

修士論文

道路標識内部のエッジを利用した 実時間道路標識抽出に関する研究



平成 21 年度修了
三重大学大学院 工学研究科
博士前期課程 情報工学専攻

福田 剛久

目次

第 1 章	はじめに	1
1.1	研究の背景	1
1.2	関連研究	1
1.2.1	運転支援を目的とした道路標識の認識に関する研究について	2
1.2.2	悪条件化での抽出精度向上を目的とした研究について	2
1.2.3	データベース作成を目的とした研究について	2
第 2 章	提案手法	3
2.1	2 値化	3
2.1.1	HSV 色空間の特徴	3
2.1.2	赤色領域の抽出	4
2.2	道路標識の候補領域の抽出処理	6
2.2.1	道路標識の内側エッジ	6
2.2.2	道路標識のグループ分け	7
2.2.3	抽出手順	8
2.2.4	外側のエッジトレース	9
2.2.5	内側のエッジトレース	9
2.2.6	矩形領域内の色情報	9
2.2.7	駐車禁止, 駐停車禁止の結合処理	13
2.2.8	道路標識の候補領域の判定処理	15
2.2.9	道路標識内部の赤の比率調査	16
2.2.10	抽出手順のまとめ	20
2.3	認識処理	22
2.3.1	シンボル部分の抽出	22
2.3.2	駐車禁止, 駐停車禁止の認識方法	24
2.3.3	速度制限の認識のための数字領域の抽出	24
2.3.4	速度制限の認識方法	26
2.3.5	画像の修正	28

第 3 章 実験	30
3.1 実験環境	30
3.1.1 抽出結果	31
3.1.2 抽出に関する考察	34
3.1.3 認識結果	37
3.1.4 認識に関する考察	38
第 4 章 まとめと今後の課題	39
4.1 まとめ	39
4.2 今後の課題	39
謝辞	40
参考文献	41
付録	43

第 1 章

はじめに

1.1 研究の背景

平成 20 年度に国内で発生した交通事故件数は約 76 万件 [24] で、近年では飲酒運転の取締りの強化などの対策により減少傾向にある。しかし、交通事故の年間の発生件数は 76 万件以上、負傷者は 90 万人以上と、交通事故情勢は依然として厳しい状況である。このような背景の下、危険箇所に対して道路交通標識の大型化やカラー舗装、信号機の LED 化などの対策を行ったことで、交通事故が約 8 割減少したという報告がなされている [2]。ここで、道路交通標識の大型化という対策に着目すると、車内カメラを使って道路交通標識をドライバーに通知するシステムを開発することで、交通事故の減少、防止に役立てることができると期待されている。

交通事故の減少を目的とした運転支援に関する研究では、世界中で高度道路交通システム (ITS: Intelligent Transport System) の一環として、1990 年頃から盛んに行われてきた [3]。特に、道路交通標識に関する研究では、車載カメラによってドライバーから見える道路交通標識を認識し、ドライバーに標識の通知を行うシステムに関する研究が従来から盛んに行われている。また、近年では車載カメラによって通行人や障害物などを認識し、自動車の運転支援を行うという動きが活発化している。そのため、車載カメラを使って道路交通標識の認識を行うことで、道路交通標識の大型化等の設置を行うよりも低コストでの実現が可能となる。

以上のことから、本研究ではドライバーから見える道路交通標識の見落とし防止を支援するシステム開発を目的とする。しかし、道路交通標識を全てリアルタイムで認識することは困難であるため、本研究では、規制標識の中で赤色を含む標識に種類を限定して行う。

1.2 関連研究

道路交通標識を車載カメラで認識しようとする試みは、1990 年代から国内外問わず活発に行われている。そして、現在では研究目的によって以下の三つに大別することができる。

1. 運転支援を目的とした道路標識の認識に関する研究 [4]～[16]
2. 悪条件化での抽出精度向上を目的とした研究 [17]～[19]
3. データベース作成を目的とした研究 [20]～[21]

上記の研究についての概要の説明を行う。

1.2.1 運転支援を目的とした道路標識の認識に関する研究について

道路標識の認識に関する主な処理手順を以下に示し、各ステップの処理の説明を行う。

1. 対象の色情報を利用して入力画像を 2 値化もしくはエッジ抽出を行う。
2. 対象の幾何学情報を利用して標識の候補領域の抽出を行う。
3. 候補領域内部のシンボル情報の抽出を行う。
4. シンボル部分の認識処理を行う。

まず初めに、色情報を利用した標識部の抽出には、RGB 表色系を HSV 表色系に変換して行うのが一般的である。HSV 表色系が利用されるのは、明るさに対して柔軟に対応できるからである。その他には、松浦氏らの研究 [6] で RGB 減算法を用いた抽出方法などが提案されている。

次に、色情報を使って標識領域の情報を抽出した画像に対して、標識の候補領域の抽出処理を行う。候補領域の抽出には輪郭追跡の結果を利用する手法 [7][9][11][13] やエッジ点の勾配情報を使った手法 [5] などが提案されている。

最後に、抽出した候補領域から標識のシンボル部分を抽出し、抽出したシンボル部分の解析を行うことで標識の種類を特定する。シンボル部分の抽出には色度と輝度値を利用した抽出方法 [8] や反対色フィルタを利用した抽出方法 [13] などがある。シンボル部分を抽出した後は、テンプレート画像との正規化相互相関を利用した判定方法 [5] や形状特徴値配列を用いた手法 [8] がある。

また、近年では認識処理の部分に自己組織化マップ (SOM) を利用した研究 [15] など盛んに行われている。

1.2.2 悪条件化での抽出精度向上を目的とした研究について

道路標識が設置されている環境には、道路上の他の物体によって標識の一部が隠されてしまうことがある。そのため、標識の隠れや欠けを補う研究もなされている。例えば、標識の前に障害物が存在し、認識に悪影響を及ぼす環境下での認識精度向上に関する研究 [18] がある。この研究では、連続するフレームからそれぞれ標識の領域を抽出して重ね合わせることで、隠されている領域を補間する方法が提案されている。他には、図形の特徴を利用して抽出した標識の領域の欠けを補う研究 [19] なども行われている。

1.2.3 データベース作成を目的とした研究について

道路標識を認識する研究では、車載カメラを使ってドライバーに通知する目的以外にも、カーナビに登録するデータ収集などを目的としたデータベース作成に関する研究も行われている。株式会社パスコからは PS-Cruiser[19] という道路標識自動認識システムが開発されている。このシステムは、車に 2 台のカメラとコンピュータを搭載して行われる。また GPS も搭載しているため、標識の位置情報も同時に取得することができる。しかし、処理時間や搭載設備の観点から、現状では通常の車に搭載して認識処理を行うのは困難である。

第 2 章

提案手法

この章では車載カメラの映像から得られる画像から，道路標識を抽出，認識する方法の説明を行う．また，処理は入力画像の 2 値化，道路標識の候補領域の抽出，候補領域から道路標識のシンボル部分の抽出，シンボル部分の認識処理の順で行われる．

2.1 2 値化

道路標識に使用される色は赤，白，青，黄，黒と限定的である．したがって，道路標識の認識では，色情報を手がかりにしてラベル付けを行うのが一般的である．そして，様々な色空間がある中で，HSV 色空間が最もよく使用される．HSV 色空間が使用される理由は，光の変化に強いことが挙げられる．本研究でも，この HSV 色空間を用いて 2 値化を行い，道路標識の候補領域の抽出，認識を行っていく．

2.1.1 HSV 色空間の特徴

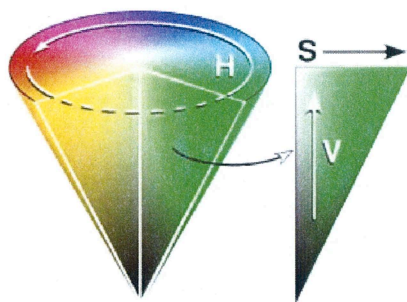


Fig.1 HSV 色空間の模式図

HSV 色空間は，色相 (Hue)，彩度 (Saturation, Chroma)，明度 (Value, Brightness, Lightness) の 3 つの成分からなる表色系のことである．RGB 表色系では物体の色を直接想像するのが困難であるが，HSV 表色系では人間の感覚にあった色表現が可能となる．Fig.1 に HSV 色空間の模式図を示す．

HSV 表色系では Fig.1 に示されるように，円錐の円周に沿って色相が，中心からの距離に応じて彩度が，円錐の高さに沿って明度が表されている．RGB 表色系から HSV 表色系への変換には，次に示す式を使う．

以下では，RGB 表色系における成分をそれぞれ R, G, B, HSV 表色系における成分をそれぞれ H, S, V と表現する．

・RGB → HSV への変換式

$$\begin{aligned}
 MAX &= \max\{R, G, B\}, \quad MIN = \min\{R, G, B\} \\
 H &= \begin{cases} \frac{G-B}{MAX-MIN} \times 60 + 0 & \text{if } R = MAX \\ \frac{B-R}{MAX-MIN} \times 60 + 120 & \text{if } G = MAX \\ \frac{R-G}{MAX-MIN} \times 60 + 240 & \text{if } B = MAX \end{cases} \\
 H &= H + 360 \quad \text{if } H < 0 \\
 S &= \frac{MAX - MIN}{MAX} \\
 V &= MAX
 \end{aligned}$$

本研究では上記の式で変換を行った後，H, V 成分の値を 0 から 1 の間に正規化を行う．

2.1.2 赤色領域の抽出

本研究では道路標識の中でも赤色を含む規制標識を対称とするため，前項で説明をした変換式で算出された値を使って，赤色の画素を抽出して 2 値化を行う．この 2 値化を行うための閾値は実験的に求め，本研究では以下のように設定した．

$$H \leq 0.11, \quad 0.65 \leq H, \quad S \geq 0.08, \quad V \geq 0.1$$

上記の閾値は，標識の明度が低下している場合にも対応できるように範囲を広く設定している．この閾値を使用して 2 値化を行った画像を Fig.2 に示す．

また，2 値化を行った画像にはノイズが多く含まれているため，先行研究 [13] と同様に 2 値化した画像のノイズ除去を行う．ノイズ除去では，赤色とラベル付けされた画素の 8 近傍を調査し，赤色とラベル付けされた画素が 3 画素以下の場合，注目画素のラベルを 0 に戻す．これにより，画像内のノイズを除去することができる．

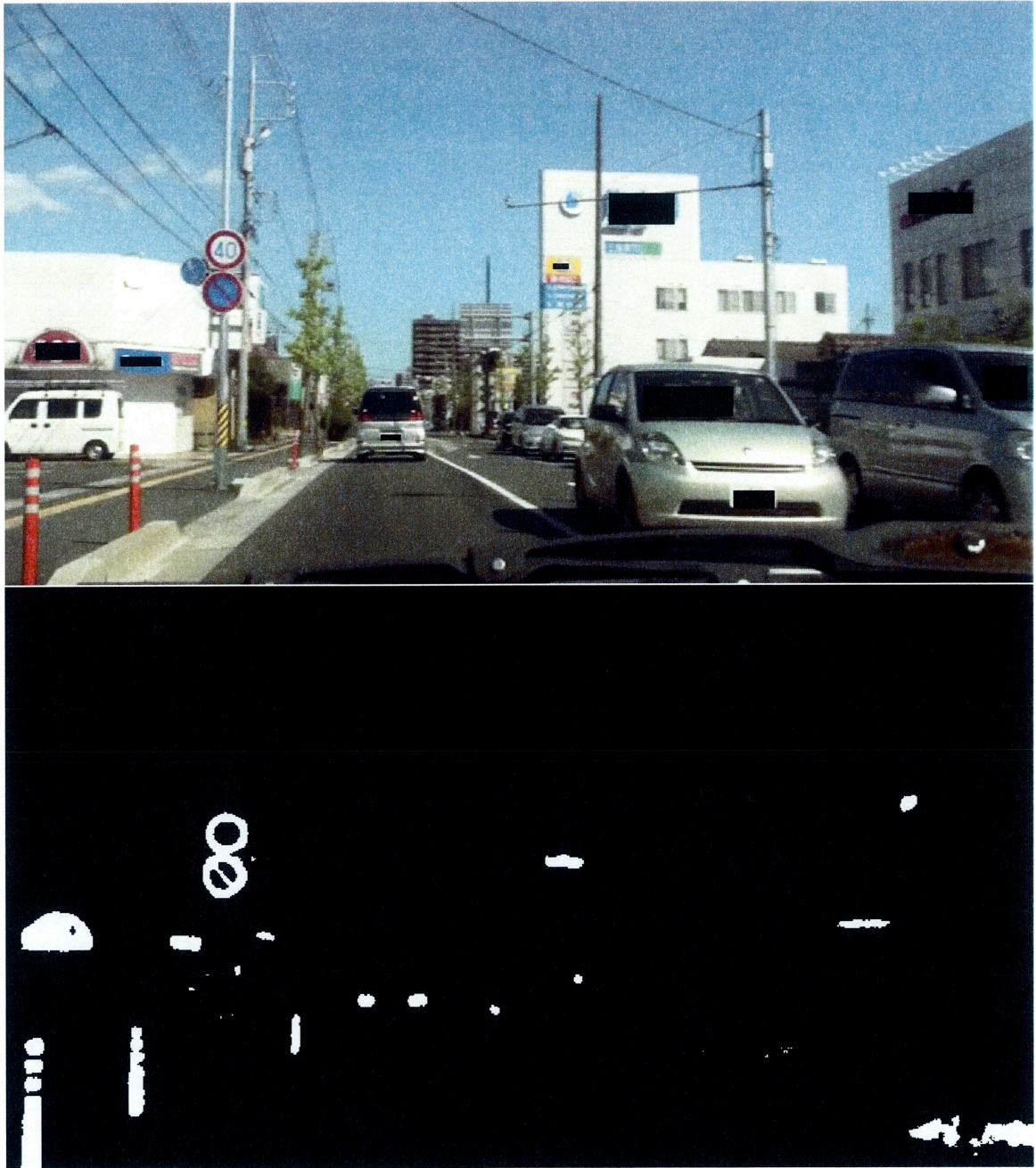


Fig.2 上:入力画像 下:2 値化後の画像

2.2 道路標識の候補領域の抽出処理

本論文で対象としている赤い道路標識の候補領域の抽出を行うために、前節で2値化した画像の中から、円形の道路標識であると思われる領域の抽出を行う。また、従来研究ではこの候補領域の抽出処理において、候補領域内に背景など道路標識以外の情報が多く含まれてしまう。そのため、候補領域から標識の種類を決めるシンボル部分を抽出するために、反対色フィルタ [13] などの処理を行う必要がある。しかし、この反対色フィルタというのは、シンボル部分の抽出が可能だが、処理時間がかかるという難点がある。そこで本研究では、認識処理の段階でシンボル部分の抽出処理を軽減するために、道路標識内部にできるエッジを利用して候補領域内に写る雑音を少なくする手法を提案する。

2.2.1 道路標識の内側エッジ

候補領域の抽出に利用する道路標識内部にできるエッジについての説明を行う。

今回対象とする道路標識の特徴には、赤色を含むという点以外に「2色以上の色が使われている」という点が挙げられる。そのため、2.1で説明を行った2値化を行うと、道路標識部は次の2つの領域に分けられる。

- ・赤の領域 (2値画像では白)
- ・赤以外の領域 (2値画像では黒)

また、Fig.3, Fig.4に例を示す。

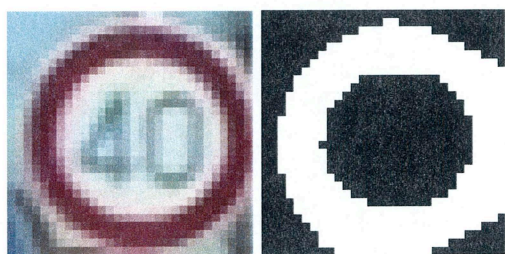


Fig.3 左:入力画像 右:2値画像

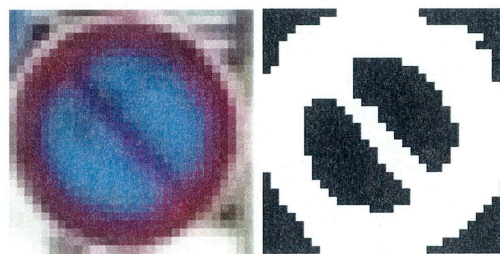


Fig.4 左:入力画像 右:2値画像

Fig.3, Fig.4の2値画像を見て分かるように、道路標識の外側の白い領域と内側の黒い領域の間に境界ができる。本論文ではこの境界を道路標識の内側のエッジと呼ぶ。この道路標識の内側のエッジを利用することで、Fig.5, Fig.6に示すような候補領域を抽出することが可能となる。また、Fig.5, Fig.6から分かるように、候補領域内に標識以外の雑音が混ざらないという特徴がある。

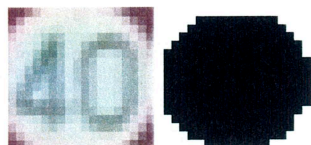


Fig.5 左:抽出領域 右:2 値画像

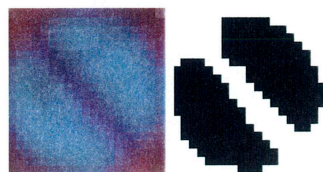


Fig.6 左:抽出領域 右:2 値画像

2.2.2 道路標識のグループ分け

道路標識内部にできるエッジを利用する場合，道路標識の種類に応じた抽出を行う必要がある．そのため，道路標識内部の特徴から対象とする道路標識を次の三つのグループに分類する．また，Fig.7, Fig.8, Fig.9 に各グループの例を示す．

・グループ 1

外側に赤いリング状の領域が存在し，内部に赤い斜線が存在しない，または赤い斜線が途切れている道路標識

・グループ 2

外側にリング状の赤い領域が存在し，内部に赤い線が途切れることなく存在している道路標識 (駐車禁止，駐停車禁止)

・グループ 3

グループ 1, 2 以外の道路標識 (進入禁止，一時停止)



Fig.7 グループ 1

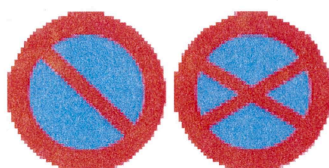


Fig.8 グループ 2

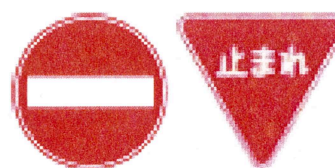


Fig.9 グループ 3

本研究では，赤色を含む規制標識を上記の三つのグループに分類したが，今回はグループ 3 の道路標識の抽出，認識方法を確立できなかったため対象外とする．しかし，今回提案する抽出方法において，グループ 3 に属する道路標識がグループ 1 やグループ 2 として抽出されることはないため，先行研究の抽出方法を参考にして追加処理を加えることで抽出・認識ができると考えられる．

2.2.3 抽出手順

道路標識内部にできるエッジを利用した候補領域抽出手順のフローチャートを Fig.10 に示す。また、フローチャート中の分岐条件等の詳細については後で説明を行う。

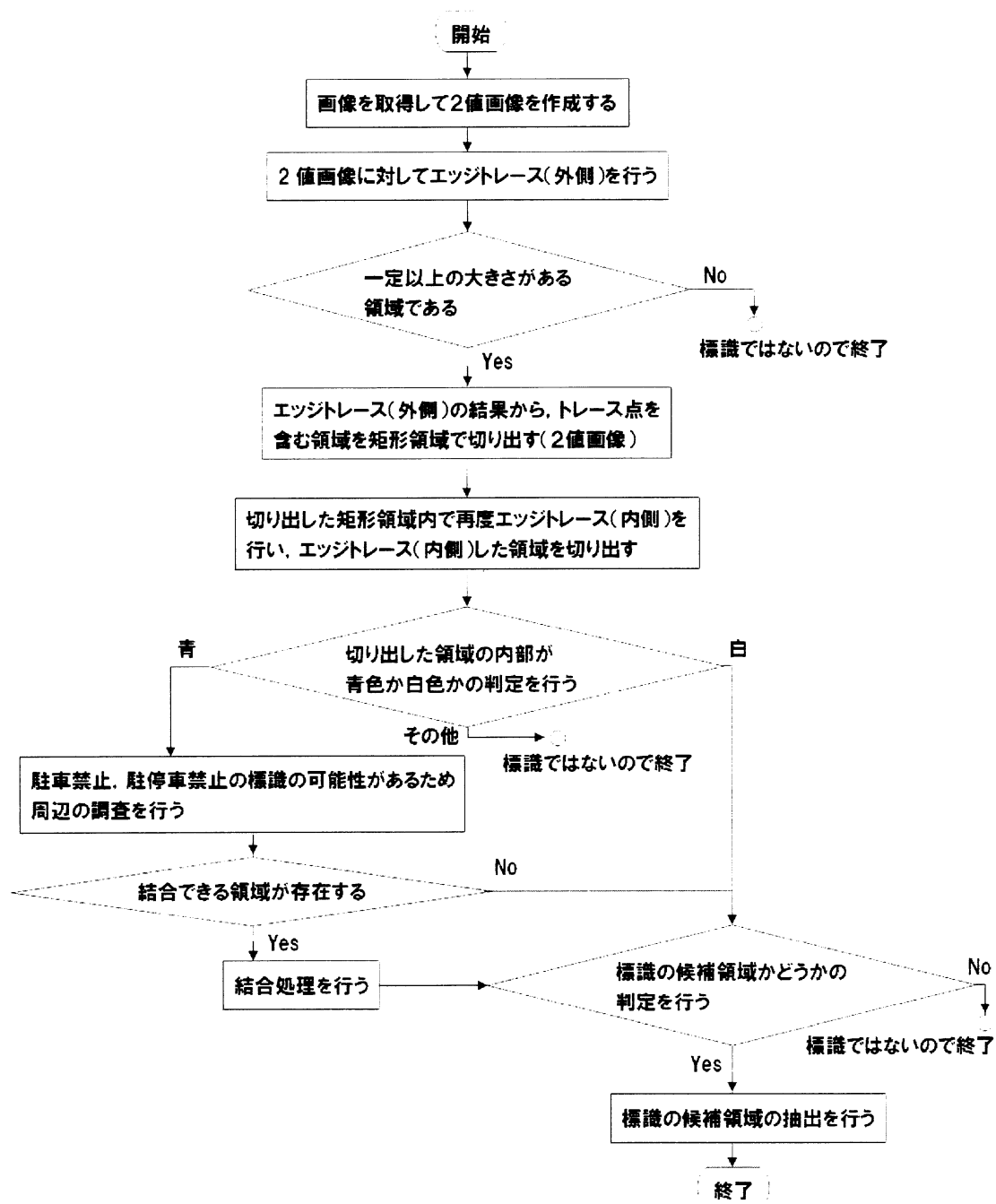


Fig.10 抽出処理のフローチャート

2.2.4 外側のエッジトレース

道路標識の候補領域を抽出するため、まず初めに馬場氏らの手法 [13] と同様に、2 値化した画像に対してエッジトレースを行う。そして、本論文ではこの最初のトレースを外側のエッジトレースと呼ぶ。

次に、先行研究の認識結果では、道路標識を認識するために直径が 20 画素は必要であるとされている。そのため、トレース点が少ない領域は雑音である可能性が高いと判断できる。そこで、トレース点が極端に少ない領域は雑音として処理をする。それ以外の領域には道路標識が存在している可能性があるため、トレース点を含む 2 値画像の領域を矩形領域で切り出し、切り出した矩形領域内で再度トレースを行うことで候補領域を抽出する。矩形領域の切り出しには、トレース中に記録しておいた上下端、左右端の座標情報を利用する。この切り出しの例を Fig.11 に示す。Fig.11 を見ると、切り出した領域には道路標識内部にできるエッジが存在していることが分かる。本研究では、この内部にできるエッジを使って、候補領域内に写る道路標識以外の雑音を除去し、グループ分けを行っていく。

2.2.5 内側のエッジトレース

前項で説明をしたように、外側のエッジトレースの情報を基に切り出した矩形領域には、道路標識内部にできるエッジが存在する。そこで、このエッジを利用することで候補領域に写る雑音の除去、グループ分けを行っていく。

内側のエッジトレースを行うときは、2.2.2 で説明をしたグループに応じた抽出方法を実装する必要がある。また、内側のエッジの発見を目的とするので、外側のエッジトレースでトレースをしたエッジ点は、内側のエッジトレースで対象外とする。

内側のエッジトレースではまず初めに、外側のエッジトレースで切り出した矩形領域内で再度エッジトレースを行う。そして、エッジトレースの結果を利用して矩形領域を切り出す。このとき、入力画像 (カラー画像) と 2 値画像の 2 枚の画像を切り出す。そして、この 2 枚の画像の情報を使って切り出した領域が道路標識の候補領域かどうかの判定を行う。

2.2.6 矩形領域内の色情報

内部エッジを利用して道路標識の候補領域を抽出する場合、2.2.2 でグループ 2 に分類した駐車禁止、駐停車禁止の標識では、Fig.14 に示すようにシンボル部分が複数に分かれて抽出されてしまう。そのため、複数に分かれて抽出される領域を結合して正確に抽出するには、内側のエッジトレースをして抽出された領域がグループ 2 に属する標識の一部かどうかを判定する必要がある。



Fig.11 上:2 値画像 下:切り出した矩形領域の例

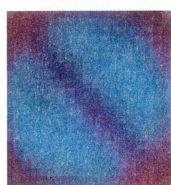


Fig.12 切り出した 2 値画像の例

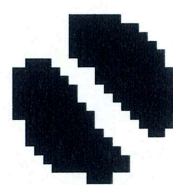


Fig.13 切り出した入力画像の例

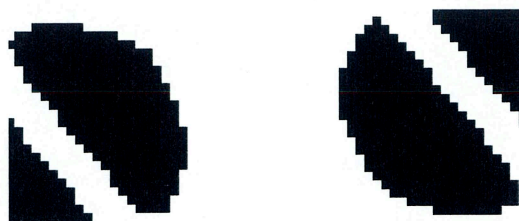


Fig.14 別々に切り出された標識

本研究ではこの判定のために道路標識内部の色情報を利用する。そして、内部の色の違いによって次の三つに分類を行う。また、以下のように分類を行うことで認識処理をより限定的に行うことができる。

- ・内部が白い (グループ 1 に属する道路標識)
- ・内部が青い (グループ 2 に属する道路標識)
- ・上記以外 (グループ 3 に属する道路標識)

グループ 1 とグループ 2 に属する標識の例をそれぞれ Fig.15, Fig.16 に示す。



Fig.15 内部が白い例



Fig.16 内部が青い例

Fig.15, Fig.16 からグループ 1 に属する標識は、シンボル部分以外の領域が白もしくは赤で構成されていることが分かる。そして、グループ 2 に属する標識は、赤と青のみで構成されていることが分かる。そのため、本研究ではこの色の偏りを利用するために、カラー画像の RGB 値を使用して分類を行う。

分類を行う前に各色と RGB 値の対応関係を Table1 に示す。なお、Table1 内では、各成分の中で値が大きいものを○、値が小さいものを×で表現している。例えば、赤色では R 成分の値が大きく、G,B 成分の値が小さいので、R 成分は○、G, B 成分は×と表現される。

Table1 色と RGB 値の対応関係

色	R 成分	G 成分	B 成分
白	○	○	○
赤	○	×	×
青	×	×	○
緑	×	○	×

Table1 から分かるように、グループ 1 に属する標識の内部では白と赤が大半を占めるため、R 成分の値が大きいという特徴がある。次に、グループ 2 に属する標識の内部では青が大半を占めるため、B 成分の値が大きいという特徴がある。さらに、グループ 2 の標識は赤と青で構成されているため、G 成分の値が非常に小さいという特徴がある。反対に、グループ 1 では標識内部に白の領

域が多いため、G 成分の値が大きいという特徴がある。本研究では抽出した領域内で各成分の総和を求め、先の特徴を考慮した色判定を行う。色判定には G 成分の値に対する R 成分、B 成分の比率を求めることで行う。今回設定した判定のための条件を次に示す。

・判定のための条件

$\text{ratio_BG} = \text{領域内の B 成分の総和} / \text{領域内の G 成分の総和}$

$\text{ratio_RG} = \text{領域内の R 成分の総和} / \text{領域内の G 成分の総和}$

とする

白の条件式 $0.8 \leq \text{ratio_BG} \leq 1.1$ かつ $0.8 \leq \text{ratio_RG} \leq 1.1$

青の条件式 $1.3 \leq \text{ratio_BG} \leq 1.7$ かつ $1.1 < \text{ratio_RG}$

グループ 1 の場合、全体的に白の領域が多いため各成分の値が等しくなる。実際には、領域内には赤や青の領域があるため、G 成分の値が少し低くなることが予想されるが、赤の領域では B 成分が低くなり、青の領域では R 成分が低くなるので、全体的にどの成分の総和もほぼ等しい値になる。そのため、ratio_BG, ratio_RG の値は 1.0 付近の値をとる。

グループ 2 の場合、領域全体の中で青の領域が大半をしめるため、B 成分の値が高くなる。そして、若干の赤の領域が存在するため、R 成分は G 成分よりも高い値をとることになる。そのため、B 成分は G 成分よりも高くなり、R 成分も G 成分より高い値となる。

また、標識までの距離が遠い場合、駐車禁止や駐停車禁止の内部に存在する赤い斜線を 2 値化の段階で検出するのは困難である。その原因は、内部の斜線は細く、かつ周囲を青の領域で覆われているため、画像上では色が曖昧になってしまうからである。その例を Fig.17 に示す。

Fig.17 を見ると、2 値化を行った際、外周のリング状の領域は抽出ができていたが、内部の斜線がほとんど抽出されていないことが確認できる。ここで、内側のエッジトレースを行った後に先ほどの色判定を行うと、領域内には青の領域が多いため青と判定される。そのため、2 値化の段階で内部の斜線を抽出できなくとも、内側のエッジが存在していれば、色判定処理でグループ 1 とグループ 2 の差別化を行うことが可能となる。したがって、グループ 1 とグループ 2 の認識処理は独立して行うことができ、無駄な処理を省くことができる。



Fig.17 左:遠くに写る駐車禁止の標識 右:2 値画像

2.2.7 駐車禁止，駐停車禁止の結合処理

内側エッジトレースの情報から抽出した矩形領域において，前項の色判定で青と判定された場合，駐車禁止と駐停車禁止の可能性があるので，周辺を調査して領域の結合を行う．結合の処理手順を以下の Fig.18 に示す．また，Fig.18 では，各処理に該当する画像を載せた．対応関係は※印の番号を参照してもらいたい．

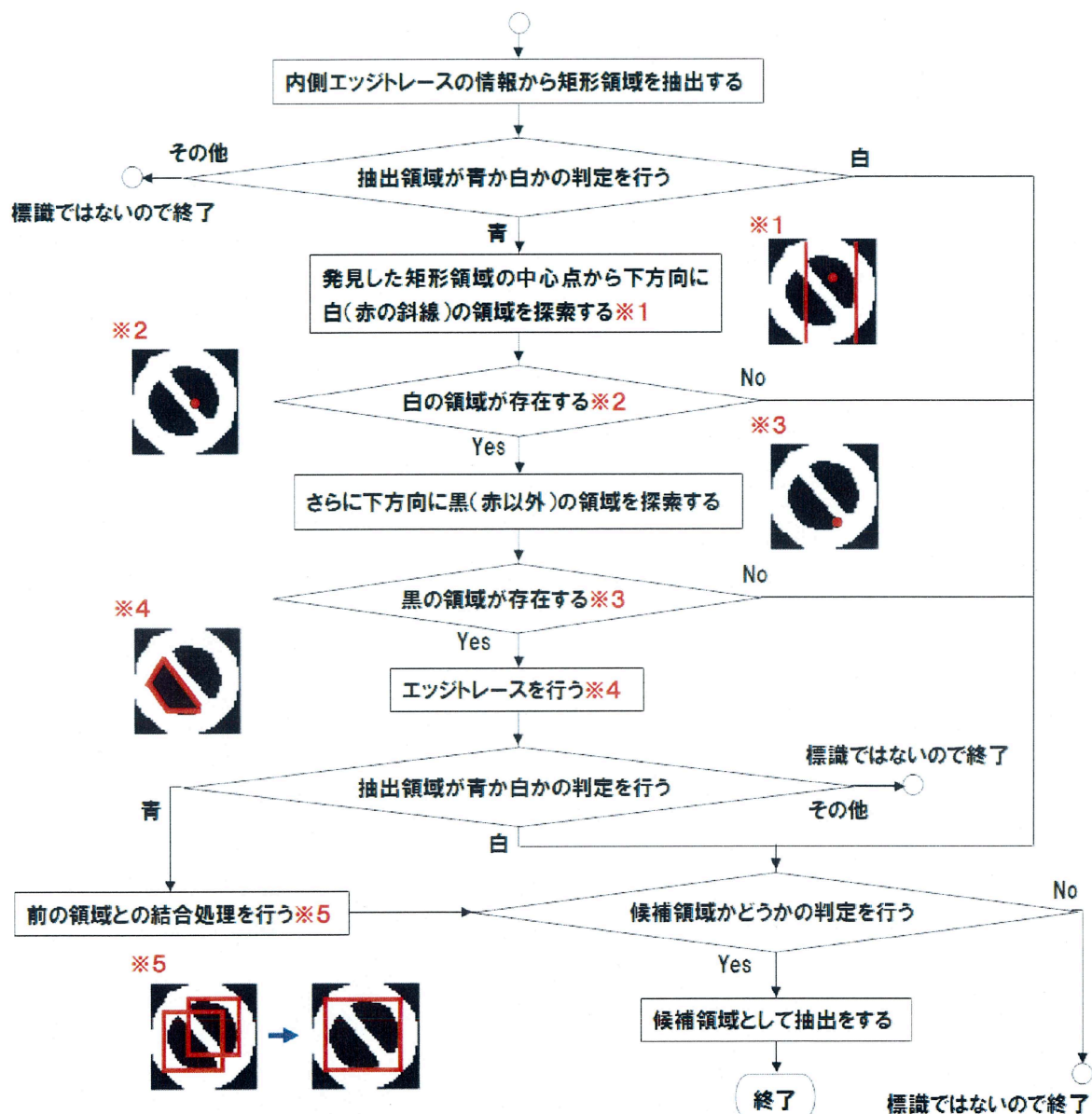


Fig.18 内側エッジを用いた処理のフローチャート

領域の結合処理では、2 値化の段階で標識内部の領域が分離していることを前提として考える。また、前項の Fig.17 のように 2 値化の段階で分離していない場合は、最初の内側エッジトレースの領域内にシンボル部分が全て含まれている。したがって、結合処理をしなくともトレースした範囲だけで認識に十分な領域を得ることができる。

結合処理ではまず初めに、発見した領域の下側を探索することから始める。これは、候補領域のトレースを画像の左上から開始するため、上にある領域が先に発見されるからである。下側の探索を行うための開始点には、切り出した矩形領域の中心点を使用する。

次に、開始点から下方向に探索を行い、白い画素を発見するまで下に探索を行う。このとき発見される白い画素は、赤の斜線の領域に該当する。そして、赤の斜線は幅をもっているため、次の黒い画素を発見するまで下方向に探索を続ける。こうして発見された黒い画素は、標識内部にある赤の斜線によって分離された青い領域の一部であるため、その点を始点として内側のエッジトレースを開始する。このトレースによって発見された領域に対して、再度内部の色確認を行う。そして、領域内の色が白である場合はグループ 1 の候補領域の可能性があるため、後の道路標識の候補領域かどうかの判定処理に回す。領域内の色が青である場合は、最初のトレースで得られた領域との位置関係が、上下の関係にあるかどうかを調べる。

最後に、二つの領域が上下関係にあると判定された場合、その領域は駐車禁止と駐停車禁止の標識の可能性が高いといえる。ここで本研究では、認識処理をより限定的に行えるように、二つの領域の重なり具合から駐車禁止と駐停車禁止の判定を予め行う。判定条件には次の特徴を利用する。

- ・駐車禁止の場合

二つの領域は重なり合う

- ・駐停車禁止の場合

二つの領域が重なり合わない

上記の様子を Fig.19 に示す。



Fig.19 左:駐車禁止 右:駐停車禁止

Fig.19 から分かるように、駐車禁止の場合は二つの領域が重なり合い、駐停車禁止では上下の二つの領域が重ならないことが分かる。また、今回の提案手法では、駐停車禁止の左右の領域は考慮しないこととする。この理由は、従来手法 [13] においても認識段階で左右の領域を使用することなく精度の高い認識率を出しているため、上下の領域の情報のみで十分な認識が行えると判断したためである。また、左右の領域を抽出しなくとも、候補領域の中心点と縦の長さから横への拡張が可能である。よって、本研究では抽出段階で駐停車禁止の左右の青い領域は考慮しない。

以上の処理を行うことで分離した領域を結合することができる。また、抽出段階でグループ 2 の標識をさらに細分化することができるため、認識処理にかかる時間の短縮にもつながる。

2.2.8 道路標識の候補領域の判定処理

前項までの方法でグループ分けを行いながら領域を抽出した後、道路標識とは関係のない領域を取り除くために、本研究では以下の二つの情報を利用する。

- ・縦横比
- ・標識内部の赤色の比率

まず、一つ目の縦横比について説明を行う。抽出しようとしている道路標識は円形であるため、その候補領域は正方形に近い形をしている。そのため、正方形から著しく逸脱する領域は道路標識である可能性が低い。また、これは今回対象外としたグループ 3 の道路標識についても同様のことがいえる。一時停止の標識は三角形をしているが、道路標識が写る領域を矩形領域で切り出せば、その領域は正方形に近い形となる。しかし、実際にはカメラで撮影を行うと、特に画面の端に写る道路標識は歪んで写ってしまう。このため、この歪みを考慮して、本研究では以下の条件を満たすものを道路標識の候補領域である可能性が高いと定めた。

$$\text{短辺} > \text{長辺} \times 0.7$$

短辺、長辺は切り出した矩形領域の縦横の長さを指す。

次に、二つ目の条件についての説明を行う。今回抽出対象としているグループ 1 とグループ 2 の標識は、外周に赤のリング状の領域が存在するが内部には赤が存在しない、または赤の斜線が 1, 2 本存在するという特徴がある。そのため、標識の内部に範囲を限定すると、赤の領域が非常に少なくなっている。本研究では、この内部の赤色の偏りを使って、道路標識ではない抽出領域を除去する。

2.2.9 道路標識内部の赤の比率調査

道路標識とは関係のない領域を除去するため、各標識の内部に占める赤い領域の比率を調査した。調査方法は、道路標識の一覧データからテンプレート画像を作成し、内側のエッジトレースによって得られる矩形領域を切り取る。切り取った画像の中心点を円の中心として円を描き、その円の円周上における赤い画素の比率を求めた。円の半径の大きさは、矩形領域に円が接するまでとした。この調査の様子を Fig.20

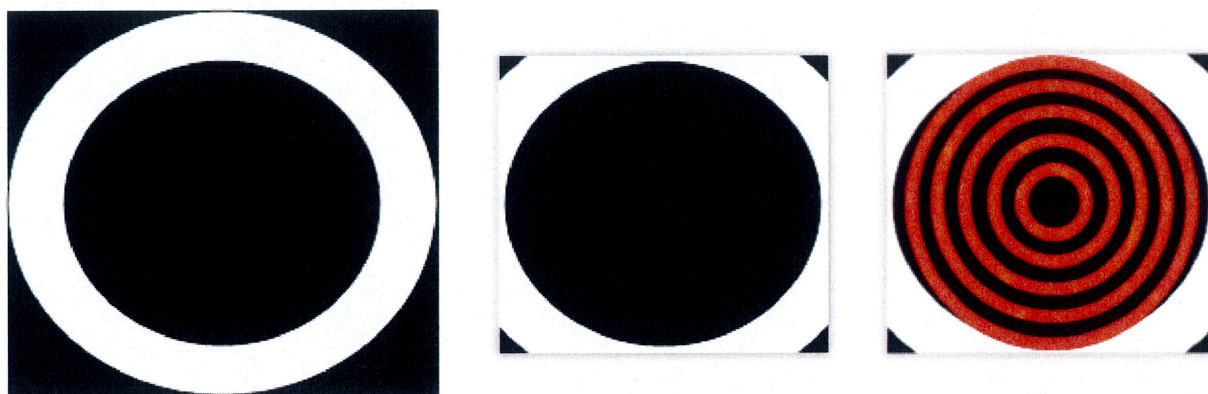


Fig.20 左:外側エッジトレースの結果 中:内側エッジトレースの結果 右:調査する円周のイメージ図

Fig.20 では、左の図が外側のエッジトレースによって抽出される領域であり、真ん中の画像が内側のエッジトレースによって抽出される領域である。そして、真ん中の画像に対して右側の図中にある赤い円の円周上を調査し、各半径毎に赤と赤以外の比率をまとめる。

この調査結果を Fig.21 から Fig.25 に示す。各グラフは横軸が半径、縦軸が円周上における赤以外が占める割合を示している。また、半径はテンプレート画像の短辺の半分までの値をとっている。さらに、この調査はテンプレート画像を縦や横に伸ばした画像についても調査を行った。そして、今回は横の長さを 25% 伸ばした場合の結果を掲載した。そのため、今回使用する値はカメラの歪みを十分に考慮していると考えられる。

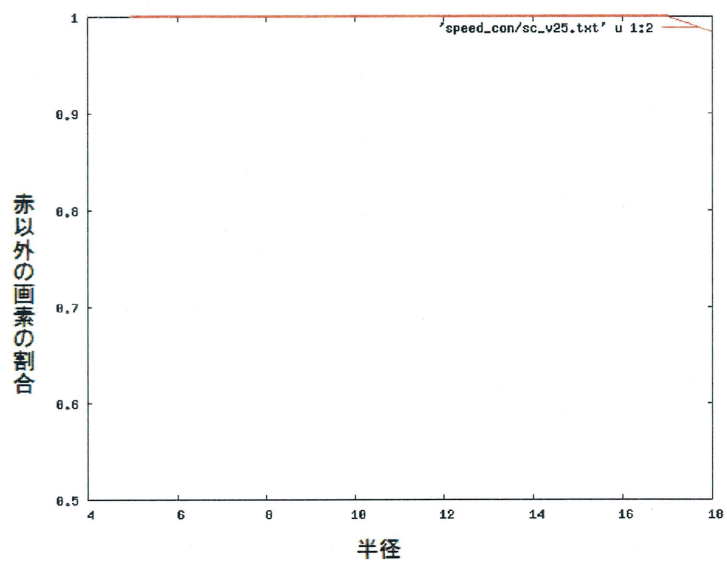


Fig.21 速度制限

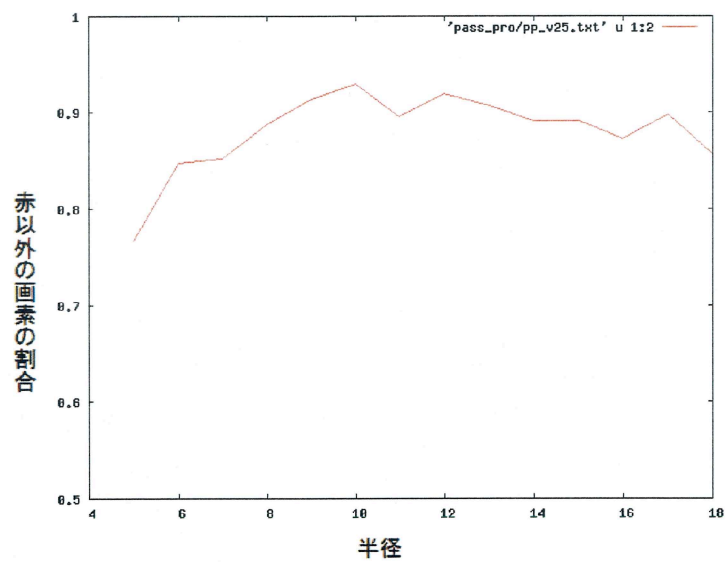


Fig.22 追い越し禁止

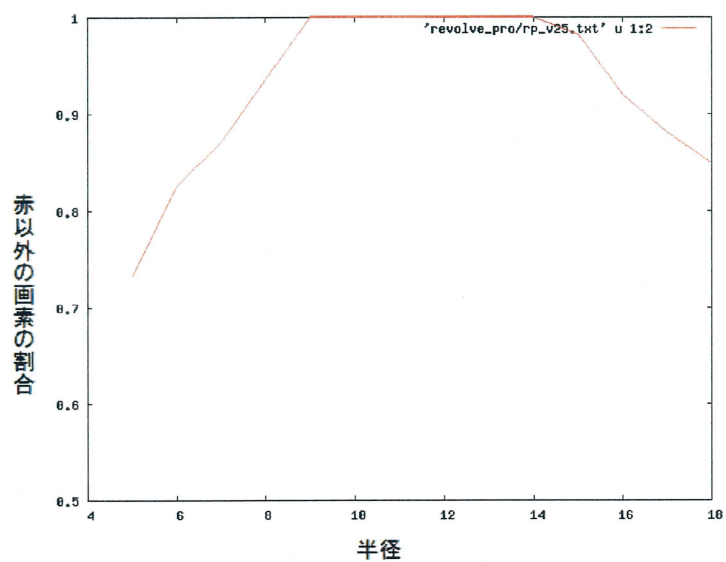


Fig.23 転回禁止

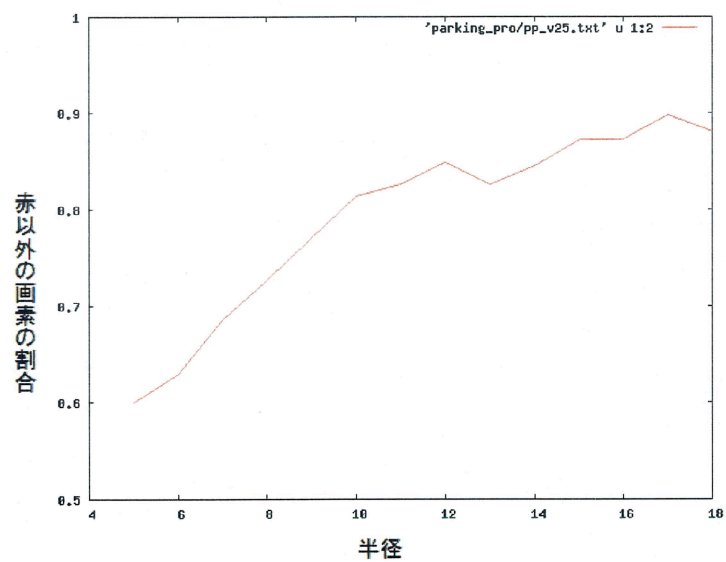


Fig.24 駐車禁止

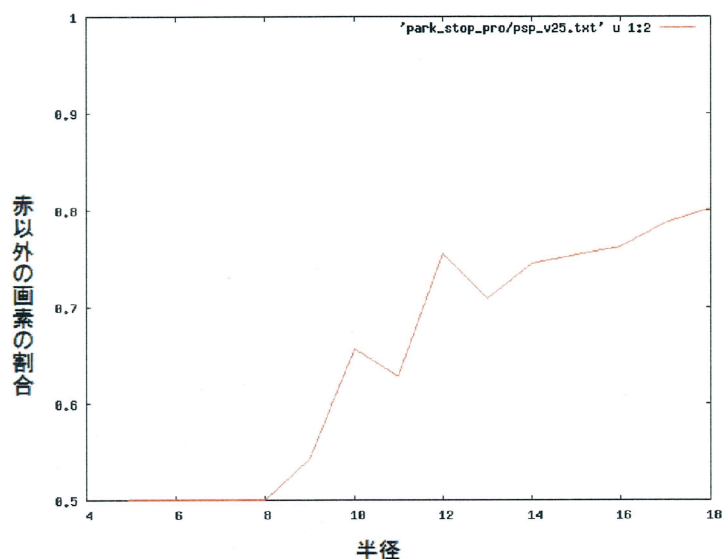


Fig.25 駐停車禁止

Fig.21 から Fig.25 を見ると、最大半径 (=短辺の半分) 付近の円の円周上における赤以外の画素の割合は、だいたい 8 割から 9 割である。そして、内部に赤の斜線が 2 本あり、一番赤い領域が多い駐停車禁止の標識でも、最大半径付近における赤以外の画素の割合は 8 割弱である。しかし、今回の計測には理想的な画像を使用しているため、実際の画像では色のボケやにじみといった要因により、境界線が曖昧になることが予想される。また、境界線のエッジトレースを行っているため、内側のエッジトレースによって得られた矩形領域の短辺を直径 R とした場合、半径 $R/2$ の円周上には赤い画素が多く含まれる可能性がある。そこで本研究では、半径 $(R/2-2)$ の円の円周上を調査し、円周上に占める赤以外の割合が 70% 以上である場合に道路標識である可能性が高いと判断する。

この様子を Fig.26 に図示する。Fig.26 の右側の図中に示す赤の円が、半径 $R/2$ の円である。そして、青で示す円が半径 $(R/2-2)$ の円に当たる。このように、半径を少し小さくすることによって、道路標識の外周にある赤いリング状の領域を避けることができる。

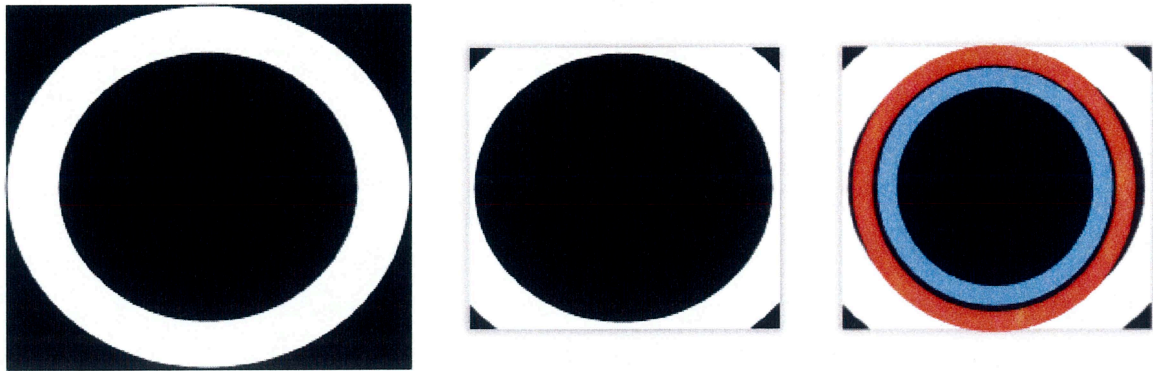


Fig.26 左:外側のエッジトレース結果 中:内側のエッジトレース結果 右:調査する円周

2.2.10 抽出手順のまとめ

ここまでの内側エッジトレースを用いた候補領域の抽出方法の流れを、例を用いて Fig.27 のフローチャートにまとめた。

まず初めに、内側のエッジトレースによって切り出された矩形領域は内部の色判定によって、グループ1とグループ2の候補に分けられる。このとき、駐車禁止や駐停車禁止の内部に存在する赤の斜線の抽出が不十分であっても、内部の色判定を行うことでグループ2の候補として分類することが可能である。そして、グループ1の候補領域と判定された場合、矩形領域の縦横比、内部の赤色と赤以外の色の比率によって最終的な候補領域の判定を行う。

グループ2の候補領域と判定された場合、1回の内側のエッジトレースで抽出される領域だけでは不十分であるため、領域を結合するために周辺の調査を行う。そして、結合条件に合う領域が存在した場合には、新たに発見した領域と結合をして最終的な判定を行う。また、標識内部の赤い斜線の抽出が不十分な場合、この処理で結合されることはない。しかし、赤の斜線が抽出されない場合は、領域が複数に分かれることがないため問題はない。そして、内部の色判定によって青と判定されているため、グループ2の候補領域として抽出することができる。

以上の処理によって、グループ1とグループ2の候補領域を別々に抽出することが可能となる。また、グループ2の標識に関しては、駐車禁止と駐停車禁止の分類も行うことができる。

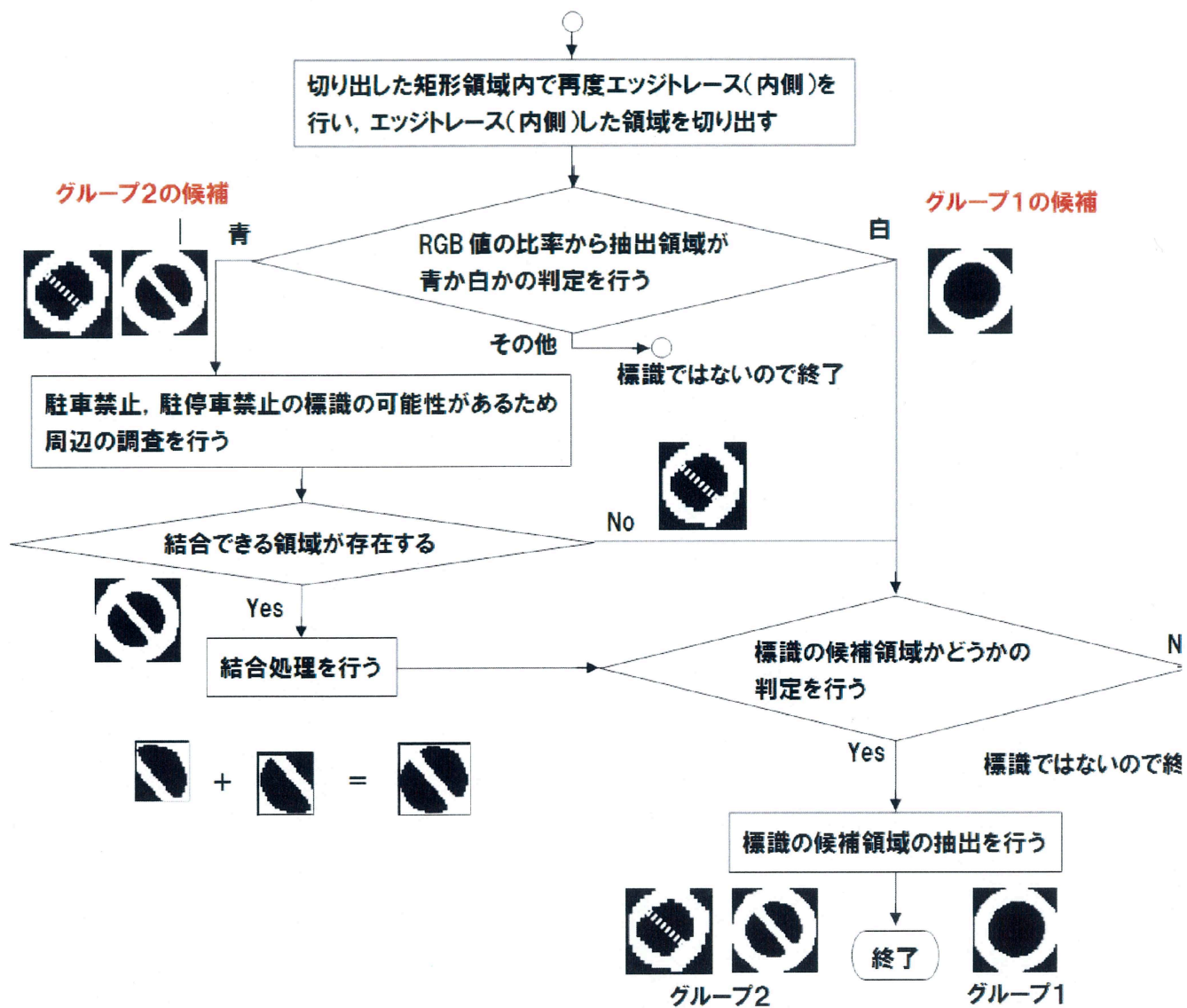


Fig.27 内側エッジを用いた処理のフローチャート

2.3 認識処理

前章で提案した道路交通標識の抽出方法の有効性を証明するために、抽出した領域からシンボル部分を抽出して認識結果に影響がでるかどうかの調査を行った。

シンボル部分の抽出には判別分析法を使い、生成された 2 値画像に対して認識処理を施す。

しかし、先行研究 [13] の認識処理にはいくつかのパラメータに不明な点が存在したため、十分な実装ができなかった。そのため、本研究で用意した認識プログラムを使用して、提案手法によって得られる候補領域から生成される 2 値画像が十分な情報を含んでいるかどうかの調査を行った。

2.3.1 シンボル部分の抽出

前章で提案した抽出方法によって切り出される矩形領域は、Fig.28 に示すように、道路標識以外の余分なものは写っておらず、標識の中でも範囲がかなり限定されている。また、内部の色も赤、青、白の 3 色もしくは赤、青の 2 色とカラーバリエーションが少ないことも特徴である。そこで本研究では、認識処理の中で使用する 2 値画像を生成するために、判別分析法を用いることにした。そして、判別分析法を用いるために使用する濃淡画像には、候補領域のカラー画像から R 成分のみを取り出した画像を使用する。

Fig.28 に示した画像に対して、R 成分のみの濃淡画像と判別分析法を用いて 2 値化を行った結果を Fig.29 に示す。Fig.29 を見ると、シンボル部分 (青色の領域) がきれいに抽出されていることが確認できる。

しかし、Fig.29 の左側の図を見ると分かるように、速度制限のようなグループ 1 に属する標識の候補領域の場合、四隅に標識の赤いリング状の領域の一部も抽出されてしまっていることが分かる。そのため、グループ 1 の候補領域の場合、この雑音除去を行ってから認識処理を行う。

雑音は Fig.29 の左図から分かるように、矩形領域に内接する円の外側の領域に存在する。このことから、内接円の外側の領域を黒で塗りつぶすことで雑音を除去することができる。しかし、候補領域は必ずしも正方形で切り出されるとは限らないため、矩形領域の短辺を円の直径として内接円を使用する。以上の処理により雑音除去された画像を Fig.30 に示す。

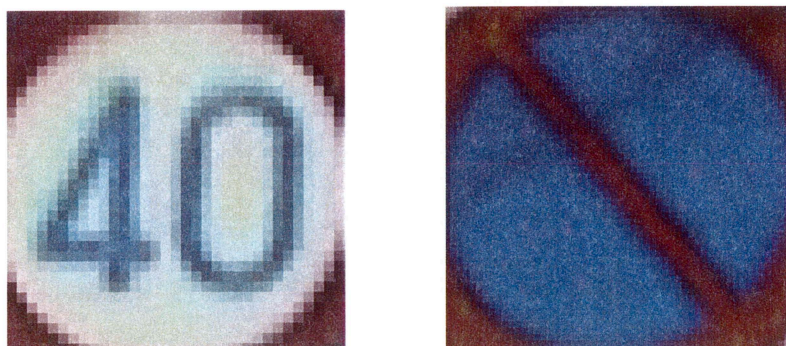


Fig.28 候補領域として切り出される画像の例

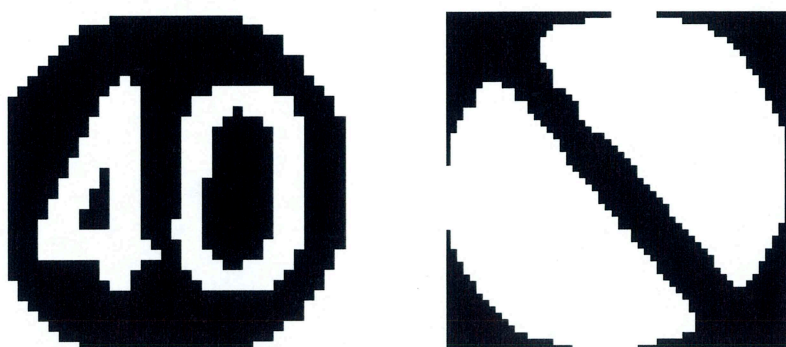


Fig.29 判別分析法を用いた結果の例

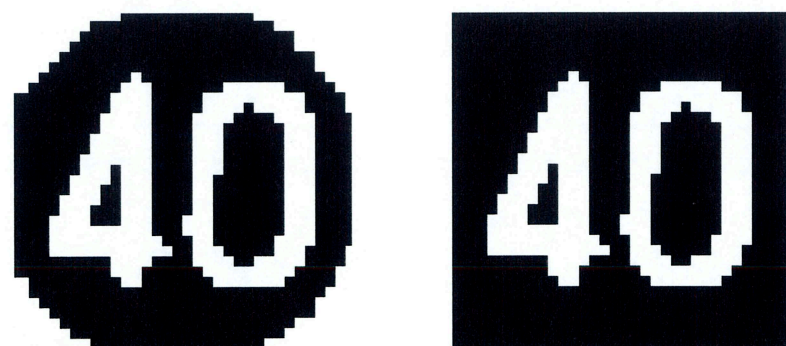


Fig.30 左:雑音除去前の画像 右:雑音除去後の画像

2.3.2 駐車禁止，駐停車禁止の認識方法

駐停車禁止と駐車禁止の認識には，領域内に存在する斜線を検出することで認識を行う．

前項の Fig.29 から分かるように，本手法によって切り出される候補領域は，シンボル部分が画像のほぼ中央に写っている．そのため，Fig.31 に示すように，画像の中心点から放射線状に黒い画素 (=赤の斜線) をカウントして，最も黒い画素が多い線の傾きによって判定を行う．また，調べる斜線の角度は 15 度刻みで行い，斜線は-75 度から 90 度の範囲で調査を行う．そして，各認識の基準は以下のように定めた．

- ・ 駐車禁止

斜線の角度が-75 度以上，-15 度以下の場合

- ・ 駐停車禁止

斜線のピークが-75 度以上，-15 度以下と 15 度以上 75 度以下の二つに存在する場合

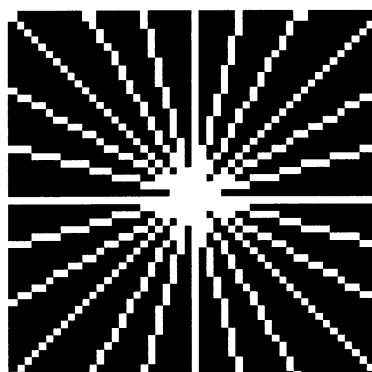


Fig.31 調査する斜線の図

2.3.3 速度制限の認識のための数字領域の抽出

数字の認識を行うためには，まず初めに二つの数字を分離することが必要となる．数字の分離には，候補領域の 2 値画像を x 軸方向へ投影したときのヒストグラムによって行う．候補領域の画像データを x 軸，y 軸それぞれの方向へ投影したときの様子を Fig.32. Fig.33 に示す．

Fig.32 は速度制限 40 の場合の軸方向への投影結果を示しており，Fig.33 は速度制限 30 の場合の軸方向への投影結果を示している．これら二つの投影結果を見て分かるように，速度制限の標識の場合，二つの数字の間に大きな谷ができる．この谷を検出することで二つの数字を分離することができる．また，数字の高さについては y 軸方向へ投影をして，画素がある領域を取り出すことで，上下の無駄な領域を除去できる．



Fig.32 速度制限 40 の場合



Fig.33 速度制限 30 の場合

2.3.4 速度制限の認識方法

速度制限の認識には制限速度を表す 2 桁の数字の内、十の位の数字の判定を行った。なお、今回は 0 の認識を行っていないので、今後は 0 の認識処理を追加する必要がある。

数字の認識は、前項までの方法で範囲を限定した領域に対して行う。そして、左上、右上の 2 方向から下側に向かってエッジトレースを行うことで 3, 4, 5 の数字の判定を行う。この様子を示す図を Fig.34, Fig.35 に示す。以降の説明には、この図を参照してもらいたい。



Fig.34 速度制限の認識方法の解説 (左からのトレース)

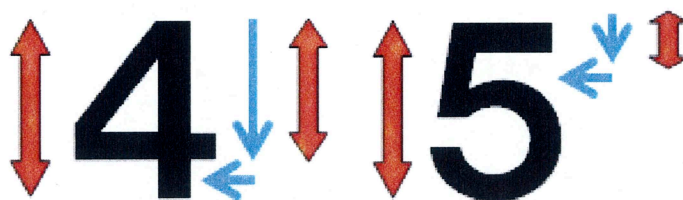


Fig.35 速度制限の認識方法の解説 (右からのトレース)

数字の判定には、二つの高さの比率によって行う。一つ目の高さには、数字全体の長さを使用する。二つ目の高さは、数字の左上もしくは右上から数字の輪郭のトレースを行い、トレースが内側に入り込む点までの長さを求める。そして、この二つの長さの比によって数字の判定を行う。Fig.34, Fig.35 の赤の矢印が二つの高さを表している。

数字の 3 の判定には、左上からのトレース結果を利用する。数字の 3 の場合、左上からトレースを開始すると、全体の高さの半分にも満たない早い段階で内側に入り込む。そのため、左上からのトレースを行った結果、全体の高さに対して比率が小さい場合、数字の 3 と判定を行う。

しかし、数字の 4 と 5 の場合どちらも全体の高さの半分前後となり判定が難しくなる。そこで、右上からも同様の処理を行う。

数字の 4 の場合、右上からトレースを開始しすると、左上からトレースしたときと同様に、トレースした結果の長さは全体の高さに対して比率が大きくなる。反対に、数字の 5 の場合は、左上からトレースをしたときの数字の 3 と同様に、全体の高さに対してトレースした高さの比率が小さ

くなる。よって、右上からのトレースの高さが全体の高さに対して小さくなる場合は、数字の 5 と判定することができる。

以上のことを次にまとめる。

- 数字の 3 の判定

左上から下方に向かってトレースをしたときの高さは、数字全体の高さに比べて比率が小さい。

- 数字の 4 の判定

左上、右上から下方に向かってトレースをしたときの高さは、共に数字全体の高さに比べて比率が大きい。

- 数字の 5 の判定

左上から下方に向かってトレースをしたときの高さは、数字全体の高さに比べて比率が大きい。が、右上から下方に向かってトレースをしたときの高さは、数字全体の高さの比率に比べると小さい。

以上の特徴から、本研究では数字の判定のための閾値を次のように定めた。

Height : 数字の高さ

Left_Height : 左上からトレースして内側に入り込むまでの高さ

Right_Height : 左上からトレースして内側に入り込むまでの高さ

- 数字の 3

$\text{Left_Height} / \text{Height} \leq 0.3$

- 数字の 4

$\text{Left_Height} / \text{Height} \geq 0.6 \ \&\& \ \text{Right_Height} / \text{Height} \geq 0.65$

- 数字の 5

$\text{Left_Height} / \text{Height} \geq 0.4 \ \&\& \ \text{Right_Height} / \text{Height} \leq 0.3$

2.3.5 画像の修正

ここまでの処理で問題とするシーンについては前処理で修正を行う。修正箇所は次の2点である。

- ・入力画像を2値化したときにできる道路標識部の欠けの修正
- ・グループ2に属する標識のシンボル部分抽出方法

まず初めに、入力画像の2値化を行うとき、Fig.36の中図のように道路標識の部分に欠けが生じてしまうことがある。このような欠けが存在すると、提案手法では外側と内側のエッジの区別がつかないため抽出率が大きく低下してしまう。そこで、このような欠けを修正するために馬場らの手法[13]を利用する。馬場らの手法と同様にして、外側のエッジトレースの情報から円形と思われる領域を抽出し、トレース点の情報から円を推定して2値画像に円を描く。これにより、欠けた円を修正することができる。修正した結果をFig.36の右図に示す。

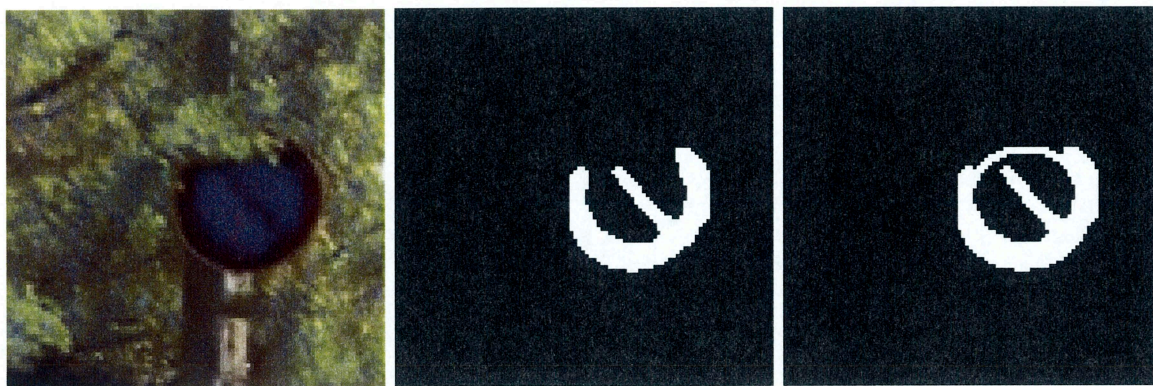


Fig.36 左:入力画像 中:修正前 右:修正後

次に、グループ2に属する道路標識ではFig.37のように標識までの距離が遠く、明度が低い場合、R成分の画像に対して判別分析法を適用してシンボル部分の抽出を行っても、Fig.38の右図に示すように、シンボル部分の抽出が不十分なことがある。そこで、入力する濃淡画像をG成分、B成分、 $Y(= R \times 0.299 + G \times 0.587 + B \times 0.114)$ 成分として、比較調査を行った。その結果をFig.39に示す。この結果から、条件の悪い状態にある標識に対しては、R成分の画像を使用するよりもG成分の画像を使用した方がよいという結果が得られた。そこで本研究では、グループ2に属する標識で悪条件下にあるものに対してはG成分の画像を使用することとした。なお、悪条件下というのは、Fig.38の中図に示すように、内部の斜線が抽出できないケースを指すこととする。また、この変更によって改善は見られても改悪になるものは存在しなかった。



Fig.37 入力画像

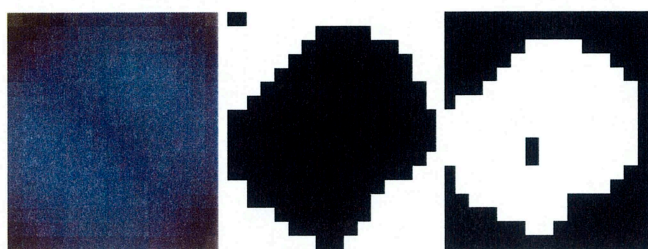


Fig.38 左:抽出された候補領域 中:2 値画像 右:シンボル部分の抽出結果

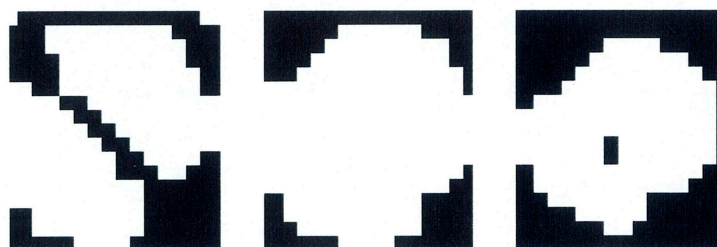


Fig.39 左:G 成分 中:B 成分 右:Y 成分

第3章

実験

3.1 実験環境

提案手法の有用性を確かめるために、車内から見える実際の映像を撮影して評価実験を行った。撮影は11月上旬の晴れた日の午前中に行った。車内からの撮影には、SONYのデジタルビデオカメラハンディカムHDR-CX12を使用し、カメラは車の中心から少し左側に位置する場所に設置した。ハンディカムで撮影した動画は付属のソフトを使用してMPEG-2 TS動画に変換した。そして、Any Video Converter フリー版[22]を使ってavi形式の動画に変換した後、Bara-Baby[23]を使って10fpsの間隔でbmp画像を作成した。画像サイズは720×406である。また、比較対象として馬場らの手法[13]を筆者らが実装したものをを用いた。

使用した動画の撮影時間は約5分程度であるが、標識が設置されている環境には以下のようなものがあり、条件の悪い環境についての調査も行った。

- ・ 順光である（順光とは逆行の反対の言葉であり、撮影者の後ろに光源があることを指す）
- ・ 木の葉っぱなどの影により明度が低下している
- ・ 標識までの距離があり、明度が低下している
- ・ 標識の一部が隠れている
- ・ 標識が縦に2つ並んでいる
- ・ 標識が離れて2つ設置されている

また、明度が低下しているものの中で一番暗いものは、RGB値がそれぞれ30～40程度の値という環境であった。

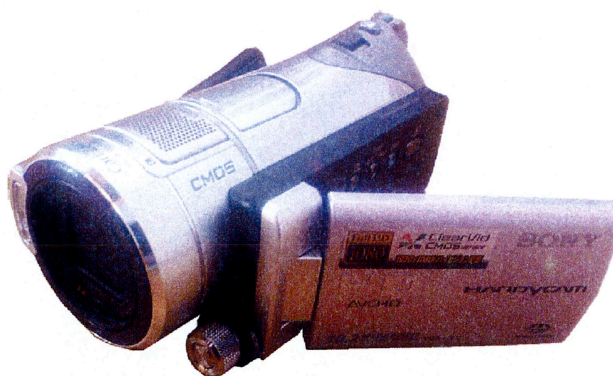


Fig.40 使用したハンディカム：SONY HDR-CX12

3.1.1 抽出結果

先行研究において道路標識の抽出可能な条件というのが曖昧であるため、本研究では抽出可能な標識の定義を以下のように定める。

- ・開始点：提案手法もしくは従来手法 [13] のどちらかで抽出が可能となったフレーム
- ・終了点：道路標識に接近し標識に欠けが生じる手前のフレーム

抽出結果の例を Fig.41, Fig.42 に示し、抽出率を Table2 に示した。



Fig.41 入力画像

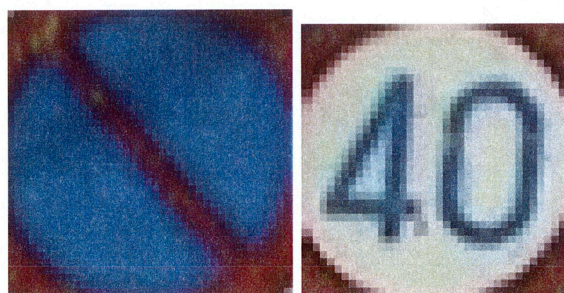


Fig.42 抽出された候補領域

Table2 抽出結果

	提案手法	従来手法
抽出可能な道路標識の個数 [※]	336	336
抽出に成功した標識の個数	312	312
抽出率	92.8%	92.8%

※抽出可能な道路標識の個数とは設置されていた道路標識の個数ではなく、フレーム単位で見た場合の個数である。

今回の実験では、従来手法と抽出率が同じという結果になったが、道路標識の設置場所によって抽出率は異なった。設置場所によって抽出率が異なった原因には次のようなことが考えられる。また、設置場所によって抽出率に差が生じるのは数フレーム程度であった。

まず初めに、提案手法では道路標識の内側のエッジを利用するため、道路標識と背景との間のエッジが明確にでる必要はない。そのため、従来手法よりも道路標識部を抽出するための2値化処理において、閾値の設定を広くすることが可能である。^{*1}

したがって、従来手法では抽出しにくい明度の低い画素も抽出が可能となり、明度が低下するような環境下での抽出率が向上した。

しかし、2値化の閾値を広くしたことによって、順光かつ距離が離れているような条件下では、標識内部がつぶれてしまうことがあった。そのため、従来手法に比べて抽出率が低下するという結果になった。

以上のことから、提案手法は従来手法に比べて暗い環境下での抽出に強いと考えられる。

また、1フレームあたりの平均処理速度はTable3のような結果になり、従来手法に比べて約23%の高速化が実現できた。なお、従来手法において一部の閾値の詳細が不明だったため、速度制限の認識処理を加えた場合、処理時間を正確に計測することは困難と判断した。そのため、従来手法の処理時間には速度制限の認識処理時間が含まれていない。したがって、実際にはTable3の結果よりも数ms増加することが予想される。提案手法の処理時間には速度制限の認識処理時間も含まれている。

Table3に示す平均処理時間は、まず初めに各フレーム毎に5回処理を行い、フレーム毎の平均処理時間を求める。そして、求めた各フレームの平均処理時間の平均を各手法の平均処理時間とし提案手法では、抽出段階でグループ分けを行っているため、限られた種類の認識処理だけを行えばよい。一方、従来手法では全ての候補領域に対して決められた順序で認識処理を行う必要がある。このように、認識処理のプロセスが異なるが、速度制限以降の認識処理手順は同じであるため、今回計測した範囲の処理時間の差が最終的な処理時間の差となる。

^{*1} 3.1.2を参照。

Table3 1 フレームあたりの平均処理速度

	提案手法	従来手法
平均処理速度	88ms	114ms

また、明度が低いなどの条件が悪い環境にある道路標識は全体の約 35% であった。そして、抽出された候補領域の大きさは、道路標識全体の直径が約 20～80 画素の範囲であり、右折左折により表れた道路標識を除いて、直径が約 20～25 画素の範囲から道路標識の抽出が可能だった。このことから、抽出できるサイズは従来研究のものと遜色ないものと考えられる。

処理時間の高速化には、道路標識内部のエッジを利用することで、候補領域中に写る雑音を除去できたことが大きく影響をしていると考えられる。従来手法では雑音が多く写る候補領域からシンボル部分のみを抽出するために、反対色フィルタを使用した処理を行っていたが、反対色フィルタの演算は計算コストが高いというデメリットがあった。しかし、本研究では道路標識内側のエッジと判別分析法を用いることで、道路標識以外の情報を削減し、認識処理前に候補領域を大別できたことによって、処理時間を大幅に短縮することができた。

3.1.2 抽出に関する考察

従来手法 [13] と比較して，提案手法で改善できた例を Fig.43 に示す．



Fig.43 上:改善されたシーン 1 下:2 値画像

Fig.43 に示すシーンでは，光源が横にあり，標識までの距離が遠いため，道路標識の領域の明度が低下している．このようなシーンにおいては，提案手法は従来手法に比べて閾値の範囲を広く設定をしても抽出できる手法を用いているため，抽出率が向上した．

しかし，閾値の範囲を広げたことにより，道路標識と背景の間にできるエッジが極端に少なくなってしまう例を Fig.44 に示す．



Fig.44 上:入力画像 中:境界が消えてしまう例 下:提案手法による抽出結果

認識結果については、Fig.45 のような枠を抽出した領域に対して付け加えることで、抽出された範囲と認識結果が分かるようにした。

速度制限の認識結果は、枠の横にある縦線の数で表現している。また、駐車禁止は内部に縦線を一本入れるという表現にした。



Fig.45 左:速度制限 30 の認識結果 右:駐車禁止の認識結果

Fig.44 中の画像から、道路標識と背後の柱との間にある境界が消えてしまっていることが分かる。しかし、提案手法では内側のエッジを利用して抽出するため、Fig.44 ような条件下においても抽出が可能である。また、同シーンにおいて従来手法の閾値を使用すると Fig.46 のような結果になる。



Fig.46 閾値を厳しくした場合の 2 値画像

従来手法において、Fig.44 のように道路標識のほんの一部しかエッジトレースができない条件下では、道路標識の抽出が困難であるとされている。そのため、従来手法では閾値の範囲を広く設定することはできず、本研究と同様の閾値設定をした場合は抽出率が低下してしまう。

また、今回の結果から 2 値化の閾値を広く設定しても抽出率に影響がないことが確認でき、背景に同色の物体が存在するようなシーンにおいても抽出が可能ながことが証明された。

3.1.3 認識結果

各手法ごとに、それぞれ抽出できた道路候補領域に対して、現段階で実装できている認識処理を施した。道路交通標識の認識には前章で説明をした方法を使用している。

しかし、今回参考にした馬場らの手法 [13] において、候補領域からシンボル部分を抽出する処理で、パラメータの一部に不明な点があった。そのため、可能な範囲内でプログラム作成を行った。したがって、今回筆者らが論文をもとに実装したプログラムによる認識結果と先行研究 [13] の認識結果には誤差が生じている可能性が高いことを予め断っておく。また、従来研究では認識対象を直径が 30 画素以上の標識に限定しているが、今回使用した画像の約半数は直径が 30 未満であった。

Table4 認識結果

	提案手法		従来手法	
	個数	率 (%)	個数	率 (%)
正しく認識できた個数	307	98.3	288	92.3
誤認識された個数	1	0.3	0	0
認識されなかった個数	4	1.28	24	7.6

今回提案した認識方法を使用して、提案手法によって抽出した道路標識のほとんどが認識可能だった。しかし、一部誤認識をしてしまったケースや、認識ができなかったケースが存在した。誤認識をしてしまった原因は、Fig.47 に示す赤の矢印のように、数字の高さの検出処理が若干の凹凸の影響を受けてしまったためであった。このことから、若干の凹凸に左右されないような認識方法を構築する必要があることが分かった。また、認識されなかったシーンでは、Fig.48 のように、道路標識のシンボル部分に電線の影が写り込んでしまった例があった。この電線の影によって数字の形が変化してしまった。しかし、このシーンにおいては、従来手法を用いても電線の影を取り除くことができなかった。そして、道路標識の前に雑音が存在する場合の認識については、従来研究においても問題視されているため、認識処理に従来研究の手法を用いても、提案手法による抽出方法で認識率が低下することはないと考えられる。



Fig.47 誤認識した例

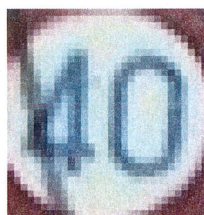


Fig.48 認識できなかった例

3.1.4 認識に関する考察

従来研究において認識できた道路標識の大きさを明記しているものでは，認識には直径で 30 画素以上を対象としている研究 [13] や認識には直径で 30 画素は必要だとしている研究 [12][16] がある．そこで，提案手法で抽出された道路標識を 30 画素未満と 30 画素以上の 2 つに分類したときの認識率を Table5 に示す．

Table5 サイズ別の認識率

	30 画素未満	30 画素以上
標識の個数	149 個	163 個
認識できた個数	149 個	159 個
認識率	100%	97.5%

今回の実験では，30 画素以上の方が認識率が悪いという結果になったが，正しく認識ができなかった画像は，道路標識のシンボル部分に電線の影などが写りこんでしまっている環境下であった．このようなシーンでは，従来手法の認識方法を用いても正しく認識することは困難である．また，直径が 20 画素程度の大きさの道路標識のシンボル部分の抽出結果*2を Fig.49 に示す．

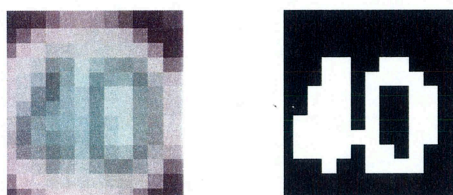


Fig.49 直径が 20 画素の道路標識の抽出結果

直径が 20 画素程度しかない道路標識では，Fig.49 のように数字の内部がつぶれてしまうことがあるが，輪郭はきれいに出的ているため，十分に認識が可能と考えられる．

以上のことから，今回提案した方法で作成されるシンボル部分の 2 値画像は，直径が 20 画素程度であっても認識に十分な情報を含んでいると考えられる．そのため，抽出段階で提案手法を用いることで，従来手法に比べて早期の段階で道路標識の認識が可能であると予想される．また，今回認識に失敗したシーンにおいては，過去の標識の抽出画像を重ね合わせることで隠れた領域を補う山内らの手法 [17] などを使用することで，認識率を向上させることができると考えられる．

*2 道路標識全体の直径は 20 画素程度だが，内側エッジを用いて抽出した結果を示しているため，Fig.49 に写っている画像は 20 画素より小さい

まとめと今後の課題

本論文では、道路交通標識を実時間で認識するシステムにおける抽出方法の提案をした。道路標識の赤色を含む規制標識を対象とし、HSV 表色系を用いて 2 値画像を作成した。道路標識の候補領域の抽出には、対象となる道路標識を 2 値化した際にできる外側と内側の 2 つのエッジと、標識内部の構造と色を利用した。標識のシンボル部分の抽出には判別分析法を用いた。そして、シンボル部分の構造的な特徴を用いた認識処理を実装し、認識処理にも影響がないことを確認した。

また、従来手法では使われていなかった内側のエッジをトレースすることで、標識以外の情報を含まない候補領域の抽出が可能となった。さらに、道路標識の候補領域を抽出する段階での大別が可能となり、従来手法と比べて約 23% もの高速化が実現できた。そして、候補領域内の雑音を軽減できたことにより、道路標識の大きさが 20 画素程度であっても認識が可能であった。

以上より、本手法は従来手法に比べてより早期に標識を認識できるため、運転者に対しての警告等の遅れを短縮でき、標識の見落としや事故の可能性を減らすことができると期待される。

まず、従来手法よりも抽出精度が低下したシーンの改善が挙げられる。これは、2 値化を行ったときに道路標識部分に欠けが生じたり、道路標識の内部がつぶれてしまい内側のエッジトレースができないことが問題であった。前者の問題に対しては、外側のエッジトレースの情報を利用して穴埋めをする処理を追加したが、現状では処理が不十分であるため修正する必要がある。また、様々な色の道路標識を認識対象とする場合、入力画像に対して赤、青、白、黄、黒などの多値化が行われる。そこで、本研究でも多値化を行うことで、後者の問題に対して、外側のエッジトレースを赤い画素で行い、内側のエッジトレースを青もしくは白の画素でトレースを行うことで抽出率が向上することが期待できる。

また、赤色を含む規制標識の内、一時停止と侵入禁止の標識については、新しい抽出方法を確立できなかったため、これらの抽出方法を追加実装する必要がある。

最後に、今回は認識処理が全体的に不十分であったため、他の種類の認識処理の追加、認識精度向上の処理の追加を行う必要がある。

謝辞

日ごろから多くの御指導を頂きました太田義勝教授，鈴木秀智准教授に深く感謝いたします。そして，日頃何かとお世話になりました落合美子事務員に感謝いたします。また，本論文作成にあたって特にお世話になりました鈴木秀智准教授に深く感謝いたします。最後に，日頃から熱心に討論して頂いた研究室の諸氏に感謝いたします。

参考文献

- [1] ”警視庁ホームページ”, <http://www.npa.go.jp/toukei/index.htm#koutsuu>
- [2] <http://www.pref.saitama.lg.jp/s-gikai/gaiyou/h1502/1502j050.html>
- [3] ”国土交通省道路局 ITS ホームページ” <http://www.mlit.go.jp/road/ITS/j-html/>
- [4] Arturo de la Escalera, Luis E.Moreno, Miguel Angel Salichs, Jose Maria Armingol, ”Road Traffic Sign Detection and Classification”, CICYT Project, TAP94-0711-C03-02, 1997.
- [5] 井上泰夫, 石川直人, 中島真人, ”道路標識の自動認識”, 電子情報通信学会技術研究報告, ITS2001-44, pp.67-72, 2002.
- [6] 松浦大祐, 山内仁, 高橋浩光, ”特定色判別と領域限定を用いた円形道路標識の抽出”, 電子情報通信学会論文誌, D-II, Vol.J85-D-II, No.6, pp.1075-1083, 2002.
- [7] 石塚裕, 平井有三, ”色情報を用いた実時間道路交通標識の切り出し”, 電子情報通信学会技術研究報告, ITS2002-66, pp.139-144, 2003.
- [8] 莫舸舸, 青木由直, ”カラー画像における道路標識の認識”, 電子情報通信学会論文誌, D-II, Vol.J87-D-II, No.12, pp.2124-2135, 2004.
- [9] 石塚裕, 平井有三, ”Opponent-Color フィルタを用いた道路標識認識システム”, 電子情報通信学会技術研究報告, PRMU2003-258, pp.13-18, 2004.
- [10] 小橋雄一郎, 石川直人, 中島真人, ”道路標識の自動認識一夜間における道路標識認識”, 電子情報通信学会技術研究報告, ITS2003-98, pp.57-62, 2004.
- [11] 馬場今日子, 平井有三, ”車載カメラ画像による道路交通標識のリアルタイム認識”, 電子情報通信学会技術研究報告, ITS2004-48, pp.45-50, 2005.
- [12] 高木雅成, 藤吉弘亘, ”SIFT 特徴量を用いた交通道路標識認識” 第 13 回画像センシングシンポジウム予稿集, SSII07, LD2-06, 2007.
- [13] 馬場今日子, ”反対色フィルタを用いた道路交通標識の実時間認識に関する研究”, 修士論文, 筑波大学大学院, 2007.03.
- [14] 岡田大輝, 和田俊和, ”動的色境界の提案と道路標識追跡・認識への応用”, 情報処理学会 研究報告, 2008-CVIM-162(29), pp.173-180.
- [15] Miguel S.Prieto, Alastair R.Allen, ”Using self-organising maps in the detection and recognition of road signs”, Image and Vision Computing, Volume 27, pp.673-683, 2009.
- [16] 高木雅成, 藤吉弘亘, ”SIFT 特徴量を用いた交通道路標識認識”, 電学論 C, 129 巻 5 号, pp.824-831, 2009.

- [17] 山内仁, 小島篤博, 宮本貴朗, 高橋浩光, 福永邦雄, "連続フレームからの抽出画像の重ね合わせによる障害物遮蔽を考慮した道路標識認識", 電子情報通信学会技術研究報告, ITS2005-32. pp.1-6, 2005.
- [18] 山内仁, 小島篤博, 宮本貴朗, 高橋浩光, "連続フレームからの抽出画像の重ね合わせによるロバストな道路標識認識手法", 電子情報通信学会論文誌, D. Vol.J90-D. No.2. pp.494-502. 2007.
- [19] Pedro Gil Jimenez. Saturnino Maldonado Bascon. Hilario Gomez Moreno. Sergio Lafuente Arroyo. Francisco Lopez Ferreras. "Traffic sign shape classification and localization based on the normalized FFT of the signature of blobs and 2D homographies". Signal Processing. Volume 88, pp.2943-2955, 2008.
- [20] 杉本真崇, 長橋怜, 菅谷保之, 金澤靖, 伊藤吉弘, "地理情報システムのための車載カメラ映像からの道路標識の認識", 電子情報処理学会技術研究報告, PRMU2008-21, pp.13-18, 2008.
- [21] "株式会社パスコホームページ". <http://www.pasco.co.jp/products/survey/ground/pscruiser/>
- [22] "AnySoft ホームページ". <http://jp.any-video-converter.com/>
- [23] "LNSOFT OFFICAL WEB 2010" <http://www.lnsoft.net/index.htm>
- [24] 田村秀行, "コンピュータ画像処理", 株式会社オーム社, 2002.

付録

[プログラムの保存場所]

/home/fukuta/fuku/Sign

[使用しているファイル]

RecogSignMain.c. RecogSignProcess.c.RecogSignProcess.h

全ての処理は RecogSignMain.c から呼び出される。そして、RecogSignProcess.c には各関数を実装し、RecogSignProcess.h には関数の宣言や構造体の定義がしてある。

[実行方法]

上記ファイル内に Makefile があるため、make コマンドを実行すると RecogSign という実行ファイルが生成される。また、入力画像の指定は RecogSignMain.c 内で行っている。

[プログラムを理解するのに必要となる知識について]

今回作成したプログラムは C 言語を使用している。そのため、C 言語と画像処理の基本的な知識が必須となる。また、C 言語については、画像データ (2 次元データ) を 1 次元配列で扱ったり、動的確保などの処理を行っているため、以下に示す知識も必要となる。

- 配列 (画像の取扱い)
- ppm, pgm 形式の画像処理 (読み込み, 書き込みなど)
- 動的確保 (malloc)
- ポインタ (ダブルポインタは使用していない)
- 構造体
- リスト

[グローバル変数について]

プログラム内で使用サイズが確定しているもの、例えば入力画像などはグローバル変数で定義をしている。グローバル変数については、各ファイルの先頭を見れば分かる。

[問題点]

今回作成したプログラムにはいくつか問題点が残っている。この問題を解決するには、処理内容を理解し、プログラムを一度書き直した方が良いだろう。また、領域を動的に確保しているため、領域の解放処理などの見直しも必要となる。詳しくはプログラムが保存してあるフォルダ内に README.txt を残しておく。