

1. 元画像の復元

どのような種類の画像にまで対応するかには悩んだが、断片画像の辺だけ見ても元画像を復元可能なものとして解くことにした。全ての断片画像の辺同士の隣接度(元画像で隣り合っている可能性のような値)を求め、断片画像を組み合わせて元画像を作成した際、隣り合った辺同士の隣接度の合計が最も良い値を取るように画像を復元する。

隣接度の計算は、現在は隣り合ったピクセル同士の色差を用いているが改善の余地がありそうだと感じている。

隣接度を用いての復元はビームサーチを用いた。今回の問題では画像の回転が存在するため、単純なビームサーチであれば状態数が多くなってしまい、正確性/速度ともにあまり良くない。そこで、貪欲法を交えつつ遷移を工夫することで高速に解を求められるようにした。

2. 復元手順の構築

具体的な手順の構築プログラムは、これから実装し結果を見てフィードバックしながら作成するため、詳細には記述できない。

ただ、どのようなプログラムであっても、復元手順を求めるプログラムをそのまま用いて手順を部分的に改善することが可能である。何らかの手順の構築では手順の部分的な変更は難しいものが多い中でこの性質はとても重要であると考ええる。

3. 回答の管理

本競技において3台までPCを使うことができる。無駄な回答送信の削減や計算資源を最大限活用するために、複数のPCが作成した回答をあるPCが取りまとめ、管理し伝達するようにした。