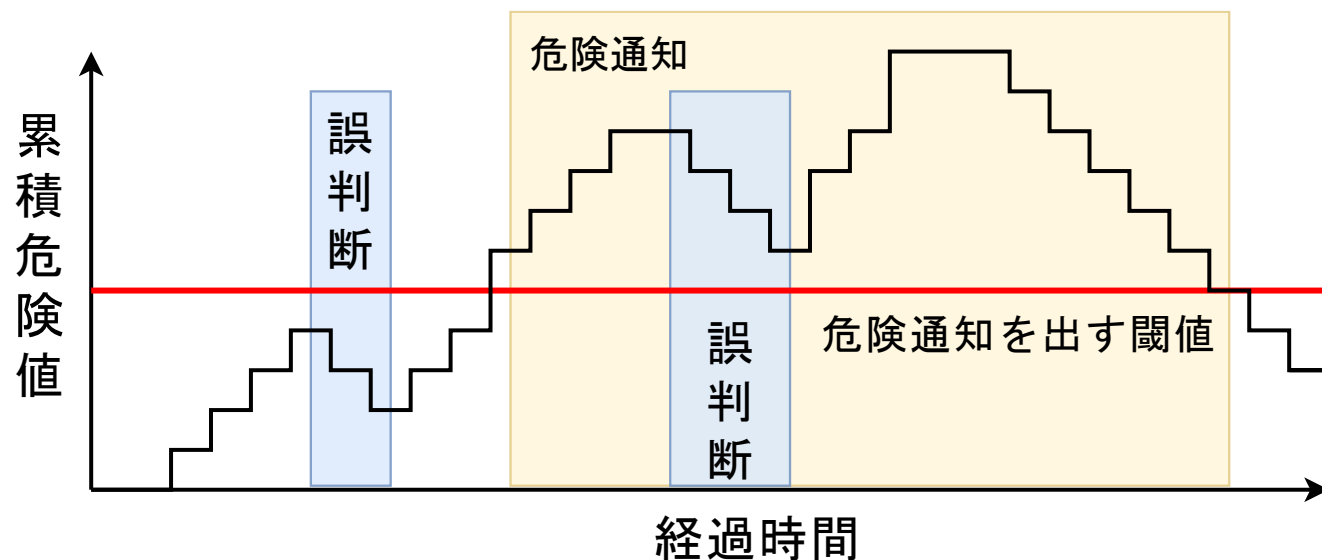
累積危険値

不安要素

- LiDARや物体検出の誤情報
- 数フレームだけでの判断

累積危険値とは...

- 危険な状況に応じて増減値を設定
- 閾値を超えているときに警告



増減例

- ホーム端 : 1m以内 \Rightarrow +3 1.5m以内 \Rightarrow +2 2m以内 \Rightarrow 0 2m以上 \Rightarrow -1
- 障害物 : 1m以内 \Rightarrow +2 1.5m 以内 \Rightarrow +1 1.5m以上 \Rightarrow -1
- 何もない : -2

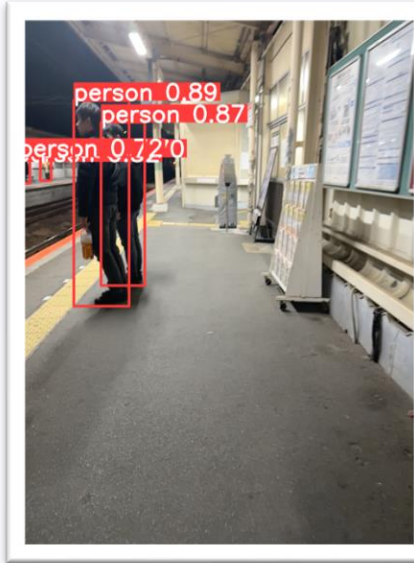
危険判断アルゴリズム

1



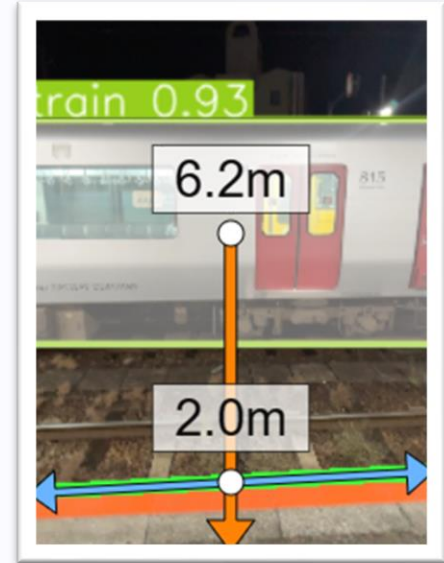
ホーム端 検出

2



障害物 検出

3



電車の位置の
判断

危険判断アルゴリズム

ホーム端の検出

検出された線分の座標と距離情報を組み合わせる

距離 ●

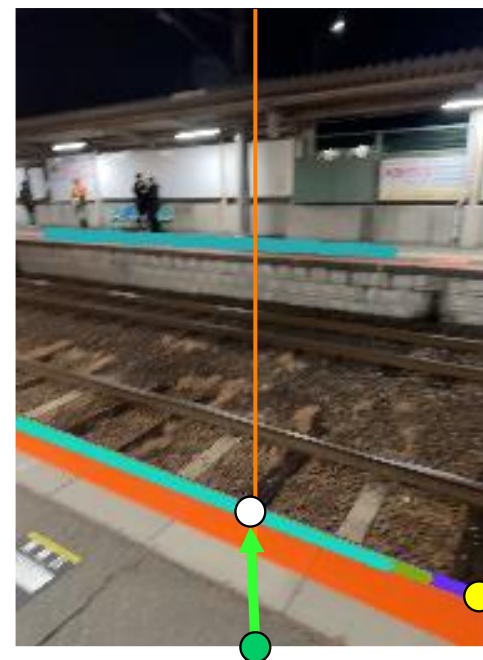
検出した線分の中で一番距離が近い点の距離情報

方向 ○

画面下部中央 ● から最短距離点 ● を含む

線分の中点 ○ までの角度を計算する

画面中央 | を角度0、画面左を-、画面右を+ とする



距離1.5m



距離2.1m

危険判断アルゴリズム

障害物の検出

物体検出の結果（位置と種類）と距離情報を組み合わせる

距離 ●

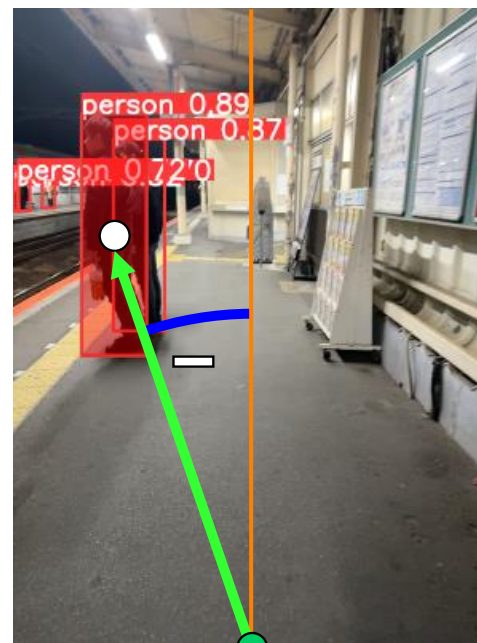
検出された障害物の中心座標の距離を比較
一番近い障害物を採用

方向 ○

画面下部中央 ● から最短距離点 ● を含む

線分の中点 ○ までの角度を計算する

画面中央 | を角度0、画面左を-、画面右を+ とする



物体検出

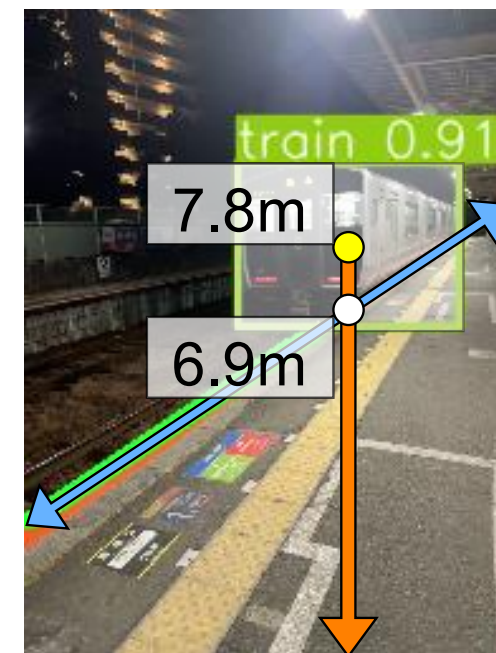
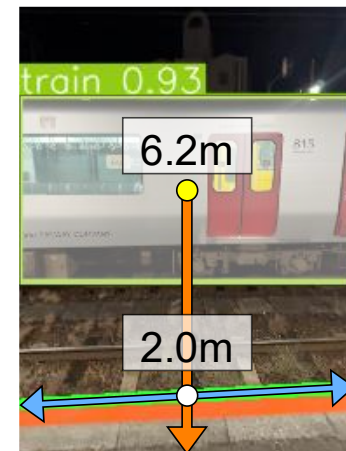


距離情報

危険判断アルゴリズム

電車の位置の判断

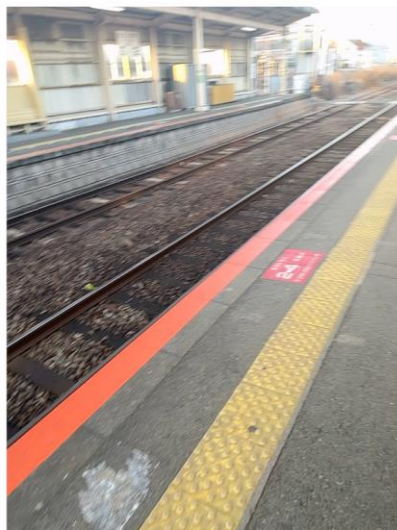
- 電車を検出した際、
 - ホーム端を検出していない場合
 - 電車が3m以内であれば、向かい側にいると判断
 - ホーム端を検出した場合、次の二直線の交点 \bigcirc を求める
 - 電車の中心座標から下への垂線 \downarrow
 - 検出したホーム端の延長線 \updownarrow
- 検出した電車の中心座標 \bullet と 交点 \bigcirc の距離の差を求める
- 距離の差が一定値以上だった場合、反対側のホームとして判断
- 音と振動を鳴らす



アルゴリズムの実行動画



中心座標距離: 5.785,192,256



障害物: 右:21, 距離:99.90
段差: 左:27, 距離:1.63

なし, 0
1.634m
safe
99.00 m

ホーム端の検出



中心座標距離: 1.940,192,256



障害物: 右:21, 距離:99.90
段差: 左:28, 距離:1.32
方向: 右 0.222, 音量: 0.826

段差 low, 30
1.322m
safe
99.00 m

障害物の検出



中心座標距離: 7.629,192,256



障害物: 右:21, 距離:99.90
段差: 左:12, 距離:1.62

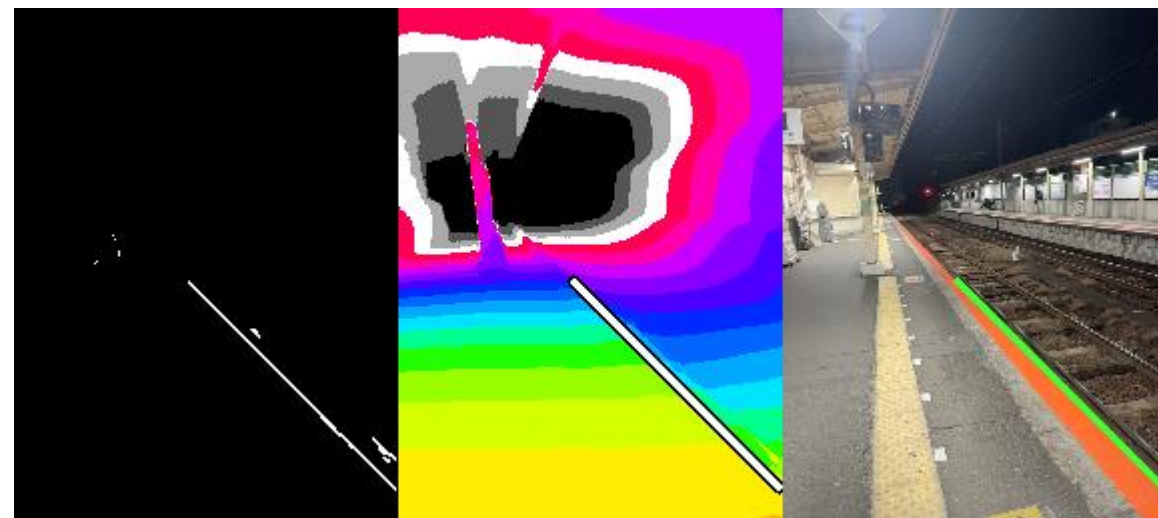
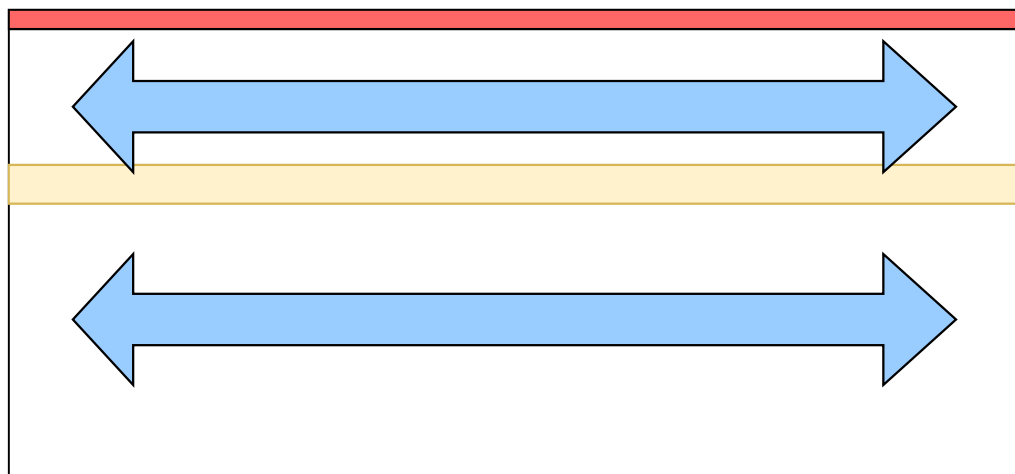
なし, 0
1.625m
safe
3.89 m

電車の位置の判断

ホーム端の検出精度の検証

検証1

1. 駅ホームで自然に歩いてデータを収集・保存（約5000枚）
2. データをランダムで選び、処理を行う
3. 表示されているホーム端の最短距離をその画像の段差の距離とし、検出の有無を記録

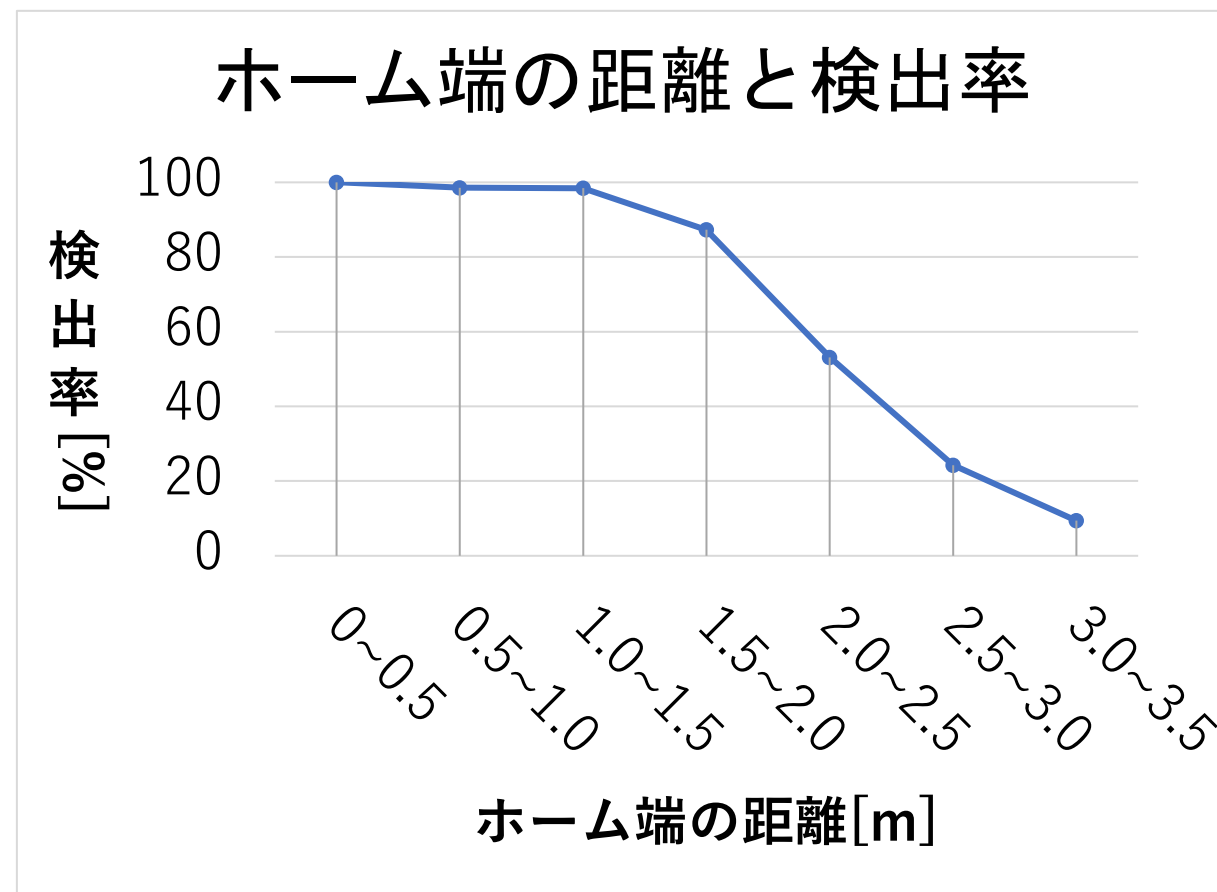


1.5m ~ 2.0 mの区間で検出あり

ホーム端の検出精度の検証

検証1：普段通りの移動経路のデータをランダムで抽出

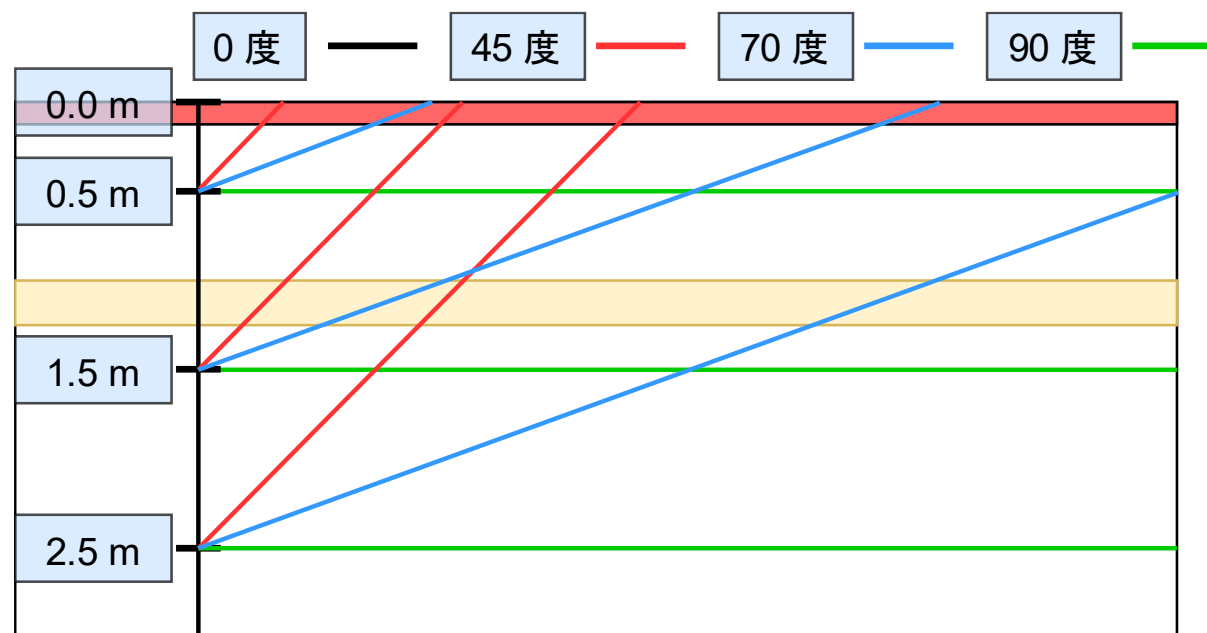
距離(m)	母数	検出数	検出率
0.0 ~ 0.5	1 枚	1 枚	100 %
0.5 ~ 1.0	70 枚	69 枚	99 %
1.0 ~ 1.5	263 枚	259 枚	98 %
1.5 ~ 2.0	337 枚	295 枚	88 %
2.0 ~ 2.5	102 枚	52 枚	51 %
2.5 ~ 3.0	34 枚	8 枚	24 %
3.0 ~ 3.5	33 枚	3 枚	9 %
合計	840 枚	687 枚	81 %



ホーム端の検出精度の検証

検証2

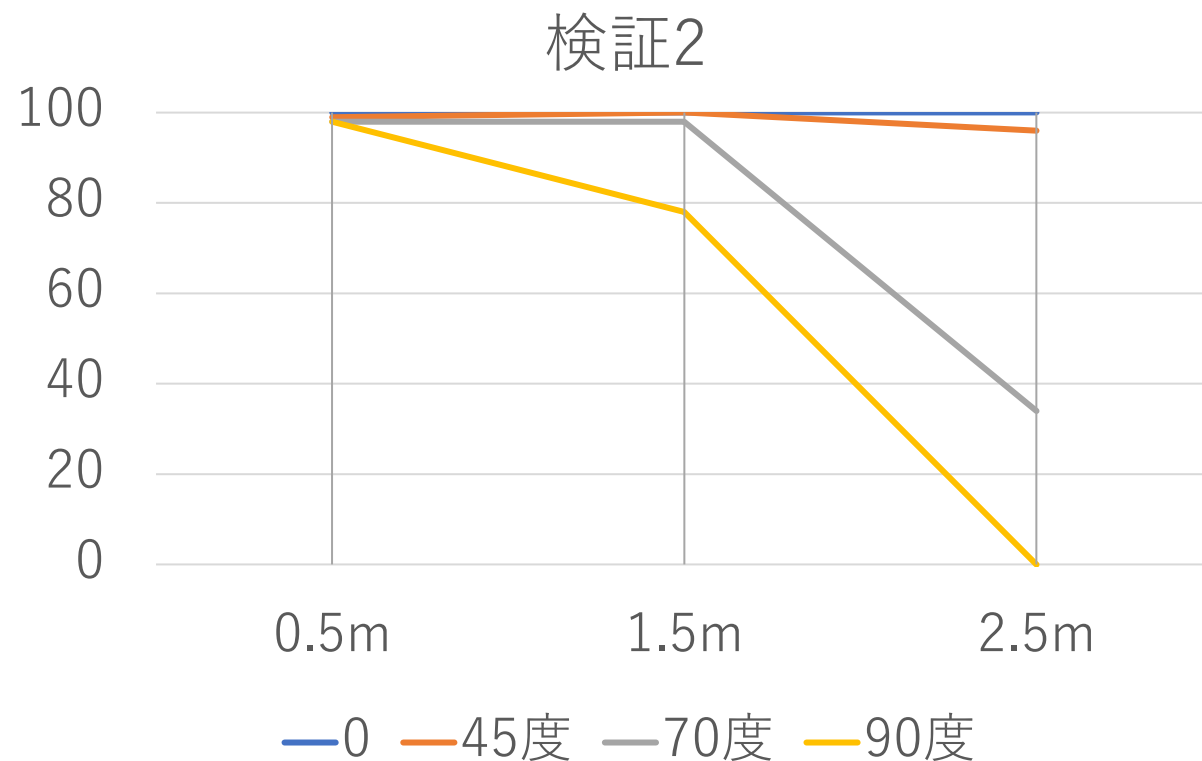
1. ホーム端からの**最短距離** を [0.5m, 1.5m, 2.5m]
人が見る**角度** を [0度、45度、70度、90度]
とし、それぞれの組み合わせ12通りで各100個ずつデータを収集
2. 段差の検出の有無のみを検証する



ホーム端の検出精度の検証2

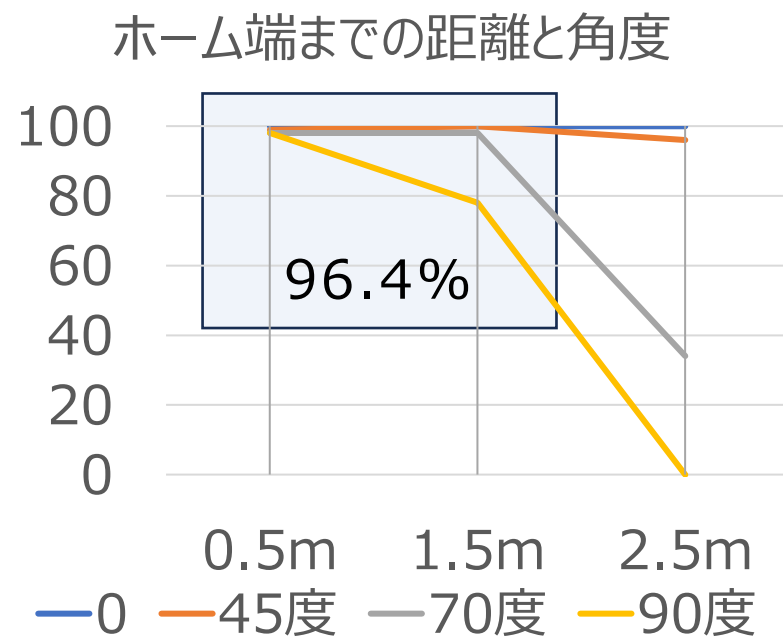
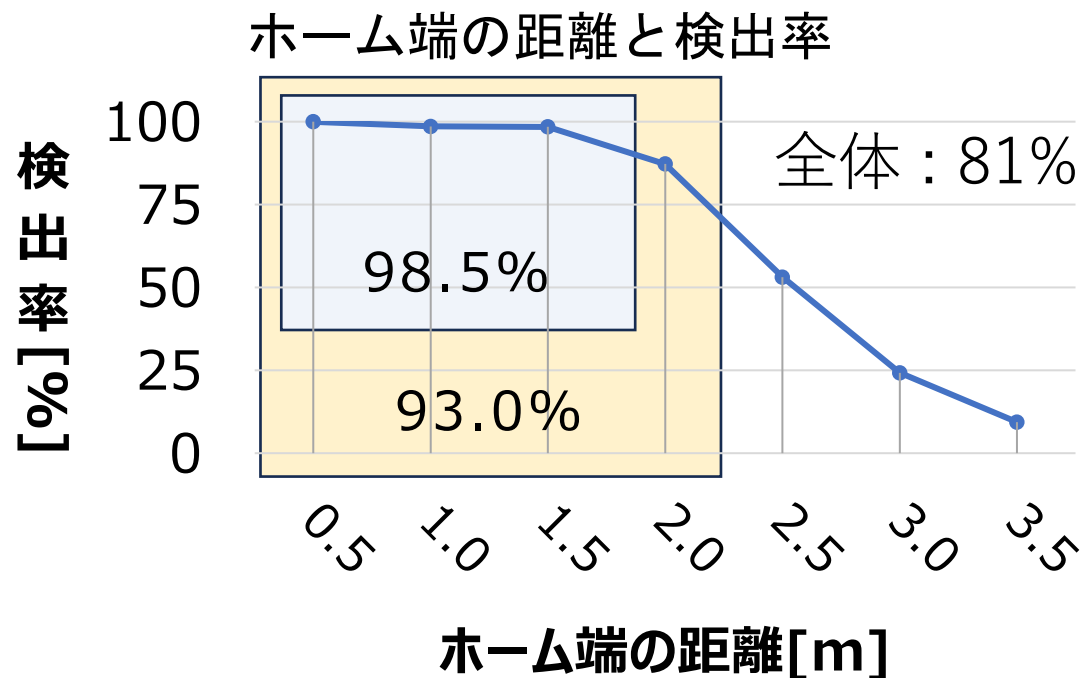
検証2：駅ホーム端までの **最短距離** と 人が見る **角度** を固定（100枚）

距離	0度	45度	70度	90度	合計
0.5 m	100%	99%	98%	98%	99%
1.5 m	100%	100%	98%	78%	94%
2.5 m	100%	96%	34%	0%	58%
合計	100%	98%	77%	59%	83%



考察

精度について



点字ブロックとホーム端の間隔

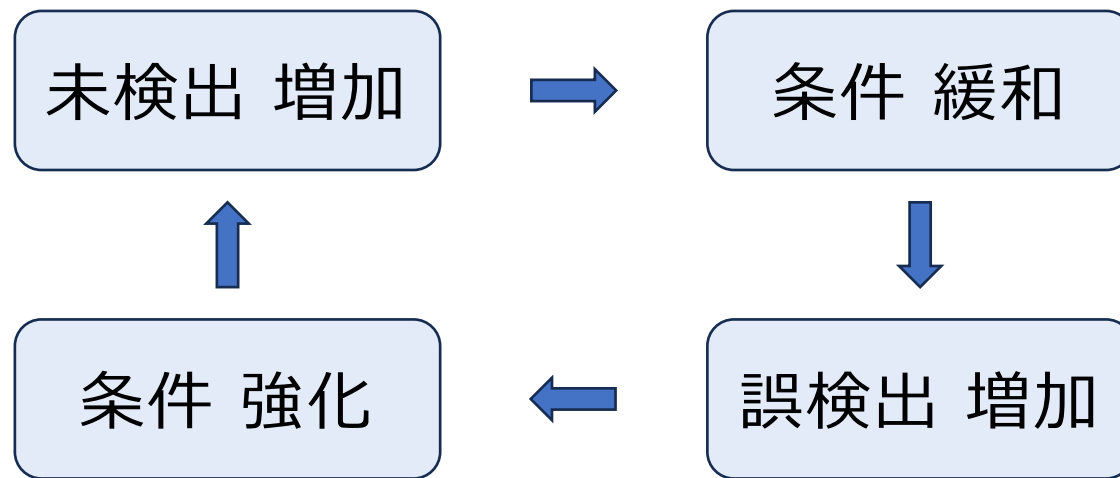
点字ブロックはホーム端から0.8m~1.0mに設置

1.5 m までは、高精度で検出されるため、段差検出アルゴリズムの有効性あり

考察

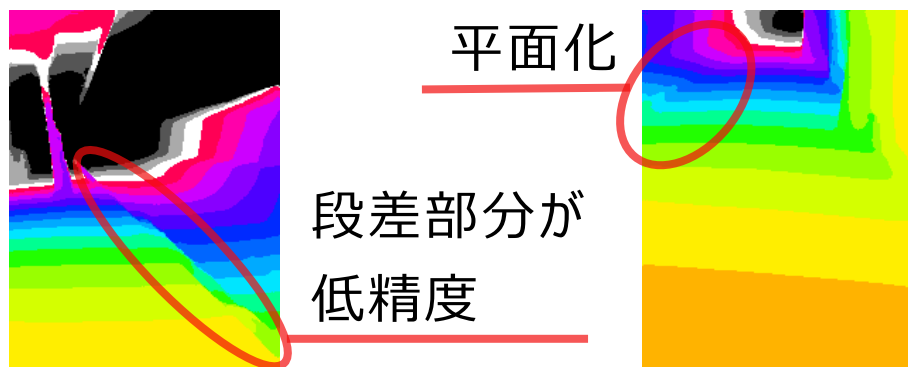
未検出について

原因1: 線分検出の閾値の調整不足



アルゴリズムでの対策が必要

原因2: LiDARの精度



距離が急激に変わる境界値

⇒ 精度が悪化する傾向がある

対策は困難

⇒ RGB画像などによる補助？

考察

誤検出について

原因1：除外のためのアルゴリズムの不足



柱

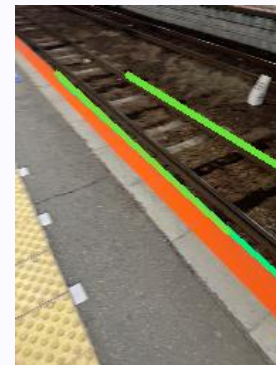


電車

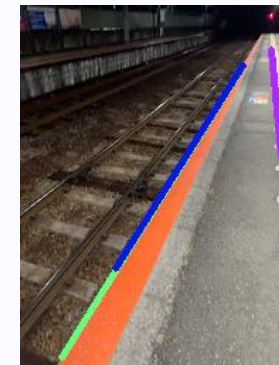


反対ホーム

原因2：LiDARによる誤情報



線路

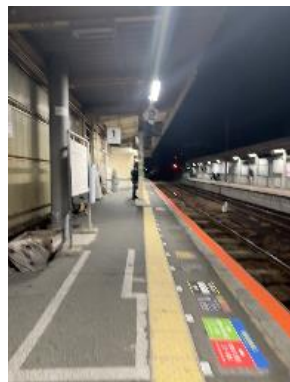


点字ブロック

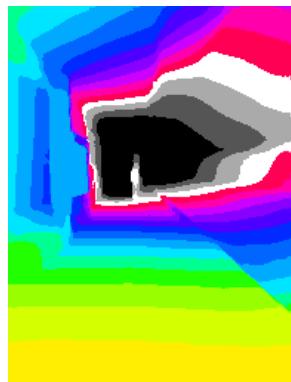
対策

- アルゴリズムの追加・改善

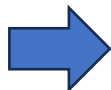
まとめ



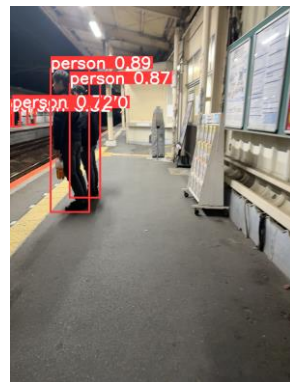
RGB画像



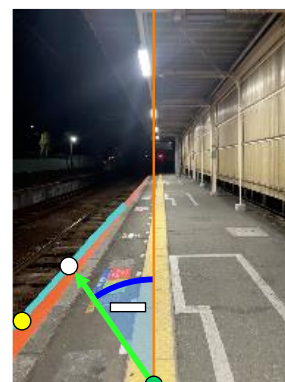
距離情報



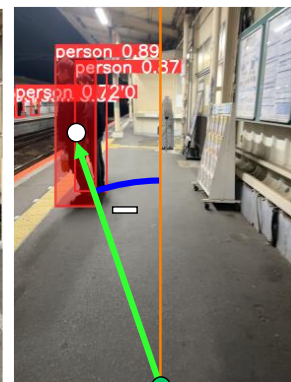
段差検出



物体検出



ホーム端
警告

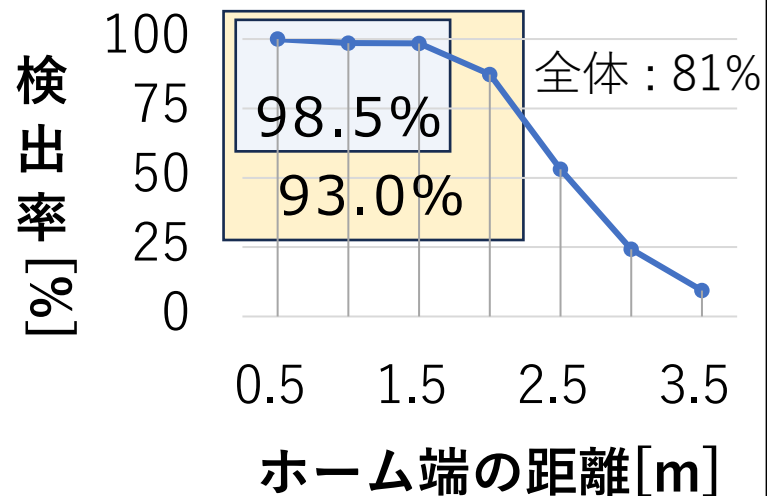


障害物
警告

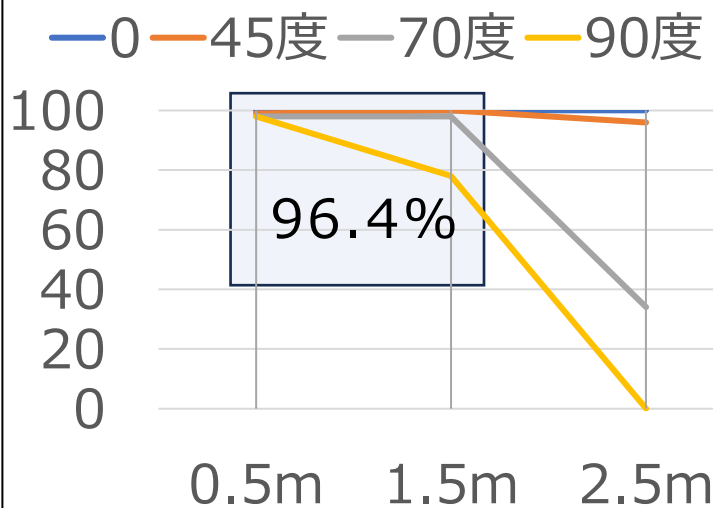


電車の
位置判断

ホーム端の距離と検出率



ホーム端までの距離と角度



今後の展望

- アルゴリズムの追加・改善
 - 特に階段検出
- 警告手段の検討
- 安全地帯への誘導

1.1 研究背景

背景 視覚障害者が駅ホームで転落する事故が繰り返し起きている

視覚障害者の駅ホームでの転落事故について

2012～2021年度
視覚障害者の転落事故
677件 [1]

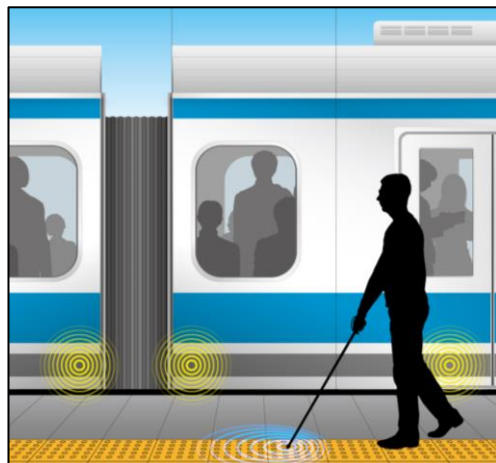
転落事故における
視覚障害者の割合
2.3% [1]

日本の視覚障害者の
人口比
0.246%

転落経験がある
視覚障害者の割合
36% [2,3]

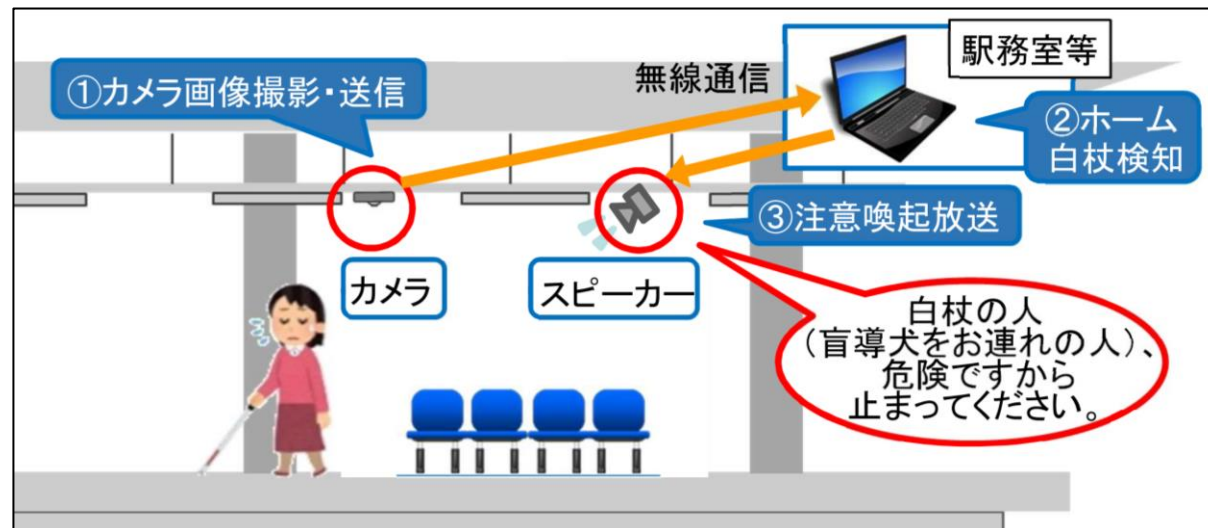
視覚障害を持つ多くの人たちが
駅ホームで危険な状況に遭遇している！

1.3 既存の研究とその課題



引用元

https://www.kyocera.co.jp/rdopeninnovation/beta-project/poc_va_system.html



引用元: <https://www.mlit.go.jp/common/001519720.pdf>

	スマート白杖 [5]	固定カメラでの 検知・報告 [4]	QRコード・RFID による誘導 [4]
ホーム端の把握	○	×	○
障害物の回避	△	×	△
人との衝突回避	△	×	×
設備	なし	あり	あり

2.2 LiDAR

LiDAR

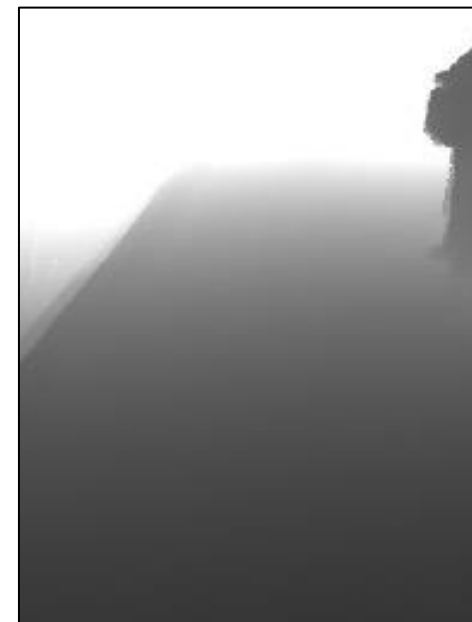
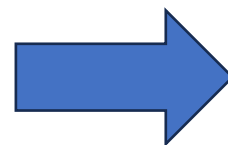
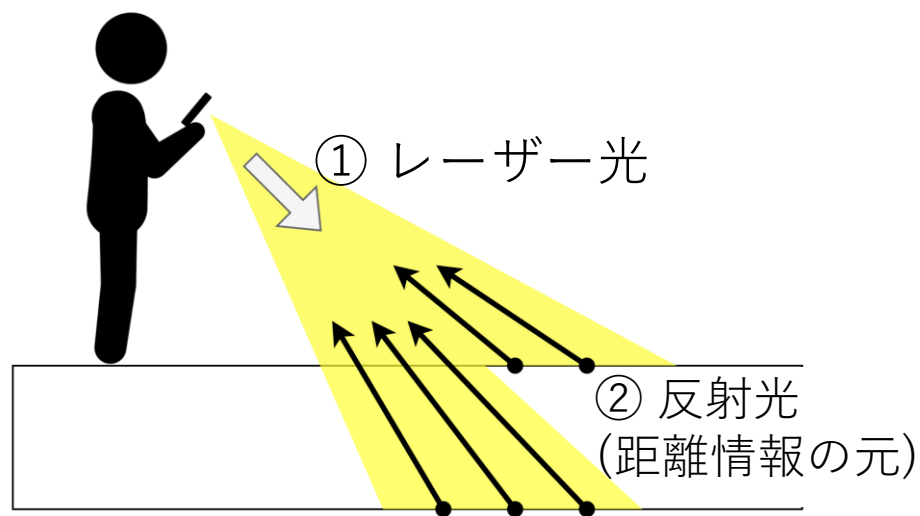


レーザー光によって距離を測定する技術

- ① レーザー光を照射
- ② 反射光を取得
- ③ 照射と反射光の取得時間の差から距離を取得

本研究での役割

- 段差の検出
- 障害物までの距離の取得



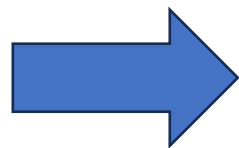
深度画像

2.3 物体検出

物体検出アルゴリズム

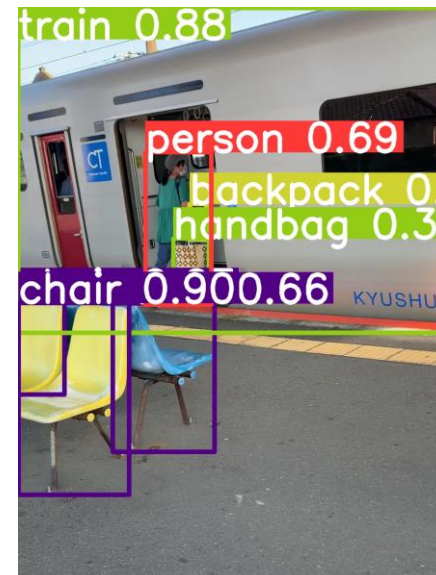
- YOLO
- 高速処理
- 複数の物体の同時検出

検出例



本研究での役割

- 障害物の種類と位置の取得



識別モデル

- YOLOは学習済みモデルを提供している
 - 検出可能な物体が80種類
- 線路や点字ブロック、自動販売機などには未対応
 - 追加学習で検出できるように対応させる

検出したい物体

- | | |
|----------|----------|
| • 人 | • ベンチ |
| • 電車 | • スーツケース |
| • 点字ブロック | • その他障害物 |
| • 線路 | |

※青文字は提供モデルで検出可能

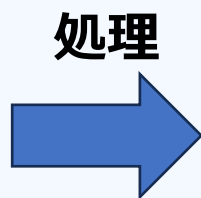
4.1 障害物の検出

アプローチ

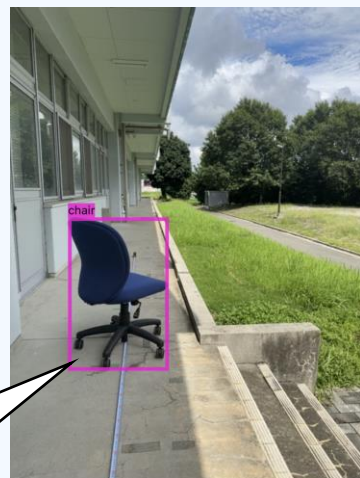
1. RGB画像に対して、物体検出処理を施し、障害物の種類と位置を取得
2. 物体検出による位置情報 と LiDARからの距離情報 を組み合わせて障害物までの距離を取得
3. 障害物までの距離と種類から危険の度合いを計算する



RGB画像



種類
位置



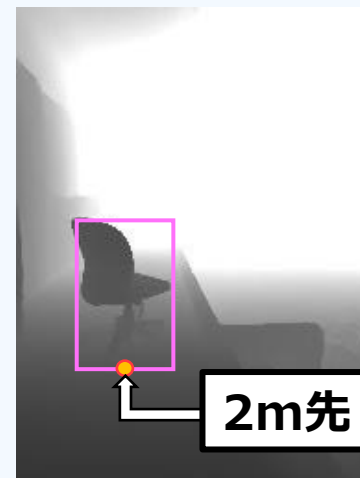
物体検出



深度画像



距離情報
(2m)



2m先

4.2 LiDARによる段差検出

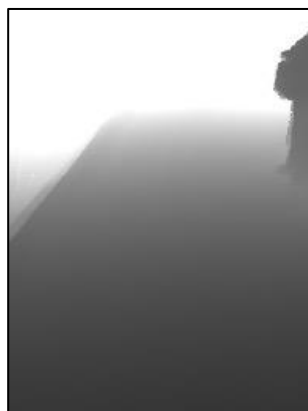
アプローチ

1. 段差部分は隣接部分との距離の差が大きい
2. この段差部分は深度画像にするとエッジ処理によって取得することができる
3. ホーム端や階段など直線的に伸びている段差を線分検出によって取得する
4. 段差部分の座標の距離情報から危険の度合いを計算する

段差検出アルゴリズム



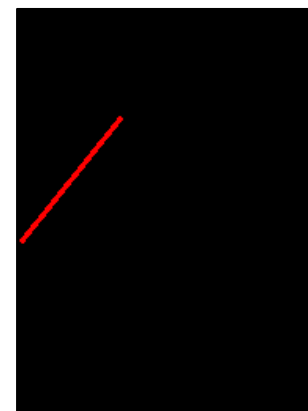
RGB画像



深度画像



エッジ抽出



線分検出



危険判断

5. 危険判断のアルゴリズム

基本

危険物や段差までの距離から判断

問題

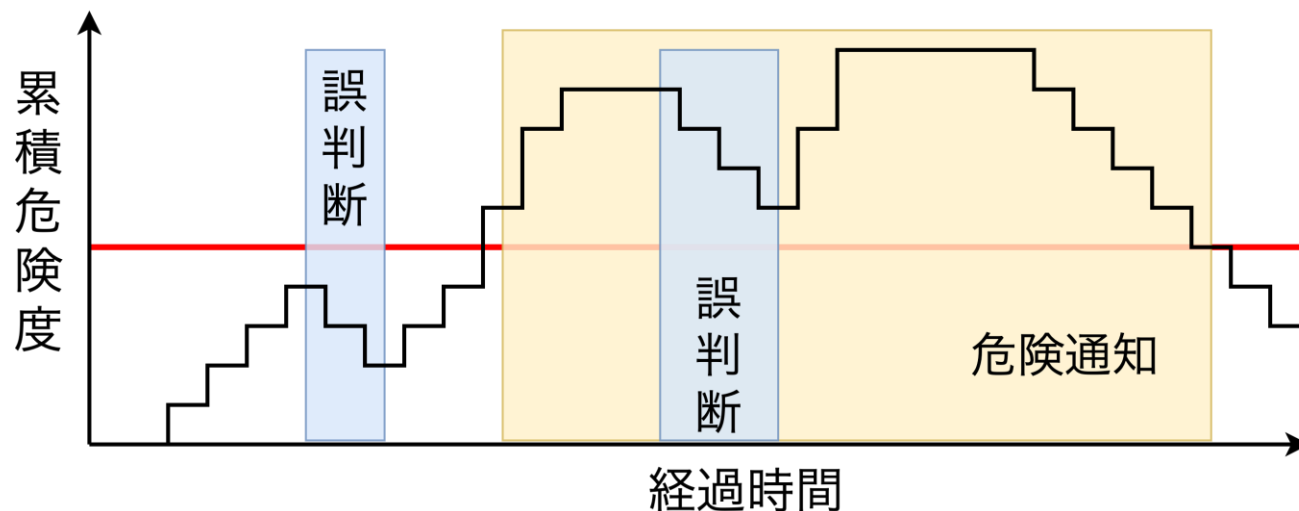
- ・ 歩行者などによる情報の遮蔽
- ・ 誤情報の取得

対策

- ・ 危険度に対して累積値を取り入れる
 - ・ 累積値が一定値を超えると通知を行う
- ⇒ 情報の欠損や誤情報による判断率の低下を防ぐことができる

累積危険度の例

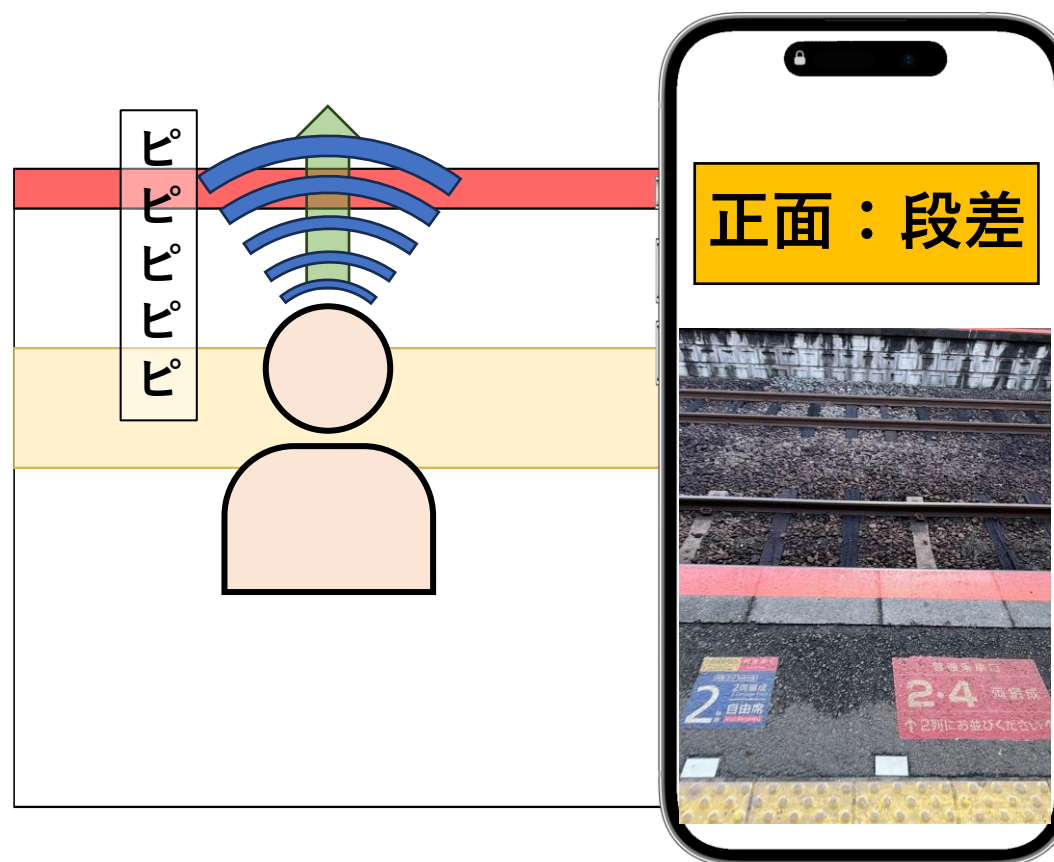
- ・ 危険度が高い場合 ⇒ 累積値 + 2
- ・ 危険度が低い場合 ⇒ 累積値 + 1
- ・ 安全である場合 ⇒ 累積値 - 1



6. 危険の報告方法

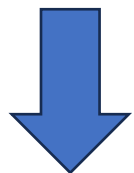
要約

- 環境音を聞ける音声出力機器を使う（骨伝導イヤホンなど）
- 危険物の方向： 左右の音の出力によって方向を伝える
- 危険物の距離： 音程や通知の速度で危険度を伝える

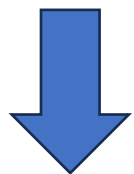


7. iOSアプリ

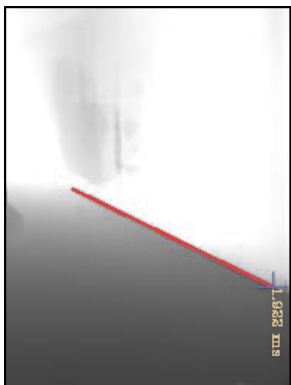
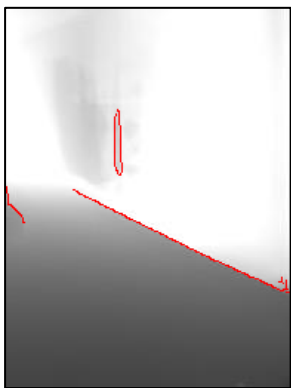
LiDARの距離情報から
深度画像を作成



エッジ検出処理



直線の段差の検出
段差までの最短距離

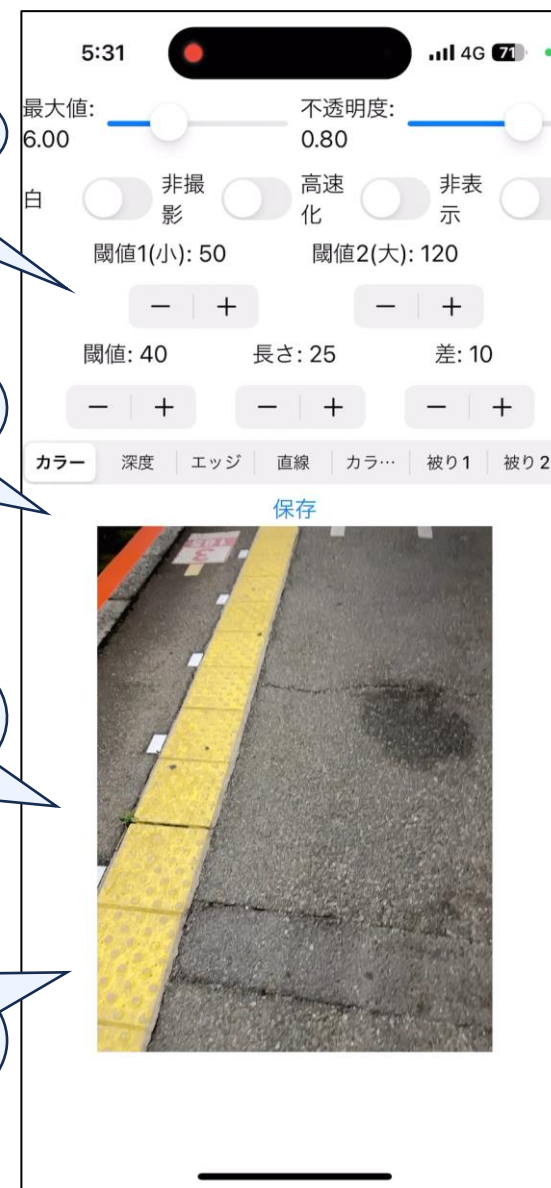


画像の保存

保存画像の
再処理 & 再表示

処理に関する
値の変更

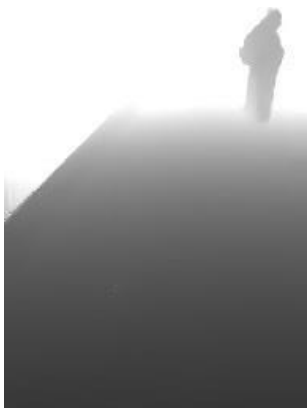
段差の検出
&
距離取得



8. 精度検証

OpenCVを使った場合、iOSアプリ(Swift)とPC(Python)で実行結果に違いはなかった
そのため、検証がしやすいPythonで精度検証を行った

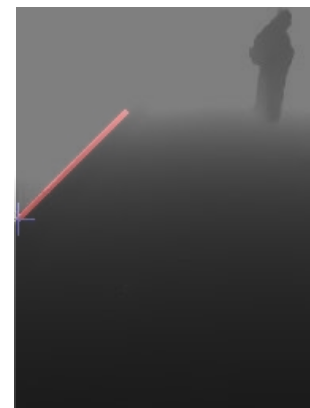
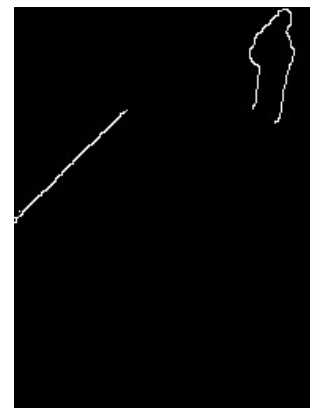
元データ



Pythonでの実行結果

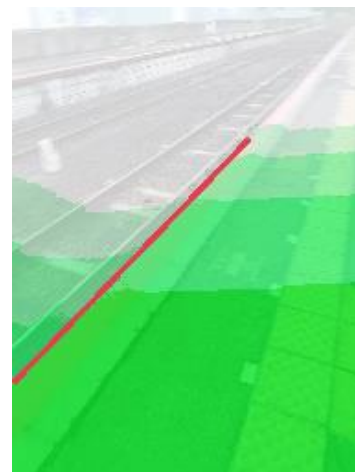


iOSアプリでの実行結果



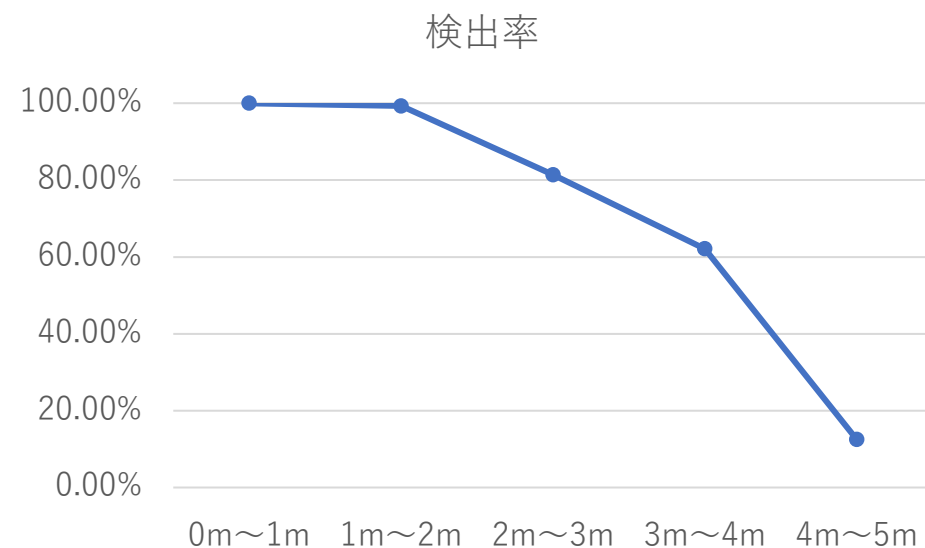
検証手順

- ・ 駅ホームにて利用客が少ない時間帯に10分間(4350枚)撮影
- ・ ホーム端が写った画像をランダムに選び、段差検出処理を実行
- ・ 1m単位の深度画像に検出した段差（直線）を表示
- ・ 段差検出の有無と段差までの最短距離を集計



8. 精度検証

線分までの距離	検出	未検出	検出率
0m～1m	17回	0回	100.0 %
1m～2m	423回	3回	99.3 %
2m～3m	686回	157回	81.4 %
3m～4m	113回	69回	62.1 %
4m～5m	4回	28回	12.5 %
合計	1243回	257回	82.9 %



結果から分かること

- **全体の検出率が 82.9 %** とかなりの確率で検出ができています
- **距離が離れるほど検出率が悪くなる**
- **3m以下での検出率がすべて 80%を超えている**ため、高確率で危険を事前に伝えられる
- **2m以下では約99%の確率で検出ができています**ため
転落の恐れが高い場合にはほぼ確実に危険を伝えられる

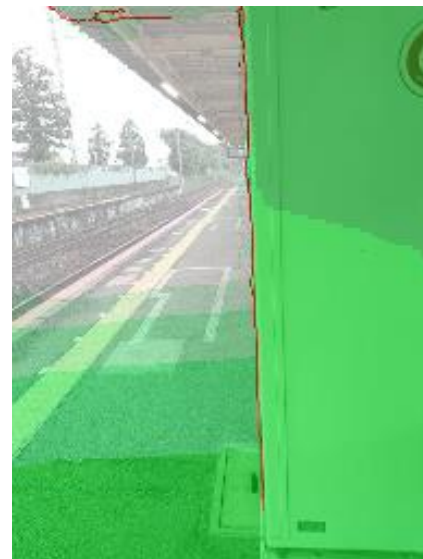
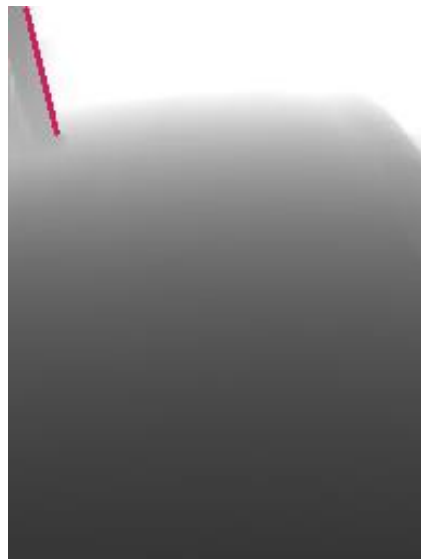
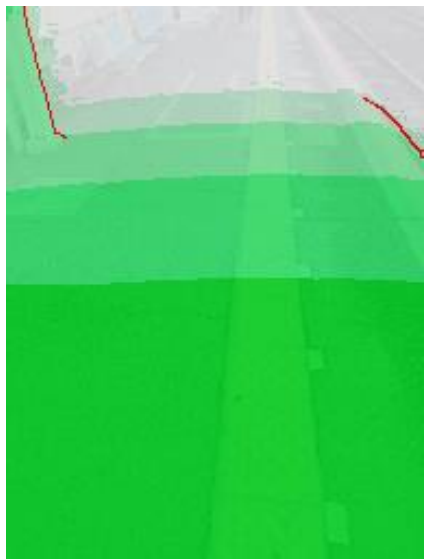
9. 考察

遠距離になると検出率が悪くなる原因

1. 実際の距離は同じでも、遠くに行くほど取得される画像では短くなる
→ 検出する線分が短くなるため、線分として検出されていない
2. 有効距離を超えてしまい、深度画像に段差が写っていない



3m以内であれば80%以上、2m以内であれば99%以上の精度であるため
現在は、この精度悪化は問題ないと判断



10. 問題点と今後の課題

問題点

- 電車がどちらの線路に来ているかを判断するためのアルゴリズムが不確定
 - 勘違いによる転落を防ぐ事ができない
 - 安全性などを考慮
- 障害物による不要な線分が除去できていない
 - 重なった検出線
 - 人の手足
 - 柱や柵
 - 障害物(スーツケースや椅子、自動販売機、ゴミ箱など)

今後の課題

問題点の解決

物体検出

危険判断
アルゴリズム

危険通知

11. まとめ

画像の保存

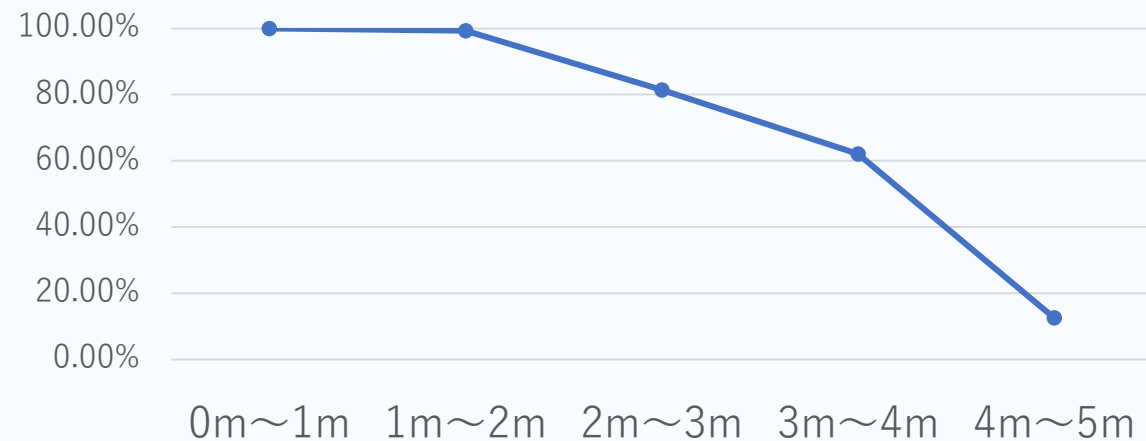
保存画像の
再処理&際表示

処理に関する
値の変更

段差の検出
&
距離取得



検出率



今後の課題

- 物体検出を用いた障害物の検知と距離取得の実装
- 危険判断アルゴリズムと危険通知の実装
- 段差検出の精度向上と誤判断率の低下
- 実際に視覚障害者の方々に使っていただき、改善

12. 参考文献

- [1] 国土交通省鉄道局: “鉄軌道輸送の安全に関わる 情報(令和 3 年度)”
- [2] 毎日新聞・日本盲人会連合:
“視覚障害者の鉄道 駅に関するアンケート調査 調査結果”
- [3] 日本視覚障害者団体連合: “転落事故に関するア ンケート調査結果”
- [4] 国土交通省鉄道局:
“新技術等を活用した駅ホームにおける視覚障害者の安全対策検討会 第12回検討会”
- [5] 日経クロストrend: “視覚障害者の「白杖」がスマート化 ホーム転落を防ぐ”