

第3回画像処理講義

～ 画像処理プログラミング① ～

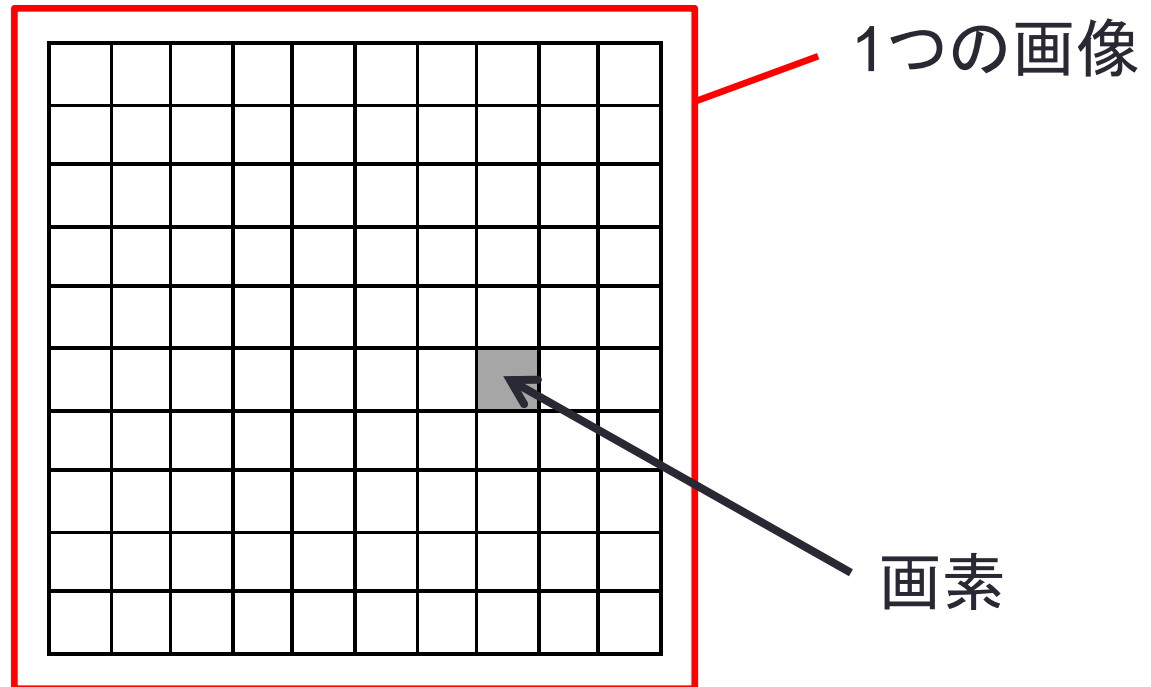
講義の流れ

- デジタル画像の基本
- 2値化の手法
- 2値画像の形状特徴パラメータ

デジタル画像の基本

デジタル画像

たくさんの画素が集まって1つの画像を構成




個々の画素が持つ光の強さや色の値を画素値という

デジタル画像：カラー画像

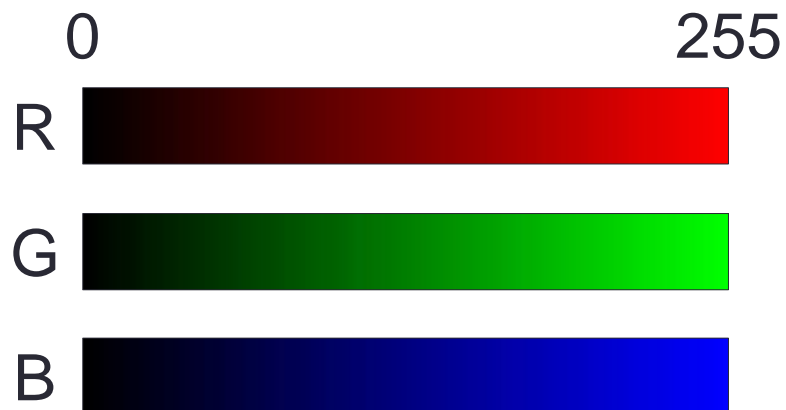
R(赤)G(緑)B(青), 3つの画素値を持つ



RGB画像

 { R : 58
G : 73
B : 89

画素値は256段階



デジタル画像：グレースケール画像

明るさに関する1つの画素値を持つ



■ { 70

画素値は256段階

0

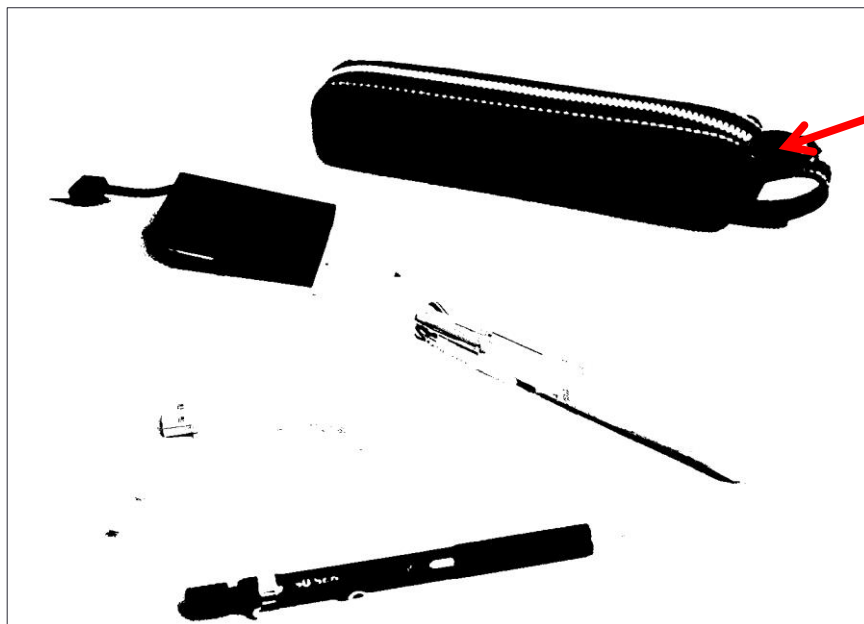
255



グレースケール画像

デジタル画像：2値画像

白または黒の1つの画素値を持つ



画素値は2段階



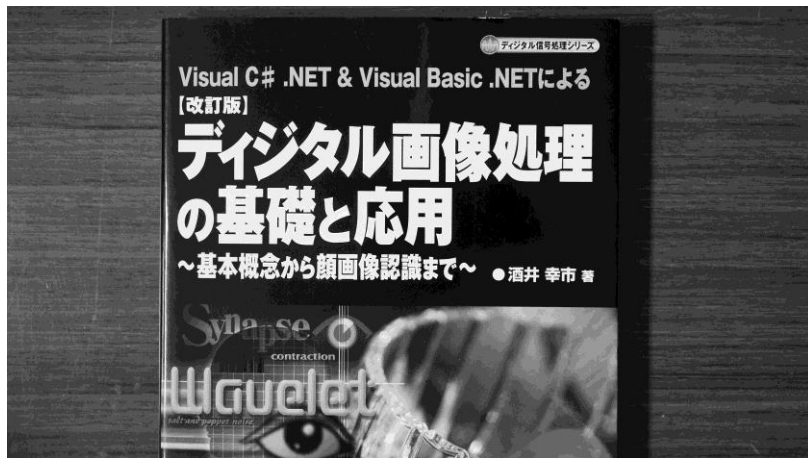
2値画像

2値化の手法

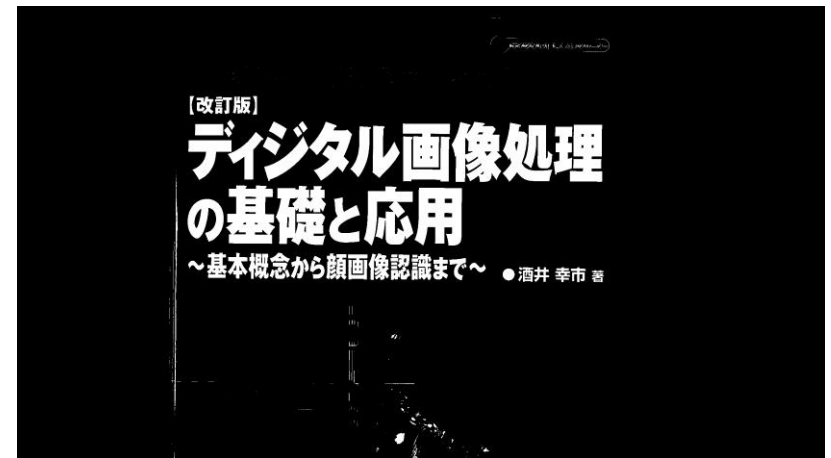
しきい値処理

ある画素値を基準としてグレースケール画像を2値化

例：画素値225 をしきい値として, しきい値処理



グレースケール画像



しきい値処理後の2値画像

0

255



225

例①: しきい値105

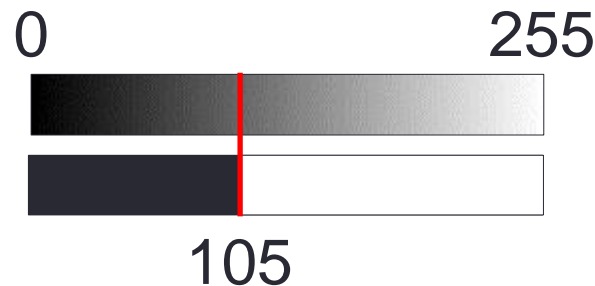
10	60	110	160	210
20	70	120	170	220
30	80	130	180	230
40	90	140	190	240
50	100	150	200	250

グレースケール画像



0	0	1	1	1
0	0	1	1	1
0	0	1	1	1
0	0	1	1	1
0	0	1	1	1

しきい値処理後の2値画像



例②: しきい値105, 155

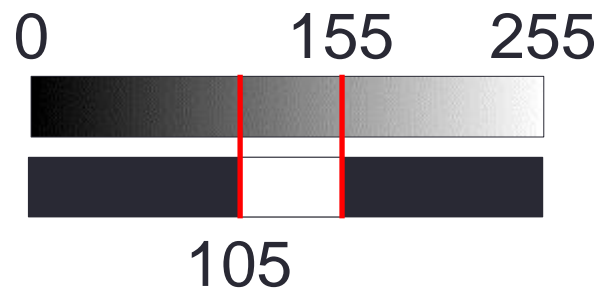
10	60	110	160	210
20	70	120	170	220
30	80	130	180	230
40	90	140	190	240
50	100	150	200	250



0	0	1	0	0
0	0	1	0	0
0	0	1	0	0
0	0	1	0	0
0	0	1	0	0

グレースケール画像

しきい値処理後の2値画像



MATLABにおけるしきい値処理

不等号としきい値を設定して2値化方法を指定

- 画素値105より大きい画素を白画素

例) `binaryImage = 105 < grayImage;`

- 画素値205より小さい画素を白画素

例) `binaryImage = grayImage < 205;`

- 画素値60より大きく, 180より小さい画素を白画素

例) `binaryImage = 60 < grayImage & grayImage < 180;`

2値画像の形状特徴パラメータ

形状特徴パラメータ

まとまった領域では**形状**に関する情報を取得可能

0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	0
0	0	0	0	1	1	1	1	0
0	1	1	0	1	1	1	1	0
0	1	1	0	1	1	1	1	0
0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

○ 重心

○ 面積

○ 周囲長

○ 外接長方形

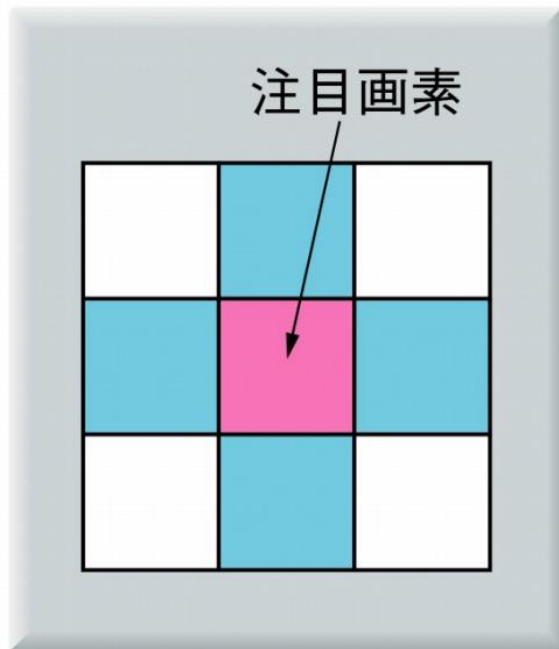
○ 円形度

着目したい領域を白画素化(画素値1)すればよい

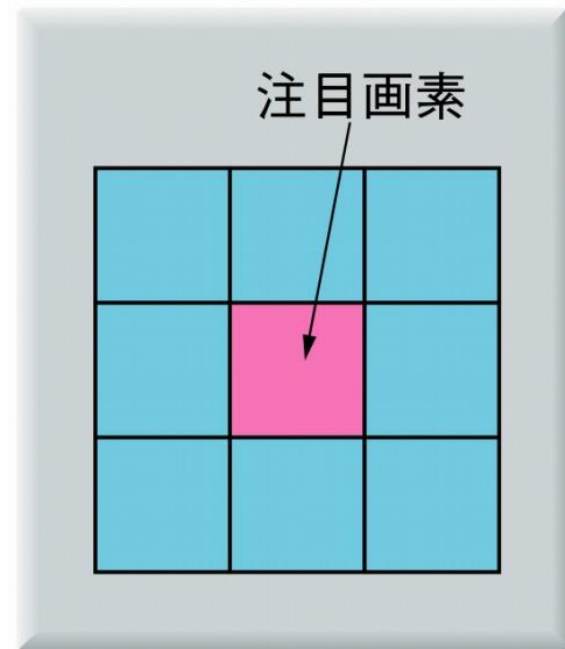
連結の定義

4連結と8連結, 2種類の定義が存在

4連結



8連結



4近傍の画素が連結対象

8近傍の画素が連結対象

連結している画素の集合 → 連結成分

定義方法による連結成分の変化

4連結で連結性を定義

0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0
0	0	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	0
0	0	0	0	1	1	1	1	0
0	1	1	0	1	1	1	1	0
0	1	1	0	1	1	1	1	0
0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

連結成分は3つ

定義方法による連結成分の変化

8連結で連結性を定義

0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0
0	0	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	0
0	0	0	0	1	1	1	1	0
0	1	1	0	1	1	1	1	0
0	1	1	0	1	1	1	1	0
0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

連結成分は2つ

形状特徴パラメータの取得方法

関数「**regionprops**」を2値画像に適用する

□ **regionprops**

入力: 2値画像, **抽出したい特徴**を示す文字列

出力: 各連結成分の形状特徴パラメータ

使用方法: stats = regionprops(2値画像, **抽出したい特徴**);

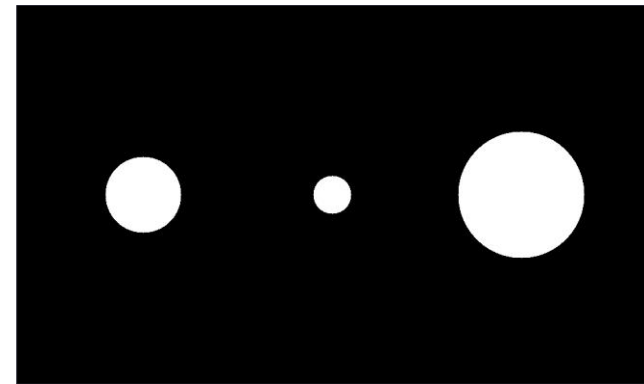
抽出したい特徴を示す文字列の例

- 'Area': 面積(ピクセル数)
- 'Centroid': 重心座標
- 'BoundingBox': 外接長方形

形状特徴パラメータの取得例

3つの円の面積, 重心座標, 外接長方形を取得

```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```



binaryImage

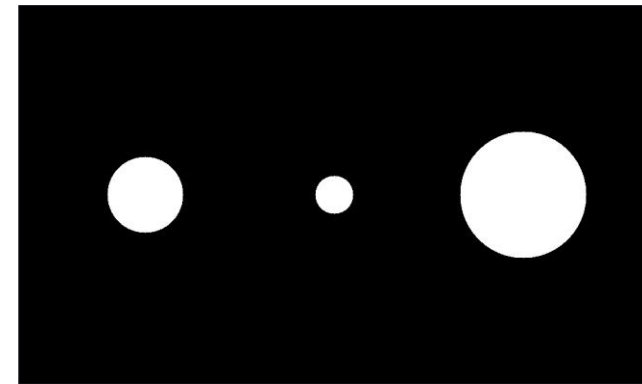
形状特徴パラメータの取得例

変数「stats」に3行3列の行列データが格納される

```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,270.5000,61,61]
3	31417	[801,301]	[700.5000,200.5000,201,201]

変数「stats」の中身



binaryImage

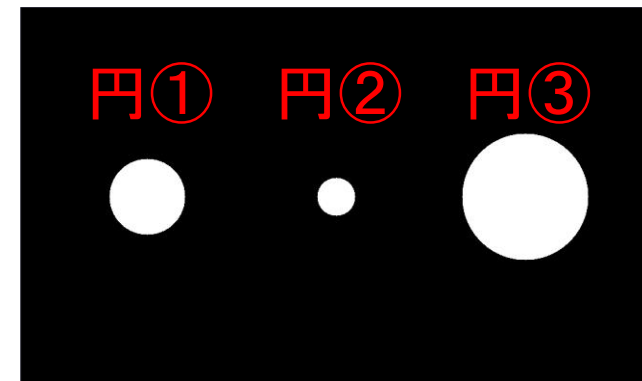
形状特徴パラメータの取得例

各行が2値画像の各円に対応

```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1 円①	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,270.5000,61,61]
3	31417	[801,301]	[700.5000,200.5000,201,201]

変数「stats」の中身



binaryImage

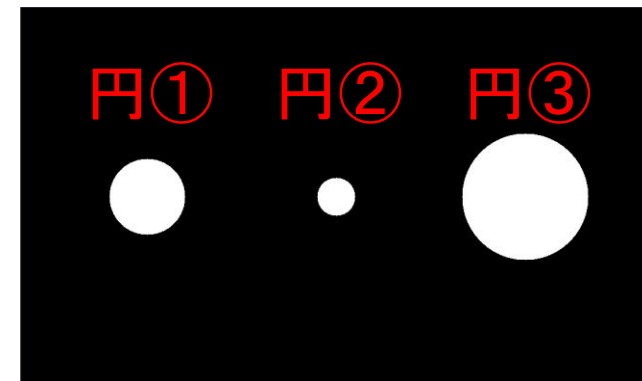
形状特徴パラメータの取得例

各行が2値画像の各円に対応

```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2 円②	2821	[501,301]	[470.5000,270.5000,61,61]
3	31417	[801,301]	[700.5000,200.5000,201,201]

変数「stats」の中身



binaryImage

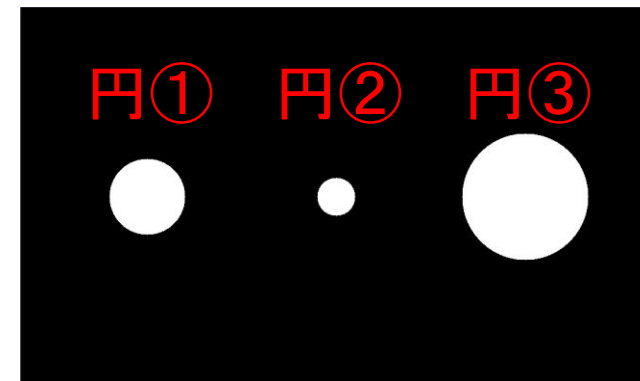
形状特徴パラメータの取得例

各行が2値画像の各円に対応

```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,270.5000,61,61]
3 円③	31417	[801,301]	[700.5000,200.5000,201,201]

変数「stats」の中身



binaryImage

形状特徴パラメータの取得例

各列が取得した形状特徴パラメータに対応

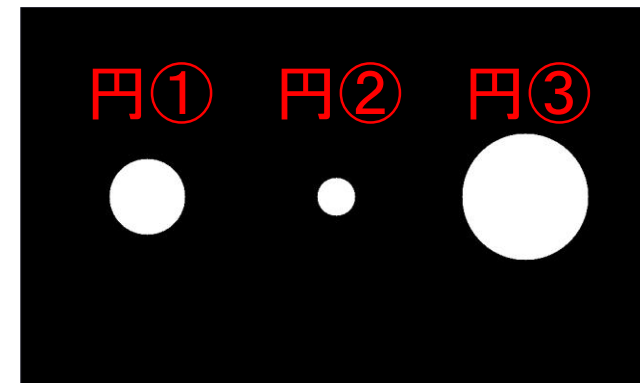
```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

Area ⇒ 白画素のpixel数

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,270.5000,61,61]
3	31417	[801,301]	[700.5000,200.5000,201,201]

面積

変数「stats」の中身



binaryImage

形状特徴パラメータの取得例

各列が取得した形状特徴パラメータに対応

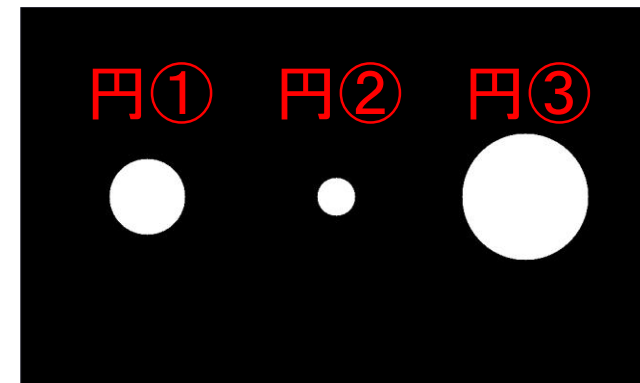
```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

Centroid \Rightarrow [重心 x 座標, 重心 y 座標]

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,270.5000,61,61]
3	31417	[801,301]	[700.5000,200.5000,201,201]

重心座標

変数「stats」の中身



binaryImage

形状特徴パラメータの取得例

各列が取得した形状特徴パラメータに対応

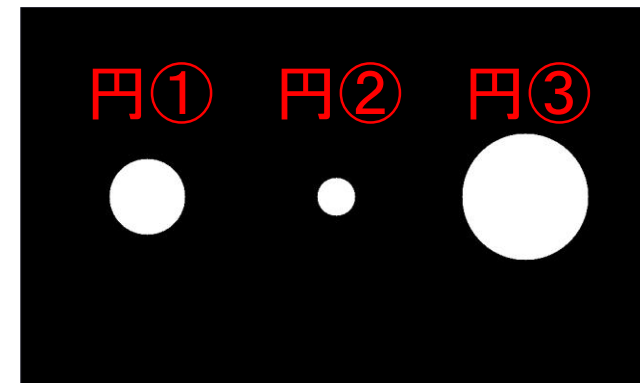
```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

BoundingBox ⇒ [左上の角 x 座標, 左上の角 y 座標, 幅, 高さ]

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,270.5000,61,61]
3	31417	[801,301]	[700.5000,200.5000,201,201]

外接長方形

変数「stats」の中身



binaryImage

各形状特徴パラメータの格納例

円③の面積を変数に格納

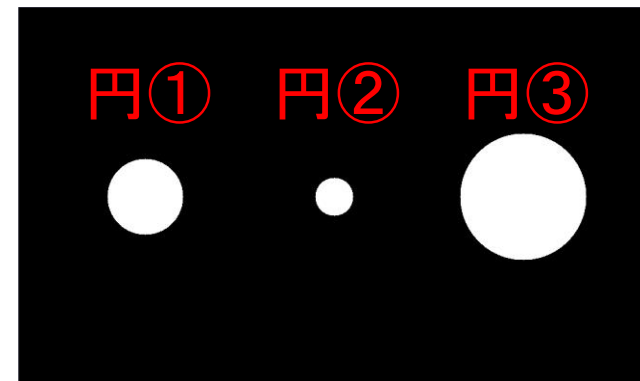
```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

```
area_circle3 = stats(3).Area;
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,270.5000,61,61]
3	31417	[801,301]	[700.5000,200.5000,201,201]

円③の面積

変数「stats」の中身



binaryImage

各形状特徴パラメータの格納例

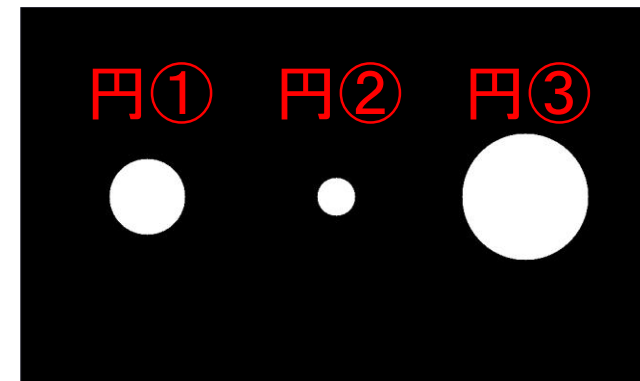
円①の外接長方形を変数に格納

```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

```
boundingbox_circle1 = stats(1).BoundingBox;
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,370.5000,61,61]
3	31417	[801,301]	[700.5000,200.5000,201,201]

変数「stats」の中身



binaryImage

各形状特徴パラメータの格納例

円②の重心 x 座標を変数に格納

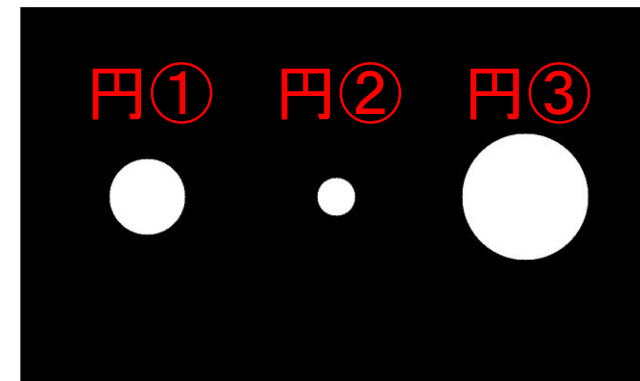
```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

```
centroid_x_circle2 = stats(2).Centroid(1);
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11239	[140.5000, 240.5000]	[121, 121]
2	2821	[501, 301]	[470.5000, 270.5000, 61, 61]
3	31417	[801, 301]	[700.5000, 200.5000, 201, 201]

円②の重心 x 座標

変数「stats」の中身



binaryImage

各形状特徴パラメータの格納例

円②の重心y座標を変数に格納

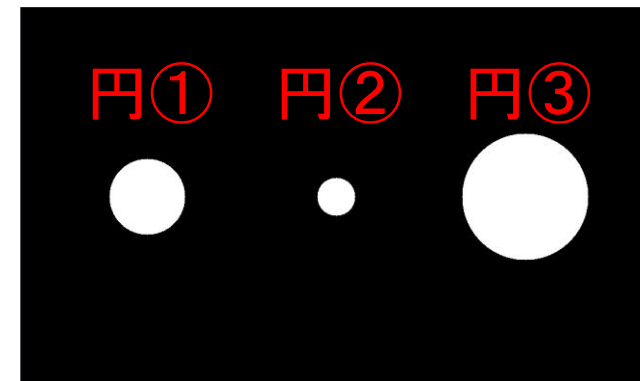
```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');
```

```
centroid_y_circle2 = stats(2).Centroid(2);
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11285	[500,240]	[500,121,121]
2	2821	[501,301]	[470,500,270,500,61,61]
3	31417	[801,301]	[700,500,200,500,201,201]

円②の重心y座標

変数「stats」の中身



binaryImage

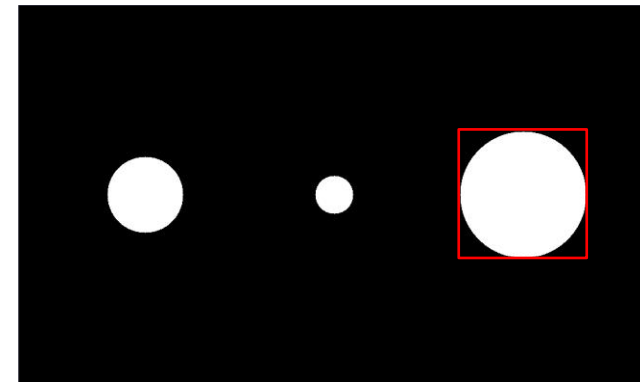
取得した形状特徴パラメータの利用例

関数「rectangle」により円③を赤枠でかこむ

```
stats = regionprops(binaryImage, 'Area', 'Centroid', 'BoundingBox');  
imshow(binaryImage);  
rectangle('Position', stats(3).BoundingBox, 'EdgeColor', 'red')
```

stats			
3x1 struct 3 フィールド			
フィールド	Area	Centroid	BoundingBox
1	11289	[201,301]	[140.5000,240.5000,121,121]
2	2821	[501,301]	[470.5000,270.5000,81,81]
3	31417	[801,301]	[700.5000,200.5000,201,201]

変数「stats」の中身



binaryImage