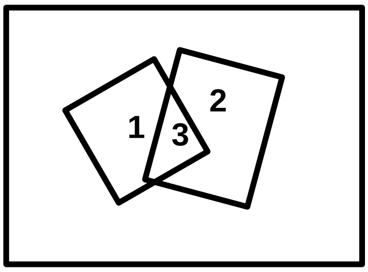
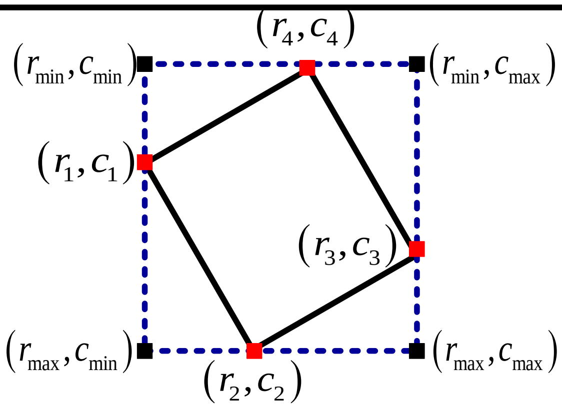
Aims:

Compute Overlap Rate Between two quadrilateral



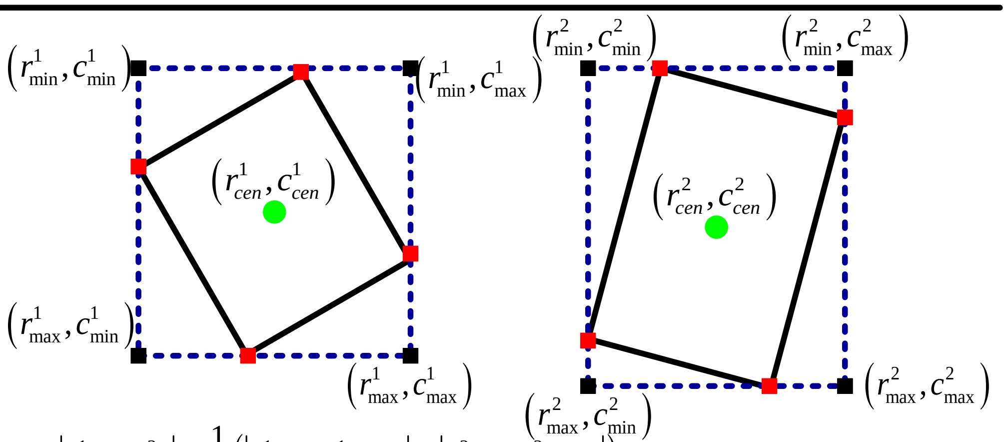
$$overlapRate = \frac{Area(3)}{Area(1)+Area(2)+Area(3)}$$

Step (1): Compute Bounding Box



$$r_{\min} = \min\{r_1, r_2, r_3, r_4\}$$
$$c_{\min} = \min\{c_1, c_2, c_3, c_4\}$$
$$r_{\max} = \max\{r_1, r_2, r_3, r_4\}$$
$$c_{\max} = \max\{c_1, c_2, c_3, c_4\}$$

Step (2): Overlap or Not

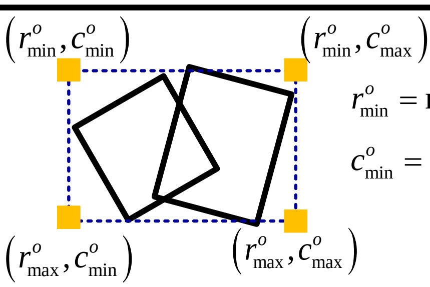


if
$$\left|r^1_{cen} - r^2_{cen}\right| > \frac{1}{2}\left(\left|r^1_{\max} - r^1_{\min} + 1\right| + \left|r^2_{\max} - r^2_{\min} + 1\right|\right) \textbf{ or}$$
$$\left|c^1_{cen} - c^2_{cen}\right| > \frac{1}{2}\left(\left|c^1_{\max} - c^1_{\min} + 1\right| + \left|c^2_{\max} - c^2_{\min} + 1\right|\right)$$

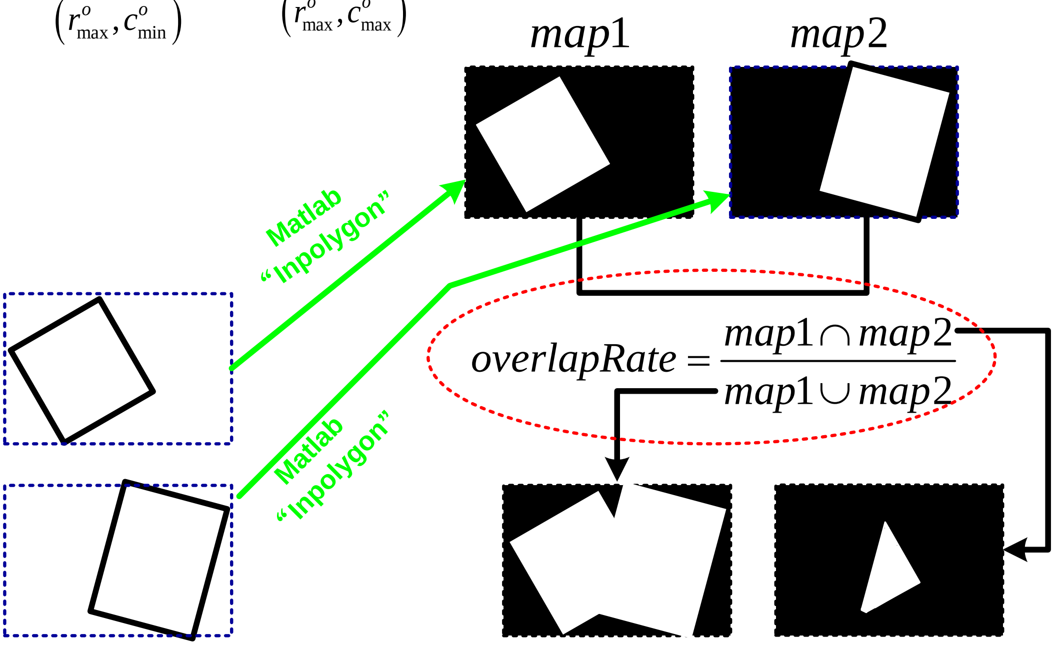
$overlapRate = 0$

else **Go to Step (3)**

Step (3): Compute Overlap Rate



$$r^o_{\min} = \min\{r^1_{\min}, r^2_{\min}\}; r^o_{\max} = \min\{r^1_{\max}, r^2_{\max}\}$$
$$c^o_{\min} = \min\{c^1_{\min}, c^2_{\min}\}; c^o_{\max} = \min\{c^1_{\max}, c^2_{\max}\}$$



inpolygon

Points inside polygonal region

Syntax

```
IN = inpolygon(X,Y,xv,yv)
[IN ON] = inpolygon(X,Y,xv,yv)
```

Description

`IN = inpolygon(X,Y,xv,yv)` returns a matrix `IN` the same size as `X` and `Y`. Each element of `IN` is assigned the value 1 or 0 depending on whether the point `(X(p,q),Y(p,q))` is inside the polygonal region whose vertices are specified by the vectors `xv` and `yv`. In particular:

`IN(p,q) = 1`	If `(X(p,q),Y(p,q))` is inside the polygonal region or on the polygon boundary
`IN(p,q) = 0`	If `(X(p,q),Y(p,q))` is outside the polygonal region

`[IN ON] = inpolygon(X,Y,xv,yv)` returns a second matrix `ON` the same size as `X` and `Y`. Each element of `ON` is assigned the value 1 or 0 depending on whether the point `(X(p,q),Y(p,q))` is on the boundary of the polygonal region whose vertices are specified by the vectors `xv` and `yv`. In particular:

`ON(p,q) = 1`	If `(X(p,q),Y(p,q))` is on the polygon boundary
`ON(p,q) = 0`	If `(X(p,q),Y(p,q))` is insic

INPOLYGON True for points inside or on a polygonal region.
IN = INPOLYGON(X,Y,XV,YV) returns a matrix IN the size of X and Y.
IN(p,q) = 1 if the point (X(p,q), Y(p,q)) is either strictly inside or
on the edge of the polygonal region whose vertices are specified by the
vectors XV and YV; otherwise IN(p,q) = 0.
[IN ON] = INPOLYGON(X,Y,XV,YV) returns a second matrix, ON,
which is the size of X and Y. ON(p,q) = 1 if the point (X(p,q), Y(p,q)) is on
the edge of the polygonal region; otherwise ON(p,q) = 0.
INPOLYGON supports non-convex and self-intersecting polygons.
The function also supports multiply-connected or disjoint polygons;
however, the distinct edge loops should be separated by NaNs. In the
case of multiply-connected polygons, the external and internal loops
should have opposite orientations; for example, a counterclockwise
outer loop and clockwise inner loops or vice versa.

Example 1:

```
% Self-intersecting polygon
xv = rand(6,1); yv = rand(6,1);
xv = [xv ; xv(1)]; yv = [yv ; yv(1)];
x = rand(1000,1); y = rand(1000,1);
in = inpolygon(x,y,xv,yv);
plot(xv,yv,x(in),y(in),'.r',x(~in),y(~in),'.b')
```

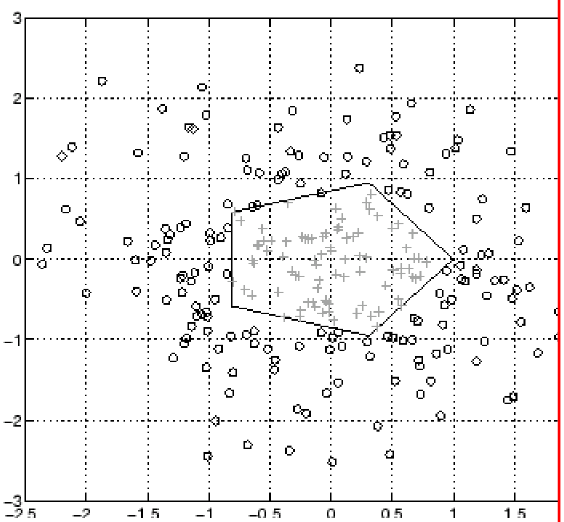
Example 2:

```
% Multiply-connected polygon - a square with a square hole.
% Counterclockwise outer loop, clockwise inner loop.
xv = [0 3 3 0 0 NaN 1 1 2 2 1];
yv = [0 0 3 3 0 NaN 1 2 2 1 1];
x = rand(1000,1)*3; y = rand(1000,1)*3;
in = inpolygon(x,y,xv,yv);
plot(xv,yv,x(in),y(in),'.r',x(~in),y(~in),'.b')
```

Class support for inputs X,Y,XV,YV:
float: double, single

Examples

```
L = linspace(0,2.*pi,6); xv = cos
xv = [xv ; xv(1)]; yv = [yv ; yv(
x = randn(250,1); y = randn(250,1
in = inpolygon(x,y,xv,yv);
plot(xv,yv,x(in),y(in),'r+',x(~in
```



inmem