

囲碁における盤面識別システムの検討

橋本 燎 (指導教員 井上 優良)

令和 3 年 1 月 19 日

1. 緒言

囲碁は、2 人のプレイヤーが碁石を碁盤へ配置するボードゲームである。碁石は各プレイヤーが使用する白黒の石であり、碁盤は通常 19×19 の格子状の盤面のことを指す。このゲームは両プレイヤーが碁石を交互に碁盤上に設置し、碁石で囲まれた陣地を多く確保したプレイヤーが勝利する。相手の陣地として成立している領域でなければどこにでも碁石を置くことができる、相手プレイヤーの碁石を陣地内に収めることでその碁石を奪い使用することで相手陣地の縮小を行うことができる等、ルール上の制約が極めて少ないといった特徴を持つため、他のボードゲームと比較すると可能な総局面数は膨大になる。

近年では、2015 年に Google DeepMind 社が開発した、ディープラーニングを実装した囲碁プログラム *AlphaGo*[1] が初めてプロ棋士に対しハンデ無しでの対局で勝利した。コンピュータが人間に打ち勝つのは難しいとされていた分野で勝利を果たしたことは、人工知能の有用性を広く知らしめるものとなった。以降にも様々な人工知能が登場しプロ棋士との対局が行われてきたが、これらの対局の様子を見ると、プロ棋士の置いた石の配置をコンピュータに入力する役割を持った人間が存在していた。そこで、人の手による記録作業を行わずに対局の状況を記録できるシステムの構築を目指す。

本研究では、対局中の囲碁の画像から碁石の配置を識別するシステムの検討を行う。具体的には、盤面を含む画像から射影変換を用いて盤面を切り抜き、ノイズ処理を適用して盤面上の線を曖昧なものにした後に、輝度値をもとに碁石の配置を識別するシステムの構築を行う。その後、複数の画像に対してシステムを試し、結果を考察する。

2. 理論

2.1 変換

2.1.1 同次座標

座標 (x, y) に対し、その要素を 1 つ増やした座標 (ξ_1, ξ_2, ξ_3) を、以下の関係式を満たすように定義する。

$$\begin{aligned} x &= \frac{\xi_1}{\xi_3} \\ y &= \frac{\xi_2}{\xi_3} \end{aligned} \quad (1)$$

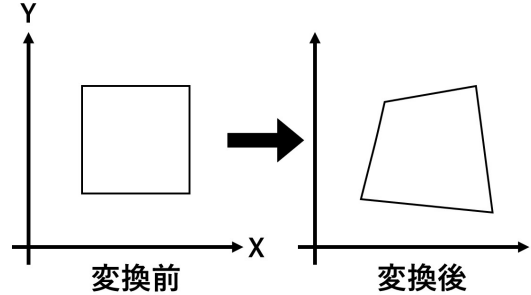


図 2.1 射影変換の模式図

ただし、 ξ_1, ξ_2, ξ_3 のうち少なくとも 1 つは 0 ではないとする。このように定義される座標を同次座標と呼ぶ [2]。

同次座標においては $\lambda \neq 0$ なる任意の λ に対して、 (ξ_1, ξ_2, ξ_3) と $(\lambda\xi_1, \lambda\xi_2, \lambda\xi_3)$ は通常の座標に直したときともに $(\xi_1/\xi_3, \xi_2/\xi_3)$ となるため、同じ点を表している。つまり、同次座標による表現では、定数倍をしても変わらないとみなすことができる。このような表現を同値であるとして、これを式では以下のように表す。

$$\begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix} \sim \begin{pmatrix} \lambda\xi_1 \\ \lambda\xi_2 \\ \lambda\xi_3 \end{pmatrix} \quad (2)$$

ここで記号 \sim は同値関係を表し、定数倍の違いを許して等しいことを意味する。

2.1.2 射影変換

同次座標を利用することにより、一般的な変換を表現することができる。これは以下の式で表現されるもので、射影変換と呼ばれている [2]。

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \sim \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (3)$$

射影変換においては、線分の直線性は保たれるものの、平行性は失われる。別の言い方をすると、任意の四角形を別の任意の四角形に移すような変換であるといえる。射影変換の模式図を図 2.1 に示す。

3. システム内容

システムは、与えられた画像に対して以下の前処理を行う。

- 射影変換（盤面部分の切り抜き）

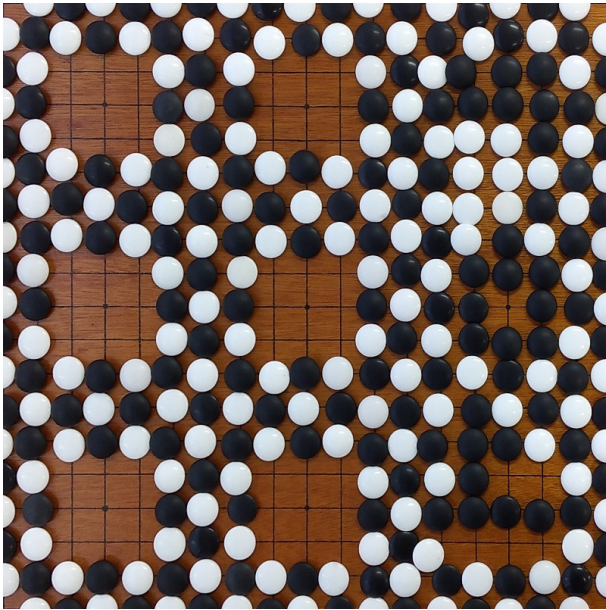


図 4.1 事例 2 の盤面

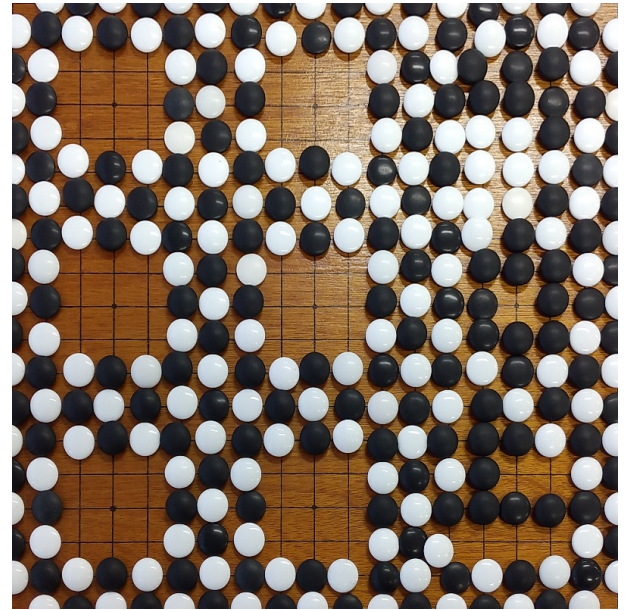


図 4.4 事例 3 の盤面

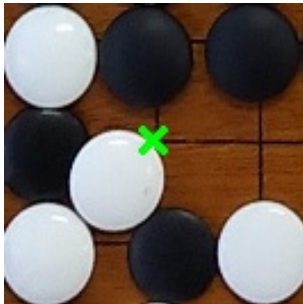


図 4.2 事例 2 の誤り部分

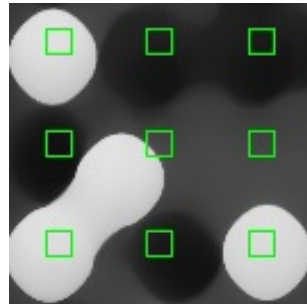


図 4.3 誤り部分の領域を可視化した図

- グレースケール化
- ノイズ処理

前処理が完了した画像に対して、システムは 19×19 個の小さな領域を付与する。この領域内に存在する画素値の平均と、設定したしきい値をもとに、システムは各領域の位置に置かれている石の種類の識別を行う。

4. 実験

システムの有効性を検証するため、実際に盤面を含む画像を使ってシステムを適用し結果を考察する。

4.1 事例 2

この盤面画像（図 4.1）に対し、システムを適用した。識別結果に誤りがあった部分にマークを付け、その周囲をトリミングしたものを図 4.2 に示す。

図 4.2 と同様に、輝度値の平均を取得する領域を可視化すると、中央の石の大部分が領域からはみ出ているのが分かる。このことから、石が正しく盤面の交点上に配置されていない場合、システムは正常に石を識別できないと考える。図 4.2 の範囲に対し、領域を可視化した図を図 4.3 に示す。

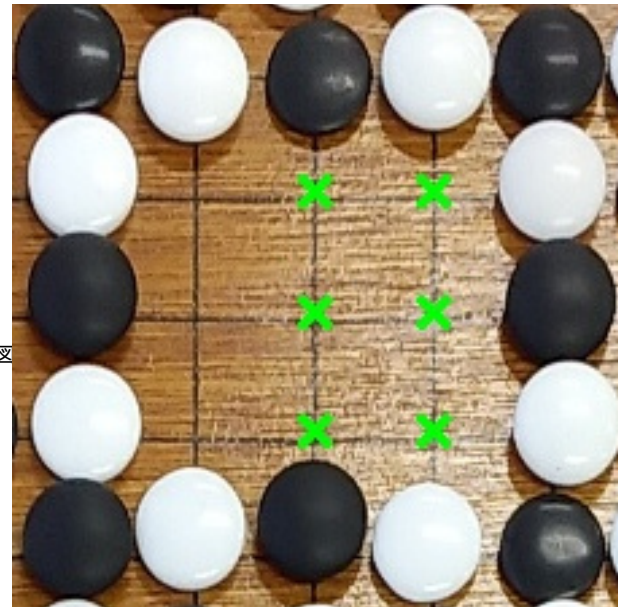


図 4.5 誤り部分の一部

4.2 事例 3

この盤面画像（図 4.4）に対し、システムを適用した。識別結果に誤りがあった部分にマークを付け、その一部をトリミングしたものを図 4.5 に示す。

図 4.5 の範囲に対し、白石の識別に用いた閾値よりも暗い部分を 0、明るい部分を 1 として `inRange()` 関数で二値画像を生成すると、白石が無い部分も白石があるとみなしていることが分かる。このことから、盤面や黒石に反射光が含まれていると、システムは誤って白石と識別してしまうと考える。

5. 結言

けつげん

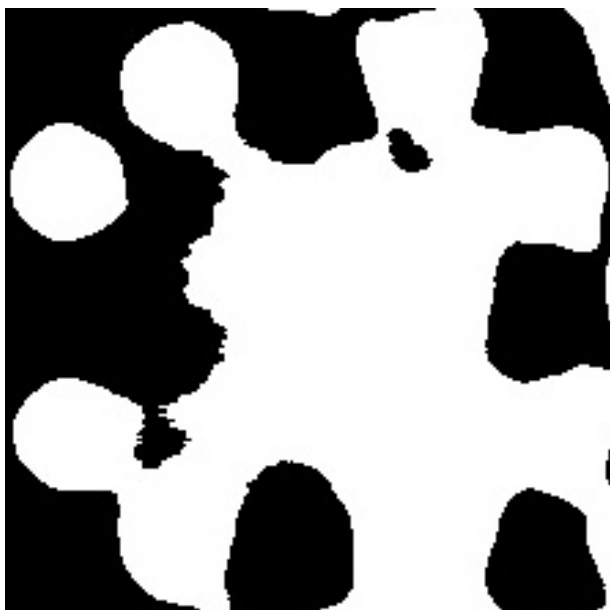


図 4.6 誤り部分をしきい値で二値化

参考文献

- [1] DeepMind「AlphaGo — DeepMind」(最終閲覧日: 令和 3 年 1 月 13 日)
<https://deepmind.com/research/case-studies/alphago-the-story-so-far>
- [2] デジタル画像処理 [改訂第二版], 松阪 喜幸 (2020)