

点群マッチング

(Ver.1.0)

2011 年 9 月

株式会社 アイディール

目次

eyemMatchBasic	1
機 能 2点群のマッチング	1
eyemMatchDelta	3
機 能 (ユーティリティ) 基準点の移動量, 姿勢およびスケール	3
eyemMatchPoints.....	4
機 能 (ユーティリティ) Master 点群の各点位置および対応点間距離.....	4

eyemMatchBasic

機 能	2点群のマッチング														
形 式	#include “eyemLib.h” int eyemMatchBasic (int iPtNum, EyemOcsDXY taWork[], EyemOcsDXY taMaster[], EyemOcsDXYQ *tpBase, int iCalcMode, double dRobustCoef, EyemOcsDXYQS *tpResult);														
解 説	<p>各点の対応がとれている2つの2次元点群 (Master 点群および Work 点群とします) に対し、Master 点群を Work 点群に合わせこみ、そのときの Master 点群における『基準点の位置、姿勢(回転角)およびスケール(相似比)』を推定します。</p> <p>マッチング手法は、「最小二乗法」、「ロバスト推定」または「ミニマックス法」を選択できます。ミニマックス法とは、各対応点間の距離の最大値が最も小さくなるように合わせ込みを行う手法です。</p>														
引 数	iPtNum	データ(座標)点の数です。2以上を入力してください。													
	taWork[]	Work点群の2次元座標値です。													
	taMaster[]	Master点群の2次元座標値です。													
	*tpBase	Master点群の基準点位置とその姿勢(初期回転角:rad単位)です。													
	iCalcMode	処理選択のフラグです。以下の選択となります。													
		<table><tr><td>EYEM_MATCH_LSQ</td><td>最小二乗法・スケール推定なし</td></tr><tr><td>EYEM_MATCH_LSQ_S</td><td>最小二乗法・スケール推定あり</td></tr><tr><td>EYEM_MATCH_ROBUST</td><td>ロバスト推定・スケール推定なし</td></tr><tr><td>EYEM_MATCH_ROBUST_S</td><td>ロバスト推定・スケール推定あり</td></tr><tr><td>EYEM_MATCH_MINMAX</td><td>ミニマックス法・スケール推定なし</td></tr><tr><td>EYEM_MATCH_MINMAX_S</td><td>ミニマックス法・スケール推定あり</td></tr></table>		EYEM_MATCH_LSQ	最小二乗法・スケール推定なし	EYEM_MATCH_LSQ_S	最小二乗法・スケール推定あり	EYEM_MATCH_ROBUST	ロバスト推定・スケール推定なし	EYEM_MATCH_ROBUST_S	ロバスト推定・スケール推定あり	EYEM_MATCH_MINMAX	ミニマックス法・スケール推定なし	EYEM_MATCH_MINMAX_S	ミニマックス法・スケール推定あり
EYEM_MATCH_LSQ	最小二乗法・スケール推定なし														
EYEM_MATCH_LSQ_S	最小二乗法・スケール推定あり														
EYEM_MATCH_ROBUST	ロバスト推定・スケール推定なし														
EYEM_MATCH_ROBUST_S	ロバスト推定・スケール推定あり														
EYEM_MATCH_MINMAX	ミニマックス法・スケール推定なし														
EYEM_MATCH_MINMAX_S	ミニマックス法・スケール推定あり														
	dRobustCoef	Work点群のばらつきを表す係数です。ロバスト推定を選択のときのみ有効となります。この係数の目安は、Master点群を合わせ込んだときに、Work点がMaster点から標準偏差で σ のばらつきを持つと推定されるとき、 $k\sigma$ 以上のデータ点を無視したい場合には、この係数‘ k ’を入力してください(推奨値:2.7)。													
	*tpResult	マッチング結果(合わせ込んだときのMaster点群における『基準点の位置、姿勢(回転角:rad単位)およびスケール(相似比)』)が格納されます。													
戻り値	エラー報告です。														
	<table><tr><td>FUNC_OK</td><td>正常終了</td></tr></table>			FUNC_OK	正常終了										
FUNC_OK	正常終了														

FUNC_NOT_ENOUGH_MEM	メモリ不足
FUNC_ILLEGAL_ARGUMENT	パラメータが不正
FUNC_CANNOT_CALC	マッチングパラメータの推定失敗

留意事項

‘スケール推定なし’を選択した場合は、スケールは1に固定されます。

eyemMatchDelta

機 能 (ユーティリティ) 基準点の移動量, 姿勢およびスケール

形 式

```
#include "eyemLib.h"

int      eyedMatchDelta ( EyemOcsDXYQ *tpBase, EyemOcsDXYQS *tpResult
                        double *dpDeltaX, double *dpDeltaY, double *dpTheta, double *dpScale );
```

解 説 eyedMatchBasic で得られた結果をもとに, Master 点群における基準点の位置移動量と基準点周りの回転角 (基準姿勢からの回転角) およびスケールを求めます (スケール値は変化しません).

引 数

*tpBase	Master点群の基準点位置とその姿勢 (初期回転角:rad単位) です.
*tpResult	マッチング結果です. eyedMatchBasic関数の出力値を入力してください.
*dpDeltaX	基準点 x 座標の移動量が格納されます.
*dpDeltaY	基準点 y 座標の移動量が格納されます.
*dpTheta	基準点周りの回転角 (基準姿勢からの回転角:rad単位) が格納されます.
*dpScale	基準点中心のスケール (相似比) が格納されます (スケール値は, 入力値からの変化はありません).

戻り値 エラー報告です.

FUNC_OK	正常終了
FUNC_ILLEGAL_ARGUMENT	パラメータが不正

留意事項 特にありません.

eyemMatchPoints

機 能 (ユーティリティ) Master 点群の各点位置および対応点間距離

形 式

```
#include "eyemLib.h"

int      eyemMatchPoints ( int iPtNum, EyemOcsDXY taWork[], EyemOcsDXY taMaster[],
                           EyemOcsDXYQ *tpBase, EyemOcsDXYQS *tpResult,
                           EyemOcsDXY taPoints[], double daDist[] );
```

解 説 eyemMatchBasic で得られた結果をもとに, Master 点群の各点位置および対応点間の距離 (マッチング誤差) を求めます.

引 数

iPtNum	データ(座標)点の数です. 2以上を入力してください.
taWork[]	Work点群の2次元座標値です.
taMaster[]	Master点群の2次元座標値です.
*tpBase	Master点群の基準点位置とその姿勢(初期回転角:rad単位)です.
*tpResult	マッチング結果です. eyemMatchBasic関数の出力値を入力してください.
taPoints[]	Master点群の各点のマッチング位置が格納されます.
daDist[]	マッチング結果における, 各点の対応点間の距離(マッチング誤差)が格納されます.

戻り値 エラー報告です.

FUNC_OK	正常終了
FUNC_ILLEGAL_ARGUMENT	パラメータが不正

留意事項 特にありません.

改訂履歴

Version No.	内 容
1.0	• 新規発行