

## 1. はじめに

今回の競技は、正方形から切り出した多角形を復元し、なるべく多くの「ピース」を「わく」に収めることを目的とした競技である。構築するシステムは「ピース」を画像化する入力部分と「ピース」を「わく」に収めるように並べるアルゴリズムの処理部、結果を表示する出力部によって構成される。それぞれの処理について以下に説明する。

## 2. 入力：画像処理について

今回はデータ入力の正確性を考慮し、A3 スキャナを用いて画像化を行う。取得した画像は「OpenCV」を用いた2値化を行い、線の検出を行う。この画像処理において「わく」及び「ピース」を検出しデータ化する。

## 3. 処理：アルゴリズムについて

データ化した「わく」及び「ピース」は、深さ優先探索を用いた全頂点に対して「ピース」を置く、全探索を基本

とする。ただし、「ピース」が多い場合には枝刈りを行うことで探索時間の短縮を図る。探索のオーダは「わく」と「ピース」の頂点数や多角形自体の数に大きく依存するため、解の候補が複数得られる可能性があり、その際は解の判断を人間が行う。

## 4. 出力：UI について

今回の競技では、出力結果の見やすさ、データと現実の「ピース」との紐づけが非常に重要となる。そのため、解答を表示する GUI を用意し、人間はその GUI に沿って実際に「わく」に「ピース」を設置する。

## 5. 開発環境

言語：Java 言語

環境：Eclipse

ライブラリ：OpenCV 2.4.12、PDFBox 2.0.2