

# CONTENTS

---

ビジュアルヘルプ - ImageThreshold.....	2
ImageThreshold コマンドのヘルプ .....	2

# ビジュアルヘルプ – ImageThreshold

## ImageThreshold コマンドのヘルプ

**ImageThreshold** [*flags*] *imageMatrix*

閾値処理を実行し、グレースケール *imageMatrix* を二値画像に変換します。

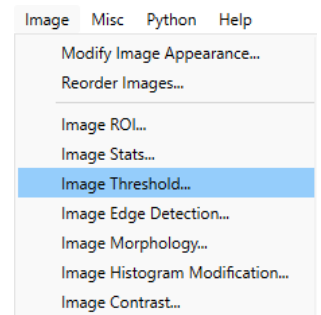
このコマンドはすべてのデータ型に対応しています。

ただし、ソースウェーブは2次元行列でなければなりません。

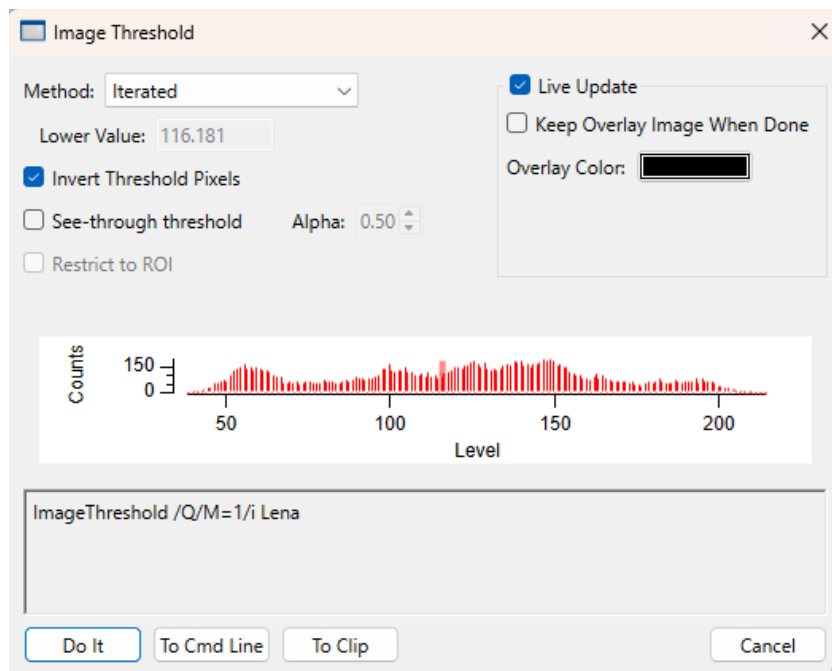
*imageMatrix* に NaN が含まれている場合、NaN 値に対応するピクセルは値 64 にマッピングされます。

On ピクセルと Off ピクセルの値は、それぞれ 255 と 0 です。

結果の画像は、ウェーブ *M\_ImageThresh* に格納されます。



メニュー Image→Image Threshold では視覚的に操作するためのダイアログを表示できます。



### フラグ

*/C*

元の画像と閾値処理によって生成された画像との相関係数を計算します。相関係数は履歴エリアに出力されます (*/Q* フラグが指定されていない場合)。また、変数 *V\_correlation* にも格納されます。

*/DEST=destWave*

このフラグは、コマンドによって作成される出力ウェーブを指定します。このフラグを使わない場合、コマンドの結果は現在のデータフォルダー内のウェーブ *M\_ImageThresh* に保存されます。

*srcWave* と *destWave* の両方に同じウェーブを指定するのはエラーです。

関数内で使う場合、ImageThreshold は宛先ウェーブに対して実数型のウェーブ参照を作成します。詳細は、ヘルプ Automatic Creation of WAVE References を参照してください。

Igor Pro 10.0 で追加されました。

/FREE

宛先ウェーブ *destWave* をフリーウェーブとして作成します。

/FREE は関数内でのみ使用可能であり、かつ *destWave* が単純な名前またはウェーブ参照のフィールドである場合に限り使用できます。

詳細は、ヘルプ Free Waves を参照してください。

Igor Pro 10.0 で追加されました。

/I

このフラグを使うと、画像に書き込まれる値が反転します。つまり、閾値を超えるすべてのピクセルが 0 に設定されます。

/M=*method*

計算結果は、履歴エリアに出力され (/Q オプションが指定されていない場合)、変数 *V\_threshold* に格納されます。

*method*=0      デフォルト。この場合、/I フラグを使って、手動で選択した閾値を指定する必要があります。

*method*=1      反復法を用いて閾値を自動的に算出します。

*method*=2      画像のヒストグラムは、単純な二峰性分布を示します。

*method*=3      適応型閾値処理。各行の最後の 8 ピクセルに基づいて閾値を評価し、行を交互に処理します。出力ウェーブ *M\_ImageThresh* の数値型は、入力ウェーブと同じです。特に、入力の数値型が符号付きバイトの場合、On ピクセルの値は 127、Off ピクセルの値は 0 となります。

なお、このメソッドは、ImageEdgeDetection コマンドの一部として使う場合、サポートされていません。

*method*=4      「曖昧さ」の尺度としてエントロピーを用いたファジー閾値処理です。

*method*=5      対象物と背景の平均グレーレベルを含む「曖昧さ」指標を最小化する手法を用いたファジー閾値処理です。

*method*=6      データにヒストグラムを作成し、画像をクラスターの集合として表現した上で、クラスターが 2 つになるまで反復的に削減することで、理想的な閾値を決定します。その後、閾値は、より低い方のクラスターの最高レベルに設定されます。

この手法は、A.Z. Arifin と A. Asano による論文（下記参考文献参照）に基づいています

が、ヒストグラムが比較的平坦な画像を処理できるように修正されています。

画像のヒストグラムから2つ未満のクラスターしか得られない場合、この方法では閾値を決定することができず、閾値は NaN に設定されます。

Igor Pro 7.0 で追加されました。

*method=7* 「対象」と「背景」の間の総分散を最大化するよう、最適な閾値を決定します。  
<[http://en.wikipedia.org/wiki/Otsu%27s\\_method](http://en.wikipedia.org/wiki/Otsu%27s_method)> を参照してください。

Igor Pro 7.0 で追加されました。

*/N* 背景レベルを 64 (つまり NaN) に設定します。

*/O* 元の画像を、計算された閾値画像で上書きします。

*/O* フラグを指定しない場合、閾値処理後の画像はウェーブ *M\_ImageThresh* に書き込まれます。

*/P=layer* *imageMatrix* が 3D ウェーブの場合、*/P* は閾値計算を行う特定のレイヤーを選択します。*layer* は 0 から始まるレイヤーインデックスです。

*layer* が -1 (デフォルト値) の場合、*imageMatrix* のすべてのレイヤーに対して閾値が計算されます。

*/P* フラグは */O* フラグと互換性がありません。

Igor Pro 7.0 で追加されました。

*/Q* 計算された相関係数 (*/C*) と計算された閾値 (*/M*) を履歴エリアに出力しないようにします。

*/R=roiSpec* このフラグを使って、ROI (関心領域) を指定します。ROI は、unsigned byte 型 (*/b/u*) のウェーブによって定義されます。ROI ウェーブの行数と列数は、画像ウェーブと同じでなければなりません。ROI 自体は、値が 0 のエントリ/ピクセルによって定義されます。ROI 外のピクセルは、0 以外の任意の値をとることができます。ROI は連続している必要はなく、任意の形状をとることができます。ROI ウェーブの作成に関する詳細は、*ImageGenerateROIMask* コマンドのヘルプを参照してください。

一般的に、*roiSpec* は {*roiWaveName* , *roiFlag*} という形式をとります。ここで、*roiFlag* は以下の値をとることができます：

- 0 : ROI の外側のピクセルを 0 に設定する
- 1 : ROI の外側のピクセルを元の画像と同じ設定にする
- 2 : ROI の外側のピクセルを NaN (=64) に設定する

デフォルトでは、*roiFlag* は 1 に設定されており、この場合、短縮形式の */R=roiWave* を使って */R* フラグを使うことができます。

`/T=thresh`

希望する閾値が分かっている場合は、このフラグを使ってください。

`/W=Twave`

閾値の区間を適用したい場合に、このフラグを使います。各区間は、ウェーブ Twave 内の値のペアによって指定されます。各ペアの最初の要素が下限値、2番目の要素が上限値となります。指定されたすべての区間の外側にあるピクセル値は、0 に設定されます。

### 参考文献

自動閾値設定法 ( $M=1$ ) については、次の文献に記述されています：

T. W. Ridler and S. Calvard, IEEE Transactions on Systems, Man and Cybernetics, SMC-8, 630-632, 1978

$M=6$  で使われる閾値化手法については、次の文献に記述されています：

A.Z. Arifin and A Asano, "Image segmentation by histogram thresholding using hierarchical cluster analysis", Pattern Recognition Letters 27 (2006) 1515-1521

### 参照

コマンド ImageGenerateROIMask、ImageEdgeDetection

ヘルプ Image Processing