

光沢を考慮した点字画像の2値化アルゴリズム

岡村 宙輝† 伊藤 祥一‡

長野工業高等専門学校 電子情報工学科†‡

1 はじめに

点字とは指先の触覚を利用して読み取ることができる文字であり、視覚障害者が情報を得るための手段の1つである。読売新聞の記事によると、バリアフリー化が進む現代において身近な場面で点字を目にすることが多くなっている一方で、間違っただ点字も増えてきてしまっているという¹⁾。このような間違いを点字が読めない盲人も直すことができるように、先行研究では盲人向けの点字翻訳アプリケーションの開発が行われた²⁾。このアプリケーションでは、点字を撮影した画像から点字を読み取って翻訳を行う。画像に対して二値化処理を行っており、画像に点字よりも明るい部分がある場合（白飛びしている場合）や複数の光源から光が当たっている場合などに、精度よく点字の凸点を抽出することができないという課題があった。本研究ではこれらの課題のうち、白飛びしている場合の画像に対して点字の凸点を抽出できることを目的として二値化処理アルゴリズムの改良を行う。

2 二値化処理

二値化処理とはグレースケール画像の画素の階調値と閾値を比較し、すべての画素を白か黒にする処理のことである。この処理を行うことで点字の凸点とそうでない部分をはっきりさせることができる。閾値の値によっては白一色または黒一色の画像になってしまうため、閾値の算出には注意が必要である。

2.1 閾値の算出

一般的に使われる閾値の算出方法として大津の手法がある³⁾。大津の手法では閾値 t を仮に決めて、 t を基準にすべての画素を2つのグループに分ける。次にグループ内の分散とグループ間の分散を求め、これらの値から2つのグループの分離度を求める。分離度が最も大きくなる t を最終的な閾値とする。白飛びしている領域は点字の凸点と同じような階調値を持つ画素が多く、ヒストグラムでは同じグループになってしまうため、大津の手法で閾値を決めて二値化を行うと白飛びしている領域も黒く残ってしまう。

(a) カメラで撮影された画像



(b) 大津の手法



(c) 階調値の最頻値と最高値との中間値



図 1: 二値化処理の例

これを受けて先行研究では階調値の最頻値と最高値との中間値を閾値としている²⁾。このように閾値を決めると、点字の凸点と同じような階調値が多い場合でもある程度は凸点のみを抽出できる。一方で白飛びしている画像（以下では8ビットスケール画像を考える）では階調値の最頻値が255ということがあり得るので、この場合は閾値が255になり二値化したあとに白一色の画像になってしまう。大津の手法と先行研究のアルゴリズムで閾値を算出した二値化処理の例を図1に示す。二値化した画像は点線で囲ってある。白飛びしている画像は階調値の最頻値が255に近い値になるので、先行研究の方法で閾値を決めて二値化するとほとんど白一色の画像になる。

2.2 コントラスト調整

先行研究では式(1)のシグモイド関数 $f(x)$ ：調整後の階調値、 x ：元の階調値、 a ：パラメータ）を使ってコントラスト調整を行ってから、二値化している²⁾。コントラスト調整を行うと階調値の幅を狭めることができ、階調値が0か255の領域が増える。

$$f(x) = \frac{255}{1 + e^{-a\left(\frac{x-128}{255}\right)}} \quad (1)$$

3 提案手法

白飛びしている画像でも点字の凸点を抽出できるように、コントラスト調整を行ってから二値化を行うまでに、白飛びしている領域を消す処理を加えることを考える。そこで本研究では先述した大津の手法やコントラスト調整を利用して、画像の白飛びしている領域をそれ以外の領域と同じような色に塗りつぶすようにした。具体的な手順を次に示す。

Binarization Algorithm for Braille Image with Lusters
 † Okamura Hiroki: National Institute of Technology,
 Nagano College
 ‡ Shoichi Ito: National Institute of Technology, Nagano
 College

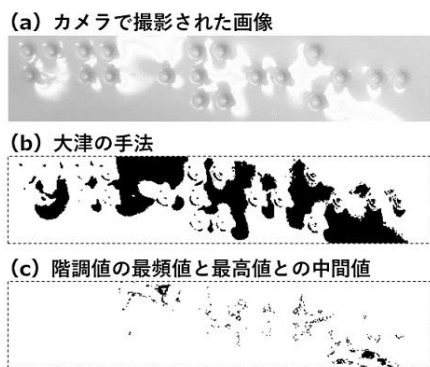


図 2: 提案手法による処理結果

1. コントラスト調整を行う
2. 大津の手法で二値化を行う
3. 輪郭を検出して、輪郭ごとにその内部領域の面積を求める
4. 面積が大きい領域を白飛びしている領域として塗りつぶす

まずは白飛びしている領域とそうでない領域との境目をわかりやすくするために、手動でコントラスト調整を行う。次に大津の手法を用いて二値化を行うことで、白飛びしている領域が残りやすいようにする。手順 3 では二値化した画像に対して、OpenCV の関数 `cv::findContours()` を使って輪郭の検出を行う。この関数を実行すると、画像に含まれる輪郭 1 つ 1 つに対して包含関係と座標が得られる³⁾。輪郭の包含関係には入れ子になった輪郭に対してどの輪郭が内側、外側であるかという情報や、一番外側にある輪郭がどれかといった情報が含まれている。次に輪郭の座標の値から輪郭の面積を求め、画像の大きさの 5% 以上の大きさがあったものを白飛びしている領域とする。このときに白飛びしている領域の中に点字がある場合があるため、輪郭の包含関係を参照して点字の部分だけ取り除くようにする。最後に白飛びしている領域を、その領域以外の階調値の平均値に塗りつぶす。

4 提案手法の処理結果

図 1 のカメラで撮影された画像に対して、提案手法の手順 1 から 4 の処理を行ってから二値化した結果を図 2 に示す。なお、手順 1 のコントラスト調整を行うときのパラメータ a は 9 とした。手順 1 から 4 の処理を行った画像に注目すると、点字の凸点の中心というよりも凸点の影が目立つようになっている。これはカメラで撮影された画像が白飛びしている場合はそれほど強い光が当たっているということであり、凸点の影もより濃くなっているためだと考えられる。よって手順 1 から 4 の処理を行ったあとに二値化を行うときは、階調値の低い画素が黒く残るようにしている。また、閾値は大津の手法を用いて算出している。

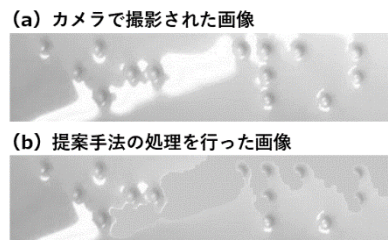


図 3: 白飛びしている領域が小さい場合

5 評価と今後の展望

図 1(a) に対して、先行研究と同じように二値化処理を行うと白一色の画像が出力として得られた。このとき、コントラスト調整のパラメータ a は 10 とした。出力が白一色の画像になった原因として白飛びしている画像は階調値が全体的に高く、階調値の最頻値と最高値との中間値を閾値とすると、ほとんどの画素の階調値が閾値を下回ってしまうということが考えられる。ただ提案手法では白飛びしている領域が画像の大きさの 5% よりも小さい場合は、その領域を塗りつぶすことができないという問題がある。例えば図 3 のカメラで撮影された画像には左下に白飛びしている小さな領域があるが、提案手法の処理を行ったあとでもその領域が白く残っている。よって白飛びしている領域が画像の大きさの 5% よりも大きい場合は、提案手法によって二値化処理を行うことで点字の凸点のみを抽出しやすくなったことがわかる。

提案手法の改善すべき点として、白飛びしている領域と判断した領域に対して塗りつぶしを行っているため、その領域に点字の凸点またはその一部が含まれていた場合にその部分の情報が失われてしまうことがある。この問題を改善するためには階調値が高い画素に対してのみに適用できるような処理を考える必要がある。また、白飛びしている領域を領域の面積によって判断すると前述したような問題が起こるため、改善する必要がある。この手法で正しく点字の凸点を抽出できるようになれば、金属などの光沢がある素材に点字が印字されている場合も点字の検出が行えるようになる。

参考文献

- [1] 点字案内 目立つミス：読売新聞 1999 年 8 月 6 日 (金) 発行
- [2] Shoichi Ito and Yoshinori Fujisawa: Tender – Smartphone Application for Braille Reading, th ACIS International Conference on Applied Computing & Information Technology (ACIT2019) proceedings, pp.69-74, 2019.
- [3] 詳解 OpenCV3 – コンピュータビジョンライブラリを使った画像処理・認識：Adrian Kaehler・Gary Bradski (著), 松田晃一・小沼千絵・永田雅人・花形理 (訳) 株式会社オライリー・ジャパン, 2018.