

# 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

## Development of a Hazard Avoidance Smartphone App for Visually Impaired at Train Platform using LiDAR and Object Detection

白井 達也  
電子情報システム工学専攻

**Abstract:** Since there have been repeated accidents involving the visually impaired falling from train platforms, this study tried to develop a smartphone app specifically for use at railway stations. By acquiring distance information with a LiDAR implemented in smartphone, it will be easier to detect the height difference between station platforms and rail tracks, and the detection accuracy is expected to be improved. In addition, I consider that it can measure the distance of objects and steps and can provide detail information about the situation. In this paper, I first analyze dangerous situations for visually impaired persons. Then, I explain the detection method to steps and objects detection and propose algorithms that can judge dangerous situations. Finally, I validate and confirm the effectiveness of the step detection accuracy and the processing speed of the app.

**Keywords:** Visually impaired, Object detection, Smartphone app, Hazard avoidance, LiDAR

### 1. まえがき

視覚障害者が駅ホームから転落する事故が繰り返し発生している。視覚障害者の転落事故は 2013 年度から 2022 年度までの 10 年間で 656 件発生しており、新型コロナウイルスの影響で件数は減ったものの、全体の転落事故における視覚障害者の割合は毎年 2.3%程度である[1]。また、日本総人口に対する視覚障害者の人口比は、約 0.25%[2, 3]であるのに対して、転落事故における視覚障害者の割合が約 2.3%であることから、視覚障害者の転落リスクが高いことがわかる。

そのような背景から、先行研究では、視覚障害者の周辺機器を強化する研究と駅ホームを改善する研究が行われている。前者では、代表的な研究として、スマート白杖[4]がある。後者では、QR コードによる誘導や監視カメラによる警告がある[5]。著者らはこれまでに、視覚障害者が歩行する際の支援として、RGB 画像と深度画像から段差を検出する手法の提案や、機械学習による物体検出を応用して RGB 画像から障害物を検知して利用者に警告するスマホアプリの開発を行ってきた。[6]

本研究では、これまでの物体検出手法に加え、LiDAR からの距離情報を活用する。スマートフォンのカメラから取得される RGB 画像を利用した場合、線路や点字ブロックなどを検出することによって、利用者の位置情報を予測することができる。しかし、環境光の影響などによって安定した予測をすることは困難であった。それに対して、LiDAR から得られる距離情報は、環境光の影響を受けないため、より安定した高精度な距離情報を取得できる。また、距離情報を使うことで、危険な状況までの距離をより正確に伝えることが可能になる。

本論文では、まず LiDAR から取得できる距離情報を用いて危険な状況を検知する手法について説明する。その後 LiDAR を搭載している Apple 社の iPhone 14 Pro で動作するアプリとして提案手法を実装し、動作検証を行い、アプリの有効性と LiDAR からの距離情報を活用する有効性を示す。

### 2. 視覚障害者の駅ホームでの現状と分析

前述のように全体の転落事故における視覚障害者の割合は毎年 2.3%程度である。またアンケート調査[5, 7, 8, 9]によると視覚障害者の約 30%が転落を経験していることが分かる(表 1)。また「転落しそうになった」と回答した人も含めると約 60%になる[7, 9]。アンケートから転落原因として以下のことが挙げられている。

- ・ 何らかの原因で方向感覚を失った
- ・ ホーム端に気付かなかった
- ・ 勘違いをして踏み外した

以上のことから、図 1 に示すように以下の 4 つの点を危険な状況として定義した。

- ① ホーム端に気付かない
- ② 電車の位置の勘違いによる転落
- ③ 障害物の回避・衝突による方向感覚の喪失
- ④ 階段からの転落

本研究では、①から③の危険を判断できるシステムを提案する。

表 1 転落に関するアンケート調査

文献	母数[人]	転落数[人]	割合[%]
日盲連 [7]	252	92	36.5
毎日新聞 [8]	222	70	31.5
論文 [9]	297	108	36.4
国土交通省[5]	303	109	36.0

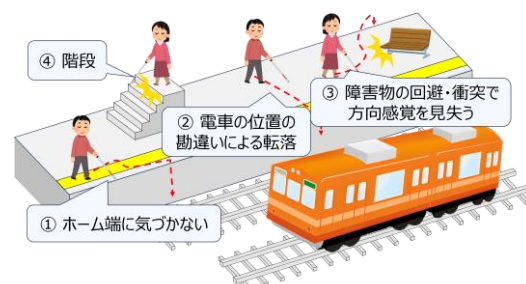


図 1 駅ホームにおける視覚障害者の危険な状況

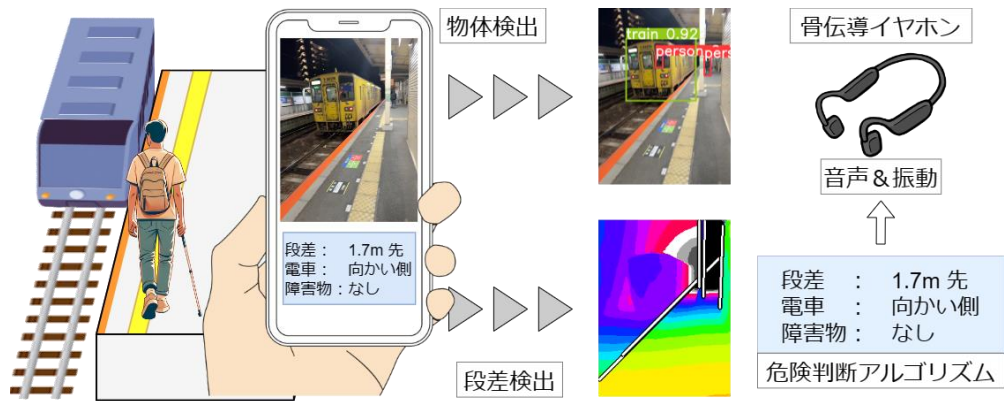


図 2 システムの全体像

### 3. システム全体像

本研究で開発するアプリケーション（以下、本アプリ）に関して説明する。本アプリでは、使用者が手に持ち、自ら状況を把握したい場所に対して、スマホを向けることで、障害物や段差がないかという状況を調べることができるよう支援することを目的としている。

システムは、障害物の種類と位置を把握するための機械学習による物体検出と LiDAR から取得される距離情報を組み合わせたものであり、それらの情報から段差や障害物までの距離や方向を判断し、使用者にアラート音や振動を使って感覚的に知らせる。アラート音の出力デバイスは周りの音が聞こえるものを使用することを想定している。図 2 に本アプリのシステム全体像を示す。

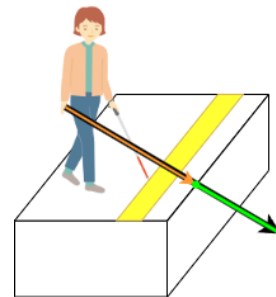


図 3 駅ホームの段差

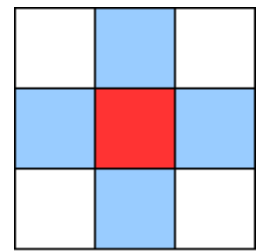


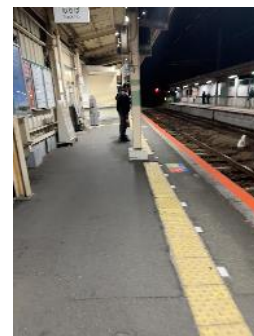
図 4 段差の決め方

### 4. LiDAR による段差検出

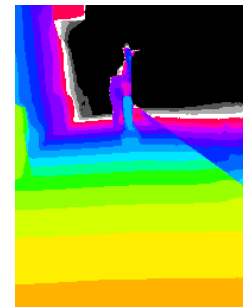
駅ホームからの転落を防ぐためには、ホーム端を正しく検出する必要がある。図 3 に示すようにホーム面(オレンジの矢印)と線路面(黒の矢印)には大きな段差(緑の線分)があるので、距離の差から段差を求めることが可能である。そこで LiDAR の距離情報を用いることで段差検出が容易になると考える。

#### 4.1 LiDAR について

LiDAR とは、レーザー光を照射し、その反射光を取得することで、照射時間と反射光の取得時間の差から距離を計算する距離測定技術である。LiDAR は、レーザー光を使用するため、環境に左右されにくく、安定して高精度な情報を取得することができる。ただし、ガラスや鏡などは精度が悪くなる。図 5(a)はカメラから取得される画像であり、同位置から LiDAR で取得した距離情報を用いて 50cm ごとに色を変えた深度画像を生成すると同図(b)のようになる。



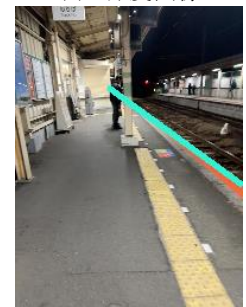
(a) RGB 画像



(b) 深度画像



(c) エッジ画像



(d) 線分検出

図 5 段差検出のプロセス

#### 4.2 エッジ画像

まず段差を検出するにあたり、2つの値を設定する。一つは段差を検出するための距離（有効距離）である。iPhone に搭載された LiDAR の有効照射距離は 5m で、それ以降は精度が悪くなる傾向がある。また遠くにある柱や屋根、反対のホームなども検出してしまう。そのため、段差を検出する距離を制限する。本研究では 7m としている。もう一つは段差として判定するための距離の差（段差距離）である。これは駅ホームと線路の高さが最低で 760mm であり、誤差を考慮して 500mm に設定している。どちらもアプリ内で変更可能である。

LiDAR からの情報は、等倍のカメラと同じ画角で「192×256」の配列に格納された少数値である。距離情報と同じサイズの行列（エッジ行列）を作成し、各ピク

セル（図 4 の赤）とその上下左右（図 4 の青）の距離情報の差を求める。4つの値の最大値が段差距離を超えている、かつ、中心座標（図 4 の赤）が有効距離内である場合、その中心座標を段差と判断し、エッジ行列を 1(白)とする。そうでない場合は、0(黒)とする。最後にエッジ行列を二値化画像に変換してエッジ画像を生成する。図 5(c)に生成されたエッジ画像を示す。

#### 4.3 線分検出

本研究において危険とする段差は、駅ホーム端の段差や階段など直線の段差である。そこで本研究では、

Hough 変換を用いてエッジ画像から線分を検出する。OpenCV の HoughLinesP 関数は、直線的に伸びているエッジ部分のみを線分(始点の座標と終点の座標)として取得できるので、人の輪郭のような曲線的なエッジは排除することができる。HoughLinesP 関数は引数として、線分判定のための閾値、線分と判断するための最短の長さ、1つの線分としてまとめる間隔、などを指定する必要がある。検出した線分を RGB 画像に重ねた画像を図 5(d)に示す。

## 5. LiDAR による段差検出

本研究では、障害物との距離を求めて、衝突防止や回避誘導を行う。そのため、障害物検出には、「カメラから得られる RGB 画像を用いて行う物体検出処理」と「LiDAR から得られる距離情報」を組み合わせる。

### 5.1 CoreML

本研究では、Apple 社製品の iPhone Pro 14 を使ってアプリ開発・検証を行う。物体検出には CoreML と呼ばれる Apple が開発している機械学習フレームワークを使用する。このフレームワークは、デバイス上で処理が行われるため、インターネットが繋がっていなくても物体検出を利用することができるようになる。また、CoreML は、Apple 社のデバイスに対して最適化されているため、高速で効率的に動作する。

### 5.2 物体検出アルゴリズム YOLO

本研究では、物体検出アルゴリズムとして YOLO を扱う。執筆時点の YOLO の最新バージョンは v8 である。YOLO は、物体の種類と位置を同時に予測することで、処理を高速化しており、リアルタイム処理に適している。デメリットとして、小さい物体が検出困難であることと他の物体検出アルゴリズムに比べてやや精度が低いことが問題としてある。本研究では、小さい物体は検出する重要性が低く、精度問題も累積危険値(後述)を導入することで影響を緩和することができると考えられる。

### 5.3 検出モデルと検出物

検出できる物体は利用する物体検出モデルによって決まる。本研究では、YOLO の開発元である Ultralytics[10]が提供している学習済みモデルを使用する。このモデルは、車や動物、家具など 80 種類の物体が検出可能であり、その中には、電車や人、椅子やスーツケースなど駅ホームで重要となるものも含む。しかし実際には、駅ホームに適した物体のみを検出できるモデルを作成し、使用することが望ましい。

## 6. 警告について

### 6.1 累積危険値に関して

LiDAR からの距離情報や物体検出の結果は、稀に誤った情報を取得する場合がある。また危険な状況が画面に写っていないくても危険な状況が続いている恐れもあり、そのフレームだけで危険かどうかを判断するのは誤った情報を提供し、返って危険な状況にさせる可能性がある。逆に数フレームだけ危険な状況が写っただけという場合もある。この場合は警告を鳴らすと余計な心配をさせてしまう。

そこで本研究では、上限と下限を設定した累積危険値というものを導入する。距離に応じて加算する値を変更する。この値が決められた閾値まで達している状態のときのみ、危険の警告を行う。これによって、誤った情報

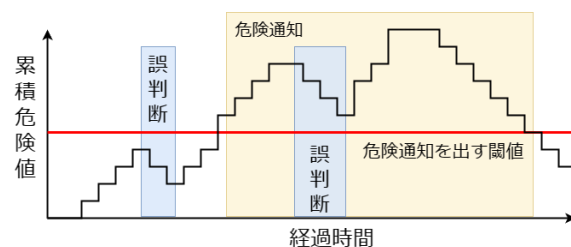


図 6 累積危険値の例

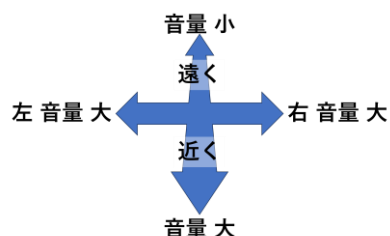


図 7 アラートの鳴らし方



(a) 左側 距離 2.1m (b) 中央 距離 1.5m

図 8 段差検出の例

を取得した際の影響や数フレームでの映り込みによる誤警報を少なくすることができると考えられる。増減値は以下のように設定している。段差は、1m 以内は+3、1.5m 以内は+2、2m 以内の段差は 0、2m 以上は-1 としている。障害物は、1m 以内であれば+2、1.5m 以内であれば+1 としている。それ以外は-2 のように増減する。図 6 に累積危険値の変動を表したグラフの例を示す。

### 6.2 警告の伝達

本研究では、ステレオオーディオ再生かつ外音取り込みが可能である骨伝導イヤホンなどの音声出力機器を使用することを想定している。本来であれば、距離も伝えたいところだが、読み上げていると時間がかかるため、音の大きさや鳴らす回数などで危険な状況(障害物や段差)までの距離を伝え、左右の音の出力の差で方向を伝える。その他にも振動を使った警告も行っている。アラートの鳴らし方を図 7 に示す。

## 7. 危険判断アルゴリズム

### 7.1 駅のホーム端の検出

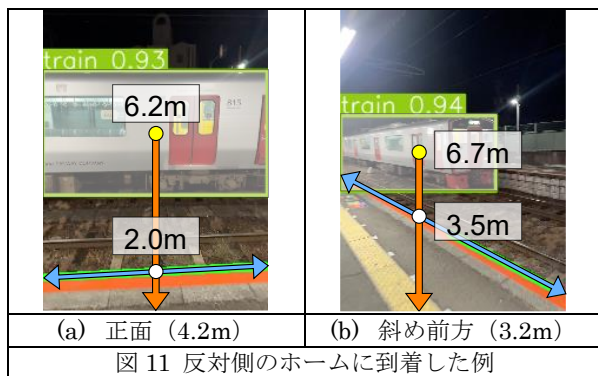
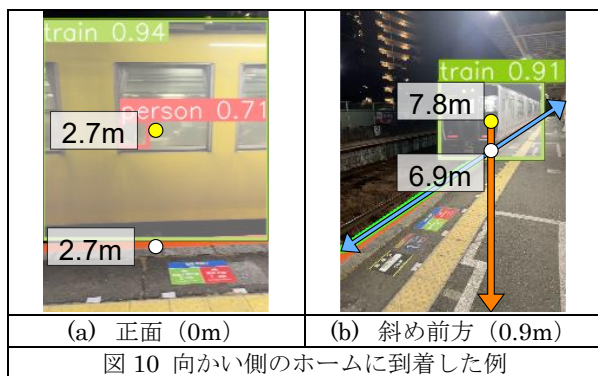
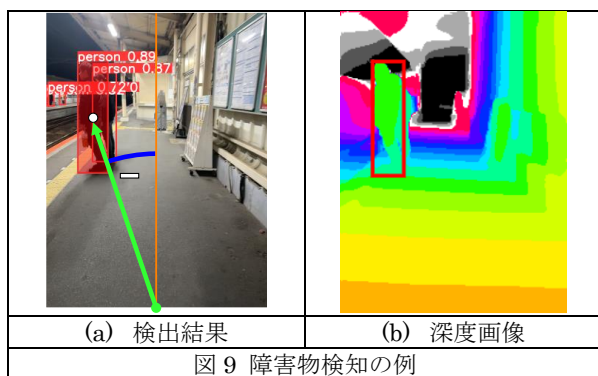
検出された線分(段差)の座標と距離情報を合わせて、段差までの距離と方向を求める。

距離： 検出された線分の点から一番近い距離である点を段差までの距離とする。

方向： 画面の最下部中央を原点、その上の垂線を角度 0 として、最短距離の線分の中点までの角度を方向とする。角度は右を正、左を負とする。

段差検出の例を図 8 に示す。





## 7.2 障害物の衝突・回避

物体検出で障害物の種類と位置を取得し、同座標の距離情報と照らし合わせて、障害物までの距離と方向を求める。

距離： 検出された障害物の中から、一番近い物体の中心座標を物体までの距離とする。

方向： 画面の最下部中央を原点、その上の垂線を角度 0 として、最短距離の障害物の中心座標までの角度を方向とし、右を正、左を負とする。

図 9 に検出結果とその際の深度画像の例を示す。図 9(a)では複数の人間が検出されており、これらの中心座標を求めて、図 9(b)から距離情報を求める。

## 7.3 電車の位置の判断

電車が向かい側に来ている際に段差を検出するかは曖昧であり、段差がなければ向かい側に到着したと判断することは出来ない。そこで物体検出による電車の有無と段差検出の有無を組み合わせ、電車がどちらのホームに来ているかを判断させる。

まず、電車が 3m 以内にあれば、向かい側に来ていると考える。3m 以上先に電車を検出した場合、一番近い段差(線分)の画面内に限定した延長線と検出された電車

表 2 段差検出の精度検証 1(ランダム抽出)

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	255 枚	251 枚	98 %
1.5 ~ 2.0	321 枚	280 枚	87 %
2.0 ~ 2.5	96 枚	51 枚	53 %
2.5 ~ 3.0	33 枚	8 枚	24 %
3.0 ~ 3.5	32 枚	3 枚	10 %
3.5 ~ 4.0	2 枚	0 枚	0 %
4.0 ~ 4.5	3 枚	0 枚	0 %
4.5 ~ 5.0	1 枚	0 枚	0 %
合計	814 枚	663 枚	81 %

ホーム端の距離と検出率

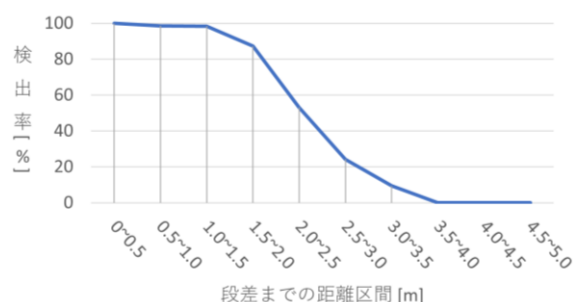


図 12 段差の検出率のグラフ

表 3 段差検出の精度検証 2(パターン分け)

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%

の枠の中心座標から下ろした垂線の交点を求める。その交点と電車の中心点の距離の差を求め、これが閾値以上であれば、反対側のホームであると判断する。それぞれのホームでのアルゴリズムでの判断を図 10 と図 11 に示す。各図の右の値は距離の差である。

## 8. 検証

### 8.1 段差検出の精度 検証 1

段差検出アルゴリズムは、LiDAR からの距離情報と OpenCV を使っている。そのため、精度検証では、確認の都合上、PC の Python で検証を行った。Python と iPhone アプリ開発言語である Swift で OpenCV を使った際の線分検出に差が無いことを確認している。

検証方法は、駅ホームで自然に歩行をし、データを収集して、それらのデータに対して段差検出アルゴリズムを施す。そして、表示されているホーム端までの最短距離と検出の有無を調べた。結果は表 2 と図 12 に示す。

2.0m 付近までは精度がかなり良いことが分かる。また検証した画像の中で誤検出は、約 33%であった。

### 8.2 段差検出の精度 検証 2

検証 1 を行っている最中にいくつかのパターンによって検出率に偏りがあると感じた。そこで、次に検出するホームまでの最短距離と利用者が向いている方向を固定し、検証した。母数は各データ 100 枚である。検証結果を表 3 に示す。

表 3 各処理と処理速度

処理名	最小値	平均値	最大値
RGB 画像	4ms	7ms	10sm
距離情報	10ms	15ms	28ms
線分検出	14ms	22ms	40ms
物体検出	20ms	30ms	46ms
アルゴリズム	0.02ms	3ms	90ms
1 フレーム	66ms	100ms	230ms
FPS	4FPS	10FPS	60FPS

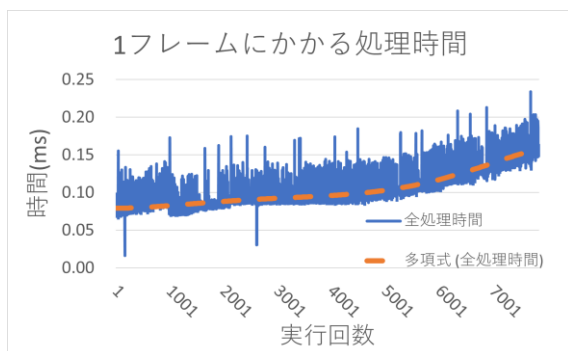


図 14 1 フレームにかかる処理時間（14 分間起動）

結果から、角度が小さいほど、また段差まで近いほど検出率が上がるのが分かる。また 2.5m の 70 度と 90 で低精度になっている。70 度の場合、ホーム端は約 4.0m、画面上でホーム端に隣接する線路面は約 8.5m であるため、LiDAR の有効照射距離を超えたことで、精度が悪化したことが原因であると考えられる。

点字ブロックの位置は、国土交通省により、ホーム端から 80cm～100cm のところに設置するようになっている[11]。そのため、100cm より近い場合は、より強い警告を出す必要がある。検証 1、検証 2 どちらも 1.5m までは高精度で検出できており、段差検出アルゴリズムの有効性が高いといえるだろう。

### 8.3 アプリの処理速度

1 フレームかかる時間と各処理に分けて、処理時間を計測した。表 3 に各処理時間に対する代表値を示し、全処理の処理時間をグラフにしたものを図 14 に示す。記録はしていないが、本アプリを長時間使用していると端末から熱が発生することがわかった。アルゴリズムに関しては非常に軽量であり、今後さらにアルゴリズムを追加しても処理速度への影響は少ないだろう。

現在、累積危険値の閾値は 20 にしており、最も危険な状況で累積危険値は+3 である。そのため、正しく検出されれば、平均で 0.7 秒以内にアラームを鳴らすことが出来るため、処理速度に関しては問題ないと考えられる。

この検証は、アプリ開発時点での処理で行っており、最適化や軽量化を行っていないため、実際に使用する段階ではこれより軽くなると予想される。

## 9. 考察

### 9.1 未検出の段差に関して

未検出の原因は大きく分けて 2 つあった。

1 つ目は線分検出の閾値の調整ミスによるものである。図 15(a)では、段差のエッジは検出できているものの、線が短く検出されていない。また図 15(b)では、直線のように見えるが直線の度合いが閾値に達しておらず、線分として検出されていない。これらを検出するために線分検出の条件を緩くすると、誤検出が多くなるため、今

後の調整やアルゴリズムなどでの対策が必要ある。

2 つ目は LiDAR からの距離情報の精度によるものである。図 16 の青色の直線は線分ではなく、実際のホーム端を表している。図 16(a)では線路面を正しく測定できず、平面のようになっている。図 16(b)では、段差部分の一部が正しく測定できず、直線上になっていないため、直線を検出することができていない。これらは急激に距離が変わる箇所などは、LiDAR からの距離情報の精度が低くなる傾向があり、このことが原因だと考えられる。またホーム端に塗られたオレンジ色のラインや点字ブロックが局所的に精度を悪化する問題があり、図 16(b)に関しては、この問題も重なって発生していると考えられる。これらの問題については対策することが難しく、LiDAR の精度向上を期待する。

しかし、遠くの段差に関しては危険度が低く、重要である近距離での段差は高精度で検出できているため、視覚障害者の転落防止には有効であるといえる。

### 9.2 誤検出の段差に関して

誤検出の例を図 17 に示す。誤検出が発生する理由は、線路や点字ブロックのような LiDAR の精度によるものと物体や反対のホーム、柱などのアルゴリズムで除外出来ていないものがあつた。本研究では、近距離の段差かつ最短距離の段差を危険として判断するため、反対側のホームや線路など段差より遠い誤検出は、通知警告には、あまり影響しないと考えられる。しかし、柱や電車などは近距離であり、誤検出と未検出が重なる場合もある。そういった場合は、誤った危険レベルだと判断されてしまう。LiDAR の精度では、線路部分より上部が全て 10m 以上先と判断されているケースがあつた。また点字ブロックのみ至近距離で判断されるといったケースもあつた。どちらも図 4 の各点で上限と下限を設けることで対策できると考える。アルゴリズムによる誤検出は、物体検出の結果を利用したり線分付近での最大値と最小値を比べたりなど、更にアルゴリズムを追加することで対策できると考える。

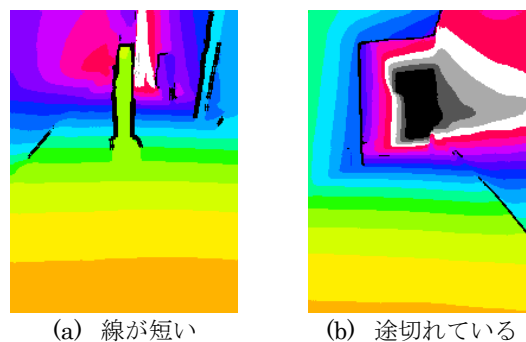


図 15 未検出の例 1

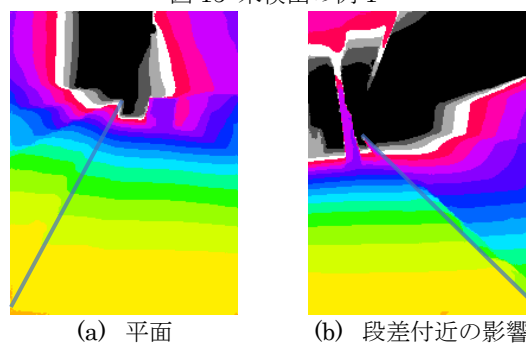


図 16 未検出の例 2（段差部分表示）

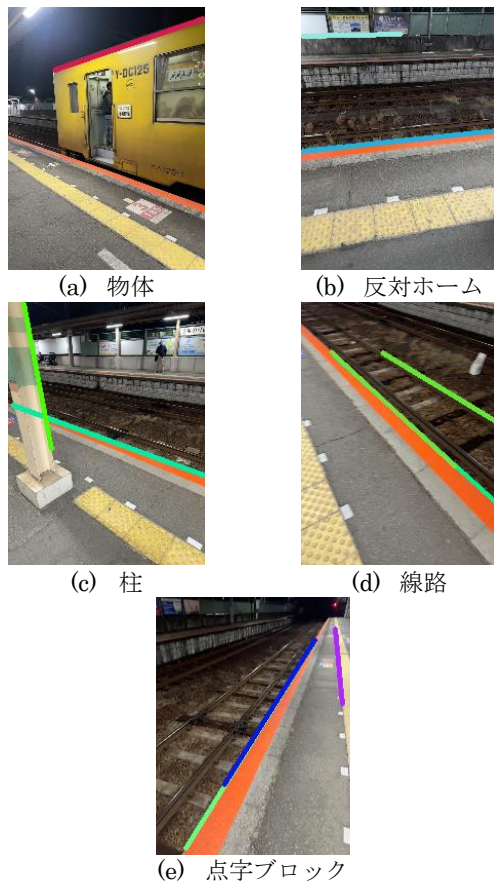


図 17 誤検出の例

## 10. むすび

### 10.1 まとめ

本研究では、LiDAR と物体検出を組み合わせることで駅ホームにおける視覚障害者の歩行支援アプリの開発を行った。普段時の使用方法で行った検証 1 では、ホーム端の検出率が全体で 81%、特に 2m 以内であれば約 92.9%、1.5m 以内であれば 98.5%となっており、LiDAR による段差検出の有効性を確認した。危険判断のためのアルゴリズムでは、駅ホームでの 3 つの危険な状況を判断するように実装した。距離情報を追加することで距離に応じて危険度を設定することが可能になり、累積危険値を導入することで誤情報や短期間での映り込みによる誤った判断や警報を軽減した。本アプリは 1 フレームにかかる処理速度は平均で 0.1 秒であり、最も危険な状況の場合、正しく検出されれば平均で 0.7 秒以内に警告することができる。そのため、本アプリの処理速度は高速ではないが、危険判断や警告には問題ないといえるだろう。しかし、実際に視覚障害者に利用していただき、検証することは出来なかった。検証までに改善すべきことを次の節で述べる。

### 10.2 今後の展望

本研究では、Ultralytics が提供している学習済みモデルを使用した。しかし、駅ホームには自動販売機などがあり、これらに対応していない。そのため、駅ホームでの障害物を検知できるようなモデルを作成し、使用することが望ましい。段差や障害物の検出とそれまでの距離の計算は実装できたが、アルゴリズムや警告の方法、UI などは最適解とはいえず、現段階では、誤検出や未検出も多く、これらはアルゴリズムで改善できる見込みがある。実際に視覚障害者の方に利用してもらい、意見を反

映していくことが重要である。また執筆時点では、階段を検出するためのアルゴリズムが実装できていない。階段からの転落は大怪我につながるため、転落を回避できるように支援することは重要である。深度データによる階段検出は研究として以前から行われている[12]ため、それらの研究を参考にアプリに実装する必要がある。

また本研究では、危険の検出に焦点を当てたが、視覚障害者を安全な場所に誘導することも大切である。段差部分と RGB 画像を組み合わせると、点字ブロックなどに誘導できるようにすると更に視覚障害者を支援できるようになる。また RGB 画像との組み合わせは段差検出の精度向上に役立つと考える。

## 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 4 年度）”，2023。  
<https://www.mlit.go.jp/tetudo/content/001712991.pdf>
- [2] 厚生労働省，“平成 28 年生活のしづらさなどに関する調査（全国在宅障害児・者等実態調査）”，2018。  
[https://www.mhlw.go.jp/toukei/list/seikatsu\\_c housa\\_b\\_h28.html](https://www.mhlw.go.jp/toukei/list/seikatsu_c housa_b_h28.html)
- [3] 総務省統計局，“人口推計（2022 年（令和 4 年）10 月 1 日現在） - 全国：年齢（各歳），男女別人口・都道府県：年齢（5 歳階級），男女別人口 - ”，2023。  
<https://www.stat.go.jp/data/jinsui/2022np/ind ex.html>
- [4] 中島祐樹，平山正治，“駅ホームにおける視覚障害者の転落防止システムに関する研究”．情報科学技術フォーラム講演論文集．2008，7（3），p. 583-584.
- [5] 国土交通省，“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について ～中間報告～”，2021 年。  
<https://www.mlit.go.jp/common/001426521.pdf>
- [6] 田口一精，渡邊賢太郎，島川学，清田公保，“機械学習を用いた視覚障害者のための障害物検出スマートフォンアプリ”，第 19 回日本知能情報フアジィ学会九州支部学術講演会，A104，pp. 15-18（2017）
- [7] 日本盲人会連合，“転落事故に関するアンケート調査結果”，2011。  
<http://nichimou.org/wp-content/uploads/2014/02/1105tenrakujukioan.pdf>
- [8] 毎日新聞東京本社社会部，日本盲人会連合，“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”，2017  
<http://nichimou.org/wp-content/uploads/2017/03/mainichiannke-to.pdf>
- [9] 辻本陽琢，佐々木大輔，横飛雅俊，森信哉，“視覚障害者の駅ホームからの転落原因の体系的整理に関する一考察”，土木学会論文集 F6（安全問題），Vol. 77，No. 2，p. I\_1-I\_11，2021.
- [10] Ultralytics，<https://www.ultralytics.com/>（Accessed 2024. 01. 16）
- [11] 国土交通省，“公共交通機関の旅客施設に関する移動等円滑化整備ガイドライン バリアフリー整備ガイドライン 旅客施設編”，2022  
<https://www.mlit.go.jp/sogoseisaku/barrierfre e/content/001475260.pdf>
- [12] Soichiro Murakami, Manabu Shimakawa, Kimiyasu Kiyota, Takashi Kato: “Study on Stairs Detection using RGB-Depth Images”, SCI&ISIS2014, pp1186-1191, 2014.