

1 動作の流れ

1. 試合開始と共に公開される問題番号を入力する事によって、自動的に問題画像をサーバより取得する。
2. 画像を取得できた事を確認したら、画像の復元を開始する。
3. 解答を得たら、サーバに送信する。これもプログラム中に実装されている機能を用いる。
4. 送信した解答の正誤をサーバより取得し、中身を解析する。解答が完全な物では無かったら、サーバより取得した正誤を元に再解析を行う。

2 アルゴリズム

本チームのプログラムは、画像に特殊な変換を施すといったような事はせず、人工的に作られた画像で無い限り、画像中の隣り合った画素には強い関連性がある事を用いて、画像の間の繋がりを求めていく。

まず画像に施された縁取りを元に、画像の四隅を決定する。次に縁取りがついている画像を、縁取りがついている場所によって4つのグループに分類する。各グループ中でそれぞれの画像に評価関数を適用し、画像を繋げていき外周を完成させる。最後に外周から内側に向けて画像を繋げていく。

2.1 評価関数

2つの画像の間に関連性があるかどうかを求める関数である。2つの隣り合った画像を取る関数で、2つの画像において隣り合った画素及びその両隣の画素の輝度値の変動の総和を関数の値とする。式で表すと以下の様になる。

$$f(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^n (|x_i - y_i| + |x_i - y_{i-1}| + |x_i - y_{i+1}|)$$
$$y_0 = y_{n+1} = 0$$

上式で得られた値を $[0, 1]$ で規格化した物を実際には用いる。値が小さいほど隣あっている確立が高い。