

博士学位論文

ウェアラブル視覚補助システムのための
画像認識・理解に関する研究

Study on Image Recognition and Image Understanding
Techniques for Wearable Low Vision Aids

2013年3月

九州工業大学大学院 工学府 電気電子工学専攻

折居 英章

目次

第 1 章	序論	2
1.1	ウェアラブルコンピューティングの概観	2
1.2	弱視者とは	3
1.3	視覚補助システムの従来研究と本研究の位置づけ	8
第 2 章	視覚情報変換による弱視者向け文字情報認識支援	9
2.1	概要	9
2.2	視覚情報からの文字領域抽出	9
2.2.1	文字領域抽出の従来研究	10
2.2.2	提案手法	26
2.2.3	実験	38
2.2.4	まとめ	46
2.3	弱視者の視覚特性に基づく事前学習による文字構造変換法	47
2.3.1	事前学習による文字構造変換の必要性	47
2.3.2	弱視	47
2.3.3	提案手法	50
2.3.4	構造変換アルゴリズム	55
2.3.5	予備実験	58
2.3.6	評価実験	64
2.3.7	まとめ	67
第 3 章	視覚情報フィルタリングによる弱視者の自律的移動支援システム	68
3.1	概要	68
3.2	ウェアラブルカメラから得られる視覚情報からの歩行可能領域抽出	69
3.3	画像特徴	70
3.3.1	SIFT 特徴量	70
3.3.2	Haar-Like 特徴量	73
3.3.3	SURF 特徴量	74
3.3.4	画像間における対応点探索によるマッチング	76
3.3.5	HOG 特徴量を用いた人検出	76
3.4	提案手法	78
3.4.1	概要	78
3.4.2	ホモグラフィ行列を用いた平面抽出	79
3.4.3	ホモグラフィ行列の最適化	84
3.4.4	領域分割を用いた平面と物体の分離	86
3.5	実験	90
3.5.1	実験環境	90

3.5.2 結果と考察	90
3.6 まとめ	91
第4章 結論	95
謝辞	97
参考文献	98

第1章 序論

1.1 ウェアラブルコンピューティングの概観

近年、コンピュータや各種センサの小型化・軽量化を背景に、人間が身につけることができるウェアラブルコンピューティングに関する研究や実用化への取り組みが盛んに行われている。ウェアラブルコンピューティングとは、コンピュータを常時装着して利用するスタイルのことを言う。このシステムの利用例として、センサを用いて行動認識を行い、ユーザに様々なサービスを行う研究が行われている。例えば健康管理の分野では、東芝が Life Minder[1] という生体センサを利用したユーザ状況の監視システムの開発を行っている。これは、脈拍センサや皮膚温センサと加速度センサを用いて、薬の飲み忘れや運動不足などの指導を行うシステムである。また、観光の分野では、現在地や状況、見ているものに応じた情報提示を行う平城京跡ナビ [2] や博物館での情報提示支援 [3] などが提案されている。このようなシステムでは、情報を取得するためのセンサとして、GPS などのほかに小型カメラなどの視覚センサが用いられることが多い。その際、ユーザの状態や周辺の環境を高度に認識し、適切な情報をユーザに提示することが要求される。

一方、ウェアラブルシステムを用いた弱視力者や視覚障害者の支援を行う研究開発も進んでいる。例えば、Bryant らは、赤外光投影できるビデオカメラを利用して近接した障害物を検出して、その位置をファイバースキャニング網膜ディスプレイに提示するウェアラブルシステム [4] を開発している。Alvaro らは周囲の危険状況を超音波センサで読み取って危険性を音声や振動で通知するシステム [5] を提案している。久野らはタッチパネル操作支援システム [6] の研究を行っている。これは、まずユーザーがマイクを通じてシステムに指示を与え、タッチパネル操作に必要な情報を装着しているステレオカメラから取得し音声でユーザに出力するものである。

本研究では、とくに、老眼や白内障といった多くの高齢者がかかえる視覚障害を含む「弱視」と呼ばれる視覚障害に着目し、「弱視者」の視覚補助システムへの応用を目指した画像認

識アルゴリズムの開発を行う。

1.2 弱視力者とは

老眼や白内障といった多くの高齢者がかかえる視覚障害を含む「弱視」または「ロービジョン」と呼ばれる視覚障害により、日常生活に不自由を感じている人は多い。弱視力者の定義は様々な見方があるが、世界保健機関 (WHO) の定義では「両眼の矯正視力が 0.05 から 0.3 のもの、または視力以外の視機能障害があり、学習や日常生活に制約があるが、主として視覚における様々な行動ができる者」と言われている。「矯正視力が 0.05 から 0.3」とは、メガネやコンタクトレンズなどの矯正器具を用いても視力が十分に出ない状態のことをいう。また、両眼の視力が 0.05 未満である場合を盲 (blindness) と呼ばれる。日本眼科医学会によると、弱視力者を含む視覚障害者は 2007 年現在約 164 万人に上ると推定されている [7]。そのうち矯正視力が 0.1~0.5 である弱視力者の人数は、144 万 9000 人いると推定されている。また、高齢化に伴って、老眼や白内障の視覚障害の人数が増加している。今後も増加が予想され、2030 年頃には 200 万人を超えると予想される。視覚障害者の中で弱視者は、まったく視力がない全盲より遥かに多く約 8 割を占めている。また弱視にも視力・視野、夜盲や羞明 (しゅうめい)、疲労度や矯正後の視力の程度など個人差が幅が広く、大変理解されにくい。全盲者とは異なり、弱視者とはまったく視覚をもたないわけではなく、日常の諸行動ができる。したがって弱視者の外見からは目が不自由であるということが相手にも分かりにくく自分から言い出せずにいる人も多いため、実態が社会的に十分理解されていないというのが現状である。

弱視力者の見え方は一様ではなく、症状によって個人差が大きい。また、一人の弱視力者でも体調や周りの環境に左右されて見え方は変化する。以下に具体的な見え方とその症状の内容を紹介する。

1) ぼやけて見える

対象物を見る際にピントが合わずぼやけて見える。症状としては、次の (a)~(c) に分けることができる。

(a) 近眼 (Myopia)

晴眼者の場合、目に入った光は網膜に映し出され、網膜でピントが合うが、近視の場合遠

くを見たときに網膜よりも手前に光の焦点が結ばれるため、ピントが合わない。そのため、対象物がぼんやり見えてしまう屈折異常のことをいう。角膜や水晶体による光の屈折率が強すぎることによる「屈折性近視」、網膜よりも前方に焦点を結んでしまう「軸性近視」がある。軸性近視は、遺伝的要素が多いと言われている。

(b) 遠視 (Hyperopia)

近視の逆で、ピントが網膜よりも後ろに焦点が結ばれて、ピントが合わないことをいう。水晶体などの屈折力が弱い場合を「屈折性遠視」という。また、白内障の手術を行い水晶体を取り去った状態を「無水晶体症」という。

(c) 老眼 (Presbyopia)

年齢とともに、調節力の主因である水晶体の弾力性が弱まり、近点も遠ざかる。近点とは、極度に調節を行ったときに明瞭に見えるもっとも近い外界の点のことを言う。遠点と近点の間の空間の点は調節により明視することができるので調節域、もしくは調節幅と呼ばれる。調節幅はその目の調節力を示すものである。角膜頂点から近点までの距離を近点距離という。遠視の場合では大きく、近視の場合だと小さい。老人ほど遠く若年ほど近い。このように、年齢共に調節力が減少し、近点が明視の距離である 30cm 以上に遠ざかったものを老眼と呼ばれる。

2) 歪んで見える

まっすぐな物が曲がって見えること。網膜剥離の患者は、この状態になることが多いと言われている。また、視界の真ん中がゆがむこともある。

3) 視野が制限される

見える範囲がどんどん狭くなり見えにくくなる状態。視野の欠ける部分は個人差があり、端だけが書ける場合や中心が欠ける場合もある。目の前にあるものがちょうど視野の欠けた部分に入ると、まったく何もないように見える。網膜剥離や緑内障などの患者に起こりやすい。

4) 虫が飛んでいるように見える

飛蚊症といわれる症状で、蚊やゴミが目の中に飛んでいるように見える。硝子体の老化が原因で起こる。これは、硝子体がはがれて起こるもので、後部硝子体剥離や網膜裂孔といっ

た病気を引き起こし失明する場合もある。

5) 光が走る

光視症と呼ばれる症状で、角膜の異常、網膜と硝子体の癒着、脳の異常などが原因といわれている。角膜に異常がある場合は、電灯などの光の周りに虹のような輪ができ、脳に異常がある場合は、中心から外に向かって光の輪が広がり消えていく。

6) 物が2つに見える

左右の目の状態が著しく違う状態の時などに起こる。片目複眼や眼筋麻痺などが多い。

7) 混濁する

白内障といい、水晶体が白く濁ることにより起こる。混濁は水晶体たんぱくの変化によって起こる。一般的に多いのが60歳以上に発生する老人性白内障である。老人性白内障は進行の度合いが異なるが、ほとんどの高齢者に現れる。初期の場合、精密な検査を行わないと気づきにくい。

8) まぶしい

暗い部屋から明るい場所へ出た時などに起こる。特に昼間の明るい時間帯はこの現象が起きやすい。

これらの各症状を持つ状態における弱視者の見え方を図1.2～図1.7に示す（出典：[8]）。図1.1の晴眼者の見え方のイメージ図に対して、図1.2は近視や遠視・老眼などの症状であるピンボケが起こった場合の見え方、図1.3は白内障の症状の場合の見え方、図1.4、図1.5、図1.6は視野狭窄の場合の見え方、図1.7は明順応障害の場合の見え方をそれぞれ示している。図を見てもわかるように、各症状に対して統一した対応方法を取ることは難しく、その支援を行うためには弱視力者それぞれの症状に合わせた対策を取ることが必要となる。



図 1.1: 晴眼者の見え方



図 1.2: ぼやけて見える場合



図 1.3: 混濁およびまぶしい場合



図 1.4: 視野狭窄 (30 度) の場合



図 1.5: 視野狭窄 (10 度) の場合



図 1.6: 中心に暗点がある場合



図 1.7: 明順応障害の場合

1.3 視覚補助システムの従来研究と本研究の位置づけ

前節で述べたように、弱視者は視覚に対して様々な制約を持ち、全盲でないにも関わらず視覚によって得られる情報は非常に限定的なものとなる。そのため、街中の看板・標識、本や雑誌の文字、機器や端末の操作法など、生活空間にあふれる重要なサインや文字情報を認識し難く、弱視者は見えているのに「見えづらい」という矛盾した環境の中で不便な生活を強いられている。このような問題に対し、さまざまな視覚支援システムの開発が取り組まれている。例えば、丸茂ら [9] や田中ら [10] は、文字認識を行いユーザへ音声で伝達する支援技術を提案している。しかし、紙面などと異なり、日常生活に存在する任意の文字情報を認識するのは技術的な課題がある。また、Sakamaki ら [11] は画像拡大技術を視覚支援に用いるシステムを提案している。これは視覚情報を拡大して提示することによりユーザの視認性を高めるというものである。しかし、これまでの拡大技術が再現性を最大の目的として発展してきたことから、ただの拡大では実現できない「ユーザの見やすさ」への配慮はなされていない。また、屈折異常を補正する器具として、古くから眼鏡、コンタクトレンズ、ルーペ、拡大鏡などが存在する。これらは、常に一定の倍率（手動で切り替え）で拡大を行う。そのため、眼鏡による補正が不必要な時も拡大されることや、加齢などにより視力が変化する度に新たなレンズを作り直す必要がある。近年、小型カメラ付きゴーグル型の電子メガネ [12] が開発されている。これは、網膜に直接映像光を投射して、視野映像を拡大する装置である。操作ボタンを押すことで見たい倍率に拡大を行う。網膜投影を行うことで、ディスプレイ提示型のヘッドマウントディスプレイよりも鮮明に見ることが可能である。しかし、網膜投影型ディスプレイは、光路中の障害物がすべて結像するため、目の中のゴミやまつ毛が映ることが課題である。もう一つは、瞳孔や眼球運動により映像の欠損が生じることも問題である。

本研究では、実世界の事物に対して直接情報を付与する拡張現実感 (Augmented Reality: AR) を利用した視覚補助システムへの応用を前提に、実世界から情報を取得する画像認識アルゴリズムの開発及び視覚補助システムへの適用検討を行う。具体的には、視覚情報変換による弱視者向け文字情報認識支援アルゴリズムの構築、視覚情報フィルタリングによる弱視者の自律的移動支援システムの開発を行う。

第2章 視覚情報変換による

弱視者向け文字情報認識支援

2.1 概要

老眼や白内障といった多くの高齢者がかかえる視覚障害を含む「弱視」と呼ばれる視覚障害により、日常生活に不自由を感じている人は多い。とくに、街中の看板・標識、本や雑誌、機器や端末の操作法など、生活空間にあふれる文字情報を見ることが困難な場合、日常生活に大きな支障をきたす。このような問題に対して、文字認識を行いユーザへ音声で伝達する支援技術が提案されているが、紙面などと異なり、日常生活に存在する任意の文字情報には「文字の修飾」や「射影ひずみ」などが存在し安定して認識・抽出を行うのには技術的な課題がある。また、抽出した文字情報をどのようにユーザへ提示すればユーザに「見やすい」と感じてもらえるかについても配慮されていないのが現状である。そこで本研究では、「ウェアラブルカメラから取得される視覚情報からの文字領域抽出法」、及び、「ユーザが“見やすい”文字像へ変換し提示するアルゴリズム」の開発を行う。そして、それらの視覚情報変換による弱視者向け文字情報認識支援システムへの応用について検討する。

以降、2.2節では生活空間における視覚情報からの文字領域抽出処理について、2.3節では弱視者の視覚特性を考慮した文字画像の「見やすい」像への変換処理について詳述する。

2.2 視覚情報からの文字領域抽出

本節では視覚情報からの文字領域抽出処理について検討する。提案手法では、情景画像内において文字情報は看板などの人工物上に存在すると仮定し、「文字存在領域選定」と「文字成分と背景成分の推定」の二つのプロセスにおいて文字抽出を実現する。「文字存在領域選定」では、文字存在領域、すなわち看板領域を選定する。屋外環境下においてカメラを用いて撮影を行う場合、情景画像には文字領域だけでなく様々なものが含まれるために、文字抽出が困難になってしまう。そこであらかじめ看板領域を選び出し、その領域に対して文字抽

出処理を行うようにすることで、処理の簡単化と抽出精度向上を実現する。画像内の看板領域は色情報、形状情報などの条件から抽出し、選定する。「文字成分と背景成分の推定」では、看板領域内において文字成分の背景からの分離を行う。情景画像では環境光などの反射による光沢や周囲環境に起因する影のような劣化が生じるため、文字抽出は困難になる。そこでそのような劣化に依存しない、2値化処理の手法を確立する。MAP-MRFによる定式化に、背景と文字それぞれの輝度値を推定する新たなアルゴリズムを導入することにより確率モデルを定義し、最適化問題をグラフカットアルゴリズムを用いて解くことで文字抽出を実現する。提案手法を種々の情景画像に適用し、その有効性を示す。

2.2.1 文字領域抽出の従来研究

2.2.1.1 Wangらの手法

Wangらは、複雑な背景下でも文字を抽出できる手法を提案している [13]。これは色クラスタリングによる領域分割を応用した手法であるが、クラスタリングの計算コストが大きくなるという問題がある。以下、Wangらの手法について詳述する。

(1) 前処理

情景画像は複数のオブジェクトで構成されている。しかし、情景画像内で同じオブジェクトを表している画素が必ずしも似たようなRGB値を保持しているとは限らない。これは照明の影響などオブジェクトの周囲環境を原因として起こる。また、情景画像を取得する際に何らかの原因によって画像中にノイズが混入してしまう状況も考えられる。文字とそれ以外の部分とを良好に分離する2値画像を作成しようとする際、画像内の同一オブジェクトが似たようなRGB値を保持していない場合、同一オブジェクトであるにも関わらず別領域として2値化されてしまうことが起こりうる。またノイズが混入している場合、同一オブジェクトを無駄に分割したり、統合してしまい良好な文字列領域の抽出を困難にさせる。そこで、上下左右4方向からのRGB値の変化分に対し、明度値にメディアンフィルタを掛けた値の差分で重み付けをして平滑化を行うWMFAD(Wighted Median Filter based Anisotropic Diffusion)を画像に施す。

WMFADによる平滑化後の画素値は以下の式により求められる。

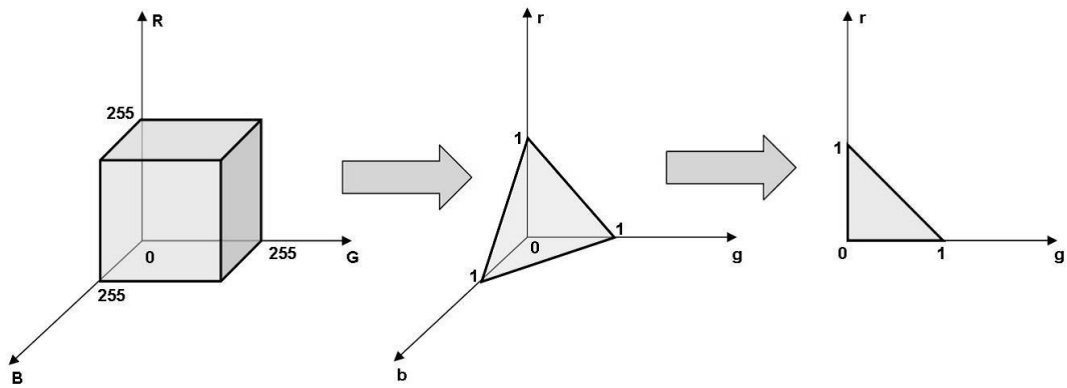


図 2.1: 正規化 RGB 空間への変換

$$I(t+1) = I(t) + \sum_D c_D \nabla_D I(t) \quad (2.1)$$

ここで $I(t)$ は注目画素の RGB 値, $I(t+1)$ は平滑化後の RGB 値, D は上下左右の 4 方向, $\nabla_D I(t)$ は 4 近傍の画素と注目画素の $I(t)$ の差, c_D は 4 近傍に対する重みを表し

$$c_D = \exp(-(\nabla_D \text{median}(Y)/K)^2) \quad (2.2)$$

で表される. $\text{median}(Y)$ は明度値にメディアンフィルタを掛けた値, λ, K は定数を表す.

(2) 複数の 2 値画像作成

ここでは, 文字というオブジェクトを構成している領域とそれ以外の部分を構成している代表色とを良好に分類するためにオブジェクトのクラスタリングを行い, 文字とそれ以外の部分とを良好に分解した 2 値画像を作成する. オブジェクトのクラスタリングはグレイスケールと正規化 RGB 空間上で行う. まず図 2.1 のように画像の RGB 値を正規化 RGB 値へと変換し, 正規 RGB 空間上でのヒストグラムを作成する. 画像の RGB 値から正規化 RGB 値 (r, g, b) へは以下の式によって変換される.

$$(r, g, b) = \begin{cases} (0, 0, 0) & \text{if } R + G + B = 0 \\ \left(\frac{R}{R+G+B}, \frac{G}{R+G+B}, \frac{B}{R+G+B} \right) & \text{otherwise} \end{cases}$$

次にそれぞれの空間でのヒストグラムを考える. ヒストグラムの形状が山になっている部分がオブジェクトが存在していると考えられるので, 山の部分の領域を取り出すように閾値

を決め、複数の2値画像を作成する。このとき、ノイズと考えられる小さい山も取り出される可能性があるため以下の手順により取り除く。

1. ヒストグラムの谷を見つける。
2. 谷の部分と左右の山との差をとり、以下のように分類する。
 - 左の山との差が TH_1 以下 \cap 右の山との差が TH_1 以下 ... class 1
 - 左の山との差が TH_1 以下 \cup 右の山との差が TH_1 以下 ... class 2

これらは、ノイズとして削除される。

3. ある2つの class 2 の谷の間に、class 1 の谷があれば、2つの class 2 の谷の中間を新しい閾値とする。

図 2.2 に、上記の処理を示す。V5 は class 1, V3, V4, V6 は class 2 であるので、これらは削除される。また、class 2 である V4, V6 の間に class 1 である V5 が存在するので、V4 と V6 の間に、新しい閾値 V8 を置く。

グレイスケールでは、上記の通りに2値画像を作成する。正規化 RGB 空間は図のように表されるので正規化 R 及び正規化 G のヒストグラムのみを考慮する。作成された正規化 R、及び正規化 G のヒストグラム上から極大値を全て探索する。そして正規化 R 及び正規化 G のヒストグラム上で極大値を示す部分がオブジェクトを表している値であると仮定し、オブジェクトの分割位置を隣り合う極大値の平均値に設定する。そして正規化 R 及び正規化 G のヒストグラム上で設定されたそれぞれのオブジェクトの分割位置の組合せをオブジェクトのクラスタとし、複数枚の2値画像を作成する。オブジェクトのクラスタは、 rV, gV をオブジェクトの分割位置 (図 2.3 参照) として、以下の式で表される。

$$cluster(m, n) = \sum_{r=rV(m)}^{rV(m+1)} r(x, y) \cap \sum_{g=gV(n)}^{gV(n+1)} g(x, y) \quad (2.3)$$

例えば図 2.3 の場合、正規化 R の分割数は5、正規化 G の分割数は4であるので、20枚の分割画像が出来上がることになる。

(3) 構成要素

複数枚の2値画像を作成後、白領域を外接方形で囲む。これらを、構成要素 (connected component:CC) とし、それぞれの外接方形で、文字領域であるかどうか解析する。

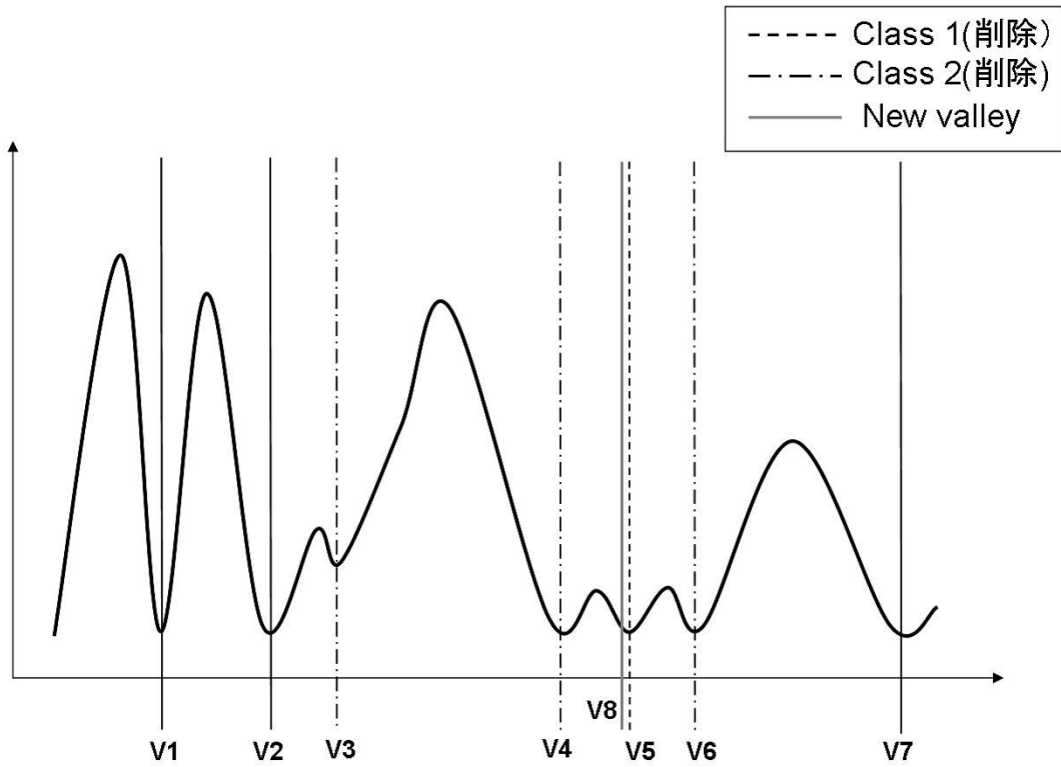


図 2.2: ノイズを含むヒストグラム

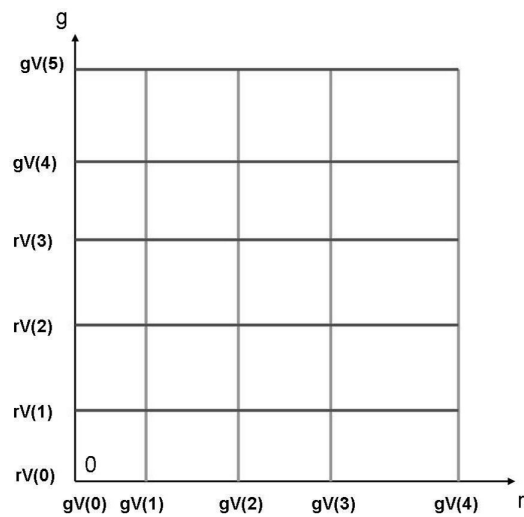


図 2.3: 正規化 RG の閾値設定

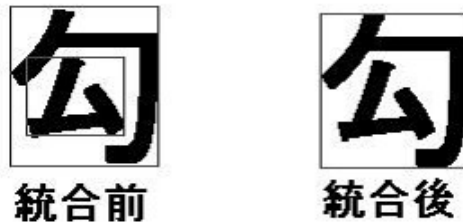


図 2.4: 領域候補の統合

領域候補の中で、複数の候補が重なっているものがある。これは、重なりあった候補が1つの文字を構成していると考えられる。これらを統合することで、以降の処理の速度を上げることができる。2つの候補が重なり合って、かつ以下の式を満たすならば統合し、1つの候補とみなす(図2.4)。

$$\frac{S}{\min(S_1, S_2)} < 0.5 \quad (2.4)$$

S は重なった部分の領域の面積, S_1, S_2 はそれぞれの候補の面積である。

(4) 文字領域の抽出

多くの言語(例えば中国語, 日本語と韓国語)では, 各々の文字は複数の形状から成り, それが互いと必ずしも重なって構成されているというわけではない。それゆえ, 1文字は複数のCCで構成されるときがあり, 各々の分割画像において, 文字全体を表す近隣のCCを複数個統合しなければならない。近傍に存在する2つの文字が十分に離れているならば, CCは位置, 面積によって定義される条件を用いて正しく統合することができる。しかし, 近傍の文字が互いに近接している場合, 統合の条件が複雑になる。そこで, wangらは実画像の文字は「縦, 横に並んでいる」, 「同じ列(行)の文字は通常同程度のサイズを持つ」という文字列の典型的特徴を仮定として導入し, 以下のAMA(Aligning and Merging analysis)法を文字領域の抽出に用いた。

(5) CCの種の選定

本手順では, 文字の構成要素であるCCの種(CC seed:CCs)を選ぶ。ここでは, CCsを領域の面積と整列関係によって選択する。以下に, CCが1文字で構成されていると仮定し

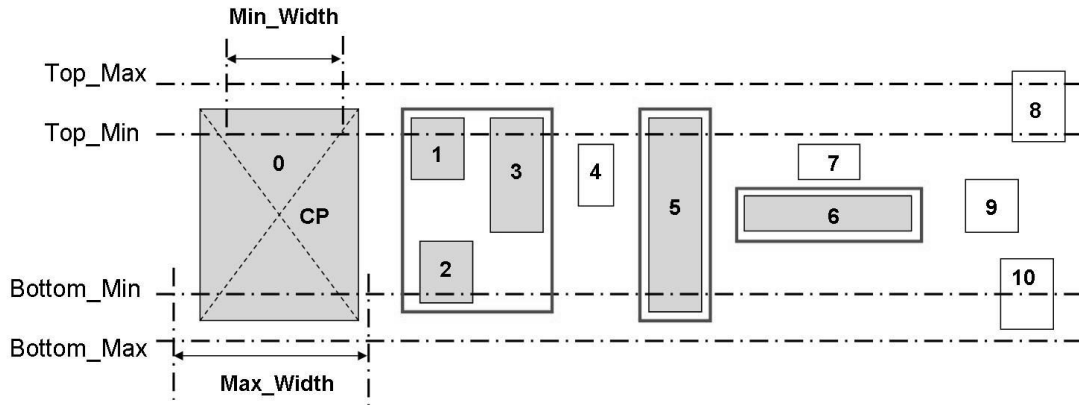


図 2.5: 文字領域の選定 (AMA 法)

た場合の, CC の高さ (CCH), 幅 (CCW) に関する定義を示す.

$$\frac{CCH}{CCW} \leq TH_2 \text{ and } \frac{CCW}{CCH} \leq TH_3 \quad (2.5)$$

(6) AMA 法

k 番目にある構成要素 CC の外接方形を $RECT(k)$ とする. $RECT(k)$ は以下のような属性を持つ.

- $RECT(k, height)$... 外接方形の高さ
- $RECT(k, width)$... 外接方形の幅
- $RECT(k, Top)$... 外接方形の上部分の座標
- $RECT(k, Bottom)$... 外接方形の下部分の座標
- $RECT(k, Left)$... 外接方形の左部分の座標
- $RECT(k, right)$... 外接方形の右部分の座標

これらを用いて, 文字領域の選定を行う. 図 2.5 のように領域要素が並んでおり, 0 番目の構成要素を CC s とし, 文字の抽出を行う. AMA 法の流れを以下に示す.

- (A) それぞれの 2 値画像で, 領域候補の外接方形の中心 (CP) を計算する (図 2.5 には 0 番目の方形のみ記述).

(B) 領域候補の種 (CCs) に関して，文字が位置すると考えられる範囲は，以下の状況で決定される

- $Top_{Max} = RECT(k, Top) - k_1 * hRECT(k)$
- $Top_{Min} = RECT(k, Top) + k_1 * hRECT(k)$
- $Bottom_{Max} = RECT(k, Bottom) + k_1 * hRECT(k)$
- $Bottom_{Min} = RECT(k, Bottom) - k_1 * hRECT(k)$
- $MaxWidth = (1 + k_2) * wRECT(k)$
- $MaxWidth = (1 + k_2) * wRECT(k)$

(C) $RECT(k, Right)$ から画像の右端まで， Top_{Max} から Top_{Min} までの範囲で，外接方形の中心がある候補を探索する．探索した候補の中心を $CP(i), i = 1 \dots n$ とし，以降のプロセスで，文字の選定，統合を行う．

(D) $CP(i)$ に対応する領域候補 $RECT(i)$ を選択肢，以下の条件に基づいて分類する．全ての選定を終えたら，(I) へ進む．

(E) $RECT(i)$ が以下の条件

- $Top_{Max} \leq RECT(i, Top) \leq Top_{Min}$
- $Bottom_{Min} \leq RECT(i, Bottom) \leq Bottom_{Max}$
- $Width_{Min} \leq RECT(i, Width) \leq Width_{Min}$

が与えられたとき，すべて満たすならば， $RECT(i)$ を文字と見なし，(D) にもどり次の選定を行う．そうでない場合は，(F) へ進む．

(F) $RECT(i)$ と $RECT(i + 1)$ を統合し， $MRECT(i, i + 1)$ とする．

(G) $MRECT(i, i + 1)$ が以下の条件

- $MRECT(i, i + 1, Top) \leq Top_{Max}$
- $MRECT(i, i + 1, Bottom) \geq Bottom_{Max}$
- $MRECT(i, i + 1, Width) \geq MaxWidth$

を1つでも満たすとき，(H) へ進む．そうでない場合，以下の条件

- $Top_{Max} \leq MRECT(i, i + 1, Top) \leq Top_{Min}$
- $Bottom_{Min} \leq MRECT(i, i + 1, Bottom) \leq Bottom_{Max}$

- $Minwidth \leq MRECT(i, i + 1, Width) \leq Maxwidth$

をすべて満たすとき、 $MRECT(i, i + 1)$ は統合された文字と見なし、(D)に戻り、次の選定を行う。そうでない場合、 $RECT(i + 1) = MRECT(i, i + 1)$ とし、(F)に戻り、次の選定を行う。

(H) $MRECT(i, i + 1)$ 内で、 $RECT(x)$ があるとき、以下の条件

- $Top_{Max} \leq RECT(x, Top) \leq Top_{Min}$
- $Bottom_{Min} \leq RECT(x, Bottom) \leq Bottom_{Max}$
- $RECT(x, Width) \leq 3 * RECT(x, Height)$

をすべて満たすとき、 $RECT(x)$ を縦長の文字と見なす。そうでない場合、条件

- $Top_{Min} \leq RECT(i, Top), Bottom_{Min} \leq RECT(i, Bottom)$
- $Bottom_{Min} \leq RECT(i, Bottom) \leq Bottom_{Max}$
- $RECT(i, Height) \leq 3 * RECT(i, Width)$

をすべて満たすとき、 $RECT(x)$ を横長の文字と見なし、(D)に戻り、次の選定を行う。

(I) 左から右へ、全ての文字の再編成を行う。一方で、文字の数をカウントする (領域候補の種 (CCs) も含む)。

(J) すべての2値画像で処理を行った場合、この処理を終了する。そうでない場合、(B)に戻り、次の画像で処理を続ける。

(7) 文字領域の認識

(6) で、文字領域の抽出を行ったが、その中には、文字でない部分も含まれると考えられる。誤検出した領域を取り除くために、以下の処理をそれぞれ適用し候補領域の選別を行う。

1. 候補領域が文字を含むならば、文字と背景と一致している階調値が極大でなくてはならない。
2. 候補領域を2値化し、白いピクセルを文字、黒ピクセルは背景とする。候補領域内において、1列ずつ左から右へ白ピクセルを探索し、白ピクセル数が、領域幅の5/6以上ある場合、1、そうでない場合0とする。そして、1が連続して存在している部分を数える。この部分は文字の1画になると考えられる。同じように、行方向に対しても計算し、画数を調べる。

3. 2. で抽出した領域の画数が小さい場合、これを誤検出と見なす。
4. 文字認識アルゴリズムを候補領域に対して適用し、文字認識ができたものだけ選択する。具体的には、あらかじめ学習した文字モデルとの距離（文字モデルとの類似度）を計算し、閾値処理により誤検出と思われる領域を除去する。

2.2.1.2 平山らの手法 [14]

ここでは、平山らの手法について詳説する。本手法では、カラークラスタリングを用いて複数枚の2値画像を生成し、その中から複雑度を基準に画像を選択することにより文字領域抽出を行う。以下に詳細な手順を示す。

(1) カラー情報による複数の2値画像作成

平滑化画像に対しまず $L_*a_*b_*$ 色空間上においてカラークラスタリングを行い、クラスタリング画像を作成する。そして作成されたクラスタリング画像の正規化 a, 正規化 b, 及び、明度値 L の3種類のヒストグラムを作成し、ヒストグラム上の極大値を利用して、オブジェクトの分割位置を複数個決定することでオブジェクトのクラスタリングを行い、2値画像を複数枚作成する。その中から複雑度を評価基準として、文字とそれ以外の部分とを良好に分解していると考えられる2値画像を選択することで複数枚の2値画像を作成する。

(2) カラークラスタリング

まず、Wang らの手法と同じように WMFAD 法を適用し、 $L_*a_*b_*$ 表色系に変換する。全画素が投影された $L_*a_*b_*$ 空間において、投影された画素における $L_*a_*b_*$ の最大値最小値を求め、それを利用して全画素を囲む直方体を作成する。次に直方体の $L_*a_*b_*$ 各辺を K 分割 (実験では $K = 30$) し、直方体の中に K^3 個の Cube を作成する。そして各 Cube 内に存在する画素数をカウントし、周囲 26 個の Cube よりも Cube 内に存在する画素数が多い Cube を選択し、選択された Cube の重心を代表色とする。ここで周囲 26 個の Cube とは注目 Cube に対して $3 \times 3 \times 3$ の Cube、つまり 27 個の Cube を考えた際に中央に存在する Cube を注目 Cube として、その周囲の 26 個の Cube のことを指す。そして、各画素において全代表色との $L_*a_*b_*$ 空間内での距離を測定し、最も距離の近い代表色へと変換する。その後代表色の $L_*a_*b_*$ 値を で RGB 値へ逆変換し、クラスタリング画像を作成する。このク

ラスタリング画像の正規化 R 正規化 G 及び明度値ヒストグラムを利用して、オブジェクトのクラスタリングを行う。

(3) オブジェクトのクラスタリング

(2) までの手順において、情景画像に対して Wang らの手法と同様に前処理を行い、同一オブジェクトを表している画素が似たような RGB 値を保持するような平滑化画像を作成し、その平滑化画像に対しカラークラスタリングを行ったことで、類似する RGB 値を保持する同一オブジェクトを表す画素は同じ代表色へと変換されている。しかし、入力画像はどのような文字の色をしているかまたどのような照明条件下で撮影されたかなどは未知である。そのため、同じ文字列上の文字でも陰影が掛かっているために平滑化画像内でも同じオブジェクトが異なる RGB 値を保持し、カラークラスタリングにおいて異なる代表色に変換されることがある。そのため、文字というオブジェクトを構成している代表色と、それ以外の部分を構成している代表色とを良好に分類するために、オブジェクトのクラスタリングを行い、文字とそれ以外の部分とを良好に分解した 2 値画像を作成する。これは図 2.3 のように Wang らの行った手法と同じく、正規化 RGB 空間に変換し、正規化 R と正規化 G のヒストグラムの山の部分を取り出すように閾値を決め、分割画像を作成する。

(4) 複雑度の計算

カラークラスタリングとオブジェクトのクラスタリングによって 2 値画像を作成する場合、入力画像によってはオブジェクトがほとんど存在しないような 2 値画像も多く作成される。そこで作成された 2 値画像の境界線の長さから複雑度を計算し、文字とそれ以外とを良好に分解していると考えられる 2 値画像を選択する。2 値画像の複雑度は以下の式で計算される。

$$complexity = \frac{boundary}{2 \times height \times (width - 1)} \quad (2.6)$$

(5) 明度情報による複数 2 値画像作成

平滑化画像からまず明度値を計算し明度情景画像を作成する。そして作成された明度情景画像を小領域に分割し、各小領域における 4 つの参照領域から NAT 法 [15] を利用して、4 つの領域閾値を設定する。次に各小領域に所属している各画素において、正規化 RGB 空間

上での平均色を計算し、各画素に対し4つの領域閾値から1つの画素閾値を決定する。このようにして全画素において閾値を設定し、1枚の2値画像を作成する。

(6) 画像閾値設定

画像分割及び参照領域決定

まず平滑化画像から明度値を以下の式により計算し、明度情景画像を作成する。

$$L = (R + G + B) / 3 \quad (2.7)$$

次に作成された明度情景画像を 4×4 pixel の小領域に分割する。そして各小領域において、注目領域が、右下、左下、右上、左上に位置するように、 2×2 の参照領域を4つ決定し、各参照領域において NAT 法を利用することで注目小領域に対し領域閾値を4個決定する。NAT 法については次で説明する。

NAT 法による領域閾値決定

NAT 法 [15] は 2001 年に Hon-Son Don により提案された手法で、文字と背景と裏書き文字の3種類のオブジェクトが存在するような、一般に用いられる明度値とヒストグラムを利用した2値化がうまく機能しない対象画像に対して、ノイズの特性を利用して2値化することで文字とそれ以外の部分とを良好に分解することを可能にした手法である。NAT 法では、明度値ごとに閾値をとり、できた2値画像の領域数をヒストグラムとして表す。これにより、ヒストグラム上で、文字、裏書き文字、背景の部分のが山となり、良好に2値化を行うことができる。ここでは各小領域に対する4つの参照領域それぞれにおいて NAT 法を用いることで、参照領域内に存在するオブジェクトを良好に2値化することのできる閾値を決定する。

色比較による画素閾値決定

各小領域において NAT 法を利用することで4個の領域閾値が決定された。これにより各小領域に所属する各画素においても4個の領域閾値が決定されたことになる。この各画素に決定された4個の領域閾値から、今度は正規化 RGB 空間上での平均色を利用することで各画素に対する画素閾値を決定する。

注目画素 $c(i, j)$ の周囲 5×5 pixel の正規化 RGB 空間上での平均色 (R_{av}, G_{av}, B_{av}) を以下の式により求める。

$$R_{av} = \frac{\sum_{a=i-2}^{i+2} \sum_{b=j-2}^{j+2} R_{ab}}{R_{ab} + G_{ab} + B_{ab}} \quad (2.8)$$

$$R_{av} = \frac{\sum_{a=i-2}^{i+2} \sum_{b=j-2}^{j+2} R_{ab}}{R_{ab} + G_{ab} + B_{ab}} \quad (2.9)$$

$$R_{av} = \frac{\sum_{a=i-2}^{i+2} \sum_{b=j-2}^{j+2} R_{ab}}{R_{ab} + G_{ab} + B_{ab}} \quad (2.10)$$

同様にして注目画素の領域閾値を決定した4つの各参照領域の正規化RGB空間上での平均色 (R_n, G_n, B_n) ($n = 1 \sim 4$) を以下の式により求める.

$$R_n = \frac{\sum_a \sum_b R_{ab}}{R_{ab} + G_{ab} + B_{ab}} \quad (2.11)$$

$$R_n = \frac{\sum_a \sum_b R_{ab}}{R_{ab} + G_{ab} + B_{ab}} \quad (2.12)$$

$$R_n = \frac{\sum_a \sum_b R_{ab}}{R_{ab} + G_{ab} + B_{ab}} \quad (2.13)$$

そして注目画素の周囲 5×5 pixel の平均色 (R_{av}, G_{av}, B_{av}) と、各参照領域での平均色 (R_n, G_n, B_n) ($n = 1 \sim 4$) の距離 D_n を以下の式で求める.

$$D_n = \sqrt{(R_n - R_{av})^2 + (G_n - G_{av})^2 + (B_n - B_{av})^2} \quad (2.14)$$

この D_n ($n = 1 \sim 4$) が最小である参照領域において求められた閾値 TH_n に対し大きな重みを付加することで、注目画素の画素閾値を決定する. 平山らは重みの値を 1.2 としている. 注目画素において D_3 が最小であった場合、注目画素の画素閾値 T_{ij} は以下の式によって求まる.

$$T_{ij} = \frac{TH1 + TH2 + 1.2 \times TH3 + TH4}{4.2} \quad (2.15)$$

このようにして、各画素に決定された4個の領域閾値から平均色を利用することで画素閾値を求める. この画素閾値を全画素において求め1枚の2値画像を作成する.

(7) 画像補正

各画素において画素閾値を決定し、2値画像を作成するとオブジェクトを余計に連結してしまうような画素が存在する場合がある. このように文字と他のオブジェクトとが余計に連結された場合、文字は他のオブジェクトと同領域と見なされ文字を抽出することを困難にさせる. そこで、注目画素に対し上下の画素値及び左右の画素値がそれぞれ同じ値を保持し、

かつ上下の画素値と左右の画素値が異なる画素値を保持している場合、注目画素を無視する。これにより、1 pixel で存在しているような線を排除することができ、オブジェクトを余計に連結してしまっているような画素を無視することができる。

(8) 文字列領域の抽出

文字列領域候補抽出

これまでの手順によって生成された画像では、文字を表す領域は黒画素または白画素の閉領域で存在している。そこで、各 2 値画像において同一の画素値を持つ閉領域を 4 連結でラベルを与えていき、同じ領域として統合する。これにより 2 値画像はラベルによる領域分割をされたことになる。次にこの領域分割された各閉領域の x, y 座標の最大値最小値を求め、その最大値最小値を始点終点として閉領域を方形で囲み、各閉領域に外接方形を与える。以上の手順により各閉領域に与えられた外接方形の形状によって、全閉領域を文字領域候補部分領域候補漢数字候補不用の 4 種類に分類することができる。分類の条件を以下に示す。

文字領域候補

文字列領域候補抽出

- 辺が 5 画素以上
- 辺と短辺の辺比率が 1.6 未満

部分領域候補

- 短辺が 5 画素以上
- 長辺と短辺の辺比率が 1.6 以上 4.0 未満

漢数字候補

- 長辺が 5 画素以上
- 閉領域において黒画素または白画素の全画素に対する比率が 8 割以上

不用

- それ以外

(9) マージング処理

(8) において、2 値画像内に存在する閉領域を外接方形により分類し、文字領域候補部分領域候補漢数字領域候補を抽出した。しかし、画像内に存在している文字には偏と旁から構成される漢字などがあり、1 つの文字が 2 つ以上の候補から成り立っている場合がある。偏と旁から構成されるような文字で、2 つ以上の候補から成り立っている文字を 1 つの方形で囲むようにするため、外接方形を統合するマージング処理を行う。マージングとは、ある 2 つの閉領域の外接方形の位置関係や形状などに基づいて、その 2 つの閉領域を統合する処理である。この外接方形のマージング処理は、馬場口らの提案した手書き文字の文字単位ごとの切り出しによる手法 [16] を応用する。本手法で行うマージング処理は、外接方形による分類において文字領域候補または部分領域候補に分類された閉領域に与えられた外接方形に対してのみ行う。この閉領域に対して以下のような条件を利用し、3 種類のマージング処理を行う。

文字領域候補と文字領域候補のマージング

このマージング処理は偏や旁が 2 つの領域に分離しており、それぞれが文字領域候補として抽出されているような場合の文字を想定して行う。

- 2 つの外接方形は交差
- 2 つの外接方形の面積比率が 1.5 未満
- 2 つの外接方形の短辺の辺比率が 1.6 未満かつ長辺の辺比率が 1.6 未満
- 2 つの外接方形の上下左右の辺いずれか 1 つでも 2 画素以内のずれ
- 2 つの閉領域を統合して新しく作成された外接方形が文字領域候補の条件を満たす

文字領域候補と部分領域候補のマージング

このマージング処理は、偏と旁のいずれかが部分領域候補として抽出されており、残りの部分が文字領域候補として抽出されているような場合の文字を想定して行う。

- 2 つの外接方形は交差または 2 画素以内に存在
- 2 つの外接方形の面積比率が 3.0 未満
- 2 つの外接方形の短辺の辺比率が 1.6 未満かつ長辺の辺比率が 2.5 未満または短辺の辺比率が 2.5 未満かつ長辺の辺比率が 1.6 未満
- 2 つの外接方形の上下左右の辺いずれか 1 つでも 2 画素以内のずれ
- 2 つの閉領域を統合して新しく作成された外接方形が文字領域候補の条件を満たす

部分領域候補と部分領域候補のマーキング

このマーキング処理は偏と旁が2つの領域に分離しており、それぞれが部分領域候補として抽出されているような場合の文字を想定して行う。

- 2つの外接方形は交差または2画素以内に存在
- 2つの外接方形の短辺の辺比率が1.6未満かつ長辺の辺比率が2.0未満
- 2つの外接方形の面積比率が1.5未満
- 2つの外接方形の上下左右の辺のいずれか1つでも2画素以内のずれ
- 2つの閉領域を統合して新しく作成された外接方形が文字領域候補の条件を満たす

文字列領域候補及び漢数字候補抽出

ここまで抽出されている文字領域候補の中から、文字列を成している文字を表していると考えられる領域を、文字列を成す文字の連結性を利用して文字列領域候補として抽出する。文字列を成す文字の連結性を2つの文字領域候補ごとに確認し、文字列領域候補を選択する。文字列を成す文字の連結性は以下のようになる。

- 2つの外接方形の面積比率が1.5未満
- 2つの外接方形の重心のずれが5画素以内
- 2つの外接方形の重心の距離が $3 \times$ (方形の長さ) 以内
- 2つの外接方形の縦横の一辺の長さの比が共に1.2以内

これらの条件を満たす2つの文字領域候補を文字列領域候補として抽出する。このとき2つの文字列領域候補の近くに存在する漢数字候補の抽出も同時に行う。漢数字候補の抽出条件は以下のようになる。

- 2つの文字列領域候補と漢数字候補の重心のずれが5画素以内
- 2つの文字列領域候補と漢数字候補の平均色の差が30以内

平均色は情景画像における正規化RGB空間上での外接方形内の全画素の平均を用いる。漢数字候補と2の文字列領域候補は同じ文字列上の文字である場合、色が似ているとして人間が識別困難であるとされる正規化RGB空間上での平均色の差が30以内という条件にした。このようにして、各2値画像において文字列を成す文字として文字列領域候補を抽出する。

(10) 文字列領域の抽出

ここでは、各 2 値画像において抽出された文字列領域候補を利用して文字列領域を決定する。まず各 2 値画像で抽出された文字列領域候補を一つの画像に統合し、文字列統合画像を作成する。この文字列統合画像を作成することで、1 枚 1 枚の 2 値画像では文字列上の文字が欠落している場合でも、文字列統合画像において全ての文字列上の文字を抽出する文字列領域候補が存在することになる。このようにして、作成された文字列統合画像から文字列領域を決定する。まず文字列統合画像の水平方向及び、垂直方向における文字列領域候補の外接方形の量を示すヒストグラムを作成する。この 2 のヒストグラムにおいて、まず画像内に存在する文字列が縦一列の文字列かまたは横一列の文字列かを決定する。水平方向及び垂直方向それぞれのヒストグラムにおいて、最大値を探索し、比較することで縦一列か横一列の文字列か決定する。文字列が存在する範囲において、文字列領域候補を囲む方形は、垂直または水平方向に並んで存在し、その範囲内において 2 のヒストグラムにおける最大値を示すため、2 のヒストグラムにおいて最大値の存在するヒストグラムの方向に文字列が存在する。仮に垂直方向のヒストグラムにおける最大値の方が水平方向のヒストグラムにおける最大値よりも大きい場合、その情景画像内に存在する文字列は縦一列の文字列となる。

次に、文字列が存在する方向のヒストグラムにおいて、文字列領域の存在する範囲を決定する。まず文字列が存在する方向のヒストグラム上において、文字列領域候補の外接方形の量が 4 以下の場合、その位置におけるヒストグラム値を 0 にする。そして、仮に垂直方向のヒストグラムを考えた場合、最大値から右方向及び左方向それぞれにおいて最初にヒストグラム値が 0 を示す位置をそれぞれ文字列左端位置及び文字列右端位置として、その範囲内に文字列が存在するとする。ヒストグラムから文字列領域を決定する際に、ヒストグラム値が 4 以下の場合 0 にすることで余計な外接方形を無視することができ、余計な範囲を文字列領域として抽出しなくて済む。そして、この範囲内に存在する文字列領域候補を全て囲む新しい方形を作成し、文字列領域を決定し、文字列領域抽出画像を作成する。このようにして、情景画像中から文字列領域を決定する。

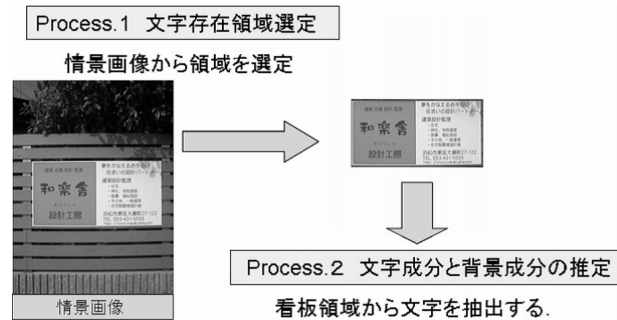


図 2.6: 文字抽出プロセス

2.2.2 提案手法

前節までで述べたように、従来研究では、文字・文字列の特徴や色の情報を用いて情景画像中からの文字領域抽出が行われてきた。すなわち、紙面などからの文字抽出の延長として「情景画像中からの文字抽出」が行われてきたのが現状である。それに対して本研究では、情景画像中に存在する「文字情報」は看板や標識などの人工物上に存在することが多いということに着目し、「文字存在領域選定」と「文字成分と背景成分の推定」の二つのプロセスにおいて文字抽出を実現する。これは「情景画像中からの文字抽出」という問題を分割し処理の簡単化と抽出精度向上を狙ったものである。提案手法では、画像は看板を含んだ情景画像という特定のシーンを想定し、「文字存在領域選定」と「文字候補領域内における文字成分と背景成分の推定」の二つの主要なプロセスにおいて文字抽出を実現する。手法の流れを図 2.6 に示す。以下、2つのプロセスについて詳述する。

2.2.2.1 文字存在領域選定

本節では、「文字存在領域選定」について述べる。屋外環境下においてカメラを用いて撮影を行う場合、情景画像には文字領域だけでなく様々なものが含まれるために、文字抽出が困難になってしまう。そこであらかじめ文字を含む可能性のある領域を文字候補領域として選び出し、その領域に対して文字抽出処理を行うことで、処理の簡単化と抽出精度向上を実現する。提案法では、文字存在領域として看板を設定し、情景画像から看板領域を検出する。図 2.7 に、提案手法の流れを示す。以降、提案手法の各段階の具体的なアルゴリズムを説明する。

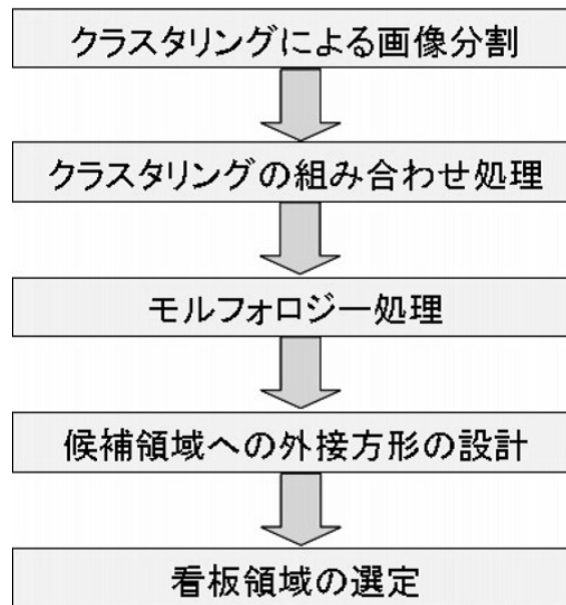


図 2.7: 看板検出の処理手順

(1) 色情報のクラスタリングによる画像分割

提案手法では、まず入力画像の各ピクセルを色情報を用いてクラスタリングする。クラスタリングには、 $L^*a^*b^*$ 表色系を特徴として用いる。 RGB 表色系から $L^*a^*b^*$ 表色系への変換は以下の式によって行う。

$RGB \rightarrow XYZ$

$$X = 0.412453 \times R + 0.35758 \times G + 0.180423 \times B \quad (2.16)$$

$$Y = 0.212671 \times R + 0.71516 \times G + 0.072169 \times B \quad (2.17)$$

$$Z = 0.019334 \times R + 0.119193 \times G + 0.950227 \times B \quad (2.18)$$

$XYZ \rightarrow L^*a^*b^*$

$$x_r = X/X_r \quad (2.19)$$

$$y_r = Y/Y_r \quad (2.20)$$

$$z_r = Z/Z_r \quad (2.21)$$

$$f_x = \sqrt[3]{x_r} \quad x_r \geq 0.08856 \quad (2.22)$$

$$f_x = (903.3 \times x_r + 16) / 116 \quad x_r \leq 0.08856 \quad (2.23)$$

$$f_y = \sqrt[3]{y_r} \quad y_r \geq 0.08856 \quad (2.24)$$

$$f_y = (903.3 \times y_r + 16) / 116 \quad y_r \leq 0.08856 \quad (2.25)$$

$$f_z = \sqrt[3]{z_r} \quad z_r \geq 0.08856 \quad (2.26)$$

$$f_z = (903.3 \times z_r + 16) / 116 \quad z_r \leq 0.08856 \quad (2.27)$$

ここでは、 RGB 表色系から $L^*a^*b^*$ 表色系への変換を、 XYZ 表色系を介して行なっている。

次に、 $L^*a^*b^*$ の値を特徴量として K -平均法によりクラスタリングを行う。そして、 K 個の各クラスタに対して、次式のように、そのクラスタに属するピクセルのみを抽出した 2 値画像 B_k を作成する。

$$B_k(i, j) = \begin{cases} 1 & \text{if } C(i, j) = k \\ 0 & \text{otherwise} \end{cases} \quad (k = 1, 2, \dots, K) \quad (2.28)$$

ここで、 i, j は画像座標、 k はクラスタラベル、 $C(i, j)$ は画素 (i, j) のクラスタである。これにより、 K 枚の 2 値画像が生成される。例として、看板を含む原画像とクラスタリングによる画像分割の結果を図 2.8 に示す。

(2) クラスタの組み合わせ処理

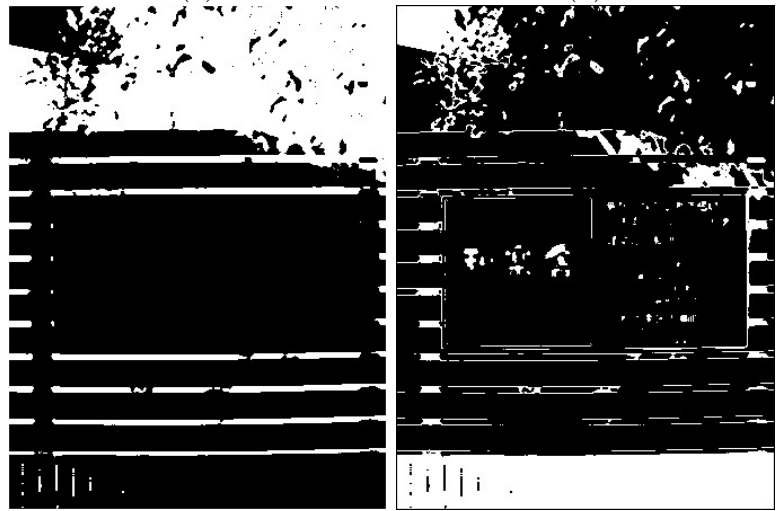
図 2.8 を見るとわかるように、看板が複数の色で構成されていた場合、単なる色のクラスタリングだけでは看板全体を抽出することができない。そこで本研究では、様々な種類の看板に対応するため、色クラスタを抽出する際に組み合わせを許容し異なる色を持つ領域を一つのクラスタとみなす。これは、次式のように行う。

$$B_k(i, j) = \begin{cases} 1 & \text{if } C(i, j) \in S_t \\ 0 & \text{otherwise} \end{cases} \quad (t = 1, 2, \dots, T) \quad (2.29)$$



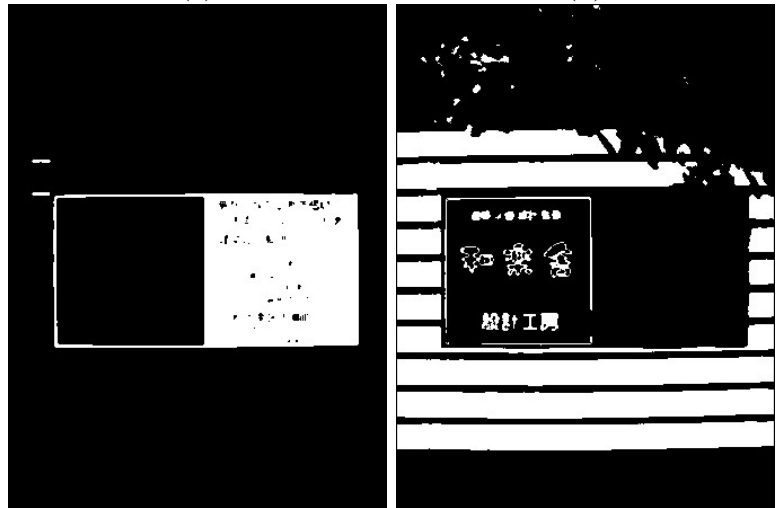
(a)

(b)



(c)

(d)



(e)

(f)

図 2.8: K 平均法による 2 値化画像. (a) 原画像, (b)~(f) 2 値化結果.

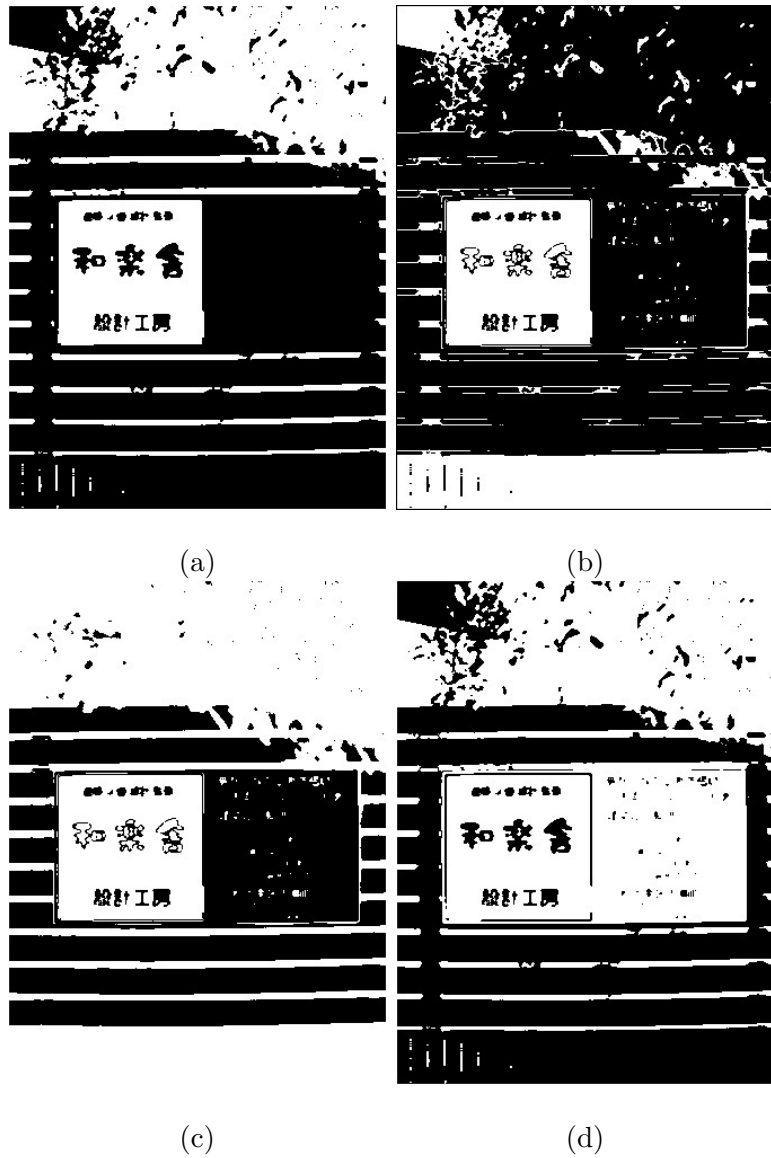


図 2.9: 組み合わせにより生成された 2 値画像

ここで、 S_i はクラスタラベルの集合で、 T は集合の個数である．例えば、 $K = 5$ の時に組み合わせのパターンをそれぞれ単独のものから 3 枚のものまで考慮した場合、集合の個数 T は 25 となる．この組み合わせにより得られる画像の例を図 2.9 に示す．

領域内の黒画素数が20%を超えてしまっているため条件2により排除されてしまう。



図 2.10: 文字領域を含む候補の排除

(3) モルフォロジー処理

前節までの処理により、看板などの人工物を特徴的な図形として含む2値画像が多く生成される。しかしながら、図 2.10 に示すように、領域候補中に含まれる文字量が多量な場合や、文字の面積が大きな場合、文字列を含む領域候補が多量の黒画素を含んだ穴あき領域になってしまい、看板領域を選定する条件を設定するのが困難になってしまう。そこで、そのような問題を回避するために、領域候補中の黒画素領域を埋める処理が必要となる。本手法ではモルフォロジー処理の一つである *closing* により、黒画素領域を埋める処理を行う。モルフォロジーは与えられた2値画像または濃淡画像からの特徴抽出を目的とした集合論的操作からなる対象図形の変形手法の一貫した理論体系であり、主に図形の形状分割や画像の平滑化等に用いられている。

2値画像に対して *closing* を行った場合、画像内に存在する構造物の外側が滑らかになり、また、集合内の穴は埋められる。提案手法では、これを用いて黒画素領域を埋める処理を行う。処理例を図 2.11 に示す。

(4) 外接方形による看板領域の選定

得られた2値化画像に対し、それぞれの白画素に外接する方形を作成する。これを領域候補とする。図 2.8 の原画像より得られた領域候補例を図 2.12 に示す。

外接方形で囲まれた看板領域には、次のような特徴があると考えられる。

1. 方形の面積が、画像サイズと比較してある程度大きい。
2. 外接方形内において、白画素が領域内の多重を締めている。

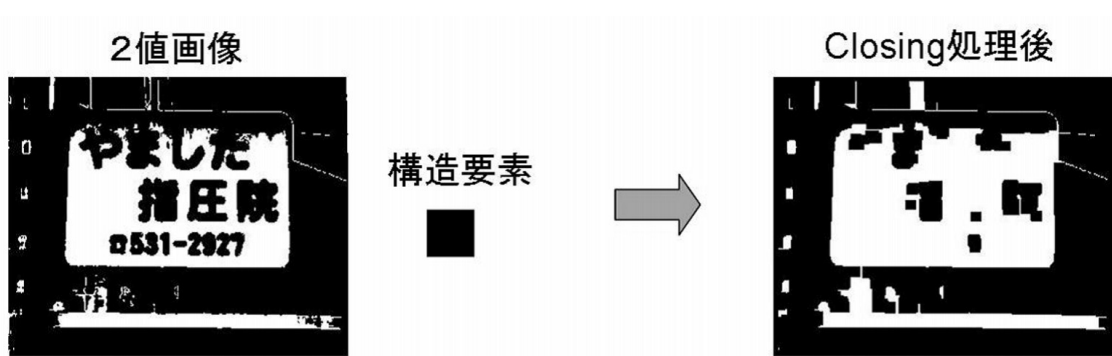


図 2.11: *closing* 処理の例

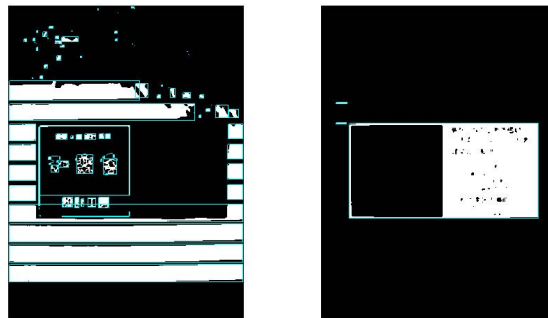


図 2.12: 看板領域候補

3. 看板は、外接方形の中心に存在している。

これらの特徴より、作成された外接方形に対し、以下の条件を設定した。

1. 候補面積が原画像面積の 1/20 以上
2. 候補内の白画素が候補内領域の 80 パーセント以上を占める。
3. 領域の重心と外接方形の中心の距離が 10 ピクセル以内。

図 2.13 に示すように、条件 1 により、明らかに小さな領域候補を排除し、条件 2 により、文字を含むことが出来ない L 字型の領域などを排除する。最終的に残った領域候補を文字を含む領域とする。

抽出される看板領域結果を図 2.14 に示す。

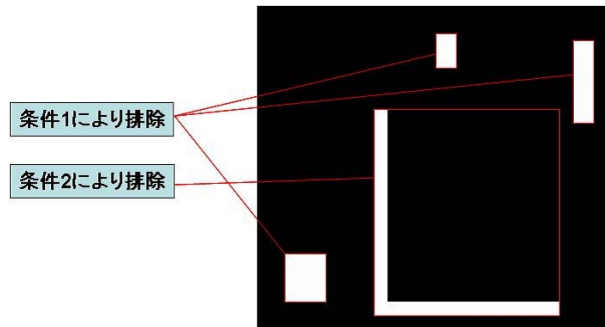


図 2.13: 条件による候補の排除



図 2.14: 看板領域抽出

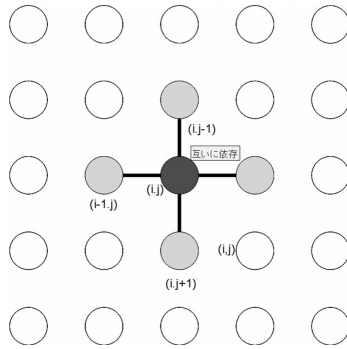


図 2.15: マルコフ確率場

2.2.2.2 領域内における文字抽出

本節では，特定された領域内において文字成分の背景からの分離を行う．情景画像では環境光などの反射による光沢や周囲環境に起因する影のような劣化が生じるため，文字抽出は困難になる．そこでそのような劣化に依存しない，2 値化処理を実現するため文字領域の抽出処理をエネルギー最小化問題として定式化しそれをグラフカットアルゴリズムを用いて解く．

MAP-MRF

まず，文字領域の抽出処理をエネルギー最小化問題として解くため，画像をマルコフ確率場として取り扱う MAP-MRF (Markov Random Field modeling with Maximum A Posteriori probability) として定式化する．マルコフ確率場とは，無向グラフで確率変数の間の依存関係を表したグラフィカルモデルである．マルコフ確率場では，図 2.15 に示すように，各ノード X の他の全ノードに関する条件付確率が X にエッジを持つまわりのノードだけに依存する．

2 値化を確率モデルによる最適化問題として定式化する場合，2 値化対象グレイスケール画像を I とし，座標 p における 2 値化ラベル f_p を次式のように定義する．

$$f_p = \begin{cases} 1 & \text{if } I_p \text{ is text} \\ 0 & \text{if } I_p \text{ is background} \end{cases} \quad (2.30)$$

グレイスケール画像 I が与えられたとき，2 値化ラベル f の事後確率 $p(f | I)$ を最大に

するラベル f^* を最適基準とすると、次式のように記述できる.

$$f^* = \arg \max_f p(f | I) \quad (2.31)$$

ベイズの定理より、上式は次のように展開できる.

$$p(f | I) \propto p(I | f)p(f) \quad (2.32)$$

ここで、 $p(I | f)$ はラベル f が与えられたときの所与画像に対する尤度であり、 $p(f)$ はラベル f の事前確率である. したがって、 $p(I | f)$ と $p(f)$ が計算できれば、 $p(f | I)$ を最大にするラベル f^* を求めることが可能になる.

まず、尤度 $p(I | f)$ を計算するために、ピクセル p における輝度値 I_p についての観測モデルを次式のように定義する.

$$I_p = \rho_p^b (1 - f_p) + \rho_p^t f_p + n \quad (2.33)$$

ここで、 ρ_p^b は画素 p をテキストと仮定した場合の推定輝度値、 ρ_p^t は画素 p を背景と仮定した場合の推定輝度値、 n は平均 0、分散 σ に従うガウスノイズである. このモデルは、文書画像がテキスト成分または背景成分にノイズを加えたものとして観測されることを仮定している. ρ_p^b と ρ_p^t は 2 値化の結果を左右する重要なパラメータであり、本論文で提案する算出方法については次節で詳しく述べる. 式 2.33 の観測モデルより、尤度は次式のように記述できる.

$$p(I | f; \rho_p^b, \rho_p^t) = \prod_{p \in V} \frac{1}{\sqrt{2\pi\sigma}} \exp^{-\frac{(I_p - \rho_p^b(1-f_p) - \rho_p^t f_p)^2}{2\sigma^2}} \quad (2.34)$$

ここで、 V は画像内の全てのピクセルの集合である.

次に、事前確率 $p(f)$ について述べる. 事前確率は、

- 隣り合うピクセルにおいて輝度値が似ている場合、同じラベルにする
- 隣り合うピクセルにおいて輝度値が離れている場合、違うラベルにしても良い

という知識をモデリングするために、Potts Model を用いる. 一般的な Potts Model は次のように表される.

$$p(f) = \frac{1}{Z} \exp^{-\sum_{\{p,q\} \in E} \lambda w_{p,q} \delta(f_p, f_q)} \quad (2.35)$$

表 2.1: グラフカットのためのエッジに対する重み係数の設定

edge	weight
(s, p)	$(y_p - \rho_p^b)^2$
(t, p)	$(y_p - \rho_p^t)^2$
(p, q)	$\lambda w_{p,q} \delta(f_p, f_q)$

$$\delta(f_p, f_q) = \begin{cases} 1 & \text{if } f_p \neq f_q \\ 0 & \text{if } f_p = f_q \end{cases} \quad (2.36)$$

ここで、 Z は分配関数、 E は画像中の隣り合うピクセルの全集合、 λ は平滑化のための調整係数、 $w_{p,q}$ はピクセル p とピクセル q の輝度値から決まる重み係数である。

最後に、式 (2.34) と式 (2.35) を式 (2.32) に代入し負の対数尤度をとると、式 2.31 の事後確率最大化問題は最小化問題となり、次式のように表すことができる。

$$f^* = \arg \min_f \sum_{p \in V} (y_p - \rho_p^b(1 - f_p) - \rho_p^t f_p)^2 + \sum_{\{p,q\} \in E} \lambda w_{p,q} \delta(f_p, f_q) + \text{const}, \quad (2.37)$$

ここで、 $\lambda \leftarrow 2\sigma^2\lambda$ とした。また、式 (2.37) では、 const が f に依存していない定数であるため無視できる。

以上の定式化により、式 (2.37) を解くことで最適なラベル f^* を求められる。提案手法では、式 2.37 をグラフカットアルゴリズム [17][18] を用いて解く。

グラフカットアルゴリズム

グラフカットアルゴリズム [17] では、グラフ G をノード集合 $\Lambda = \{V, s, t\}$ とエッジ集合 $\Phi = \{E, E_s, E_t\}$ により定義し、Min cut/Max flow algorithm を用いて、コスト総和が最小となるようにグラフをカットすることで、テキストと背景のグラフに分割する。ここで、 $E_s = \{(s, v) | v \in V\}$ 、 $E_t = \{(t, v) | v \in V\}$ である。カットの後に、ターミナル s と繋がっているピクセルはテキスト、同様にターミナル t と繋がっているピクセルは背景とすることで、2 値化が実現する [図 2.16]。エッジに与える重みを表 2.1 に示す。また、ピクセル同士は隣接する 8 近傍とつながるエッジのみに重みを与えた。

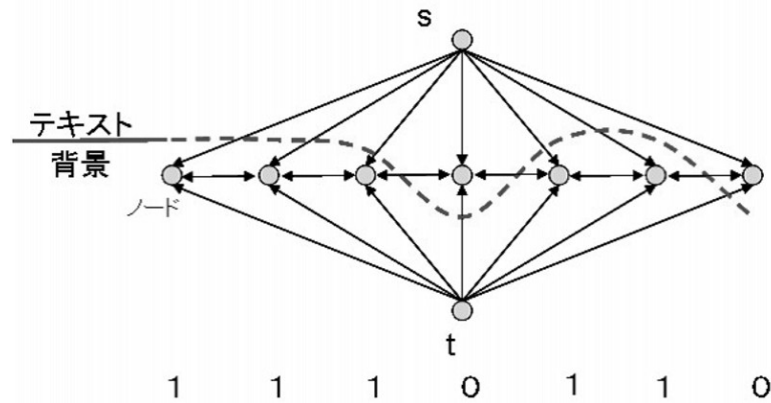


図 2.16: グラフカット

局所領域画像からのテキスト推定値と背景推定値の算出

提案手法では、文字は背景に対するイレギュラーな輝度値であるという仮定の下で推定を行う。以下、提案する背景成分とテキスト成分の推定方法について詳しく述べる。また、図 2.17 は提案する推定方法の全体像を図式的に示したものである。

背景成分画像の推定

背景の推定には、メディアンフィルタを用いる。メディアンフィルタは、局所領域における中央値をフィルタリング結果画像として採用するもので、インパルス性の雑音などを除去することができる。提案法では、劣化文書画像において文字は背景に対する雑音であると仮定し、文字よりも十分大きな局所領域を設定し、元画像にメディアンフィルタを施すことにより、背景成分画像を得る。

具体的には、対象画像 I に対してメディアンフィルタを施したものを IM として、ピクセル p に対応する背景成分の推定値 ρ_p^b を次のように定義する。

$$\rho_p^b = IM_p \quad (2.38)$$

テキスト成分画像の推定

テキストの推定には、対象画像 I とメディアンフィルタ画像 IM を用いる。まず、 I と IM の絶対値差分画像 ID を次式により求める。

$$ID = |I - IM| \quad (2.39)$$

IM は文字成分が除去された画像であるから、 ID において各ピクセル値はそのピクセルの局所領域におけるイレギュラー度を表していることになる。したがって、 ID において大き

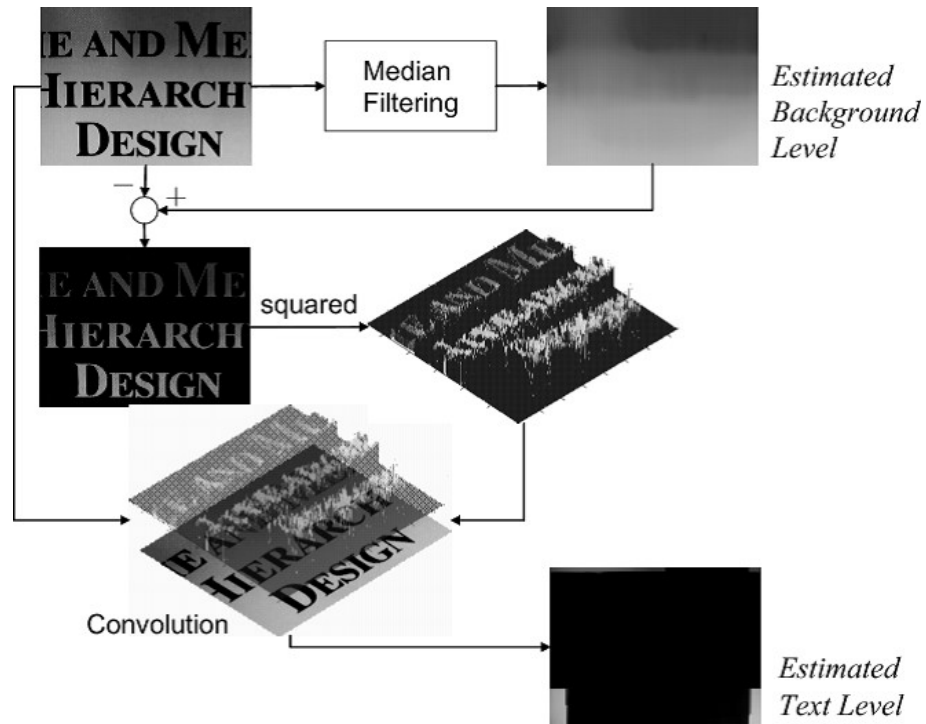


図 2.17: 背景成分とテキスト成分の推定法の全体像

なピクセル値となっている場所は文字の可能性が高いと考えられる。以上より、ピクセル p に対応するテキスト成分の推定値 ρ_p^t を次のように定義する。

$$\rho_p^t = \frac{\sum_{q \in W_p} \{I_q \times ID_q^2\}}{\sum_{q \in W_p} ID_q^2} \quad (2.40)$$

ここで、 W_p はピクセル p を中心とする局所領域、 q は W_p 内の任意のピクセルである。式 (2.40) では、各ピクセルのイレギュラー度 ID_q の 2 乗を重みとして、輝度値 I_q の重み付き平均をとることで、テキストの推定輝度値としている。

以上が照明条件や汚れなどの文字像劣化に対して頑健な文字抽出アルゴリズムである。図 2.18 に本アルゴリズムをテキスト画像に適用した結果を示す。図において、文字の色が異なる、照明条件による劣化が存在している、などの悪条件下で良い文字抽出が実行できているのが確認できる。

2.2.3 実験

提案手法の有効性を確認するため、2 色看板及び複数色看板を含む情景画像 25 枚に対して看板領域抽出及び文字領域抽出の実験を行った。実験に利用する情景画像の条件を画像内



(a) 入力画像 1



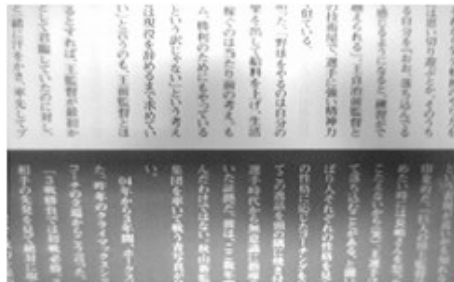
(b) 結果 1



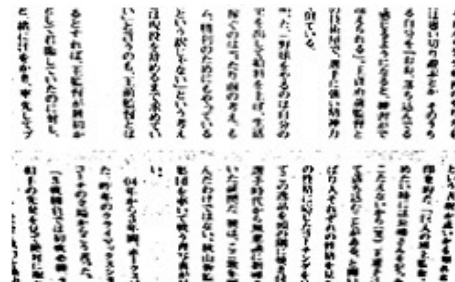
(c) 入力画像 2



(d) 結果 2



(e) 入力画像 3



(f) 結果 3

図 2.18: 照明変化・文字色変化を含む画像に対する文字領域抽出結果

表 2.2: 看板領域抽出率

	単色看板を含む画像	複数色看板を含む画像
抽出率	17/20 = 85%	4/5 = 80%

表 2.3: 文字抽出率

抽出文字数	抽出率
236/273	86.4%

に存在する文字情報が目視にて読み取れることとし、文字の種類や大きさに関係なく実験を行った。

2.2.3.1 看板領域の抽出

看板領域抽出結果を表 2.2 に示す。ここでは、評価基準として次式を用いる。

$$\text{抽出率} = (\text{正しく抽出された画像数}) / (\text{適用画像数})$$

表より、実験に用いた全画像の八割に対して看板領域を正確に抽出することができた。

また、看板領域の抽出結果の例を図 2.19 から図 2.24 に示す。(a) が原画像であり、(b) が看板領域の抽出結果である。図 2.21 のような単色の看板領域だけでなく、図 2.21 のような複数色を含む看板領域においても正しい抽出が実行できていることが確認できる。

また、図 2.25 に看板領域抽出に失敗した例を示す。本入力画像からは提案手法を適用しても看板領域は抽出できなかった。これは、本画像中で看板が全体的に暗く映り、明度が低くなって看板の近傍に存在する領域と分離ができなかったためであると考えられる。

2.2.3.2 看板領域内での文字抽出

次に、各看板領域抽出画像に対して、文字領域の抽出実験を行った。文字抽出率を表 2.3 に示す。ここで、文字抽出が正しく行えたかの評価には市販 OCR ソフトを用い、次式により文字抽出率を算出した。

$$\text{抽出率} = (\text{OCR により正しく認識された文字数}) / (\text{全文字数})$$

文字列領域の抽出結果を図 2.26 から図 2.28 に示す。図 2.26 や図 2.27 のように、横に書かれた文字だけでなく、縦に書かれた文字も抽出することができた。また、日本語だけでなく、英字で書かれた看板の文字も抽出することができた。



(a)



(b)

図 2.19: 看板領域抽出結果 1



(a)



(b)

図 2.20: 看板領域抽出結果 2



(a)



(b)

図 2.21: 看板領域抽出結果 3



(a)

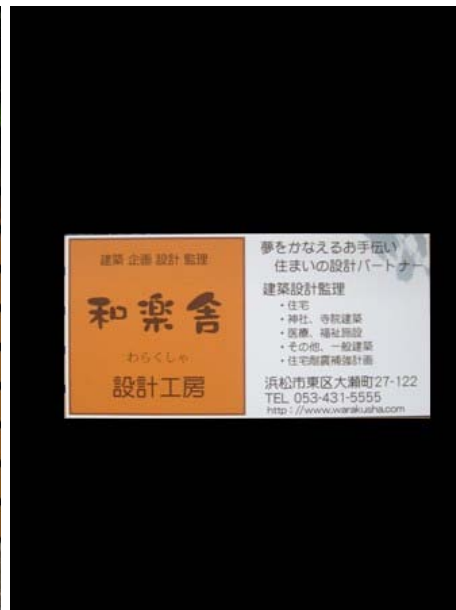


(b)

図 2.22: 看板領域抽出結果 4



(a)



(b)

図 2.23: 看板領域抽出結果 5



(a)



(b)

図 2.24: 看板領域抽出結果 6



図 2.25: 看板領域抽出に失敗した原画像

MIDDLEBOROUGH

図 2.26: 文字抽出結果 1

ニューメディア
HOME 管理
アイ住宅サービス
8882-1001

図 2.27: 文字抽出結果 2

三山本工業
TEL 3746

図 2.28: 文字抽出結果 3

2.2.4 まとめ

本節では、視覚情報からの文字領域抽出手法について提案した。提案手法では、ユーザが必要とする文字情報は看板や標識などの人工物上に描かれているという知見に基づき、情景画像中から看板領域を特定する「文字存在領域選定」と照明条件などに影響を受けにくい文字抽出を実現する「文字候補領域内における文字成分と背景成分の推定」の二つの主要なプロセスにより文字領域抽出を行った。「文字存在領域選定」では、看板領域は単色または複数色の領域で構成されているという特徴を用いて画像分割を行い、取り出された各領域に対して条件付けによる選別を行うことで看板領域（文字存在領域）の抽出を行った。また、「文字候補領域内における文字成分と背景成分の推定」では、文字の2値化をMAP-MRFに基づく最適化問題として定義し、グラフカットアルゴリズムを用いてその問題を解いた。実験では、提案した文字抽出法を実際に情景画像に適用し、看板領域と文字抽出共に良好な結果を示した。しかしながら、看板領域抽出に関しては、看板が全体的に影に覆われている等、他の領域との色情報による分離が困難な場合においてうまくいかない例もあった。今後の課題としては、色情報の他に距離などの情報を用いることで、看板領域抽出の精度向上を行なっていきたい。

2.3 弱視者の視覚特性に基づく事前学習による文字構造変換法

前節までは、視覚情報からの文字情報抽出について述べた。本節では、抽出された文字情報をユーザに提示する方法について議論する。

2.3.1 事前学習による文字構造変換の必要性

「弱視」と呼ばれる視覚障害により、日常生活に不自由を感じている人は多い。とくに、街中の看板・標識、本や雑誌、機器や端末の操作法など、生活空間にあふれる文字情報を見ることが困難な場合、視覚によって得られる情報は非常に限定的なものとなる。図 2.29 に文字画像に対する弱視者の見え方のシミュレーションを示す。図からわかるように、文字の太さや構造、大きさと認識のしやすさが大きく変化することがわかる。

このように、弱視の症状やその度合いによっては、文字の大きさや太さの違いの影響で大きな文字でも何が書いているのか認識できない場合があると考えられる。また、文字にも書体によっていろいろな特徴があり、症状と文字書体の組み合わせによってはより認識しづらくなる可能性がある。例えば、明朝体は縦線に比べ横線が細く、文字の太さが一定ではないため白内障では認識しづらくなる。他にもゴシック体は文字の太さが一定で読みやすいが屈折異常により文字がぼやけて文字の穴が潰れてしまい認識しづらくなってしまふ。

そこで本テーマでは、「像がぼけて見える」、「視野が狭くなる」といった個々の見え方に対して視覚情報処理を行うことでユーザに「認識しやすい」像を提供することを目的とし、デジタル文字画像を文字のフォント種に依らず弱視者が認識しやすい文字像へ変換するアルゴリズムを提案する。提案法では、事前に用意された各種文字像とその各種文字に対するぼけても認識しやすい理想的な文字像とのペアからそれらの関係性を学習し変換を行う。以下、弱視の特徴やその見え方について 2.3.2 節で再考し、提案手法について 2.3.3 節で詳述する。

2.3.2 弱視

弱視は日本において昔から一定限度以下の視力を有する者はすべて弱視とするという定義があり、学校保健法などに記載されている弱視もこれに準じている。さらにいうと「低視力、または視力以外の視覚機能障害があり、学習や日常生活上に制約があるが、ある程度視覚における様々な行動ができる者」と浮定義が一般的である。この定義は原因のいかんを問

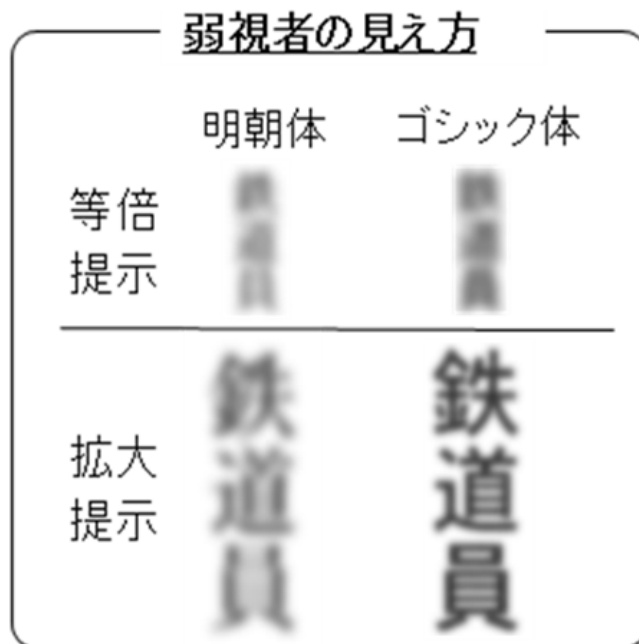


図 2.29: 弱視者が文字を認識できない場合の例

わずに一定以下の視力を有する者すべてを含んでいる。これらは一般的に社会的弱視、またはロービジョンと呼ばれている。

社会的弱視に対して、狭義の弱視で医学的弱視というものがある。医学的弱視とは視覚機能の発達過程において、先天性または後天性の障害因子によっておこるものとされる。また、現在日本において、弱視とは視覚の発達期に視性刺激遮断あるいは異常な両眼相互作用によってもたらされる片眼あるいは両眼の視力低下で、眼の検査で器質的病変はみつからず、適切な症例は予防、治療が可能なものという定義が広く受け入れられている。

弱視の定義において一定以下の視力を弱視とすると記述したが、一般的には「矯正視力が0.3未満」つまり、メガネやコンタクトレンズなどを使用しても視力が十分でない状態のことをいう。なので「裸眼視力では低くても、メガネまたはコンタクトレンズをつけると視力が1.0になる」という場合は弱視ではない。

また視力以外にも、視野が狭くなる視野狭窄や暗点などの視野障害もあり、様々な症状を合わせて持っている人もいる。

2.3.2.1 弱視の見え方の例

弱視者の視界の見え方は症状や個性差が大きく千差万別である。また、周りの明るさ、文字の大きさ、太さ、コントラストの違いによって弱視者にとっての見やすさが変わって来る。

近視と遠視

症状のひとつに近視と呼ばれるものがある。正確には弱視とは違うといわれるが見え方の例ということで記述しておく。近視とは遠くを見たときに、網膜より手前に光の焦点が結ばれてピントが合わず、ものがぼんやり見えてしまう屈折異常のことである。角膜や水晶体による光の屈折率が強すぎることによる屈折性近視と、眼球が通常より前後に長いために網膜よりも前方に焦点を結んでしまう軸性近視がある。眼に入った映像は、網膜に映し出され網膜でピントが合うのが通常であるが、これら近視の場合はピントが網膜手前であわされピントが合わずぼんやりと見える。

近視の逆の症状で遠視というものがある。近視がピントの前で合わさるのに対し、遠視はピントが網膜の後ろで合わさる。遠近いずれの目標にピントを合わせようとすると正常以上の調節力を常に働かせることになるので眼が疲れるという人が多い。

視野の制限

見える範囲がどんどん狭くなって見えにくくなる状態で、視野狭窄と呼ばれる。視界の端がだけでなく、視界の真ん中から見えなくなっていくこともある。真ん中から見えなくなるこの状態は中心暗転と呼ばれている。見えなくなる場所は人によって様々である。

この症状は「網膜剥離」などの網膜障害や、緑内障などの神経障害で起こりやすいとされる。

光量調節

暗いところから明るいところに出ると動けないほどのまぶしさを感じたり、逆に明るい場所から暗い場所に行くと目が慣れるのに時間がかかったりする。まぶしさは特に晴れた昼間に起こりやすい。

白内障

また、中には疾患によって弱視なる場合もある。疾患による弱視として有名なのは白内障という症状である。これは俗に「眼のレンズ」と呼ばれる水晶体が混濁することで視界が白く曇ってしまう疾患である。いろいろな種類があり先天性と後天性に大別されるが、原因や混濁の形状及び位置によっても分類される。後天性白内障には、老人性白内障、糖尿病性白内障、外傷性白内障などあり、ほかに層間白内、点状白内障、中心白内障などがある。とくに近年は老化に伴う老人性白内障が増加している。この老人性白内障は白内障という症状の中では最も患者が多く、高齢化社会に伴いさらに増加する可能性が懸念されている。

2.3.3 提案手法

弱視者の見え方には、像がボケて見えるという特性がある、図 2.29 はその特性を模擬したものである。この例は、晴眼者（健常視力の人）には読める文字サイズであっても、弱視者には読むことができず、拡大して提示したとしても、フォント種によっては読みにくい場合があることを示したものである。このように、弱視者のためには単純な拡大像の提示を行うのみでは不十分であり、「見やすさ」に配慮した拡大、文字の構造変換というものが必要である。

本研究では、ウェアラブルカメラを用いた視覚支援システムへの実装を前提に、「見やすさ」に配慮した文字構造変換を事前学習を用いた事例ベース処理により実現する。提案手法の概要を図 2.30 に示す。提案手法では、まず文字画像に対して細線化処理を適用することにより文字の骨格を抽出する。その後、事前の学習に基づいて太字化・構造変換処理を行う。以下、2.3.3.1、2.3.4においてそれぞれの処理について詳説する。

2.3.3.1 細線化

ここでは、構造変換の前処理として入力文字画像に対して行う細線化処理について述べる。一般に、ユーザが生活する日常空間には多種多様な太さ・フォント種の文字が存在する。本研究ではこれらの“文字の多様性”に対応するため、細線化処理を用いる。文字画像に対して細線化を行った場合、文字の装飾や太さ情報が失われて文字骨格が抽出され、この文字骨格に対する構造変換処理を学習することで、提案法は様々な太さ・フォント種の文字に適用することが可能となる。文字画像は文字領域を階調値 0、背景を 255 と事前に 2 値化され



ARシステムによるユーザへの提示

図 2.30: 提案手法の概要

ているとし、細線化処理には田村の手法を用いる。

田村の手法 [19]

細線化は文字線の中心1ピクセルだけ残すように線を細くする処理のことである。また、線が途中で切れたり、孔が開いたりしてはいけない。以下に田村の細線化アルゴリズムを詳述する。まず前処理として、入力画像の画素値を取り出し0と255で2値化する。これによって黒の画素、いわゆる文字の部分の画素値は0に、その他の文字以外の部分の画素値は255になり、文字を構成している画素と背景の画素が分けられる。

(図 2.31)の文字画像について、細線化処理を行うと、(図 2.32)のように1ピクセルで形成された画像が検出される事が期待される。

細線化の手法は古くから研究され、種々の方法が知られている。得られた線が元の図形の中心部に来るように、左上→右下、右下→左上、右上→左下、左下→右上の順に周囲から狭めてゆく方法を採用。注目する「1(黒)」のピクセルを中心に、 3×3 のピクセルの半分が「1(黒)」で他方が「0(白)」の場合は、注目する点を「0(白)」にしても良いと考えるのが基本原理である。

図 2.33 に示すように、消去できるかどうかを調べるピクセルを「center」と呼び、周囲

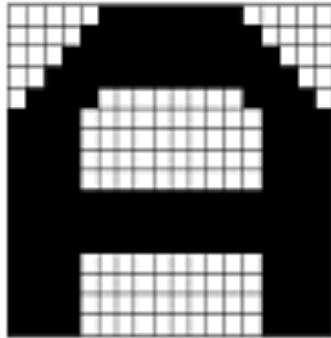


図 2.31: 2 値化画像

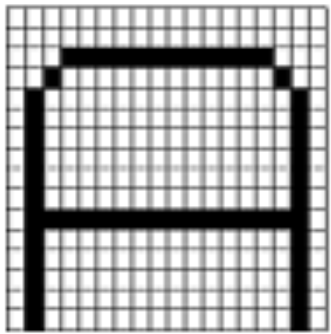


図 2.32: 細線化画像A

p[0]	p[7]	p[6]
p[1]	center	p[5]
p[2]	p[3]	p[4]

図 2.33: 消去の可否を調べるピクセルと周辺のピクセルの呼び方

0	0	0
X	①	X
1	1	1

0	0	X
0	①	1
X	1	1

0	X	1
0	①	1
0	X	1

図 2.34: 左上からの探索で消去できる条件

のピクセルは、反時計回りに、 $p[0]$ から $p[7]$ と呼ぶ。center, $p[0] \sim p[7]$ は 1 (黒) または 0 (白) の値を持つ。

画面左上から右下に向かって探索する場合に使用するパターン

画面左上から右下に向かって探索する場合には、左上が「白 (0)」に、右下が「黒 (1)」になっている場合を探す。この条件は、下記のような 3 種類の状態がある。いずれの場合も、中央のピクセル (丸で囲まれている) を「1」から「0」に変更 (消去) することができる。「X」は Don't care (「1」または「0」のどちらでも良い) を示す。

(図 2.34) より、この条件を表す式は、左から

- $p[2] * p[3] * p[4] = 1$ で、かつ $p[6] + p[7] + p[0] = 0$
- $p[3] * p[4] * p[5] = 1$ で、かつ $p[7] + p[0] + p[1] = 0$
- $p[4] * p[5] * p[6] = 1$ で、かつ $p[0] + p[1] + p[2] = 0$

である。このような条件式は、添え字を 1 ずつ増やし (7 の次は 0) ながら *for* ループで実行できる。上の場合には、添え字が 2 から始まるので、この値を *start* と呼ぶ。「1」が 3 個並んでいる、最初の場所が *start* であると考えるとわかりやすい。 $start = 2$ とは、左上から右下に向かって探索することを意味する。

画面右下から左上に向かって探索する場合に使用するパターン

画面右下から左上に向かって探索する場合には、右下が「白 (0)」に、左上が「黒 (1)」

1	X	0
1	①	0
1	X	0

X	0	0
1	①	0
1	1	X

0	0	0
X	①	X
1	1	1

図 2.35: 右下からの探索で消去できる条件

0	X	1
0	①	1
0	X	1

X	1	1
0	①	1
0	0	X

1	1	1
X	①	X
0	0	0

図 2.36: 右上からの探索で消去できる条件

になっている場合を探す。この条件は、下記のような3種類の状態がある。

(図 2.35) より、この条件を表す式は、左から

- $p[6] * p[7] * p[0] = 1$ で、かつ $p[2] + p[3] + p[4] = 0$
- $p[7] * p[0] * p[1] = 1$ で、かつ $p[3] + p[4] + p[5] = 0$
- $p[0] * p[1] * p[2] = 1$ で、かつ $p[4] + p[5] + p[6] = 0$

である。start = 6 である。

画面右上から左下に向かって探索する場合に使用するパターン

画面右上から左下に向かって探索する場合には、右上0が「白 (0)」に、左下が「黒 (1)」になっている場合を探す。この条件は、下記のような3種類の状態がある。

(図 2.36) より、この条件を表す式は、左から

- $p[0] * p[1] * p[2] = 1$ で、かつ $p[4] + p[5] + p[6] = 0$
- $p[1] * p[2] * p[3] = 1$ で、かつ $p[5] + p[6] + p[7] = 0$
- $p[2] * p[3] * p[4] = 1$ で、かつ $p[6] + p[7] + p[0] = 0$

である。start = 0 である。

画面左下から右上に向かって探索する場合に使用するパターン及び処理結果

画面左下から右上に向かって探索する場合には、左下が「白 (0)」に、右上が「黒 (1)」になっている場合を探す。この条件は、下記のような3種類の状態がある。

0	X	1
0	①	1
0	X	1

X	1	1
0	①	1
0	0	X

1	1	1
X	①	X
0	0	0

図 2.37: 左下からの探索で消去できる条件

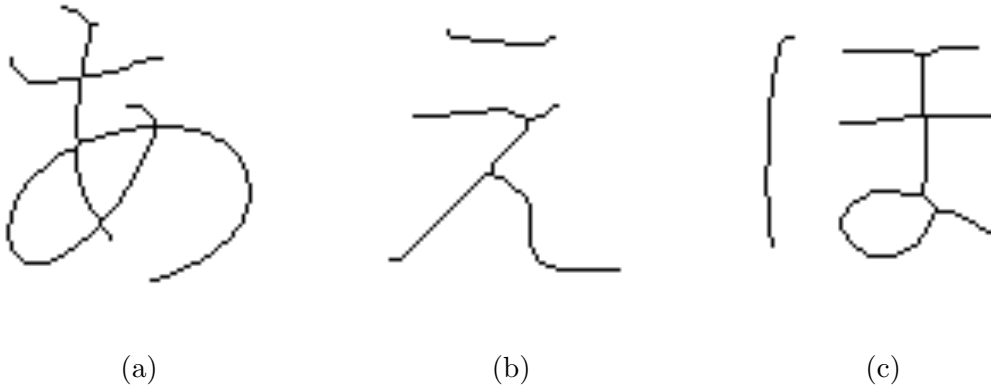


図 2.38: 各種文字に対する細線化結果. (a) “あ”, (b) “え”, (c) “ほ”.

(図 2.37) より, この条件を表す式は, 左から

- $p[4] * p[5] * p[6] = 1$ で, かつ $p[0] + p[1] + p[2] = 0$
- $p[5] * p[6] * p[7] = 1$ で, かつ $p[1] + p[2] + p[3] = 0$
- $p[6] * p[7] * p[0] = 1$ で, かつ $p[2] + p[3] + p[4] = 0$

である. $start = 4$ である.

図 2.38 に細線化の結果画像を示す.

2.3.4 構造変換アルゴリズム

ここでは, 細線化により得られた文字骨格画像に対する構造変換処理について述べる. 本研究では, ユーザが「見やすい」と感じる文字の太さや特性を構造変換に反映するため, 事前にフォント・太さの異なる同一文字のグループからユーザに最も認識しやすいと感じる文字を選んでもらい, それを用いて構造変換処理を行うためのパラメータを学習する. 以下, 2.3.4.1 では構造変換のための事前学習処理を, 2.3.4.2 では構造変換処理の詳細について述べる.

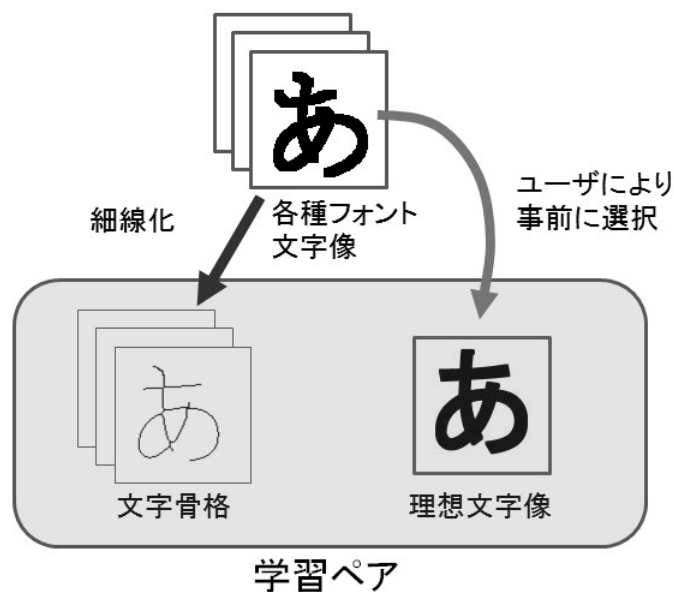


図 2.39: 学習ペアの作成

2.3.4.1 事前学習

本節では、ユーザの感性を反映した構造変換パラメータの学習について述べる。事前学習では、まず、フォント・太さの異なる同一文字のグループからユーザに最も認識しやすいと感じる文字を選んでもらい、これを数文字分行うことで認識しやすい「理想文字像」とその他の「認識し難い文字像」を得る。そして、「認識し難い文字像」を細線化し、「認識し難い文字像」の文字骨格に対する「理想文字像」という学習ペアを得る。図 2.39 にこの学習ペアの作成処理について示す。次に、これら学習ペアから学習データを作成する。学習データは図 2.40 に示すように、学習画像ペアからパッチ単位でラスタスキャンにより取得する。したがって、パッチサイズを $S_w \times S_w$ とすると、画像パッチの各ピクセルの階調値を要素とした $D = S_w \times S_w$ 次元の骨格文字像ベクトルデータ $\mathbf{y}_i, (i = 1, \dots, N)$ とそれに対応する理想文字像ベクトルデータ $\mathbf{x}_i, (i = 1, \dots, N)$ を得ることができる。

本研究ではリッジ回帰を用いて \mathbf{y}_i と \mathbf{x}_i の関係性を学習する。リッジ回帰では、次式

$$\sum_i \left\{ (\mathbf{y}_i - \mathbf{A}\mathbf{x}_i) (\mathbf{y}_i - \mathbf{A}\mathbf{x}_i)^T + \alpha \mathbf{A}^T \mathbf{A} \right\} \quad (2.41)$$

の対角要素を最小にする \mathbf{A} を文字像間関係性を表す射影パラメータ行列とする。この目的関数は通常の回帰分析における最小二乗法に正則化項を加えたもので、パラメータの二乗

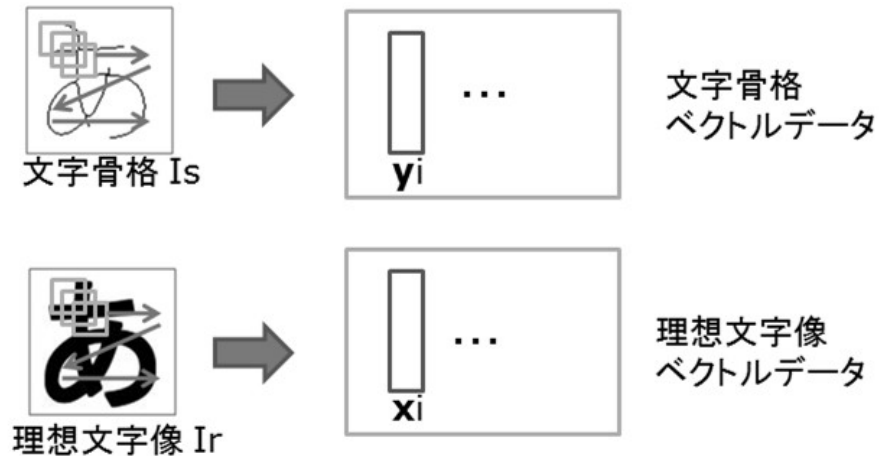


図 2.40: 学習データの取得

和がある定数以下になるよう制約をつけたものである。したがって、射影パラメータ行列 \mathbf{A} の推定値 $\hat{\mathbf{A}}$ は次式のように求められる。

$$\hat{\mathbf{A}} = [\mathbf{X}^T \mathbf{X} + \alpha \mathbf{I}]^{-1} \mathbf{X}^T \mathbf{Y} \quad (2.42)$$

ここで、 \mathbf{X} は理想文字像データ行列 $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ 、 \mathbf{Y} は骨格文字像データ行列 $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]$ 、 \mathbf{I} は $D \times D$ 次元の単位行列、 $\alpha \geq 0$ は複雑度を表すパラメータとする。

2.3.4.2 構造変換処理

本節では、文字の構造変換処理の詳細について述べる。本処理において、入力文字画像はあらかじめ細線化により骨格文字化されているとする。本処理では、まず、入力骨格文字像から学習時と同様にラスタスキャンによりパッチ単位の階調値ベクトルデータ $\mathbf{p}_i, (i = 1, \dots, N)$ を得る。そして、各骨格文字像データ \mathbf{p}_i に対して推定理想文字像 $\hat{\mathbf{q}}_i$ を次式により計算する。

$$\hat{\mathbf{q}}_i = \hat{\mathbf{A}} \mathbf{p}_i \quad (2.43)$$

ここで、 \mathbf{A} は 2.3.4.1 で作成した射影パラメータ行列である。そして、得られた理想文字像ベクトルデータ $\hat{\mathbf{q}}_i, (i = 1, \dots, N)$ の値を骨格文字像データ \mathbf{p}_i が切り出された画像上の元の

位置へ代入することで画像を再構成する。最後に、得られた画像に対して2値化処理を施し、出力画像とする。

2.3.5 予備実験

弱視者のために取得した画像を見やすい画像へと変換する過程においてまずは、弱視者がどのように見えているのかを知る必要がある。「文字の見やすさ」は周りの明るさ、文字の大きさ、太さ、コントラストの違いによって変わる。そこで、弱視者の視界を体験することができるシミュレーションレンズを用いて様々な条件下でディスプレイに映る文字を認識できるか確認した。

2.3.5.1 シミュレーションレンズ

使用したシミュレーションレンズは高田メガネのシミュレーションレンズトライアルセット<視覚障害模擬実験用>である。このトライアルセットは、屈折異常、白濁、視野狭窄などを人工的に起こすことができる。

屈折異常

0.01~0.3位まで視力を減ずるレンズで凸(+)レンズを使用している。正常眼を対象にしているので近視や遠視の方、また、眼の調節力などによって表示が異なることがある。一時的に近視の状態になるので近くのものを見たとき、物が拡大されて見える。

白濁

0.00~0.5くらいまでに白濁により視力を減ずるレンズである。眼の調節力に関係はない。

視野狭窄

見える視野を狭めるものである。3度,5度,7.5度,10度,12度,15度の6種類が存在する。

ND フィルター

見るときの光量を減ずるときに使用するフィルターである。

中心暗転

視野の中心に暗転シールが貼られている。また、暗転部のシールは取剥がして接着剤などを塗ることでより歪みのある像などを作成することができる。

表 2.4: 初期設定まとめ

	画像までの距離	使用レンズ	文字の書式
設定	70 cm	屈折異常 : 0.04 白濁 : 0.04	HGP ゴシック M HGP ゴシック E

2.3.5.2 初期設定

予備実験をするにあたって初期設定を決める。今回、弱視による見え方の違いについて文字の大きさ、太さ、コントラストの違いについて検証する。なので、文字画像までの距離、使用するシミュレーションレンズおよび使用する書式は一定にして実験を行った。

まず、文字画像を表示したディスプレイまでの距離を決める。ディスプレイまでの距離は一般的に50 cmから1 mの間が適正であることから、今回は間の70 cmの距離で予備実験を行った。

視力異常の代表的なものとして近眼と白内障の状態の検証を行うので、レンズは屈折異常と白濁を使用する。また、近眼および白濁のシミュレーションレンズは0.01~0.3位の視力まで減ずることができるので、検証できる視力の中で真ん中より低めの視力ということで0.04の視力レンズを使用する。

使用する書式は、HGP ゴシック M及びHGP ゴシック Eを使用した。ゴシック体は文字の太さが一定なので太さの違いによる見えやすさ、見えにくさが一定になるのでこれを選んだ。また、ゴシック Mは細めの文字、ゴシック Eは太めの文字なのでこの2種類の書式をつかうことで太さによる見えやすさの違いについて検証することができる。また、見る対象の文字画像はひらがな50音順に並べたものを用意した。

初期設定を以上のように決め、後は文字の大きさ及びコントラストを変えて検証を行った。なお、初期条件を(表 2.4)にまとめた。

2.3.5.3 結果

実験の結果を以下の表にまとめる。なお、実験ではディスプレイに映る文字が認識できるかどうかを確認する。

屈折異常レンズ、背景輝度:255 の場合

今回の場合ではフォントサイズが36位から認識ができる文字が出始めて、フォントサイ

表 2.5: 屈折異常レンズ 背景輝度:255

フォントサイズ	HGP ゴシック M	HGP ゴシック E
2 4	認識できない	認識できない
3 6	認識しづらい	認識しづらい
4 8	認識しづらい	認識しづらい
6 0	認識できる	認識できる
7 2	認識できる	認識できる
8 4	楽に認識できる	楽に認識できる

表 2.6: 白濁レンズ 背景輝度:255

フォントサイズ	HGP ゴシック M	HGP ゴシック E
2 4	認識できない	認識できない
3 6	認識できない	認識できない
4 8	認識できない	認識しづらい
6 0	かなり認識しづらい	認識できる
7 2	認識しづらい	認識できる
8 4	認識できる	楽に認識できる

ズ:60 でどの文字も認識できるようになった. 文字の太さによる違いはあまり見られなかった. 以上の結果を (表 2.5) にまとめた.

白濁レンズ, 背景輝度 : 255 の場合

今回の場合では太字のフォントの HGP ゴシック E でフォントサイズ:48 から文字が認識し始め, 細めのフォントの HGP ゴシック M はフォントサイズ:60 から認識でき始めた. 以上の結果を (表 2.6) にまとめた.

屈折異常レンズ, 背景輝度 : 95 の場合

今回の場合ではフォントサイズ : 48 から認識できる文字が出始め, フォントサイズ : 72

表 2.7: 屈折異常レンズ 背景輝度:95

フォントサイズ	HGP ゴシック M	HGP ゴシック E
2 4	認識できない	認識できない
3 6	認識できない	認識できない
4 8	かなり認識しづらい	かなり認識しづらい
6 0	認識しづらい	認識しづらい
7 2	認識できる	認識できる

表 2.8: 白濁レンズ 背景輝度:150

フォントサイズ	HGP ゴシック M	HGP ゴシック E
2 4	認識できない	認識できない
3 6	認識できない	認識できない
4 8	認識できない	認識できない
6 0	認識できない	かなり認識しづらい
7 2	認識できない	認識しづらい
8 4	認識できない	認識できる

でほとんどの文字が認識できるようになった。以上の結果を(表 2.7)にまとめた。

白濁レンズ, 背景輝度 : 95

文字のフォントサイズを 100 以上にしても認識することはできなかった。

白濁レンズ, 背景輝度 : 150 の場合

今回の場合では太字のフォントの HGP ゴシック E でフォントサイズ:60 で眼を細めて注視すると認識できる文字が出始めて, フォントサイズ:84 で眼を細めることなく認識できるようになった。HGP ゴシック M ではフォントサイズ:84 でも認識することができなかった。以上の結果を(表 2.8)にまとめた。

表 2.9: 屈折異常レンズ 背景輝度:119 文字輝度 : 255

フォントサイズ	HGP ゴシック M	HGP ゴシック E
3 6	かなり認識しづらい	認識できない
4 8	認識しづらい	認識しづらい
6 0	すこし認識しづらい	すこし認識しづらい
7 2	認識できる	認識できる

表 2.10: 白濁レンズ 背景輝度:119 文字輝度 : 255

フォントサイズ	HGP ゴシック M	HGP ゴシック E
2 4	認識できない	認識できない
3 6	認識できない	認識できない
4 8	認識できない	かなり認識しづらい
6 0	認識できない	認識しづらい
7 2	認識しづらい	認識できる
8 4	認識できる	認識できる

屈折異常レンズ, 背景輝度 : 119, 文字輝度 : 255 の場合

今回の場合は文字輝度を 255 の白にして実験を行った。フォントサイズ:48 から認識でき始め、フォントサイズ:60 で認識できるようになった。以上の結果を(表 2.9)にまとめた。

白濁レンズ 背景輝度 : 119 文字輝度 : 255

今回の場合も文字輝度を 255 にして実験を行った。太字のフォントの HGP ゴシック E でフォントサイズ:48 から認識でき始め、フォントサイズ:72 で認識できるようになった。また、細字のフォント HGP ゴシック M はフォントサイズ:72 から認識でき始め、フォントサイズ:84 で認識できるようになった。以上の結果を(表 2.10)にまとめた。

白濁レンズ 背景輝度 : 0 文字輝度 : 255

今回は背景を黒、文字を白にして検証を行った。太字のフォントの HGP ゴシック E でフォ

表 2.11: 白濁レンズ 背景輝度:0 文字輝度 : 255

フォントサイズ	HGP ゴシック M	HGP ゴシック E
24	認識できない	認識できない
36	認識できない	認識しづらい
48	認識しづらい	認識できる
60	認識できる	認識しやすい

ントサイズ:36 から認識でき始め、フォントサイズ:48 で認識できるようになった。また、細字のフォント HGP ゴシック M はフォントサイズ:48 から認識でき始め、フォントサイズ:60 で認識できるようになった。以上の結果を (表 2.11) にまとめた。

2.3.5.4 考察

これらの結果により文字のフォントサイズは 72 ほどあればほとんどの弱視者でも文字を認識することができると考えられる。また、コントラストにおいては、背景輝度と文字輝度の差は大きければ大きいほどいいが、差が小さい場合は最低でも 150 は必要であると考えられる。

屈折異常レンズでは背景白に文字黒と、背景黒と文字白との差はあまり感じなかった。しかし、白濁レンズでは (表 2.6), (表 2.11) より、背景黒に文字白の組み合わせのほうが見やすいことが確認できた。これは背景と文字の領域を比べると背景のほうが領域が大きい。なので、領域の大きい背景を認識しやすい背景黒と文字白の組み合わせが見えやすかったと考えられる。

太さによる違いについては、白濁レンズでの差が顕著だった。(表 2.6)(表 2.8)(表 2.10) より太い文字である方がより認識しやすいことが分かる。これは太い文字のほうが背景と文字の領域が細い文字よりも明確に分けられるので、視界が白く濁るとき文字の領域を認識しやすい太字のほうが認識しやすかったと考えられる。

屈折異常レンズの場合は太字と細字に大きな差は見られなかったが、観測者の感覚では細い方が若干認識しやすいように感じた。近眼は遠くのものを見たとき視界がぼやけるので文字が細い方がぼやけた像で文字線の隙間をつぶしにくいためと考えられる。

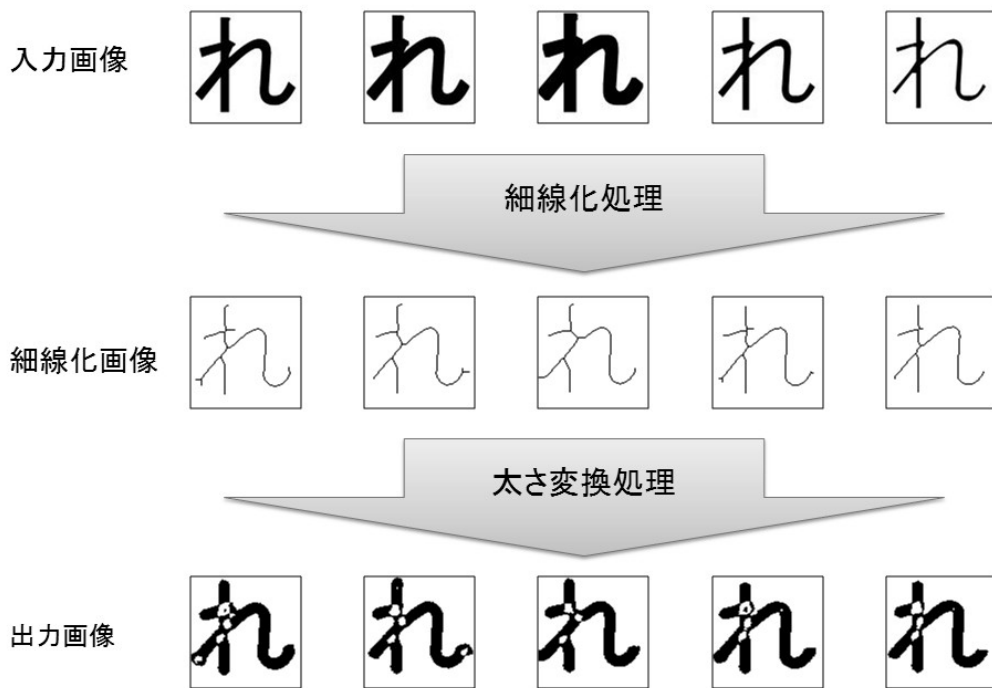


図 2.41: ひらがな“れ”に対する構造変換結果

2.3.6 評価実験

本節では、種々の文字画像に対して提案手法を適用し、その有効性を検証する。図 2.41, 2.42, 2.43 はひらがな“れ”, “ほ”, “を”の3種類に対して提案手法を適用した結果である。図からわかるように、各種太さの文字に対して、構造変換結果の文字の太さはほぼ同一となっている。しかしながら、各文字において何箇所か文字が空洞化している部分が存在する。これは学習をパッチ単位で行っているため、文字骨格をほとんど含んでいないデータの変換も同時に学習するという現象が起こり良好に学習が行えていないためと考えられる。そこで本論文では、構造変換結果に対して空洞を埋める「穴埋め」処理をしたものを最終出力として採用する。

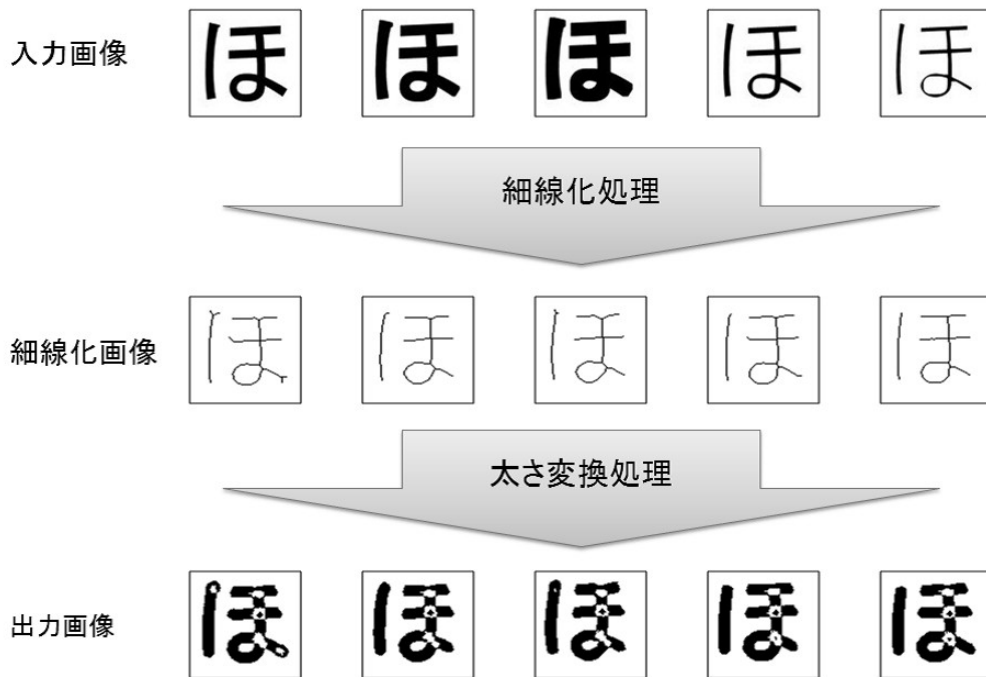


図 2.42: ひらがな“ほ”に対する構造変換結果

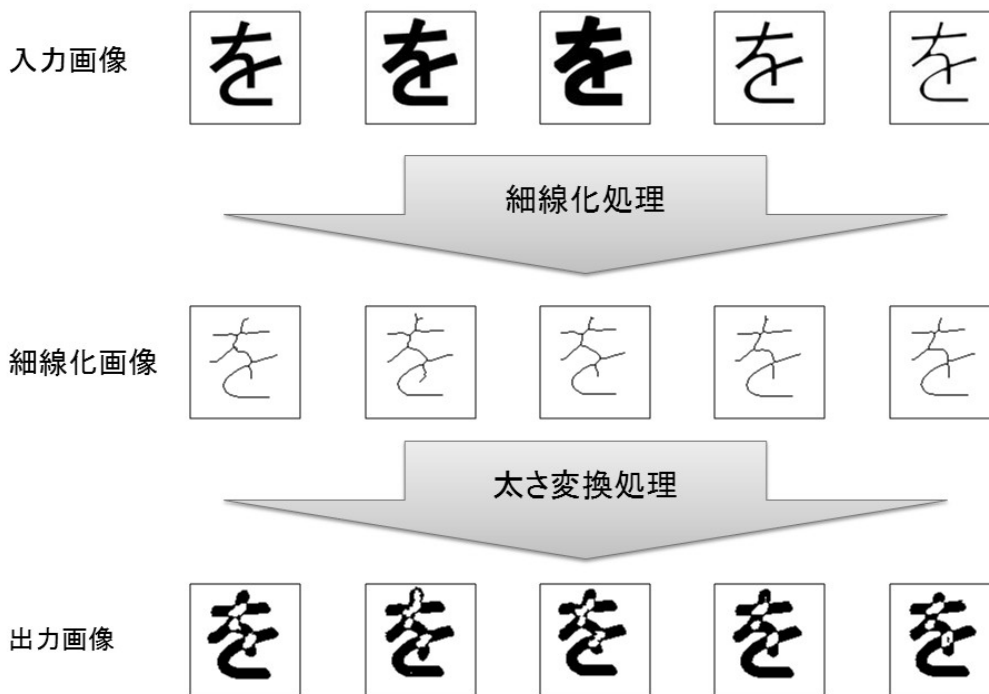


図 2.43: ひらがな“を”に対する構造変換結果

はFナンバーと呼ばれる。また、 $\cos \alpha$ の項で暗くなること（周辺減光）を表す。実際は、レンズが筒状になり、階層状の構造を有している。対象表面の明るさである放射輝度 L は、光が対象表面で反射したものである。それが、対象表面の放射輝度 L が与えられていると、あとは対象表面の放射輝度 L の対象表面での分布は、画像上で見ると、画像上の陰影から対象表面の空間での向きを推定する可能性がある。これが、陰影からの形状

反射したものと見ると、ある程度の明るさがある。これは、対象表面の放射輝度 L が与えられていると、あとは対象表面の放射輝度 L の対象表面での分布は、画像上で見ると、画像上の陰影から対象表面の空間での向きを推定する可能性がある。これが、陰影からの形状

(a)

(b)

反射したものと見ると、ある程度の明るさがある。これは、対象表面の放射輝度 L が与えられていると、あとは対象表面の放射輝度 L の対象表面での分布は、画像上で見ると、画像上の陰影から対象表面の空間での向きを推定する可能性がある。これが、陰影からの形状

(c)

(d)

図 2.44: 文書画像に対する結果. (a) 入力画像, (b) 拡大画像, (c) 構造変換結果, (d) 最終出力.

次に、文書画像に対する適用結果を図 2.44 に、ぼけに対する「見やすさ」の単純拡大画像に対する比較を図 2.45 に示す。図 2.45 はガウシアンフィルタにより弱視者の見え方を模擬したものである。図 2.44 からわかるように、穴埋め処理は有効に機能している。また、図 2.45 の弱視シミュレーション画像を見てもわかるように、構造変換を施した方が文字がくっきり、はっきりとし認識しやすいことが分かる。さらに、図 2.45 の画像の見やすさについて、晴眼者 10 人のアンケート評価を行った。アンケート評価は、非常に見やすい (5 点)、見やすい (4 点)、普通 (3 点)、見にくい (2 点)、非常に見にくい (1 点) の 5 段階で各画像を評価してもらい全評価者の平均をとることで行った。アンケート結果を表 2.12 に示す。表より、提案手法を適用した画像の方が見やすさの評価が高く、有効性が確認できる。

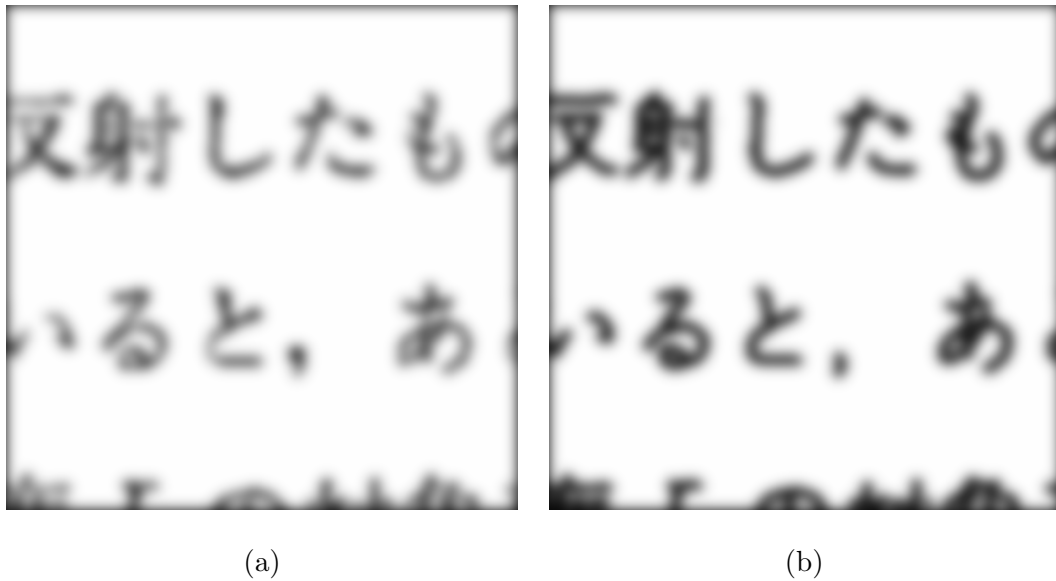


図 2.45: ボケに対する「見やすさ」の比較. (a) 単純拡大画像 (図 2.44 (b)) に対して弱視者が閲覧したときを模擬した画像, (b) 構造変換画像 (図 2.44 (d)) に対して弱視者が閲覧したときを模擬した画像.

表 2.12: 平均オピニオン評点 (10 人)

(a) 単純拡大画像	(b) 提案手法適用画像
3.1	3.7

2.3.7 まとめ

本節では、弱視者の「像がぼけて見える」という見え方に対して視覚情報処理を行うことでユーザに「認識しやすい」像を提供することを目的とし、デジタル文字画像を文字のフォント種に依らず弱視者が認識しやすい文字像へ変換するアルゴリズムを提案した。提案法では、フォント種に依らない文字の構造変換を細線化処理と事前学習を用いた事例ベース処理により実現した。実験では、まず弱視者がどのように見えているのかを知るためにシミュレーションレンズを用いた感応実験を行い、「見やすい」文字像とは何かについて検討を行った。そして、提案手法を種々の文字画像へ適用しその有効性を示した。今後は、学習の際のパッチサイズなどの各種パラメータ自動設定法及び「視野が狭くなる」等の症状に対応するためのユーザへの処理結果提示法などを検討していきたい。

第3章 視覚情報フィルタリングによる

弱視者の自律的移動支援システム

3.1 概要

弱視者は、眼鏡などの視力矯正器を用いても視力が 0.3 未満程度にしか矯正されず、全盲でないにも関わらず視覚によって得られる情報は非常に限定的なものとなる。そのため、生活空間に存在する段差や歩行を妨げる放置自転車などの弱視者にとっての潜在的な危険物を認識し難く、突発的な事故などの原因となる。このような問題に対し、さまざまな視覚支援システムの開発が取り組まれている。例えば、Fujiyoshi ら [20] は道路等に設置した環境端末及びユーザ携帯型移動端末を用い、認知・駆動・情報入手というユーザの 3 つの要素行動を補助する移動支援システムを提案している。このシステムは、監視カメラなどの環境端末とユーザが持つ携帯端末の通信により行動支援を行うため、環境端末がセンシング可能な範囲でしか支援を行なえずユーザの行動範囲が制約されてしまうといった問題がある。青木ら [21] はユーザ装着型の単一カメラによる歩行者用障害物検出システムを提案している。このシステムは従来の車載カメラ向けの障害物検出手法をベースに独自の改良を行っているが、画像中の特徴点解析を用いているためテクスチャの乏しい人工物の検出が不安定となる問題がある。また、ユーザに対する「認識しやすい」情報提示法などの検討はなされていない。

以上のように、弱視者の安全で自由な自律移動を支援する手法は確立されておらず、その実現のためには、「ユーザに装着するだけで実現できる完結したシステムであること」、「障害物検出・歩行可能領域抽出手法の頑健性」、「情報提示法への配慮」が求められる。とくに、情報の提示においては弱視者の視覚特性を考慮し「見やすさ」に配慮することが重要である。

本研究テーマでは、ウェアラブルカメラを用いた視覚支援システムへの実装を前提に、弱視者の視覚情報解析法と「見やすさ」に配慮した情報提示方法の開発を目的とする。具体的には、日常的なシーンにおけるユーザの視覚情報（ウェアラブルカメラより取得される動画）から、ユーザの行動を阻害する障害物の存在しない歩行可能領域の検出を行うアルゴリ

ズムの開発，及び，ヘッドマウントディスプレイ（HMD）を用いたユーザにとって「見やすい」情報の提示法についての検討を行う。

3.2 ウェアラブルカメラから得られる視覚情報からの

歩行可能領域抽出

弱視の定義は，両眼の矯正視力が 0.3 未満までしか視力の向上がみられず，学習や日常生活上に制約があるが，視覚における様々な行動ができる者とされている。視覚障害者の中で弱視者は，まったく視力がない全盲より遥かに多く約 8 割を占めている。また弱視にも視力・視野，夜盲や羞明（しゅうめい），疲労度や矯正後の視力の程度など個人差が幅が広く，大変理解されにくい。全盲者とは異なり，弱視者とはまったく視覚をもたないわけではなく，日常の諸行動ができる。したがって弱視者の外見からは目が不自由であるということが相手にも分かりにくく自分から言い出せずにいる人も多いため，実態が社会的に十分理解されていないというのが現状である。

弱視者に対する配慮として，階段の踏み面端部に色のコントラストの違いを付けた滑り止めや黄色の点字ブロック，点字がつけられた容器などがある。しかし，道路の点字ブロックや信号の音声案内は，まだ十分に整備されていない場所があり，缶詰やスナック菓子の袋といった容器には配慮が行われていない状況である。また，背丈の低い車止め，はみ出し看板，細い道路に立つ電柱など，弱視者にとって歩行の妨げとなる「潜在的危険物」は少なくない。したがって弱視者に対しての配慮が十分整っているとは言い難い。このような状況では弱視者は屋外において，安全な歩行環境を確保することができない。

このような現状において，弱視者にとって不自由なく歩行できる環境を確保するために，日常の生活空間において，進行の妨げとなる障害物を除いた歩行できる道路平面を抽出し，歩行できる道路平面を弱視者に伝えられるシステムは有用である。このようなシステムを用いることで，安全な歩行環境を弱視者に対して提供することができれば，弱視者に安全に歩行できる環境を提供する一助となりうる。

そこで本テーマでは，弱視者に対する安全な歩行環境の提供を目的とし，ユーザの視覚情報からの基礎的な歩行環境認識アルゴリズムについて議論，検討を行う。具体的には，ユーザの頭部等に取り付けたウェアラブルカメラから入手した歩行中動画像から，障害物の存在しない「歩行可能面」の検出を行う。本研究では，単一カメラのみを用いてユーザの歩行環

境を認識するため、画像間の特徴点对応を用いた時系列連続画像重ね合わせによる画像解析を用いる。その際、事前の人検出結果を用いることで、歩行面検出に有効な画像特徴点を効率的に発見する方法を提案し用いる。

以降、3.3節で画像特徴について本研究で用いるものも含めて概観し、3.4節で提案手法の詳細について、3.5節で実験結果とその考察について、3.6節で本テーマのまとめをそれぞれ述べる。

3.3 画像特徴

画像解析における特徴抽出とは、画像の持っている特徴を図形または数値に変換する処理のことを言い、デジタル画像処理の中で最も重要な処理のひとつである。本節では画像局所特徴量として SIFT (Scale-Invariant Feature Transform) ,SURF(Speeded Up Robust Features), SURF の特徴量記述で用いる Haar-Like 特徴量について述べる。

3.3.1 SIFT 特徴量

SIFT[22][23][24][25] の処理については特徴点 (キーポイント) の検出と回転やスケール変化、照明変化に不変な特徴量の記述の2段階に分けられる。特徴点とはコーナー点やあるピクセルを中心とした周辺領域に多くの濃淡徐情報を持つ点が特徴点として検出される。処理の流れは以下ようになる。

- (1) スケールとキーポイント検出
- (2) キーポイントのローカライズ
- (3) オリエンテーションの算出
- (4) 特徴量の記述

(1) のからスケールとキーポイントの検出においては、DOG(Difference-of-Gaussian) 画像からスケールとキーポイントを検出し、(2) のキーポイントのローカライズでは、(1) で検出されたキーポイントから特徴点に向かない点を削除し、点座標のサブピクセル推定を行う。(3) のオリエンテーションの算出では、回転やスケール変化、照明変化に不変な特徴を得る

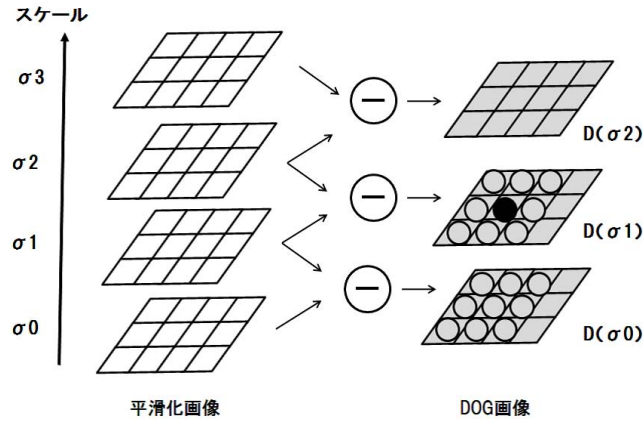


図 3.1: DOG 画像作成

ためキーポイントのオリエンテーションを求める。(4)の特徴量の記述では、(3)で求めたオリエンテーションに基づいてキーポイントの特徴量を記述する。

3.3.1.1 スケールとキーポイント検出

SIFT は、スケール変化による不変性を得るために、対象とする点の特徴をより表現できる近傍領域の範囲を自動的に決定する。範囲を決めるパラメータをスケールといい、最初にキーポイント検出時に最適なスケールを決める。スケール s のガウス関数 $G(x, y, \sigma)$ と入力画像 $I(x, y)$ を畳み込んだ平滑化画像 $L(x, y, \sigma)$ を以下の式により作成する。

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (3.1)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (3.2)$$

スケールの異なる平滑化画像間の差分処理を DOG といい、 $D(x, y, \sigma)$ で表す。 $D(x, y, \sigma)$ は、以下の式よりもとまる。

$$D_i(x, y, \sigma) = L(x, y, \sigma_{i+1}) - L(x, y, \sigma_i) \quad (3.3)$$

次に、図 3.1 に示すように DOG 画像 $D(x, y, \sigma)$ から極値の検出を行う。注目画素が 26 近傍領域において最大値もしくは最小値である場合、キーポイント候補点とする。また、このとき極値となる DOG 画像のスケールをキーポイントの特徴抽出に用いる。

3.3.1.2 キーポイントのローカライズ

検出されたキーポイント候補点には、低コントラストのような DOG 出力値が小さい点やエッジ上の点が含まれており、これらの点は開口問題が発生しやすいという問題がある。そこで、キーポイント候補点のなかから、エッジ上に存在するキーポイント候補点やコントラストの低い点を削除することにより安定したキーポイントを絞り込む。さらにキーポイントのサブピクセル推定にキーポイントの位置とスケールを推定し、より頑健な点をキーポイントとして選択する。

3.3.1.3 オリエンテーションの算出

検出された各キーポイントに対して、特徴量の記述を行う。まず、検出された各キーポイントのオリエンテーションを求める。オリエンテーションとはキーポイントにおける方向を表し、オリエンテーションにより向き正規化を行うことにより回転に不変な特徴量を求めることができる。キーポイントのオリエンテーションを求めるために、キーポイントが検出された平滑化画像 $L(u, v)$ の各画素の勾配 $m(x, y)$ とその勾配方向 $\theta(x, y)$ を以下の式より求める。

$$m(x, y) = \sqrt{f_x(x, y)^2 + f_y(x, y)^2} \quad (3.4)$$

$$\theta(x, y) = \tan^{-1} \frac{f_y(x, y)}{f_x(x, y)} \quad (3.5)$$

$$\begin{cases} f_x(x, y) = L(x+1, y) - L(x-1, y) \\ f_y(x, y) = L(x, y+1) - L(x, y-1) \end{cases} \quad (3.6)$$

式 (3.4)(3.5) で求めた局所領域における勾配の大きさ $m(x, y)$ と勾配方向 $\theta(x, y)$ から、36 方向に分割した重み付方向ヒストグラムを作成する。次に、作成した勾配方向を 36 方向に量子化したヒストグラムから、最大値 80% の以上になる方向をキーポイントのオリエンテーションとして割り当てる。

3.3.1.4 特徴量の記述

検出したオリエンテーションから 128 次元の特徴量を記述する。キーポイントの周辺領域を、割り当てられたオリエンテーション方向に回転する。特徴量を記述する際、キーポイ

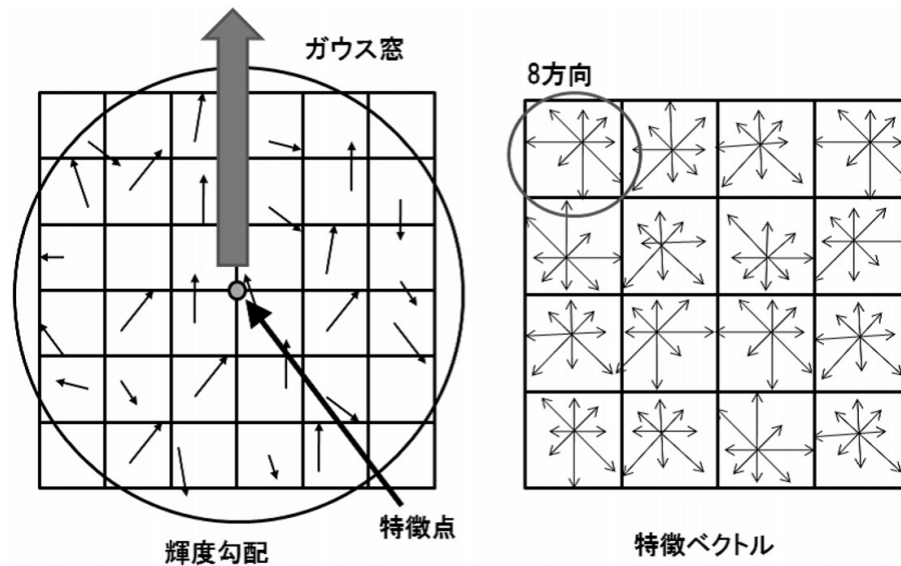


図 3.2: 特徴量の記述

ントの勾配情報を用いるため、回転に対する不変性が得られる。特徴量の記述の際には、図 3.2 に示すように、ガウス窓を用いて中心付近が強くなるように重み付けをする。このガウス窓の大きさは、キーポイントが検出された DOG 画像の平滑化スケールによって決定される。画像サイズが 2 倍であればスケールも 2 倍となり、構造的に同一の範囲における特徴量の記述を行うことになる。これにより、スケール変化に対する不変性が得られる。次に、分割した領域ごとに方向ヒストグラムを作成する。図 3.2 のように、領域を 4×4 の 16 に分割し、それぞれに 8 方向のヒストグラムを作成するため、 $4 \times 4 \times 8 = 128$ 次元の特徴ベクトルとしてキーポイントの特徴量を持つことになる。128 次元の各特徴ベクトルの長さはベクトルの総和で正規化する。したがって、キーポイントは照明変化に対しても影響の少ない特徴量となる。

3.3.2 Haar-Like 特徴量

Haar-Like 特徴量 [18] は白の領域 r_1 と黒の領域 r_2 の二つの領域の輝度差を式 (3.7) より算出して Haar-Like 特徴量としている。

$$H(r_1, r_2) = S(r_1) - S(r_2) \quad (3.7)$$

$S(r)$ は領域 r の輝度差を算出する関数である。2 つの領域のパターンは、縦方向の明暗差、横方向の明暗差をとらえることが可能である。SURF では図 (3.3) のような Haar-Like 特徴



図 3.3: SURF で用いる Haar-Like 特徴量

量を用いる.

このようなパターンとともに位置やスケールを網羅的に変化させることで膨大な特徴量を生成する.

3.3.3 SURF 特徴量

画像局所特徴量の Speeded Up Robust Features (SURF) [26][18] とは先述の SIFT と同様に画像スケール, 回転, 輝度変化, オクルージョンに耐性のある特徴量で特徴量生成手法と特徴次元数の観点で SIFT より高速な手法である. SURF は上記特性により同一オブジェクトを精度よく検出する手法である. 以下に処理の流れについて以下に示す.

3.3.3.1 キーポイント検出

SURF は SIFT のキーポイント検出と特徴量記述の処理を BOX フィルタと積分画像を用いることで近似することが可能となり大幅な高速化を実現している.

SIFT ではキーポイントとスケール検出のために σ の異なる LoG を複数回処理する必要があった. この処理は SIFT のアルゴリズムの中で最も計算コストが大きい. SURF では, キーポイント検出器のため, Hessian-Laplace 検出器を近似した BOX フィルタを用いる. BOX フィルタの出力は積分画像を用いることで高速に求めることが可能となる. Hessian-Laplace 検出器は次式で示すヘッセ行列を用いる.

$$\mathbf{H}(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (3.8)$$

$L_x x(x, \sigma)$ は画像中の点 x におけるガウシアン二次微分のたたみこみ結果である。その他についても同様である。このヘッセ行列の行列式

$$\text{Det}(H) = L_{xx}(x, \sigma) \cdot L_{yy}(x, \sigma) - (L_{xy}(x, \sigma))^2 \quad (3.9)$$

を各スケール画像に対して求め、SIFT と同様に x, y, σ の三次元空間における極大値を検出する。検出された点をキーポイントとする。

SURF ではこの Hessian-Laplace 検出器の近似として BOX フィルタを用いる。BOX フィルタとは加重平均フィルタの一種でガウシアンフィルタと二次微分フィルタをかけた結果の近似を得ることができる。得られる近似値をそれぞれ D_{xx}, D_{yy}, D_{xy} とするとき、BOX フィルタによる近似 Hessian-Kaokace 検出器 $\text{Det}(H_{approx})$ は次の式で定義される。

$$\text{Det}(H_{approx}) = D_{xx} \cdot D_{yy} - (D_{xy} \cdot 0.9)^2 \quad (3.10)$$

ただし、式 (3.10) は近似による誤差を修正した式となっている。

3.3.3.2 特徴量記述

SURF では先述の Haar-Like を計算することで輝度勾配を求める。まず、回転に対する不変性を得るためにキーポイントのオリエンテーションを求め、スケールを s としたときキーポイントを中心とした半径 $6s$ の領域から x, y 方向それぞれの Haar-Like(サイズ: $4s \times 4s$) を計算する。次にキーポイントを中心に $\sigma = 2.5s$ のガウシアンフィルタを用いて重みをかける。求められた半径 $6s$ の領域内の重み付き輝度勾配を x, y 方向ごとに総和する。以上の処理をキーポイントを中心として 60° ずつ回転させながら繰り返し、最も値の大きい方向をキーポイントのオリエンテーションとして採用する。

特徴量の記述において、キーポイントを中心とした $20s \times 20s$ の正方形の領域を 4×4 のグリッド領域に分割し、その各グリッド領域について Haar-like(サイズ: $2s \times 2s$) を計算することで輝度勾配を求める。この輝度勾配の x, y 方向の値をそれぞれ d_x, d_y とするとき、4次元ベクトル $\mathbf{v} = (\Sigma d_x, \Sigma d_y, \Sigma |d_x|, \Sigma |d_y|)$ をそれぞれ求める。したがって、16領域 \times 4次元 = 64次元の特徴ベクトルを得る。この値が SURF の特徴量となる。

3.3.4 画像間における対応点探索によるマッチング

ある2枚の画像において画像間で対応する同一の領域（対応点）を求めるためには、一般的に、各画像から局所特徴量を検出し類似度を用いた探索により実現する。例えば、SIFT特徴量を用いると、異なる画像間で抽出された各キーポイントの特徴量を比較することで、画像間の対応点探索が可能となる。以下に、対応点探索の流れを示す。画像 I_1 中のあるキーポイント k_{I_1} と画像 I_2 中のあるキーポイント k_{I_2} の特徴量をそれぞれ $\mathbf{v}^{k_{I_1}}$ 、 $\mathbf{v}^{k_{I_2}}$ とすると、特徴点間のユークリッド距離 d は以下のように求められる。

$$d(\mathbf{v}^{k_{I_1}}, \mathbf{v}^{k_{I_2}}) = \sqrt{\sum_{i=1}^{128} (v_i^{k_{I_1}} - v_i^{k_{I_2}})^2} \quad (3.11)$$

ここで、SIFT 特徴の次元数は 128 次元である。あるキーポイント 1 点に対して、異なる画像中に含まれる全キーポイントとの特徴点間の距離 d を算出し、特徴量間の距離 d が最も小さい点 k_1 と 2 番目に小さい点 k_2 について、 d_1 を k_1 のユークリッド距離、 d_2 を k_2 のユークリッド距離としたとき、 $d_1 < d_2 \times 0.6$ を満たす場合に k_1 を対応点として検出する。

3.3.5 HOG 特徴量を用いた人検出

HOG(Histograms of Oriented Gradients)[28],[29],[30] とは、アピアランスに基づいた局所領域における輝度の勾配方向をヒストグラム化した特徴ベクトルであり、物体の形状を表すことができる。近接画素の勾配を局所領域によってヒストグラム化するため、照明や影の影響を受けにくく、局所的な幾何学変化に頑健という特徴がある。以下に HOG 特徴ベクトルの算出法を示す。

3.3.5.1 セルによるヒストグラム化

式 (3.4) (3.5) と同様にして算出した勾配の大きさ $m(x, y)$ と勾配方向 $\theta(x, y)$ を用いて、ある整数 n としたとき $n \times n$ ピクセルをセルとした領域において、勾配方向を $0^\circ - 180^\circ$ とした輝度の勾配方向ヒストグラムを 20° ずつに分割して 9 方向の勾配ヒストグラムを作成する。

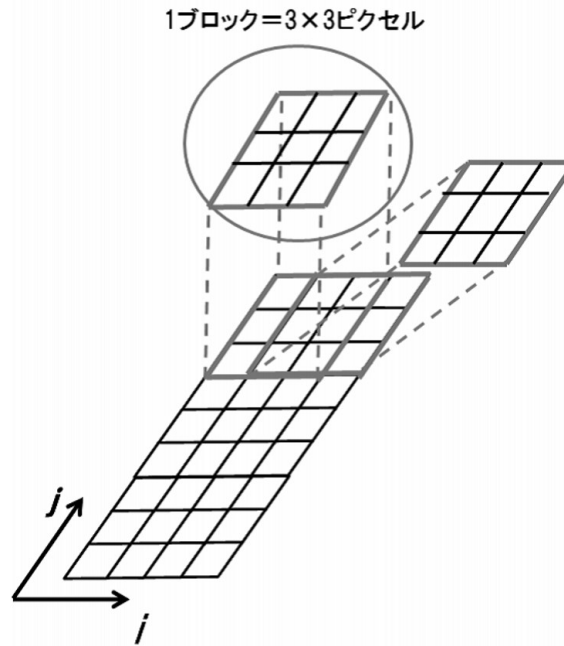


図 3.4: ブロックによる正規化

3.3.5.2 ブロックによる正規化

各セルにおいて作成したヒストグラムを 3×3 セルを 1 ブロックとして正規化を行う. i 行 j 列のセル (i, j) の 9 次元の特徴量を $F_{ij} = [f_1, f_2, \dots, f_9]$ とすると, k 番目のブロックの特徴量は

$$V_k = [F_{ij}, F_{i+1j}, F_{i+2j}, F_{ij+1}, F_{i+1j+1}, F_{i+2j+1}, F_{ij+2}, F_{i+1j+2}, F_{i+2j+2}] \quad (3.12)$$

と表現できる. 正規化後の特徴ベクトルを \mathbf{v} とすると次の式で正規化を行う.

$$\mathbf{v} = \frac{f}{\sqrt{\|V_k\|_2^2 + e^2}} \quad (3.13)$$

正規化は図 3.4 のようにブロックを 1 セルずつずらしながら全領域に対し行う. 特徴量 f は異なるブロックの領域によって何度も正規化される. 以上により特徴量を得る.

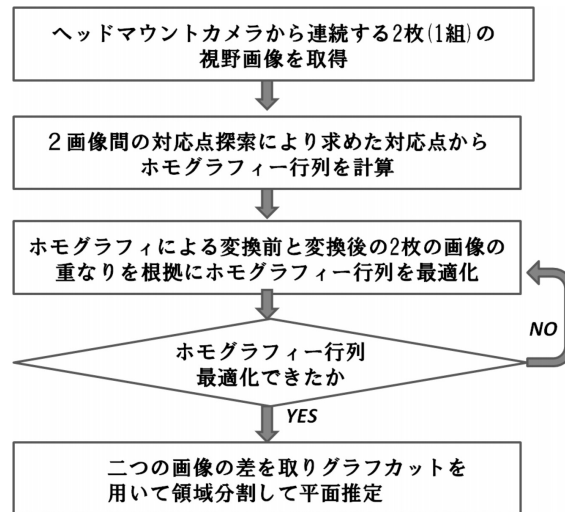


図 3.5: 提案手法の流れ



(a) SIFT によって抽出された全対応点 (b) GA によって選択された対応点

図 3.6: 対応点探索による特徴点抽出

3.4 提案手法

3.4.1 概要

本節では、提案手法のアルゴリズムについて述べる。提案手法の流れを図 (3.5) に示す。

本テーマでは、ヘッドマウントカメラ (HMC) を用いることを想定し、まずは HMC で撮影した歩行中における時間的に連続する 2 つの視野画像を取得する。次に SIFT を用いて 2 つの画像間の対応点探索を行い、対応点を取得する。SIFT によって取得した全ての対応点から遺伝的アルゴリズム (GA) の手法を用い、本実験では 20 個の対応点を選択する。SIFT によって抽出された特徴点の図の例を図 (3.6)(a) に、GA によって選択された特徴点の図の例を図 (3.6)(b) に示す。

ランダム選択された対応点からホモグラフィ行列を作成する。作成したホモグラフィ行列によって1つ目の画像を変換し、2つ目の画像との重なりを考える。平面上の物体では2つの画像を重ね合わせた際の誤差は小さくなり、平面上にないようなもの、すなわち平面から離れた立体は誤差が大きくなることを利用する。そこで各ピクセルの誤差の平均値を評価値とする。既定回数ランダムに対応点を選び、GAの手法で評価値がよい値をとるようなホモグラフィ行列に更新する。この更新の際に提案手法では、人検出結果を用いる。人の足元は平面上にあるので、人の足元の座標を抽出してその座標を対応点に選ぶようにすることで、ホモグラフィ行列作成の効率化を図ることができる。

このようにして最適化されたホモグラフィ行列を作成して、1つ目の変換された画像と2つ目の変換された画像との重ね合わせ誤差を算出する。それにより平面が誤差が小さい領域として現れ、立体が誤差の大きい領域として現れる。この情報を手がかりとしてグラフカットを用いて、取り出された平面と立体を領域分割することにより、視野画像から障害物になりうる立体の領域を除いた平面を抽出する。以上が本研究で提案する手法の流れである。以降、手法の詳細について述べる。

3.4.2 ホモグラフィ行列を用いた平面抽出

3.4.2.1 基礎行列

2つの画像を考える。1枚目の画像上の座標を (u_1, v_1) 、2枚目の画像上の座標を (u_2, v_2) 、空間の点の位置を表すワールド座標 $(X, Y, Z)^T$ としたときそれぞれの透視投影行列 $\mathbf{P}_1, \mathbf{P}_2$ を用いて、

$$\omega_i \begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix} = P_i \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (3.14)$$

のように結びつく。ここで透視投影行列 \mathbf{P} はカメラの焦点距離、画素サイズや画像中心などの内部パラメータを表す行列 \mathbf{A} とカメラの位置、姿勢を表すワールド座標 (基準座標) に対する回転行列 \mathbf{R} と平行移動ベクトル \mathbf{T} によって

$$P = A(R| - RT) \quad (3.15)$$

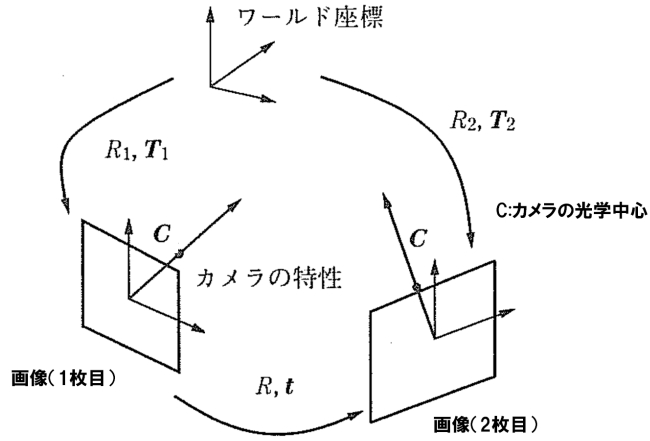


図 3.7: カメラ座標間における関係

と与えられる. したがって透視投影が既知の場合にはそれらの画像上での対応する点の位置座標の組からその点の空間での座標は式 (3.14) を $\omega_1, \omega_2, X, Y, Z$ の方程式とみて解くことで復元することができる. カメラの内部パラメータ行列を \mathbf{A} , 1 枚目の画像のカメラ座標のワールド座標に対する相対的な位置, 姿勢を $\mathbf{R}_1, \mathbf{T}_1$ とで, 2 枚目の画像の相対的な位置, 姿勢を $\mathbf{R}_2, \mathbf{T}_2$ とで表す. また, 2 枚の画像の相対的な位置関係を 1 枚目の画像のカメラ座標を基準として, 2 枚目の画像のカメラ座標への回転 \mathbf{R} と \mathbf{t} とで

$$\mathbf{R} = \mathbf{R}_1 \mathbf{R}_2^{-1} \quad (3.16)$$

$$\mathbf{t} = \mathbf{R}_1 (\mathbf{T}_2 - \mathbf{T}_1) \quad (3.17)$$

と表現できる. 空間の点の位置を $\mathbf{M} = (X, Y, Z, 1)^\top$, その像の各画像面上での位置を $\mathbf{m}_1 = (u_1, v_1, 1)^\top$,

$\mathbf{m}_2 = (u_2, v_2, 1)^\top$ とおいて式 (3.15) から式 (3.14) を書き直すと

$$\omega_1 \mathbf{m}_1 = \mathbf{A}(\mathbf{R}_1 | - \mathbf{R}_1 \mathbf{T}_1) \mathbf{M} \quad (3.18)$$

$$\omega_2 \mathbf{m}_2 = \mathbf{A}(\mathbf{R}_2 | - \mathbf{R}_2 \mathbf{T}_2) \mathbf{M} \quad (3.19)$$

これらの式から \mathbf{M} を消去して

$$\omega_1 (\mathbf{A}_1 \mathbf{R}_1)^{-1} \mathbf{m}_1 - \omega_2 (\mathbf{A}_2 \mathbf{R}_2)^{-1} \mathbf{m}_2 = \mathbf{T}_2 - \mathbf{T}_1 \quad (3.20)$$

したがって

$$\omega_1(A_1^{-1}\mathbf{m}_1) - \omega_2(RA_2^{-1}\mathbf{m}_2) = \mathbf{t} \quad (3.21)$$

これは $A_1^{-1}\mathbf{m}_1$ と $RA_2^{-1}\mathbf{m}_2$ の線形結合で, \mathbf{t} で表現されているのでこの3つのベクトルが1平面上にあることになる. でベクトル \mathbf{t} と任意のベクトル \mathbf{x} との外積演算に対して $\mathbf{t} \wedge \mathbf{x} = T\mathbf{x}$ となるような行列 T を導入する. $\mathbf{t} = (t_1, t_2, t_3)^\top$ に対し

$$T = \begin{pmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{pmatrix} \quad (3.22)$$

である. これを用いると $A_1^{-1}\mathbf{m}_1$ と $\mathbf{t} \wedge RA_2^{-1}\mathbf{m}_2$ が垂直であるからこの2つのベクトルの内積を使って

$$\mathbf{m}_1^\top (A_1^{-1}) T R A_2^{-1} \mathbf{m}_2 = 0 \quad (3.23)$$

そこで

$$\mathbf{F} = (A_1^{-1}) T R A_2^{-1} \mathbf{m}_2 \quad (3.24)$$

とおくと, この行列 \mathbf{F} は2枚の画像の結像特性と相対的な位置関係で決定され, 2つの画像上のすべての対応点 $\mathbf{m}_1, \mathbf{m}_2$ の間にはこの行列を介して

$$\mathbf{m}_1^\top \mathbf{F} \mathbf{m}_2 = 0 \quad (3.25)$$

という関係があることを意味する. 行列 \mathbf{F} を基礎行列 [31] といい, この方程式を基礎方程式という. 式 (3.25) は二つの画像上の空間の同じ点を見ている全ての対応点において成立する. したがって $\mathbf{A}, \mathbf{R}, \mathbf{t}$ が未知でも2つの画像間において多数の対応点の組が与えられれば, 逆に行列 \mathbf{F} を求めることができる. 基礎行列 \mathbf{F} は 3×3 で9個の成分をもつが, 定数倍の自由度があるのでどれか1つの成分を定数に固定することで未知数は8個であるとみなせる. つまり, 理論的には2つの画像で8組以上の点の対応が分かれば式 (3.25) に代入することで \mathbf{F} の成分を未知数とする方程式が8個でき \mathbf{F} を決定できる.

3.4.2.2 ホモグラフィ行列

3次元空間中の点の2つの画像間には, 基礎方程式を満たすという性質があり, 空間中の対象点がすべてある平面上にあるということが既知であるならばそれらの点の画像上での対

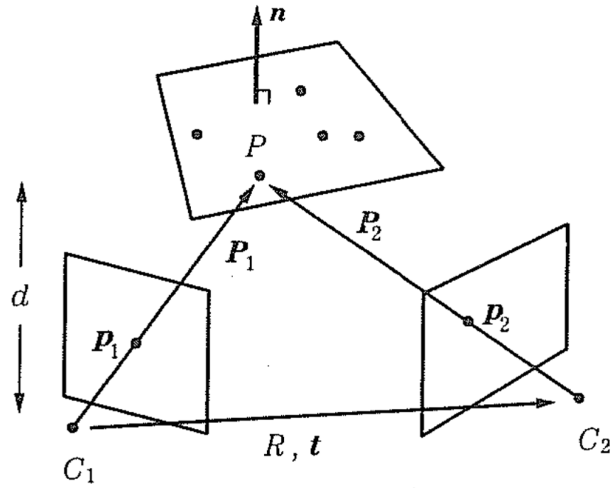


図 3.8: カメラ座標間における関係

応にはホモグラフィ行列 [31] を通して拘束がある. つまり, 全ての点の乗っている平面に対応するホモグラフィ行列が与えられると, 画像間の対応点の位置が完全にわかる. 図 (3.8) に示すように対象点は全て 1 平面上に乗っていてその画像を得ているとする. カメラ C_1 のカメラ座標を基準とする. この平面のカメラ C_1 のカメラ座標での法線ベクトルを \mathbf{n} としてカメラ中心 C_1 からこの平面に下ろした垂線の長さを d とする. この平面上の点 P をさすカメラ C_1 のカメラ座標での位置ベクトルを \mathbf{P}_1 とすると

$$\mathbf{n}^\top \mathbf{P}_1 = d \quad (3.26)$$

C_1 から見た第二のカメラ C_2 のカメラ座標への回転を \mathbf{R} , 平行移動を \mathbf{t} とすると, 点 P をさすカメラ C_2 のカメラ座標での位置ベクトル \mathbf{P}_2 は $\mathbf{P}_2 = \mathbf{R}\mathbf{P}_1 + \mathbf{t}$ であるので式 (3.26) から次式が成り立つ.

$$\mathbf{P}_2 = \left(\mathbf{R} + \frac{\mathbf{t}\mathbf{n}^\top}{d} \right) \mathbf{P}_1 \quad (3.27)$$

このそれぞれの点の画像座標を $(u_1, v_1), (u_2, v_2)$ とすると

$$\alpha \begin{pmatrix} u_2 \\ v_2 \\ 1 \end{pmatrix} = H \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix} \quad (3.28)$$

ここで

$$H \equiv A \left(\mathbf{R} + \frac{\mathbf{t}\mathbf{n}^\top}{d} \right) A^{-1} \quad (3.29)$$

は 3×3 行列でホモグラフィ行列と呼ばれる。この平面上に存在する点のそれぞれの画像への到底点の画像座標の組は全て式 (3.28) で表せる。つまり平面上にある点は、その平面のホモグラフィ行列が分かっていたら一方の画像上の点からもう一方の画像上の対応点の位置を決定できる。

3.4.2.3 ホモグラフィ行列の決定

3次元空間内にある平面上の n 個の点について二つの画像間で対応点が与えられているとする。画像 1 における i 番目の点を (u_{1i}, v_{1i}) 、画像 2 における i 番目の点を (u_{2i}, v_{2i}) とする。

この i 番目の点についての式 (3.28) の定数 α を α_i とおくと

$$\alpha_i \begin{pmatrix} u_{2i} \\ v_{2i} \\ 1 \end{pmatrix} = H \begin{pmatrix} u_{1i} \\ v_{1i} \\ 1 \end{pmatrix} \quad (3.30)$$

は H の (i, j) 成分を H_{ij} と表し、 $\mathbf{h} = (\mathbf{H}_{11}, \mathbf{H}_{12}, \mathbf{H}_{13}, \mathbf{H}_{21}, \mathbf{H}_{22}, \mathbf{H}_{23}, \mathbf{H}_{31}, \mathbf{H}_{32}, \mathbf{H}_{33})^\top$ とすることで以下のように書くことができる。

$$\begin{pmatrix} u_{1i} & v_{1i} & 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 & -u_{2i} \\ 0 & \cdots & 0 & u_{1i} & v_{1i} & 1 & 0 & \cdots & 0 & -v_{2i} \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & u_{1i} & v_{1i} & 1 & -1 \end{pmatrix} \begin{pmatrix} \mathbf{h} \\ \alpha_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (3.31)$$

左辺の行列の 3×9 の部分を U_i 、残りの 1 列を $-\mathbf{u}_{2i}$ で表すと、 n 個の対応点が取れている場合、 \mathbf{h} は次の式を満たす。

$$\begin{pmatrix} U_1 & -\mathbf{u}_{21} & \mathbf{0}^\top & \cdots & \cdots & \cdots & \mathbf{0}^\top \\ U_2 & \mathbf{0}^\top & -\mathbf{u}_{22} & \mathbf{0}^\top & \cdots & \cdots & \mathbf{0}^\top \\ \vdots & \vdots & \ddots & \ddots & \ddots & & \vdots \\ U_i & \mathbf{0}^\top & \cdots & \mathbf{0}^\top & -\mathbf{u}_{2i} & & \\ \vdots & \vdots & & & \ddots & \ddots & \mathbf{0}^\top \\ U_n & \mathbf{0}^\top & \cdots & \cdots & \cdots & \mathbf{0}^\top & -\mathbf{u}_{2n} \end{pmatrix} \begin{pmatrix} \mathbf{h} \\ \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \mathbf{0} \quad (3.32)$$

この式を解き \mathbf{h} を求めることでホモグラフィ行列 \mathbf{H} を得ることができる。

3.4.3 ホモグラフィ行列の最適化

3.4.3.1 Genetic Algorithm

本節では本論文の提案手法で用いた Genetic Algorithm[32][33] について述べる。

Genetic Algorithm(GA) は最適化問題の解を個体の染色体としてコード化し、遺伝子をもつ仮想的な生物を計算機内に設定する。それらの生物の世代交代シミュレーションによって生物を進化させ、その中で、適者生存に基づく淘汰や増殖を繰り返す。それによって実用解を得る探索手法である。図 (3.9) に GA の処理の流れを示す。初めにランダムな染色体をもつ初期個体群を発生させ、各個体の適応度を求める。個体の適応度とは、通常工学的課題における評価値となるようにコーディングを行う。

次に、適応度が高い個体ほど生き残る確率が高くなるような自然淘汰を実行して次世代の個体群を決める。続いて、ランダムに選んだ2つの個体の染色体をある部分で切断して交換する交叉という作業を行う。最後に突然変異を各個体の染色体に対して非常に低い確率で生じさせることで次世代の個体集団を得る。これは、ある確率で染色体の一部の値を変える操作で、もともと交叉だけでは、個体の親に依存するような限られた範囲の子しか生成できない。突然変異は、染色体上のある遺伝子を一定の突然変異率で、他の対立遺伝子に置き換えることにより、交叉だけでは生成できない子を生成する。その突然変異により、個体群の多様性を維持することが可能となる。これらの操作を実用解が得られるまで繰り返す。

本論文の実験においては、2枚の画像の対応点の座標データが染色体となる。SIFTを用いて生成された対応点の全座標データから対応点の座標をランダムに20組選択する。20組の対応点の座標を持つ個体を10個用意する。この10個の対応点の座標データが初期個体集団となる。評価値については、2枚の画像の重ね合わせ誤差を評価値とする。実験で用いた評価値の式は評価関数を $f(x)$ として、画像の画素数を p 、画素 i の輝度値を $L_{i,1}$ 枚目の画像から2枚目の画像へホモグラフィにより変換された画像の輝度値を H_{12} とすると

$$f(x) = \frac{|\sum_p L_2 - H_{12}|}{p} \quad (3.33)$$

となる。全ての画素に対して、ホモグラフィにより変換された画像と元の画像の誤差の絶対値を画素数で割ることで各画素に対しての誤差の平均値を算出する。これを評価値とする。

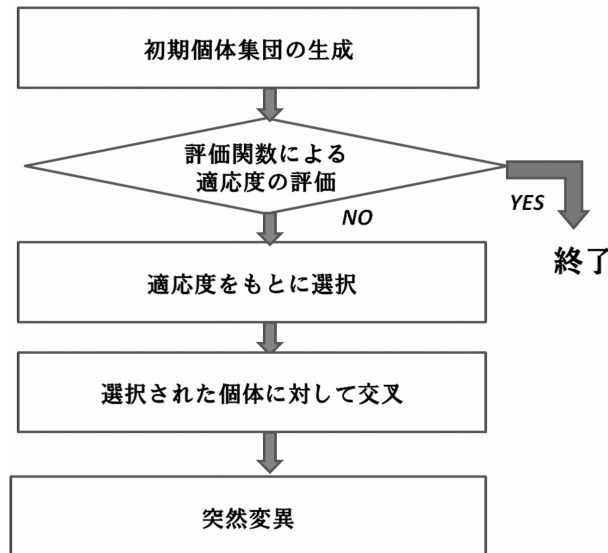


図 3.9: Genetic Algorithm の処理

3.4.3.2 人検出結果に基づいたホモグラフィ行列の最適化方法

本節では人検出結果に基づいたホモグラフィ行列の最適化の方法について述べる。まず GA における評価関数をもとに、初期個体群 10 個の中から最も適応度の高い個体 (エリート) を選択する。そのエリートの染色体に突然変異を加える。ここでいう突然変異とは、対応点座標の組であり、元々持っている 20 組の対応点の一部を全座標データから選んだものに変更し、新たな個体を作る。個体のもつ対応点座標のデータは全座標データからランダムに選ばれたものだが、ここで人検出の結果を用いる。人検出の結果から人の最下端部が足元となる。足元は平面に接しているので足元の座標は平面上の対応点として用いることができる。すなわち足元の座標を個体の染色体のひとつとして利用することで、評価値の高い個体を生成することができる。

このようにして、エリートの染色体を持つ個体を親として、次世代に複製し、遺伝子を受け継ぐような新たな個体を作成する。それを繰り返すことで適応度の高い個体を生成して初期個体群を更新する。このように初期個体群を更新することで、ホモグラフィ行列の最適化を行う。

3.4.4 領域分割を用いた平面と物体の分離

3.4.4.1 画像の領域分割

画像の領域分割とは画像中に含まれる個々の対象物に着目し、それぞれの領域が一様性の基準を満たしかつ、隣接する2つの領域の和集合が一様でないように領域をとりだす。つまり、その領域内のある種の特徴または性質が同じであるような重なりのない領域を取りだす処理のことをいう。領域の定義として以下のように大別することができる。

- メンバシップ関数やラベルなどによって指示される画素集合の連結した成分である
- 一様性条件を満たす画素の最大連結集合である。
- エッジによって囲まれる画素の連結集合

以上の3つに分類される定義に基づき、画素に基づいた画像分割法、領域に基づいた画像分割法、境界に基づいた画像分割法について述べる。

画素に基づいた画像分割法

画像分割は画像の特徴空間に基づいた解析によって行うことができる。画像内の画素に対する基本的な表現は色や輝度だけでなく、空間周波数や様々な方向をもつフィルタ集合からの応答も含まれる。画像からの特徴抽出が行われると各画素は一般に多次元の特徴空間に射影されて有効な特徴量はクラスタと呼ばれる特徴量の塊でその空間内で高い密度をもって分布する。クラスタは画像を構成する部分領域に対応する。

領域に基づいた画像分割法 領域に基づいた画像分割法とは画像の局所的な類似性に着目し、画像を構成する領域のそれぞれに対する一様性と異なる領域間の非類似性に基づく。画像分割をグラフの分割と対応付けて定式化したり確率論と組み合わせ画像をモデル化するというような分割法である。まず画像をグラフというデータ構造によって表現する。任意の特徴空間内における点集合は重みつき無向グラフとして表せる。すなわち、グラフのノード(節点)は特徴空間内の点を表しており、全てのノード対はエッジ(辺)によって結ばれている。それぞれのエッジはノード対の比類似度に対応する重みをもつ。ある適当な尺度により、類似度が同じノード集合においては高く、異なる集合間では低くなるようにノードの集合を

互いに素な集合に分割する.

境界に基づいた画像分割法

画像は領域間の境界を示すエッジによって分割される. エッジはたいていの場合において輝度や色といった低レベルの画像特徴における急激な変化として定義される. 輝度および各色成分を画像特徴とした場合空間周波数フィルタリングによる高周波成分はそのような特徴における急な変化に対応するので, ハイパスフィルタを実行することによりエッジの強調や検出を行うことができる.

3.4.4.2 グラフとその最小切断アルゴリズム

画像をピクセル単位で輝度, 視差, 領域分割などでラベリングをする際にエネルギーを最小化する問題がある. 実画像に対してラベリングを行う場合に, 画像のノイズなどの影響で不明確な部分が存在する可能性があり最適なラベリングを行うことが困難である. そこでエネルギー最小化問題に対して, グラフの最小切断アルゴリズムを用いたエネルギー最小化が近年盛んに応用されるようになってきた. 本節ではその最小切断アルゴリズム [34] について述べる.

ラベリング問題とは画像 P の各ピクセル p に対して, ラベル L_p をつけていく問題である. 図 (3.10) では, 輝度値に対してラベリングを行った結果である. このようなラベリング問題に対する解法の 1 つとしてエネルギー関数 $E(L)$ を以下のように定義し, これを最小化する方法がある.

$$E(L) = \sum_{p \in P} D_p(L_p) + \sum_{(p,q) \in N} V_{p,q}(L_p, L_q) \quad (3.34)$$

ここで N は近傍領域のピクセル, $D_p(L_p)$ は観測データに対してのペナルティ関数, $V_{p,q}(L_p, L_q)$ は近傍ピクセルとの相互関係を表した関数である.

定義したエネルギー関数 E に基づいてグラフを作成し, グラフの最小切断により最適解を求める. 最小切断アルゴリズムとは図 (3.11) のようなネットワークを想定した場合にどの程度の量を最大流することができるかを求める手法である. 重み付き有向グラフを $G = (E, V)$ と定義する. このとき, V はノード (node), E はエッジ (edge) である. このグラフをコンピュータビジョンの問題で使用する場合はピクセルがノードとなる. またピクセル以外のノードとして, ソース (source) $s \in V$ とシンク (sink) $t \in V$ と呼ばれる特殊なターミナル

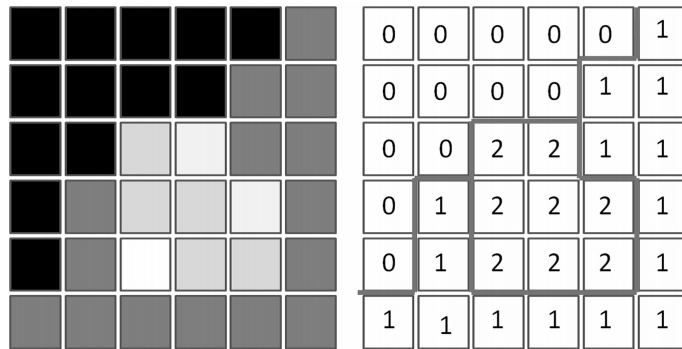


図 3.10: ラベリング問題

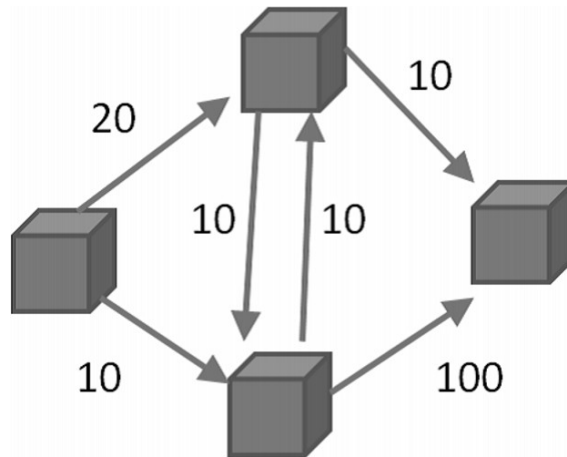


図 3.11: ネットワーク

(ラベル) を追加する. エッジはノード間の関係を表現しており, 周辺のピクセルとの関係を表したものを n-link, 各ピクセルと s, t との関係を表したものを t-link と呼ぶ. n-link のコストは, 周辺ピクセルとの連続性を表現したペナルティ関数により決定され, t-link のコストは各ピクセルがそのラベルである確率を表したペナルティ関数により決定される. n-link と t-link は, 式 (3.34) の $V_{p,q}$ と D_p に相当する. 作成されるグラフを $s \in S, t \in T = V - S$ となる 2 つのグラフへの分割を考える. 分割する際に切断されるエッジを $s-t$ カットと呼ぶ. その切断されるカットのうち S から T へ接続されているエッジのコストの総和を $s-t$ カットの容量という. この $s-t$ カットの容量が最小となる $s-t$ カット (最小カット) を求めたい. この最小カットは, 最大フロー最小カットの定理を用いることによって, グラフの最大フローから求めることが可能である. 最大フローを求める手法の代表的なものとして, Ford-Fulkerson Method と Push-Relabel Method がある. Ford-Fulkerson Method は

フローの増分路が存在しないとき、そのフローは最大であるという考えに基づいている。はじめに、フローを0として s から t への増分路を探索し、フローを増加させる。この作業を増分路がなくなるまで繰り返すことにより、最大フローを求める。Push-Relabel Method では、 s から近傍のノードへ可能な限りフローを流し、それをすべてのノードに対して繰り返すことにより最大フローを求める。

3.4.4.3 グラフカットを用いた領域分割

グラフカットアルゴリズム [35] を用いて領域分割を行う。領域分割における物体と背景のラベル L に対応付けを行う問題の解法の一つとして以下のように定義するエネルギー関数 $E(L)$ の最小値を求める方法がある。

$$E(L) = \lambda \cdot R(L) + B(L) \quad (3.35)$$

$$R(L) = \sum_{p \in P} R_p(L_p) \quad (3.36)$$

$$B(L) = \sum_{(p,q) \in N} B_{p,q} \cdot \delta(L_p, L_q) \quad (3.37)$$

$$\delta(L_p, L_q) = \begin{cases} 1 & \text{if } L_p \neq L_q \\ 0 & \text{otherwise} \end{cases} \quad (3.38)$$

ここで $R(L)$ は領域情報、 $B(L)$ は境界情報、 λ は $R(L)$ のパラメータ係数である。このエネルギー関数 $E(L)$ に基づいてグラフのエッジのコストを計算する。このとき作成するグラフは以下のように定義する。

$$R_p(\text{"object"}) = -\ln Pr(I_p|O) \quad (3.39)$$

$$R_p(\text{"background"}) = -\ln(I_p|B) \quad (3.40)$$

$$B_{p,q} \propto \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) \cdot \frac{1}{d(p,q)} \quad (3.41)$$

$$K = 1 + \max_{p \in P} \sum_{(p,q) \in N} B_{p,q} \quad (3.42)$$

ここで O, B はそれぞれ物体と背景のシードを表し、画像 P 内の注目ピクセルと近接ピクセルの輝度値をそれぞれ I_p, I_q として $d(p, q)$ をピクセル間の距離とする。そして正解ラベルをもとに物体と背景の領域分割を行う。

表 3.1: 撮影環境

カメラ	EXILIM CARD EX-S10
解像度	3648 × 2736
焦点距離	6.3mm

表 3.2: 計算環境

CPU	Pentium4 プロセッサ
Memory	2GB
OS	WindowsXP
Programing Language	C++

3.5 実験

本節では、提案手法を実画像に対して適用し、その有効性について検証を行う。以降、3.5.1 節で実験環境を、3.5.2 節で実験結果を述べる。

3.5.1 実験環境

本実験において、視野画像の撮影は市販のデジタルカメラを用いて行った。撮影環境を表 (3.1) に、計算環境を表 (3.2) に示す。また、提案手法における遺伝的アルゴリズムのパラメータは予備実験により精度と実行時間のトレードオフを考慮して決定し、遺伝子数を 20、世代数を 100 とした。各実験において、実行時間は 1500ms~2000ms であった。

3.5.2 結果と考察

図 (3.12), 図 (3.14), 図 (3.16) に撮影した入力画像列を、図 (3.13), 図 (3.15), 図 (3.13) に推定された平面領域の画像をそれぞれ示す。

実験結果 1 では、図 (3.13) をみると、人と右の木については立体であると判定されていたが、左の木に関しては、木の一部分しか立体と判別することができていなかった。実験結果 2 では人と前方にある壁を立体と判別することができていたが、横断歩道の縞模様の一部

分で平面と判別することができていなかった。これは、重ね合わせた際の少しのずれで、誤差の値が大きくなるものもあることや、撮影した画像における物体の模様などによっては、特徴点の対応がとれず誤判定が生じていると考えられる。実験結果3においては、電柱に関しては立体と判別できていなかった。これは電柱において対応点が取れなかったことが原因として考えられる。このような結果から、次のようなことがわかる。

- 人が歩行可能な道路平面領域を大まかに推定することができた
- 実際のシーンにおいて、物体・歩行面の領域分割は一部しか行えていなかった

以上のような問題に対して、

- 立体と物体のセグメンテーションにおける精度の向上のため、立体らしさ、平面らしさを推定する方法について検討する
- ホモグラフィ行列の最適化の効率化のため、対応点を選ぶ際にカメラのパラメータを利用するなどして制約をつける方法を検討する。

ということが今後の課題として挙げられる。

3.6 まとめ

本テーマでは、視覚情報フィルタリングによる弱視者の自律的移動支援システムについて、ウェアラブルカメラから得られる視覚情報からの歩行可能領域抽出法について提案しその有効性について検討した。提案法では、ユーザに装着されたカメラから得られる時系列的に連続した画像列に対して、歩行面検出のため連続するフレーム間での特徴点マッチングによる3次元解析を行った。その際、人検出結果に基づく遺伝的アルゴリズムによりマッチングした特徴点对を取捨選択することで歩行可能平面を構成する点对群を推定した。実験では、提案法の有効性を検討した。



(a) 1 枚目の画像



(b) 2 枚目の画像

図 3.12: 入力画像列 1



図 3.13: 推定された歩行可能領域 (入力画像列 1)



(a) 1 枚目の画像



(b) 2 枚目の画像

図 3.14: 入力画像列 2

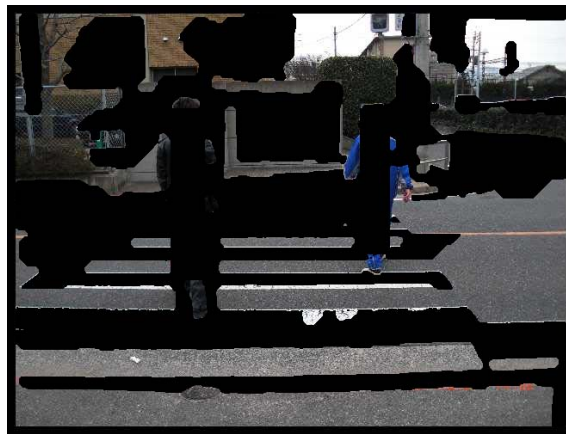


図 3.15: 推定された歩行可能領域 (入力画像列 2)



(a) 1枚目の画像



(b) 2枚目の画像

図 3.16: 入力画像列 3



図 3.17: 推定された歩行可能領域 (入力画像列 3)

第4章 結論

本論文では、弱視者のための視覚補助システムを対象とした画像認識・理解アルゴリズムの構築及びその検討を行った。これは眼鏡などの視力矯正器を用いても視力が0.3未満程度にしか矯正されない弱視者に対して、画像認識・処理システムによる視力の増強を目指したものである。本論文では主に「視覚情報変換による弱視者向け文字情報認識支援」と「視覚情報フィルタリングによる弱視者の自律的移動支援システム」について、それを実現するための画像処理・認識アルゴリズムの検討を行った。

第2章では、視覚情報変換による弱視者向け文字情報認識支援に関して、「視覚情報からの文字領域抽出」と「弱視者の視覚特性に基づく文字構造の変換」について議論した。視覚情報からの文字領域抽出では、屋外において文字情報は看板に多く含まれていることに着目し「看板領域の抽出」と「文字領域抽出」という2段階の処理による視覚情報（ウェアラブルカメラ画像）からの文字情報抽出法を提案した。さらに、文字領域の抽出においては環境光による頑健性を高めるため文字領域抽出をエネルギー最小化の問題として定式化し、ノイズの影響を受けにくい文字の2値化を実現した。実験では、提案した文字抽出法を実際に情景画像に適用したところ、看板領域と文字抽出共に良好な結果を示した。弱視者の視覚特性に基づく文字構造の変換では、弱視者の「像がぼけて見える」、「視野が狭くなる」といった個々の見え方に対して視覚情報処理を行うことでユーザに「認識しやすい」像を提供することを目的とし、デジタル文字画像を文字のフォント種に依らず弱視者が認識しやすい文字像へ変換するアルゴリズムを提案した。提案法では、事前に用意された各種文字像とその各種文字に対するぼけても認識しやすい理想的な文字像とのペアからそれらの関係性を学習し変換を行った。また、詳細な予備実験により弱視者が見やすい文字像について検討するとともに、「見やすい文字像」の条件について一定の見解を明らかにした。今後はこれらを組み合わせることで、直接ユーザに装着されたカメラから文字情報を取り出し文字構造変換を行うシステムを実現したい。

第3章では、視覚情報フィルタリングによる弱視者の自律的移動支援システムについて、

ウェアラブルカメラから得られる視覚情報からの歩行可能領域抽出法について提案しその有効性について検討した。提案法では、ユーザに装着されたカメラから得られる時系列的に連続した画像列に対して、歩行面検出のため連続するフレーム間での特徴点マッチングによる3次元解析を行った。その際、人検出結果に基づく遺伝的アルゴリズムによりマッチングした特徴点对を取捨選択することで歩行可能平面を構成する点对群を推定した。実験では、提案法の有効性を検討した。

以上のように、提案した画像認識・理解アルゴリズムによりユーザの視界情報から高度なセンシングができることを確認した。今後は、提案手法のさらなる改良や、ユーザの様々な状況に対応できる総合的なシステムの構築を行っていきたい。

謝辞

本研究を遂行するにあたり，多大なるご指導をいただいた九州工業大学大学院工学研究院 前田博 教授に心より感謝し御礼申し上げます。また，本研究に関して貴重な御助言，御鞭撻をいただいた九州工業大学大学院工学研究院 桑原伸夫 教授，芹川聖一 教授，金亨燮 教授に厚く御礼申し上げます。

研究活動における多くの場面では，九州工業大学大学院工学研究院 河野英昭 准教授をはじめ，九州工業大学大学院工学研究院 生駒哲一 准教授，垣内忍 技術職員に懇切なご指導・有益なご助言を賜り，研究の指針を示していただきました。深く感謝し御礼申し上げます。さらに，文献調査や実験，執筆などの活動へ献身的に協力していただいた九州工業大学大学院修了 野邊拓馬 氏，大浜公司 氏，九州工業大学卒業 西岡良 氏，九州工業大学大学院 富永啓太郎 氏，並びに前田博研究室，生駒哲一研究室，全ての関係者の皆様に深く感謝いたします。

また，博士後期課程の研究において，日本学術振興会 平成 24 年度 科学研究費補助金（特別研究員奨励費）12J09839 のご支援をいただきました。感謝いたします。

最後に，研究生活を行う上で精神面及び経済面で私を支えてくれた両親や家族，そしてすばらしい友人に深く感謝いたします。

参考文献

- [1] K. Ouchi, T. Suzuki, M. Doi, “LifeMinder: a wearable healthcare support system using user’s context”, in Proc. of the 2nd International Workshop on Smart Appliances and Wearable Computing (IWSAWC 2002), pp.791-792, 2002.
- [2] R. Tenmoku, Y. Nakazato, A. Anabuki, M. Kanbara and N. Yokoya, “Nara Palace Site Navigator: Device-independent Human Navigation using a Networked Shared Database”, in Proc. of 10th International Conference on Virtual Systems and Multimedia (VSMM 2004), pp.1234-1242, 2004.
- [3] T. Okumura, M. Kouroggi, N. Sakata and T. Kurata, “A Pilot User Study on 3-D Museum Guide with Route Recommendation Using a Sustainable Positioning System”, in Proc. of International Conference on Control, Automation and Systems 2007 (ICCAS 2007), pp.749-753, 2007.
- [4] R.C. Bryant, C.M. Lee, R.A. Burstein, and E.J. Seibel, “Engineering a Low-cost Wearable Low Vision aid Based on Retinal Light Scanning”, In Proc. SID 2004, pp. 23-28, 2004.
- [5] Alvaro Cassinelli, Carson Reynolds, and Masatoshi Ishikawa, “Augmenting spatial awareness with Haptic Radar”, 10th IEEE International Conference on Wearable Computers (ISWC 2006), pp.100-107, 2006.
- [6] 久野 素有, 山下 淳, 金子 透, “ステレオカメラを用いたタッチパネル操作支援システムの構築”, IEEJ Transactions on Industry Applications 131(4), pp.458-465, 2011.
- [7] キャリアブレイン 「医療介護 CB ニュース」
<http://www.cabrain.net/news/article/newsId/24377.html>, (2012.2.10 アクセス) .

- [8] 国土交通省, “旅客施設における弱視者等に配慮した施設・設備に関する調査検討報告書”, 2006.
- [9] 丸茂, 山本, 加藤, “視覚障害者支援モバイルシステムの提案-視覚と操作の統合-”, 映像技報, Vol.31, No.39, pp.15-18, 2007.
- [10] 田中, 後藤, “視覚障害者用のウェアラブルな文字認識デバイス”, 情処研報, Vol.2008, No.115, pp.125-130, 2008.
- [11] T. Sakamaki, T. Yamaguchi, K. Shimamura, K. “A Magnification Control for the Vision Support Function Based on the Magnifying Glass Metaphor”, Int. Conf. Next Era Information NEtworking (NEINE’06) NEINE06. pp.99-102, 2006.
- [12] ウェアビジョン URL: <http://wearvision.co.jp/index.html>.
- [13] K. Wang, J. A.Kangas, “Character location in scene images from digital camera”, Pattern Recognition, 23, pp.2287-2299, 2003.
- [14] 平山勝祐, 大町真一郎, 阿曾弘具, “カラー情報と明度情報を利用した情景画像中の文字列の高精度抽出”, 電子情報通信学会論文誌 D, Vol.J89-D, No.4, pp.893-896, 2006.
- [15] Hon-Son Don, “A noise attribute thresholding method for document image binarization”, International Journal on Document Analysis and Recognition, Vol.4, pp.131–138, 2001.
- [16] 馬場口登, 塚本正敏, 相原恒博, “手書き日本文字列からの文字切り出しの基礎的考察”, 電子情報通信学会論文誌 D, Vol.J68-D, No.12, pp.2123-2131, 1985.
- [17] Y. Y. Boykov, and M.-P. Jolly, “Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. Computer Vision”, IEEE International Conference on, 1:105, 2001.
- [18] 八木康史, “コンピュータビジョン最先端ガイド1, アドコム・メディア株式会社, 2008.
- [19] 田村秀行, “コンピュータ画像処理,” オーム社, 2002.

- [20] H. Fujiyoshi, T. Komura, I. Eguchi, K. Kayama, “Road Observation and Information Providing System for Supporting Mobility of Pedestrian”, ICVS2006 : IEEE International Conference on Computer Vision Systems, p.37, 2006.
- [21] 青木, 内海, 木村, “単一カメラによる歩行者用障害物検知システム”, 信学技報, IE, 画像工学, Vol.110, No.148, pp. 53-58, 2010.
- [22] D. G. Lowe, “Object recognition from local scaleinvariant features”, Proc. of IEEE International Conference on Computer Vision (ICCV), pp.1150-1157, 1999.
- [23] D. G. Lowe, “Distinctive image features from scaleinvariant keypoints”, Proc. of International Journal of Computer Vision (IJCV), 60(2), pp. 91-110, 2004.
- [24] 藤吉弘亘. ”Gradient ベースの特徴抽出 - SIFT と HOG - ”, 情報処理学会 研究報告 CVIM 160, pp. 211-224, 2007.
- [25] 都築勇司, 藤吉弘亘, 金出武雄, “SIFT 特徴量に基づく Mean-Shift 探索による特徴点追跡”, 情報処理学会 研究報告 CVIM 157, 2007 .
- [26] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, “SURF: Speeded Up Robust Features”, Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346–359, 2008.
- [27] 上野智史, 橋本真幸, 米山暁夫, “建物画像検索のための特徴点選択手法の一検討”, 情報科学技術フォーラム講演論文集, 8(3), pp.221-222, 2009.
- [28] N. Dalal, B. Triggs, “Histograms of Oriented Gradients for Human Detection”, Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.886-893, 2005.
- [29] 山内悠嗣, 藤吉弘亘, Hwang Bon-Woo, 金出武雄, “アピアランスと時空間特徴の共起に基づく人検出”, 第 10 回画像の認識・理解シンポジウム (MIRU2007), pp.1492-1497, 2007.
- [30] 安藤寛哲, 藤吉弘亘, “人検出結果に基づく自己カメラキャリブレーションと 3 次元位置推定”, 電気学会論文誌. D, 産業応用部門誌, 131(4), pp.482-489, 2011.

- [31] 出口光一郎, “ロボットビジョンの基礎”, コロナ社, pp.115-124, 2000.
- [32] 石田迫仁, 長尾智晴, “GA による画像処理フィルタの自動合成”, 電子情報通信学会技術研究報告. HCS, 97(262), pp.29-36, 1997.
- [33] 長尾智晴, “遺伝的アルゴリズムとその画像への応用”, テレビジョン学会誌, 1992.
- [34] 八木康史, “コンピュータビジョン最先端ガイド2”, アドコム・メディア株式会社, 2010.
- [35] 永橋知行, 藤吉弘亘, 金出武雄, “平滑化処理の繰り返しによる画像セグメンテーションのためのグラフカット”, 第10回画像の認識・理解シンポジウム (MIRU2007), pp.241-248, 2007.
- [36] 須田経宇, 生井浩, “新眼科学第2版”, 医学書院, pp.51-58, 1970.
- [37] 交通エコロジー・モビリティ財団 “色覚障害者, 弱視 (ロービジョン) 者に対応したサイン環境整備に係る調査研究”, 2008.
- [38] R. E. Schapire and Y. Singer, “Improved Boosting Algorithms Using Confidence-rated Predictions”, Machine Learning, No. 37, pp. 297-336, 1999.
- [39] J. Friedman, T. Hastie, and R. Tibshirani, “Additive logistic regression: a statistical view of boosting”, The Annals of Statistics, 28(2):337-374, 2000.
- [40] 安居院猛, 長尾智晴, “画像の処理と認識”, 昭晃堂, 1992.
- [41] 芦田和毅, 永井弘樹, 岡本正行, 宮尾秀俊, 山本博章, “情景画像からの文字抽出”, 電子情報通信学会論文誌 D-II, Vol.J88-D-II, No.9, pp.1817-1824, 2005.
- [42] 大津展之, “判別および最小2乗規準に基づく自動しきい値選定法”, 信学論, Vol.J63-D, No.4, pp.349-356, 1980.
- [43] 金谷健一, “これならわかる応用数学教室 最小二乗法からウェーブレットまで”, 共立出版, 2003.
- [44] “障害者白書 平成22年度版”, 内閣府, 2010.
- [45] 永田靖, 棟近雅彦, “多変量解析法入門”, サイエンス社, 2001.
- [46] 佐藤淳, “コンピュータビジョン—視覚の幾何学—”, コロナ社, 1999.