

# 2値ブロッツ解析

(Ver.1.0)

2013 年 6 月

株式会社 アイディール

## 目次

eyemBinOtsuThresholdImage .....	1
機能 判別分析法(大津の方法)による, 画像の2値化レベル .....	1
eyemBinOtsuThresholdGray .....	2
機能 判別分析法(大津の方法)による, 濃度データの2値化レベル .....	2
eyemBinOtsuThresholdHistogram .....	3
機能 判別分析法(大津の方法)による, ヒストグラムの分離しきい値 .....	3
eyemBinBinaryImage .....	4
機能 2値画像の作成 .....	4
eyemBinDilation .....	5
機能 白ブロップの指定回数分の膨張 .....	5
eyemBinErosion .....	6
機能 白ブロップの指定回数分の収縮 .....	6
eyemBinOpening .....	7
機能 白ブロップのオープニング(収縮⇒膨張) .....	7
eyemBinClosing .....	8
機能 白ブロップのクロージング(膨張⇒収縮) .....	8
eyemBinBlob .....	9
機能 2値画像のブロップ解析 .....	9
eyemBinFree .....	10
機能 ブロップ解析結果の格納領域の解放 .....	10

---

## eyemBinOtsuThresholdImage

---

**機 能** 判別分析法(大津の方法)による, 画像の2値化レベル

**形 式** `#include "eyemLib.h"`  
`int       eyemBinOtsuThresholdImage ( EyemImage *tpImage, EyemRect *tpRoi );`

**解 説** 判別分析法(大津の方法)により, 画像の2値化レベルを求めます. この2値化レベルは, それ以上を「白」, それ未満を「黒」とする境界の濃度値です.

処理領域内の濃度ヒストグラムの各度数を  $d_i$  ( $i = 0, \dots, 255$ ) とし, それらの平均を  $\mu_T$  および総和を  $N$  とします. このとき,  $i = k$  における次式のクラス間分散  $\sigma_B^2(k)$  が最大となる  $k = k^*$  を求めます.

$$\sigma_B^2(k) = \frac{\left( \mu_T \sum_{i=0}^k d_i - \sum_{i=0}^k i d_i \right)^2}{\sum_{i=0}^k d_i \left( N - \sum_{i=0}^k d_i \right)}.$$

出力される2値化レベルは,  $k^* + 1$  となります.

**引 数** `*tpImage`       処理対象となるグレイ画像の情報です.  
`*tpRoi`            処理領域(始点&サイズ)です.

**戻り値** エラー報告です.

0~255	(正常終了)2値化レベル
FUNC_ILLEGAL_ARGUMENT	パラメータが不正

**留意事項** 特にありません.

---

## eyemBinOtsuThresholdGray

---

**機 能** 判別分析法(大津の方法)による, 濃度データの2値化レベル

**形 式** `#include "eyemLib.h"`  
`int       eyemBinOtsuThresholdGray ( int n, double daGray[] );`

**解 説** 判別分析法(大津の方法)により, 濃度データ(濃度レベル:0~255)の2値化レベルを求めます. この2値化レベルは, それ以上を「白」, それ未満を「黒」とする境界の濃度値です.  
データの濃度ヒストグラムの各度数を  $d_i$  ( $i = 0, \dots, 255$ ) とし, それらの平均を  $\mu_T$  および総和を  $N$  とします. このとき,  $i = k$  における次式のクラス間分散  $\sigma_B^2(k)$  が最大となる  $k = k^*$  を求めます.

$$\sigma_B^2(k) = \frac{\left( \mu_T \sum_{i=0}^k d_i - \sum_{i=0}^k i d_i \right)^2}{\sum_{i=0}^k d_i \left( N - \sum_{i=0}^k d_i \right)}.$$

出力される2値化レベルは,  $k^* + 1$  となります.

**引 数** `n` 濃度データの個数です.  
`daGray[]` 濃度配列です. 数値は 0~255 の範囲で用意してください.

**戻り値** エラー報告です.

0~255	(正常終了)2値化レベル
FUNC_ILLEGAL_ARGUMENT	パラメータが不正

**留意事項** 特にありません.

---

## eyemBinOtsuThresholdHistogram

---

**機 能** 判別分析法(大津の方法)による, ヒストグラムの分離しきい値

**形 式** `#include "eyemLib.h"`  
`int       eyemBinOtsuThresholdHistogram ( int n, unsigned int uiHist[] );`

**解 説** 判別分析法(大津の方法)により, ヒストグラムの分離しきい値を求めます. この分離しきい値の意味は, それ以上とそれ未満の組に分けるときの境界となるヒストグラム配列のインデックスです.

$n$  レベルのヒストグラムの各度数を  $d_i$  ( $i=0, \dots, n-1$ ) とし, それらの平均を  $\mu_T$  および総和を  $N$  とします. このとき,  $i=k$  における次式のクラス間分散  $\sigma_B^2(k)$  が最大となる  $k=k^*$  を求めます.

$$\sigma_B^2(k) = \frac{\left( \mu_T \sum_{i=0}^k d_i - \sum_{i=0}^k i d_i \right)^2}{\sum_{i=0}^k d_i \left( N - \sum_{i=0}^k d_i \right)}.$$

出力される分離しきい値は,  $k^* + 1$  となります.

**引 数** `n`                   ヒストグラムのレベル数(ヒストグラムの配列要素数)です.  
`uiHist[]`               ヒストグラム(uiHist[0]~uiHist[n-1])です.

**戻り値** エラー報告です.

0~(n-1)	(正常終了)分離しきい値
FUNC_ILLEGAL_ARGUMENT	パラメータが不正

**留意事項** 特にありません.

---

## eyemBinBinaryImage

---

**機 能**                    2値画像の作成

**形 式**                    `#include "eyemLib.h"`  
`int            eyedBinBinaryImage ( EyemImage *tpSrcImg, EyemRect *tpRoi, int iBinLevel,`  
`EyemImage *tpDstImg );`

**解 説**                    処理領域内において、2値(0 または 255)のグレイ画像を作成します。

**引 数**                    `*tpSrcImg`                処理対象となるグレイ画像の情報です。  
`*tpRoi`                    処理領域(始点&サイズ)です。  
`iBinLevel`                2値化レベルです。この値以上の濃度値を白(255)とします。  
`*tpDstImg`                2値(0または255)のグレイ画像です。tpSrcImg と同じでも構いません。

**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。

---

## eyemBinDilation

---

**機 能** 白ブロップの指定回数分の膨張

**形 式**

```
#include "eyemLib.h"

int      eyedBinDilation ( EyemImage *tpSrcImg, EyemRect *tpRoi, int iBinLevel,
                           int iNum, EyemImage *tpDstImg );
```

**解 説** 画像を2値化し, 処理領域内における白ブロップの膨張を指定回数分を行います. なお, 対象ブロップは8連結として処理します.

**引 数**

*tpSrcImg	処理対象となるグレイ画像の情報です.
*tpRoi	処理領域(始点&サイズ)です.
iBinLevel	2値化レベルです. この値以上の濃度値を白(255)とします.
iNum	膨張回数です. 1以上の値を指定してください.
*tpDstImg	膨張結果の2値(0または255)のグレイ画像です.

**戻り値** エラー報告です.

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

**留意事項** 特にありません.

---

## eyemBinErosion

---

**機 能** 白ブロップの指定回数分の収縮

**形 式**

```
#include "eyemLib.h"

int      eyedBinErosion ( EyemImage *tpSrcImg, EyemRect *tpRoi, int iBinLevel,
                          int iNum, EyemImage *tpDstImg );
```

**解 説** 画像を2値化し, 処理領域内における白ブロップの収縮を指定回数分を行います. なお, 対象ブロップは8連結として処理します.

**引 数**

*tpSrcImg	処理対象となるグレイ画像の情報です.
*tpRoi	処理領域(始点&サイズ)です.
iBinLevel	2値化レベルです. この値以上の濃度値を白(255)とします.
iNum	収縮回数です. 1以上の値を指定してください.
*tpDstImg	収縮結果の2値(0または255)のグレイ画像です.

**戻り値** エラー報告です.

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

**留意事項** 特にありません.



---

## eyemBinOpening

---

**機 能** 白ブロップのオープニング (収縮⇒膨張)

**形 式**

```
#include "eyemLib.h"

int      eyedBinOpening ( EyemImage *tpSrcImg, EyemRect *tpRoi, int iBinLevel,
                          int iNum, EyemImage *tpDstImg );
```

**解 説** 画像を2値化し, 処理領域内における白ブロップのオープニングを行います. これは, まず収縮を指定回数分行い, 続けて同じ回数分の膨張を行う処理です. なお, 対象ブロップは8連結として処理します.

**引 数**

*tpSrcImg	処理対象となるグレイ画像の情報です.
*tpRoi	処理領域 (始点&サイズ) です.
iBinLevel	2値化レベルです. この値以上の濃度値を白 (255) とします.
iNum	収縮 (膨張) 回数です. 1以上の値を指定してください.
*tpDstImg	オープニング結果の2値 (0または255) のグレイ画像です.

**戻り値** エラー報告です.

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

**留意事項** 特にありません.

---

## eyemBinClosing

---

**機 能** 白ブロップのクロージング (膨張⇒収縮)

**形 式**

```
#include "eyemLib.h"

int      eyedBinClosing ( EyemImage *tpSrcImg, EyemRect *tpRoi, int iBinLevel,
                          int iNum, EyemImage *tpDstImg );
```

**解 説** 画像を2値化し, 処理領域内における白ブロップのクロージングを行います. これは, まず膨張を指定回数分行い, 続けて同じ回数分の収縮を行う処理です. なお, 対象ブロップは8連結として処理します.

**引 数**

*tpSrcImg	処理対象となるグレイ画像の情報です.
*tpRoi	処理領域 (始点&サイズ) です.
iBinLevel	2値化レベルです. この値以上の濃度値を白 (255) とします.
iNum	膨張 (収縮) 回数です. 1以上の値を指定してください.
*tpDstImg	クロージング結果の2値 (0または255) のグレイ画像です.

**戻り値** エラー報告です.

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

**留意事項** 特にありません.

## eyemBinBlob

**機 能** 2値画像のブロッブ解析

**形 式**

```
#include "eyemLib.h"

int      eyemBinBlob ( EyemImage *tpImage, EyemRect *tpRoi, int iBinLevel, int iColor,
                      int iBorder, int iAreaThrs, EyemBinBlob **tpResult, int *ipNum );
```

**解 説** 画像を2値化し、処理領域内のしきい値以上の面積をもつブロッブを解析対象とします。解析項目は、面積、重心、外接長方形および主軸傾斜角です。なお、対象ブロッブは8連結として処理します。

**引 数**

*tpImage	処理対象となるグレイ画像の情報です。				
*tpRoi	処理領域(始点&サイズ)です。				
iBinLevel	2値化レベルです。この値以上の濃度値を白(255)とします。				
iColor	解析対象ブロッブの色です。以下のいずれかを選択してください。 <table border="1"><tbody><tr><td>EYEM_BIN_BLACK</td><td>黒ブロッブ</td></tr><tr><td>EYEM_BIN_WHITE</td><td>白ブロッブ</td></tr></tbody></table>	EYEM_BIN_BLACK	黒ブロッブ	EYEM_BIN_WHITE	白ブロッブ
EYEM_BIN_BLACK	黒ブロッブ				
EYEM_BIN_WHITE	白ブロッブ				
iBorder	境界接触ブロッブの有効性です。以下のいずれかを選択してください。 <table border="1"><tbody><tr><td>ON</td><td>解析対象に含める</td></tr><tr><td>OFF</td><td>無視する</td></tr></tbody></table>	ON	解析対象に含める	OFF	無視する
ON	解析対象に含める				
OFF	無視する				
iAreaThrs	ブロッブ面積のしきい値です。この値以上の面積をもつブロッブを解析対象とします。				
**tpResult	ブロッブ解析結果格納用の配列です。解析項目は、面積、重心、外接長方形(始点・終点の座標およびx方向・y方向のサイズ(フェレ径))および主軸傾斜角(x軸から測った角度:rad単位)です。 使用の際は、NULLで初期化して入力してください。また、使用後は、必ず eyemBinFree() を実行して領域を解放してください。				
*ipNum	見つかった対象ブロッブの個数が格納されます。				

**戻り値** エラー報告です。

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

**留意事項** tpResult は、NULL で初期化して入力し、使用後は必ず eyemBinFree() を実行して領域を解放してください。

---

## eyemBinFree

---

機 能	ブロップ解析結果の格納領域の解放
形 式	<pre>#include "eyemLib.h"  void      eyedBinFree ( EyemBinBlob **tpResult );</pre>
解 説	eyemBinBlob() で確保された, ブロップ解析結果の格納領域を解放します. NULL がセットされて返ります.
引 数	<b>**tpResult</b> ブロップ解析結果の格納領域です.
戻り値	ありません.
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