

部 門	競 技 部 門	No.1 登録番号	30030
-----	---------	-----------	-------

No.2	1) 予定開発期間 : 6ヶ月 2) 予定開発人数 : 4人																																								
	<table border="1"> <thead> <tr> <th></th> <th>4月</th> <th>5月</th> <th>6月</th> <th>7月</th> <th>8月</th> <th>9月</th> <th>10月</th> </tr> </thead> <tbody> <tr> <td>問題分析</td> <td colspan="2">←→</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>設計</td> <td></td> <td colspan="3">←→</td> <td></td> <td></td> <td></td> </tr> <tr> <td>実装</td> <td></td> <td></td> <td colspan="5">←→</td> </tr> <tr> <td>試用・トレーニング</td> <td></td> <td></td> <td></td> <td colspan="4">←→</td> </tr> </tbody> </table>		4月	5月	6月	7月	8月	9月	10月	問題分析	←→							設計		←→						実装			←→					試用・トレーニング				←→			
		4月	5月	6月	7月	8月	9月	10月																																	
	問題分析	←→																																							
	設計		←→																																						
実装			←→																																						
試用・トレーニング				←→																																					

No.3	<p>実現方法</p> <p>1) 原画像の推察アルゴリズム</p> <p>正確な断片画像の並び替えを行うには、高い精度で原画像の推察を行うことが必要である。各断片画像の評価を行う関数（以下、評価関数）を作成し、遺伝的アルゴリズムによって断片画像から原画像を推察する。遺伝的アルゴリズムを用いるため、始めにランダムに初期世代集団を生成し、評価を行う。評価関数により出力された値を基に次の世代を生成し、評価を行う。これを最大世代数まで繰り返し、原画像を推察する。推察を効率的に行うためには個体数、最大世代数を適切に設定する必要があるためテスト画像を作成し、データの収集を行い最適値の設定を行う。</p> <p>評価関数は、各断片画像を隣接させた時のピクセルの非類似度を算出し、各断片画像の組み合わせの評価値を出力するものである。</p>
	<p>2) 断片画像の並び替えアルゴリズム</p> <p>最小コストで並び替えを行うためには選択回数と交換回数を少なくするため、最も効率の良い手順を探索によって導き、選択する。最初に、盤面を右上、右下、左上、左下4つの区画に分け、それぞれの断片画像が目標のマスと同じ区画に入るように移動させる。これには、目標のマスまでのマンハッタン距離などを評価関数にしたビームサーチを用いる。4つの区画に分けたあとは、それぞれの区画ごとに独立に探索を行う。ただし、分割数が少ない場合、全体を一度に探索するほうが良い手順が得られるため、この操作は行わない。</p> <p>探索には、それぞれの断片画像を順番に目標のマスに動かしていくという方法を用いる。この移動の順番を決めるために、ビームサーチを用いる。一つの断片画像を目標マスに動かす手順の計算には、ダイクストラ法を用いる。</p>
	<p>3) その他（独創的など）</p> <p>断片画像と推察した画像、探索によって得られた並び替え手順などを画面に表示し、状況を把握する GUI を作成する。探索結果に応じて並び替えの探索を行うソルバの切り替えを行う機能を作成し、問題毎に最適な探索を行う。</p> <p>並び替えアルゴリズムに Optuna というハイパーパラメータ最適化ツールを使い、問題毎に評価関数の重みづけの最適化を行う。</p>

No.4	<p>開発環境</p> <p>言語 : Java, Rust, Python 3</p> <p>IDE : IntelliJ IDEA</p> <p>OS : Windows10</p>
------	---