

第 3 章

第 2 章 前処理 演習問題

補足 : Scilab 6.0.0 でのデバッグ

Scilab 5 までは、エラーが原因で実行が停止した場合、その時点での変数の値を変数ブラウザで見ることができましたが、Scilab 6 ではこの機能がうまく働きません。Scilab 6 でデバッグを行う場合は、エラーを検出した行の先頭にカーソルを置き、SciNotes のメニューから「実行する」→「... カーソルまで実行（出力あり）」を選んでください。

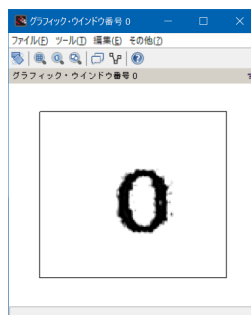
準備

1. 画像データを読み込む（`imread` 関数）
2. 画像データを倍精度表現に変換（`im2double` 関数）
3. 画像を表示する（`imshow` 関数）
4. 画像をファイルに出力する（`imwrite` 関数）
5. 中間値を求める（`median` 関数）
6. 2 重ループ

```
-->im=imread('test1.pgm'); // im は整数の 120x120 行列
```

```
-->im2=im2double(im); // 整数から倍精度表現に変換
```

```
-->imshow(im2)
```



```
-->imwrite(im2, 'out.png') // pgm, jpg, png, bmp, tiff も可
```

```
-->a = [5 8 7 2 3];
```

```
-->median(a)
ans =
    5.

-->b = [5 8 7 2 3 4];

-->median(b)
ans =
    4.5

-->for i=1:9
-->  for j=1:9
-->    mprintf("%3d",i*j)
-->  end
-->  mprintf("\n")
-->end
  1  2  3  4  5  6  7  8  9
  2  4  6  8 10 12 14 16 18
  3  6  9 12 15 18 21 24 27
  4  8 12 16 20 24 28 32 36
  5 10 15 20 25 30 35 40 45
  6 12 18 24 30 36 42 48 54
  7 14 21 28 35 42 49 56 63
  8 16 24 32 40 48 56 64 72
  9 18 27 36 45 54 63 72 81
```

実践演習 3-1

ソースコード 3.1 の (ア), (イ) を埋め、メディアンフィルタを実行する Scilab のコードを完成させよ。

ソースコード 3.1 メディアンフィルタ

```
clear;

// 画像データの読み込み
im = im2double(imread('test1.pgm'));
// 2次元配列 im のサイズ取得
[h w] = size(im);
// 結果格納用の配列 resultim を用意
resultim = ones(im);

// メディアンフィルタ適用
for y = 2:h-1
    for x = 2:w-1
        resultim( (ア) ) = median(im( (イ) ));
    end
end

// 結果の表示とファイルへの出力
imshow([im, resultim])
imwrite([im, resultim], 'out.png');
```

実践演習 3-2

ソースコード 3.1 に平均値フィルタ処理を加え、原画像、メディアンフィルタ適用後、平均値フィルタ適用後の画像を並べて結果を比較せよ。

実践演習 3-3

ソースコード 3.1 を参考に、Sobel フィルタ（第2章スライド参照）を適用するコードを作成せよ。