

セーフティナビゲーションのためのスマートフォンによる障害物検出の研究

吉田 智哉[†] 神戸 英利[†]

東京電機大学大学院理工学研究科^{††}

1. 研究背景・課題

近年の日本における視覚障害者の数は、厚生労働省が行った調査「平成 28 年生活のしづらさに関する調査」 [1]によると 31.2 万人となっている。視覚障害者は屋外での歩行時に、白杖と呼ばれる前方の路面を触擦する際に使用する白い杖や盲導犬を使用し歩行を行なっているのが現状である。しかし、盲導犬は育成にかかる費用や時間などの問題があり実働頭数は減少傾向にある。さらに、盲導犬は視覚障害者の歩行補助を行う上で非常に有用な存在ではあるが、目的地まで案内してくれるということではなく、道順は使用者がハynesを通じて指示する必要があるため、使用者の見知らぬ地での歩行は非常に困難であるという問題がある。

2. 研究目的

本研究では、健常者が視覚障害者に対して周りの状況を把握して声で伝えるように、スマートフォン(以下、スマホという)で撮影した画像から衝突の危険のある障害物や位置がわかるとより歩行が楽になるオブジェクト,あるいは視覚障害者が認識したい特定の障害物を検出する。そして、その障害物が何か,どの方向に存在するのか,障害物までどのくらいの距離かといった情報を音声により案内することで、視覚障害者の単独での安全な歩行をアシストすることを目的とする。

3. 提案手法

システム概要図を図 1 に示す。

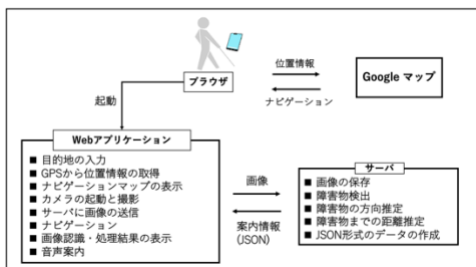


図 1 システム概要図

Research on obstacle detection by smartphone for safety navigation

Tomoya Yoshida[†], Hidetoshi Kambe[†]

[†]Graduate School of Science and Engineering,
Tokyo Denki University

スマホを図 2 のようにネクストラップを用いて首にかけ、背面カメラが進行方向を向くようにする。そして、背面カメラを用いて歩行中の対象者の前方を一定間隔で自動撮影し、その撮影された画像をサーバに送信する。サーバ側で障害物検出を行うことで画像中の障害物を検出する。障害物が検出された場合は、画像処理や深度推定を行うことで、その障害物の方向とおおよその距離の推定を行う。これらを歩行に役立つ情報として使用者に音声で提供する。



図 2 歩行中のスマホの位置

3.1. 障害物検出

スマホで撮影した画像をサーバに送信し、サーバ上で障害物検出を行う。そのために、リアルタイムに様々な障害物を検出可能な一般物体検知アルゴリズムである YoLo (You Only Look Once) を用いる。YoLo は高速であり、尚且つ認識精度も良いため世界中で広く使われている。本システムは、処理速度が非常に重要であるため、このアルゴリズムを用いる。

3.2. 距離推定

検出された障害物がどの程度離れた距離に存在するのかを推定する。これには単眼画像深度推定モデル (MiDaS) を用いる。MiDaS は入力された画像から奥行きを推定する機械学習モデルであり、任意の映像からデプス値 (奥行き) を推定することが可能である。しかし、出力として得られるデプス値は絶対値ではなく相対値であるため画像によってばらつきが生じてしまう。そのため、本研究では MiDaS によって取得したデプス値に対して実測値 (実際の距離) でキャリブレーションを行うことによって、距離推定を行う。

4. 実装

4.1. 実装システム

本システムは開発言語として Python, JavaScript を用いて, SPA 形式の Web アプリケーションとして開発を行なった. またフレームワークとして Flask, ライブラリとして jQuery などを用いた.

4.2. 障害物検出

自ら作成したデータセットを用いてモデルの構築を行った. 収集した画像の枚数は約 2000 枚. これらの画像に対してアノテーションツールを用いてラベル付を行い, データセットを作成した. 作成したデータセットを用いて画像認識モデルを構築した. 作成したモデルで検出可能なオブジェクト(17種類)を以下に示す.

- | | | | | |
|-----------|------------|-----------|---------|-----------|
| 1. 電柱 | 2. 人 | 3. ガードレール | 4. 横断歩道 | 5. 標識 |
| 6. 点字ブロック | 7. 自転車 | 8. 自転車 | 9. 信号機 | 10. 信号機_赤 |
| 11. 信号機_青 | 12. 自転車(人) | 13. ポール | 14. 車 | 15. バス |
| 16. トラック | 17. バイク | | | |

作成した画像認識モデルに画像を入力した際の出力例を以下の図 3 に示す.



図 3 画像の出力例

4.3. 距離推定

MiDaS の出力であるデプス値は相対値であるため, 実測値によるキャリブレーションを行なった. キャリブレーションの手順を以下に示す.

- 対象物の撮影を行い, 撮影地点から対象物までの距離を測定する.
- 1 で撮影した画像を MiDaS に入力し, デプス値を取得する.
- 距離を縦軸, デプス値を横軸として, 1 と 2 で取得したデータの散布図を作成する.
- 手順 3 の散布図において近似直線を引く. 実際に作成した散布図と近似直線のグラフを図 4 に示す.

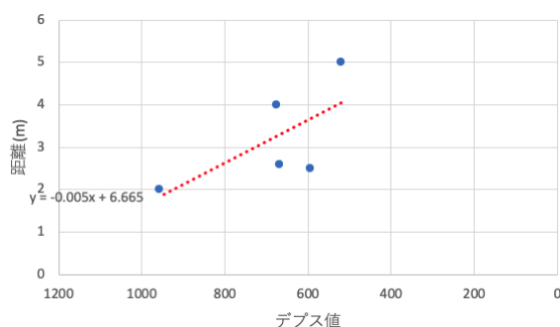


図 4 作成した散布図と近似直線

5. 評価

5.1. 障害物検出モデルの評価

4.2 節で作成した画像認識モデルに対して 87 枚のテスト画像を用いて評価を行なった. 以下にその結果をまとめた表 1 を示す.

表 1 各クラスの精度

class	precision	recall
電柱	0.724	0.861
人	0.945	0.995
ガードレール	0.685	0.779
横断歩道	0.917	0.968
標識	1	0.995
点字ブロック	0.564	0.596
信号機	0.875	0.943
信号機_赤	1	0.929
信号機_青	1	0.957
自転車(人)	0.8	0.87
ポール	0.909	0.905
車	0.901	0.939
バス	1	0.995
トラック	0.944	0.909
カーブミラー	0.792	0.881
バイク	0	0

6. 考察

5.1 節の評価からわかるように認識精度の高いモデルを作成できていない. 原因としては, 学習データの不足, データの質などが考えられる. 今後は質の良い学習データの数を増やしていく必要がある. また, 表 1 の結果から評価の数値が異常に高いものや低いものが見られる. 原因としては, テストデータの不足が考えられる. precision や recall は全ての画像の結果の平均として算出されるため, 画像の枚数が少ないほど 1 枚の画像に対する依存度が高くなってしまう.

7. まとめと今後の展望

本研究では, 視覚障害者の単独での安全な歩行を実現することを目的とし, 物体検出アルゴリズムを用いることで, 障害物を検出することができた. また, 画像処理や深度推定を行うことで, その障害物の方向と距離を推定することができた. しかし, 障害物の誤検出や方向・距離推定における実際との差異, 撮影から案内までのタイムラグなどが大きく, 精度や処理速度の向上が必要である.

参考文献

- [1] 厚生労働省, “情報バリアフリーのための情報提
https://www.mhlw.go.jp/toukei/list/seikatsu_chousa_h28.html. [アクセス日: 11 1 2022].