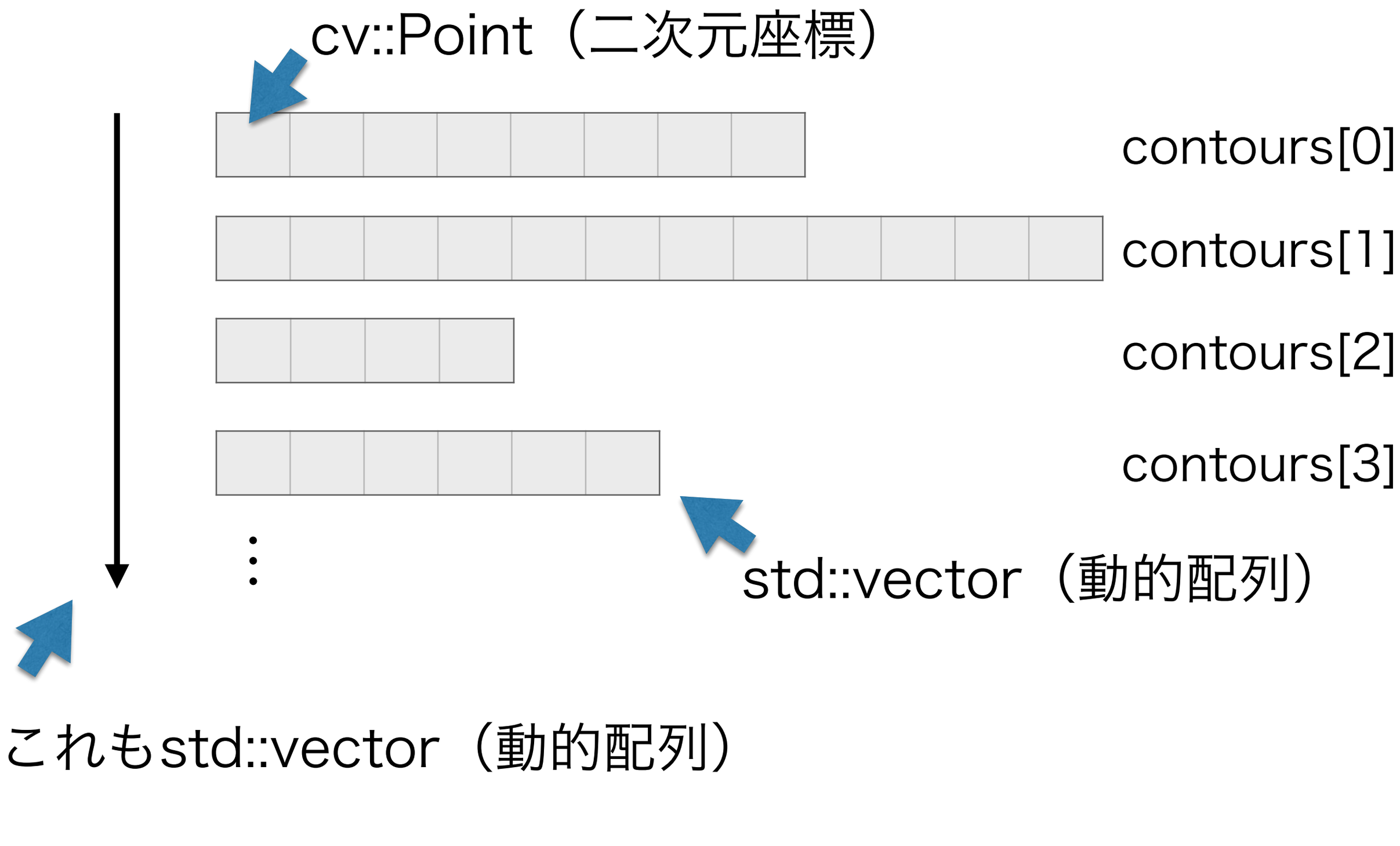
メモ

輪郭の座標リストの宣言

```
std::vector< std::vector< cv::Point > > contours;
```

- std::vector: 動的配列
  - 配列の大きさが固定されない（可変長）
- cv::Point: 座標
  - 二次元の座標
- **座標位置の 並び（＝輪郭）**をリスト化
  (cv::Point) の (std::vector) の (std::vector)

contoursの構造



メモ

輪郭追跡関数

- findContours：入力二値画像の輪郭を追跡する

```
cv::findContours(二値画像, 輪郭, 追跡モード, 輪郭近似手法);
```

- 今回の記載例

```
//7. 輪郭追跡(New!)
cv::findContours(tmp_img, contours,
   cv::RETR_LIST, cv::CHAIN_APPROX_NONE);
```

- cv::RETR_LIST: すべての輪郭追跡、リスト出力
- cv::CHAIN_APPROX_NONE: 8近傍、近似なし

輪郭追跡結果がcontoursに格納される

メモ

各輪郭へのアクセス方法

- for文でcontoursの各要素（各輪郭）にアクセス

```
//8. 輪郭の描画(New!)
for (int i=0; i<contours.size(); i++) {


}
```

- contours.size()：動的配列contoursのサイズ
  （輪郭が何個あるか）

  ✳contours[i]　でｉ番目の輪郭にアクセスできる
  （このプログラムではこの記載は使いません）

メモ

輪郭描画関数

- drawContours：輪郭を描画する

```
cv::drawContours(出力画像, 輪郭情報, 輪郭番号,
                                    輪郭の色, 描画用の線幅);
```

※線幅を負の値にすると輪郭内部も塗りつぶす

- 今回の記載例

```
//輪郭の描画
cv::drawContours(dst_img, contours, i,
                                CV_RGB(255, 0, 255), 3);
```

※CV_RGBで色を指定　B=R=255でマゼンタとしている

- この描画文をfor文の中に入れる

➡ 輪郭番号を指定して（この場合 i）描画する

メモ

周囲長と面積

・輪郭に対する周囲長と面積を求める関数

```
cv::arcLength(輪郭, 閉輪郭か否か); //出力が周囲長
cv::contourArea(輪郭);             //出力が面積
```

・記載例

```
double L,S;
//周囲長（輪郭の長さ）
L = cv::arcLength(contours[i], true);
//面積
S = cv::contourArea(contours[i]);
```

i番目の輪郭を指定

メモ

円形度

・図形領域がどれだけ円に近いか

・領域の面積をＳ，周囲長をＬとすると
　円形度Ｒは以下の式で求められる：

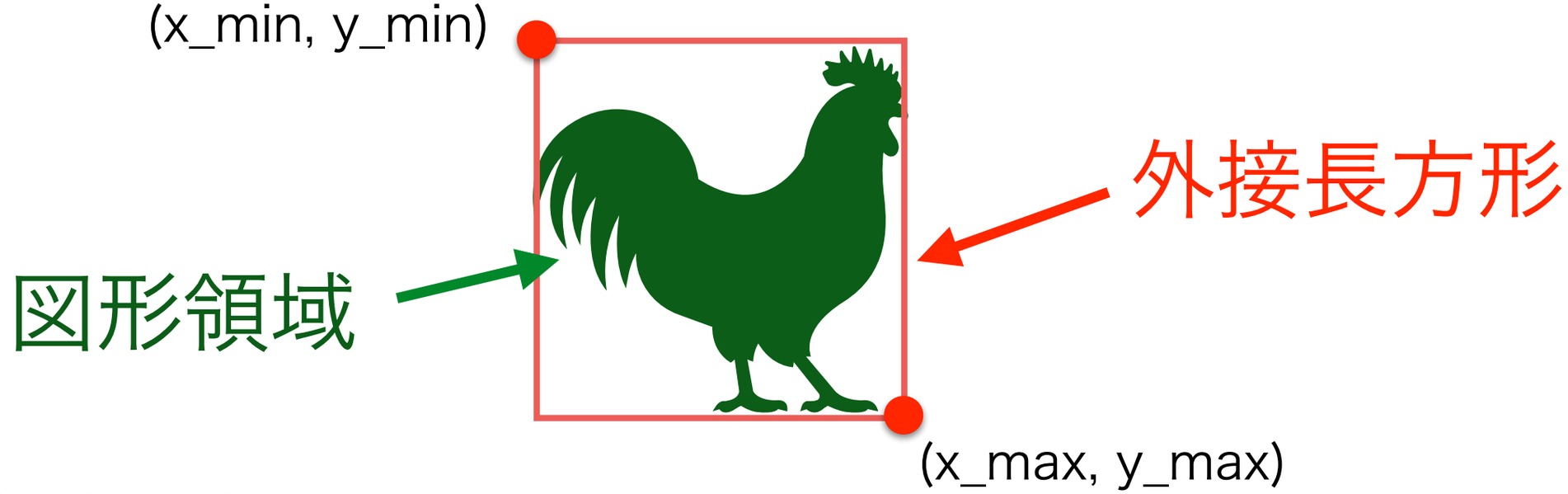
$$R = \frac{4\pi S}{L^2}$$

・理想的な円のとき，R=1

メモ

外接長方形 (バウンディングボックス)

・図形領域に接する最小の長方形



・求め方（原理）

- 図形領域の輪郭を求める
- 輪郭各画素の座標位置(x,y)からxとyの最小値・最大値を求める(x_min, x_max, y_min, y_maxとする)

➡ 外接長方形の左上の座標は(x_min, y_min)
右下の座標は(x_max, y_max)

メモ

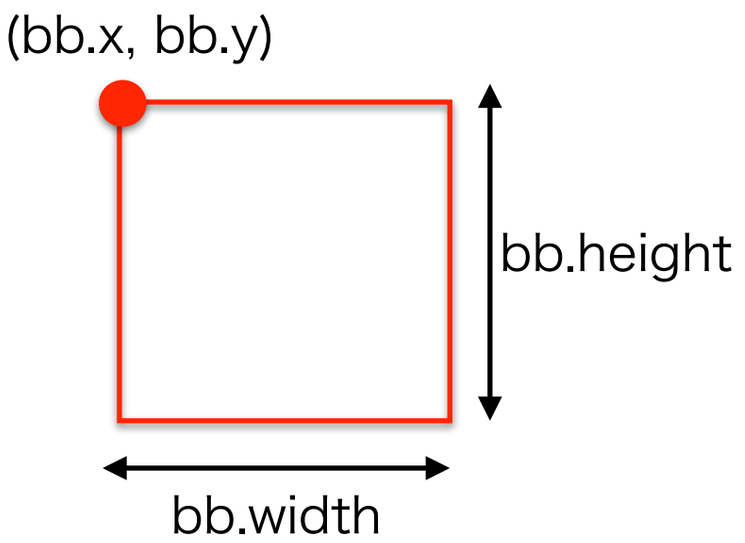
外接長方形用の関数

・外接長方形を求める関数

```
cv::boundingRect(輪郭); //出力が外接長方形
```

- 出力：構造体 Rect で定義された変数

➡ Rectのメンバ変数.x, .yは左上の座標,
.width, .heightは幅と高さを表す



・記載例

```
cv::Rect bb;
//外接長方形
bb = cv::boundingRect(contours[i]);
```

メモ

長方形描画関数

- rectangle：長方形を描画する

```
cv::rectangle(出力画像, 長方形, 色, 線幅);
```

- 長方形の指定方法　※線幅を負の値にすると内部も塗りつぶす

✓ 構造体Rectの変数（前ページのbb）

✓ 左上の頂点と右下の頂点(cv::Point型)を並べて記載

- 記載例

```
//外接長方形の描画
cv::rectangle(dst_img, bb,
                        CV_RGB(255, 0, 255), 3);
```

※CV_RGBで色を指定　B=R=255でマゼンタとしている