

Android 端末を用いた視覚障がい者歩行補助システムの検討

Study on a Vision Guide Tool for the Blind and Visually Impaired Using an Android Terminal

上條善治†, 江口一彦††

Zenji KAMIJO, Kazuhiko EGUCHI

Abstract A seeing-eye dog is one of the most helpful friends for the blind and visually impaired. However, the number of such kind of dog is very limited and it takes a lot of cost and time for training. Several developments of guide robot or auxiliary equipment to help the blind and visually impaired are reported recently but not so may yet.

In this paper it is discussed to develop a tool to recognize safe area of street by using android terminal when the blind and visually impaired walks without the guide of seeing-eye dog. The tool indicates the most safe walking direction by recognizing the edges of street and evaluating them.

1.はじめに

近年の日本における身体障がい者は、内閣府の「平成 23 年版障害者白書」^[1]によると、平成 23 年 6 月現在 366 万 3 千人、内訳をみると視覚障がい者 31.5 万人(8.8%)、聴覚・言語障害 36.0 万人(10.1%)、肢体不自由 181 万人(50.6%)、内部障害者 109.1 万人(30.5%)となっている。視覚障がい者は全員ではないものの、外での歩行時に白杖と呼ばれる杖や盲導犬を使用しなければならないのが現状である。最近では国や各自治体で障がい者のためのバリアフリーを重視したインフラの整備が行われている。しかし、盲導犬の育成にかかる時間や費用、供給できる数(H23.3.31 現在 1089 頭)などの問題や盲導犬の貸与可能な年齢、一部の施設などでは盲導犬を連れて行くことができないなどの制限を受けている。そのため近年では盲導犬の機能を工学的に再現する盲導犬ロボットなどの補助装置の研究が行われている。

盲導犬において再現すべき機能には、道の端を歩く、障害物を回避する、段差などで止める、目標物に誘導するといった役割が挙げられる。また盲導犬ユーザーは地図を思い出しつつ盲導犬の指示をトレースする必要があり、思い出すという行為を行うとともに、盲導犬のハー

ネスから伝わる僅かな動きを感じ取り、歩行を行う必要があるため、少なからずユーザーに負担がかかると考えられる。

本研究では画像処理を行え、小型持ち運びが可能であり、なおかつ GPS を搭載し地図情報の取得ができる Android 端末を用いての歩行補助システムの構築を検討した。

2.認識手法の手順

本研究では、Android 端末を用いて Fig2.1 に示す手順により道路領域を検出し、歩行補助のための指標表示を行った。

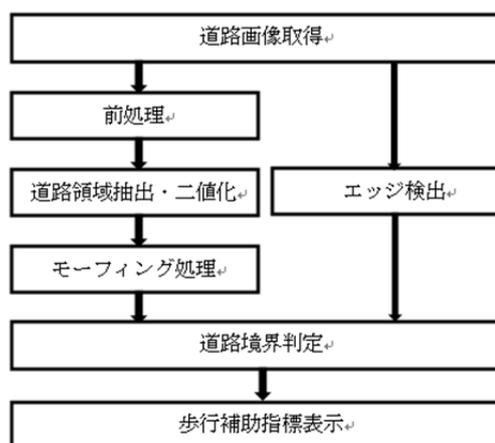


Fig2.1 認識手法

† 愛知工業大学 工学研究科電気電子工学専攻(豊田市)

†† 愛知工業大学 工学部電気学科 (豊田市)

Android 端末より画像を取得し、その画像より前処理のためにノイズの除去、平滑化を行う。前処理を行った画像を元に色情報(RGB)より道路領域の抽出、エッジの検出を行い、二つの処理結果から道路領域の判定を行う。

2.1 画像取得

実働している盲導犬はラブラドル・レトリバーやゴールデン・レトリバーなどの品種の犬が使われている。主人のいうことをよく聞き従順で温和な性質が盲導犬に向くという理由もあるが、この品種が選ばれる理由として人間と共に歩くために適当な大きさであることが理由に挙げられる。

このことを踏まえて画像の取得は道路上で、なおかつ盲導犬の視覚位置と同程度の高さから取得することにする。ゴールデン・レトリバーの雄の体高は約 60 cm、雌は体高 55 cm 前後であるので、画像の取得高さは地面より 55 cm~60 cm より取得する。

画像は 320×240pixel のサイズで取得する。

2.2 前処理

取得した画像そのままでは、色情報を抽出した際にノイズが乗ってしまい道路端を探索するのが難しいため、メディアンフィルタ^[3]を用いフィルタリングを行う。メディアンフィルタとは平滑化機能を持つ非線形フィルタであり、この出力は次式で与えられる。

$$g(i, j) = \text{median}[f(i + m, j + n)]: \\ -m_1 \leq m \leq m_1, -n_1 \leq n \leq n_1 \quad (2.1)$$

$\text{median}[\cdot]$ は中央値で与えられる関数であり、与えられた部分領域内の画素値 $f(i + m, j + n)$ ($-m_1 \leq m \leq m_1, -n_1 \leq n \leq n_1$) を大きさ順に並べたときその中央値をとる。たとえば、領域が 3×3 の場合には $-1 \leq m \leq 1, -1 \leq n \leq 1$ であり、9 つの画素を値の大きさの順に並べて、その中央値となる大きさが 5 番目の画素値を位置 (i, j) の平滑化画素値 $g(i, j)$ とする。

2.3 道路領域抽出・二値化

道路情報の抽出では取得した画像の色情報(RGB^[3])を元にして行う。

RGB(または RGB カラーモデル)は色の表現方法の一種で、赤(Red)、緑(Green)、青(Blue)の 3 つの原色を混ぜて幅広い色を再現する加減混色の一種であり、ブラウン管(CRT)や液晶ディスプレイ(LCD)、デジタルカメラなどで画像再現に使われる。

画像を取得する際道路上にいるということを仮定す

れば、取得した画像の下方中央付近は道路ということになる(Fig.2.2)。これを仮定道路領域とする。

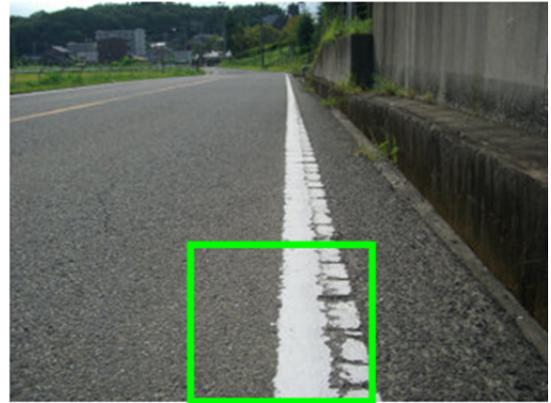


Fig.2.2 取得画像における仮定道路領域

仮定道路領域より取得した RGB の値より、赤(Red)、緑(Green)、青(Blue)の三原色の最大値及び最小値をそれぞれ取得する。取得した色情報をしきい値として再び取得した画像の処理を行い、抽出した結果から二値化処理を行う。その結果を Fig.2.3 に示す。

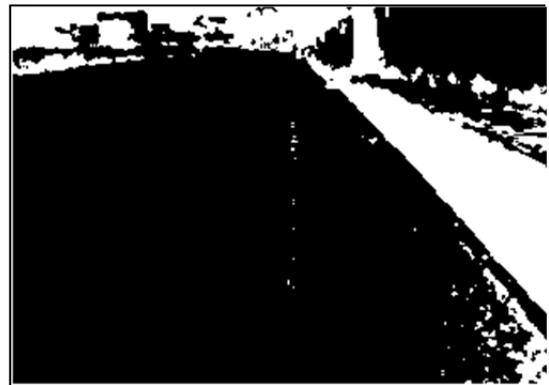


Fig.2.3 道路情報抽出および二値化結果

2.4 モーフィング処理

モーフィング(morphing)^[4]とは、ある形状から別の形状へ徐々に変化してゆく様子を動画で表示するために、その中間を補うための画像を作成することである。

モーフィングに含まれる処理には、膨張(Dilation)と収縮(Erosion, 腐食)がある。前景色を黒(論理的には 1、明るさ濃度では 0)として話を進める。膨張を用いると、切れていた線を繋ぐ、穴の開いた図形の穴を埋めるなどの処理となり、白ノイズの除去に役立つ。収縮は、小さな突起や小さな塊を除去することができ、黒ノイズの除去に役立つ。膨張を実行してから収縮を行うものをクロージ

ング(Closing)といい、収縮を先に実行してから膨張を行うものをオープニング(Opening)という。

本研究では二値化した画像にオープニングのモーフィング処理を施し、画質の向上を図っている。

2.5 エッジ検出

エッジ検出は、画像の輪郭部やエッジ部、高周波成分の部分に強調、抽出を行うためのフィルタリングである。

エッジ検出には Sobel^[3]フィルタを使用した。

Sobel フィルタとは1次微分フィルタと呼ばれ、画像を対象物体と背景に分割するために、対象物体の輪郭(エッジ)だけを抽出し、認識する手法である。

Fig2.4 に3×3画素を用いる Sobel フィルタの係数を、式(2.2)、(2.3)にフィルタ出力の式を示す。

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0^* & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0^* & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

水平方向 垂直方向

Fig2.4 マスク Sobel フィルタの係数^[3]

- ・水平方向 Sobel フィルタ出力:

$$\begin{aligned} g(i, j) = & \{f(i-1, j+1) - f(i-1, j-1)\} \\ & + 2\{f(i, j+1) - f(i, j-1)\} \\ & + \{f(i+1, j+1) - f(i+1, j-1)\} \equiv g_H(i, j) \end{aligned} \quad (2.2)$$

- ・垂直方向 Sobel フィルタ出力:

$$\begin{aligned} g(i, j) = & \{f(i+1, j-1) - f(i-1, j-1)\} \\ & + 2\{f(i+1, j) - f(i-1, j)\} \\ & + \{f(i+1, j+1) - f(i-1, j+1)\} \equiv g_V(i, j) \end{aligned} \quad (2.3)$$

これらの処理を用い、取得した道路画像よりエッジを検出すると、Fig2.5のようになる。

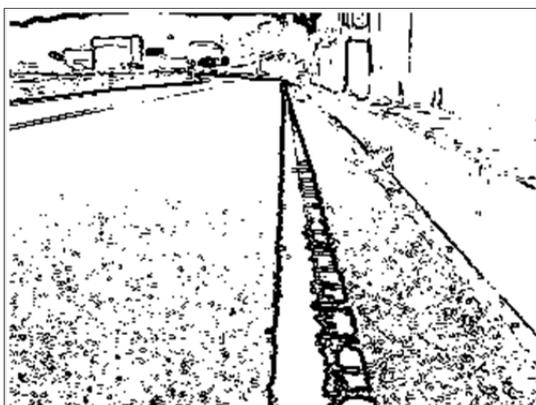


Fig2.5 エッジ検出結果

2.6 道路領域判定

エッジの検出では道路領域内に多くのノイズが残り、色情報の抽出では道路領域とそれ以外の区別が難しく、共に単体では道路領域を求めることが困難である。しかしこれら2つの処理を統合して考えれば、色情報内に発生したエッジに関しては、無視しても問題ないことがわかる。これを踏まえ、2つの処理の差分をとると Fig2.6のような画像となる。



Fig2.6 2つの処理の差分画像

さらに、歩行者から見て最も手前境界が重要になることから、道路領域は Fig2.7 のようになると判断することができる。



Fig2.7 境界線検出結果

2.7 歩行補助指標の表示

ユーザーの誘導にあたり、盲導犬の歩行経路の判断の要素の一つとして、道路上に引かれた白線があればその色と道路の色を判断基準としている。つまり、盲導犬の判断する歩行経路の選択は道路端と白線に依存すると考えられる。このことから道路端から白線程度の間隔を開けた歩行補助指標を引くことができれば、疑似的な歩行経路の選択となると判断する。

実際に歩行補助指標を描画した画像を Fig2.8 に示す。

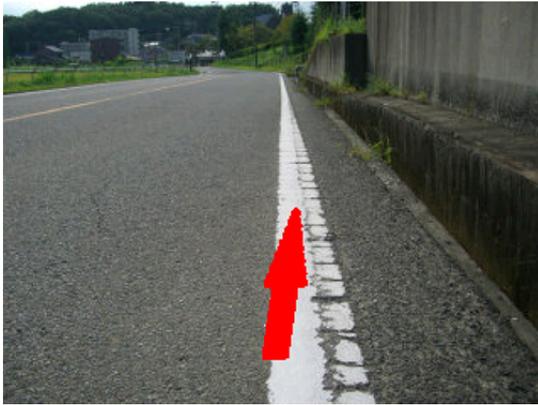


Fig2.8 歩行補助指標描画画像

歩行補助指標の描画は取得画像の(幅,高さ)(160,240)(画像の左上を(0,0)とした場合)より描画し画像の高さ 120 の位置まで描画している。

画像は道路より 60 cmほど離して撮影しているため、歩行補助指標の先端部は撮影場所から約 3m の距離が離れている。つまり Fig2.8 の指し示す歩行補助指標は 3m 先の道路上ということになる。

3.実験結果

本研究の認識手法によって得られた画像処理結果 (Fig2.8)を以下に示す。

Fig2.8 は白線があり、画像右側に溝のある道路画像となっている。

Table3.1 各処理における処理時間[ms]

処理名	処理時間			
	1 回目	2 回目	3 回目	平均値
エッジ	435	431	456	440.67
色情報	255	262	272	263.00
morphing	602	612	607	607.00
差分	67	80	66	71.00
境界判別	42	33	32	35.00
指標表示	39	29	37	35.00
総合	1440	1447	1470	1452.33

Fig2.8 の画像では、約 1400ms ほどの処理時間がかかる。

また、白線がない場合の実験結果は以下ようになる。

Fig3.1 に示す画像では木の陰により明暗の差があり、左方向に段差があることが特徴として挙げられる。



Fig3.1 画像 A

処理の結果、左方の段差を道路端と認識し段差に合わせて歩行補助指標を表示している。

Fig3.2 に示す画像は上り勾配、道路とほぼ同色の建物あり撮影位置がやや道路端より離れている。



Fig3.2 画像 B

処理の結果、現在地が道路寄りであるため歩行補助指標は左に傾いて表示されている。この指標に従って歩行を続けた場合壁に衝突してしまう。だが、矢印が指し示す位置は撮影位置より 3m 前方であり、歩行者が矢印の指し示す位置に到達する前に新たな指標が生成される。歩行者は新たに生成された指標に従うことで、衝突を回避することができる。

Fig3.3 に示す画像は道路に大きな亀裂あり、撮影位置やや中央よりとなっている。



Fig3.3 画像 C

処理の結果道路の沿道を道路境界だと判定し、沿道に沿った指標を表示する結果となった。

以上の処理結果の処理時間を Table3.2 に示す。

Table3.2 各処理における処理時間[ms]

処理名	処理時間		
	画像 A	画像 B	画像 C
エッジ	460	428	430
色情報	283	260	277
morphing	530	578	887
差分	85	61	61
境界判別	25	53	47
指標表示	29	27	28
総合	1412	1407	1730

Table3.2 より、エッジ検出、道路領域抽出、モーフィング処理に時間がかかっている。中でもモーフィングの処理時間は画像によって変化が大きく、総処理時間に影響していることがわかる。

4.結論

本論においての結論は以下のようになる。

- ① Android 端末を用いた道路領域の検出
- ② 道路境界の判定
- ③ 歩行補助指標の表示

Android 端末を用いた場合、処理時間は 1400ms～1700ms ほどとなった。成人男性の歩行速度は平均時速 3.87 km であるため、約 1.5m 毎の画像取得となる。歩行補助指標は 3m 前方を指すものであるため、処理可能な速度であると判断する。

また、ノイズ除去に使用しているモーフィング処理に

時間がかかっていることから、ノイズ除去に工夫が必要であると考えられる。

5.今後の課題

視覚障がい者の歩行補助に必要なこととして、現在地から目的地までのナビ機能の搭載が考えられる。目的地までの経路の計測を行い、その道路情報から進行方向を導出、得られた進行方向を用いることで歩行補助指標の精度をより高められると考える。これは、本研究でもともと取り組む予定の課題であり、Android における地図アプリの開発も並行して進めてきた。だが残念なことに、考案した認識手法 B の処理時間が予想以上に遅く、フローチャートに組み込むことができなかった。地図アプリを組み込むには道路情報の処理時間の短縮も同時に課題となる。

また、Android 端末などの音声アプリを用いれば、ももとの盲導犬においては不可能である音声を用いての歩行者補助も可能となる。よって歩行補助のための音声ガイダンスの構築も必要となると考えられる。

参考文献

- [1] 内閣府「障害者白書 平成 23 年度版」
<http://www8.cao.go.jp/shougai/whitepaper/index-w.html>
- [2] 荒海宇宙, 荒川貴則, 井尻健嗣:
「視覚障がい者の歩行補助機器の開発」愛知工業大学卒業論文(2010)
- [3] 末松良一, 山田宏尚: 画像処理工学コロナ社(2000)
- [4] Visual C++ の勉強部屋
<http://homepage3.nifty.com/ishidate/vcpp.htm>
- [5] 溝口早苗: Smartphone World Volume.1 CQ 出版社
(2011 年 5 月 1 日発行)
- [6] 小澤拓治: Smartphone World Volume.2 CQ 出版社
(2011 年 9 月 10 日発行)
- [7] 中西葵, 中村裕之, 高橋良司:
Android SDK 逆引きハンドブック C&R 研究所
(2011 年 5 月 6 日発行)
- [8] 金宏和實:
作ればわかる! Android プログラミング
—10 の実践サンプルで学ぶ Android アプリ開発入
翔泳社 (2011 年 10 月 14 日発行)

(受理 平成 24 年 3 月 19 日)