

## 1. 処理の流れ

撮影装置にフレームとピース(以下パズル)を順番に置き、それぞれ Web カメラで撮影する。撮影した画像を、線分検出を用いてデータ

に変換し、それをソルバーに送信、ソルバーが解答を返す。解答はディスプレイに出力されたピースの



解答の出力例

配置を確認しながら手動で行う。

## 2. 設計

問題の傾向によって最適なアルゴリズムが異なると考えられるので、異なる手法やしきい値を用いて複数のアルゴリズムを並列で実行させる。そのため3台のパソコンをサーバーとクライアントに分けて動作させる。

## 2. アルゴリズム

実数空間のパッキングアルゴリズムは計算時間の観点から難しいと判断したため、ピース及びフレームの、辺の長さや頂点の角度を用いて探索を行うことにした。

### 2.1 アルゴリズム A

パズル同士を比較、角度の一致する場所や辺の長さが一致する場所を探し、一致している場所の数を評価値とし探索を行う。評価値が一定のしきい値を超えたピース同士は探索前に結合し、ある程度の評価値をもったピース同士はピースの ID を記録しておき、探索時の枝切りに使用する。

### 2.2 アルゴリズム B

フレームの辺ごとに、辺の幅にピッタリ収まるようなピースの集合を探しだし、それぞれの集合が重ならないようにピースを置いていく。

## 3. 開発環境

Ubuntu, C++14, Qt, Boost, Opencv