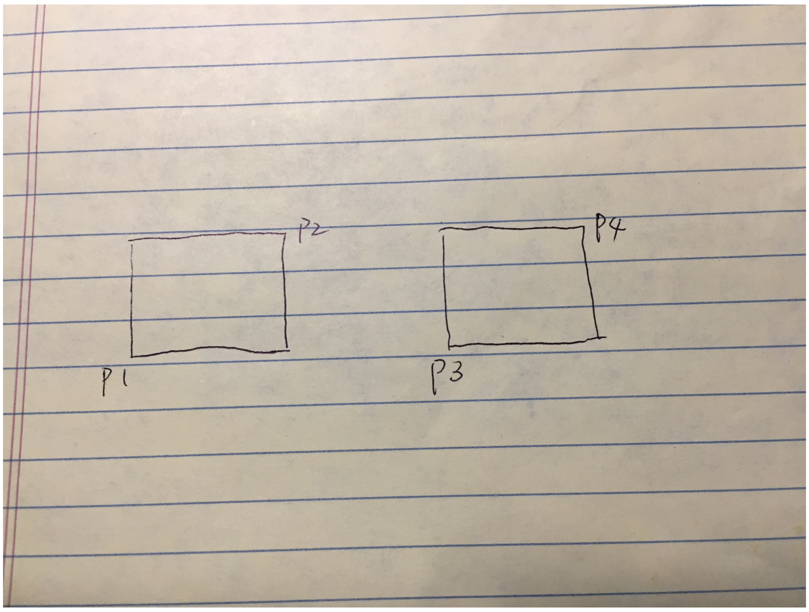
In order to trim a rectangle, we need to start thinking how we can know if two rectangles are overlapping. For example, if I have two rectangles, like I showed below,



if these two rectangles are not overlapping, (p2.y < p3.y) or (p4.y < p1.y) or (p2.x < p3.x) or (p4.x < p1.x). So if two rectangles are overlapping,

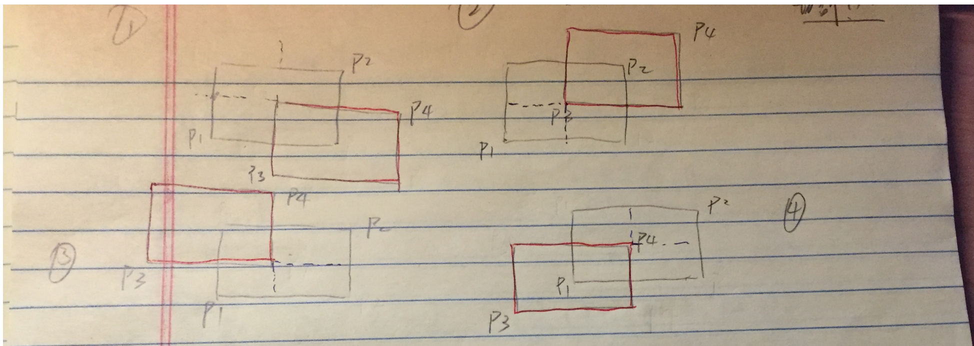
$$\neg[(p2.y \le p3.y) \vee (p1.y \ge p4.y) \vee (p2.x \le p3.x) \vee (p1.x \ge p4.x)]$$

In other words,

$$(p2.y > p3.y) \wedge (p1.y < p4.y) \wedge (p2.x > p3.x) \wedge (p1.x < p4.x)$$

Then we will say it's overlapping if those four conditions are met.

Thus, if we want to know how to trim the rectangle, we have to know relative position of two rectangles. This mainly depends on p1.x vs p3.x, p2.x vs p4.x, p1.y vs p3.y, and p2.y vs p4.y. If we know the result of these four group comparisons, we will know their relative positions and candidate rectangle position. Overall, there are 16 cases, like I showed below.



and

1 P1.x < P3.x ~~and~~ P2.x < P4.x and P1.y > P3.y and P2.y > P4.y

2 P1.x < P3.x and P2.x < P4.x and P1.y < P3.y and P2.y < P4.y

3 P1.x > P3.x and P2.x > P4.x and P1.y < P3.y and P2.y < P4.y

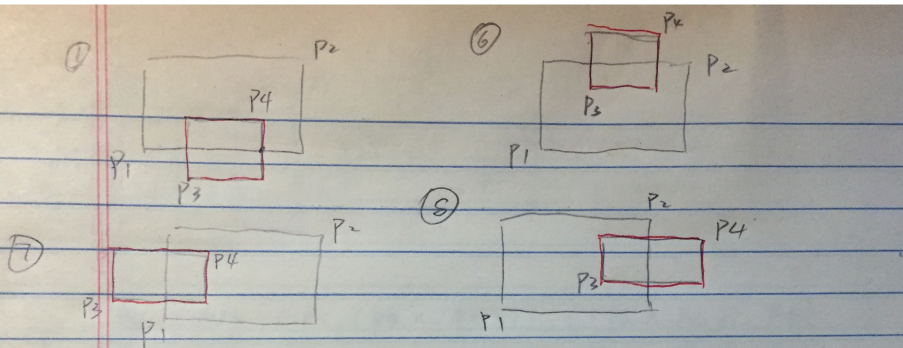
4 P1.x > P3.x and P2.x > P4.x and P1.y > P3.y and P2.y > P4.y

1 2 rec: ((P2.x, P3.y), P4), (P3, (P4.x, P1.y))

2 2 rec: ((P3.x, P2.y), P4), ((P2.x, P3.y), P4)

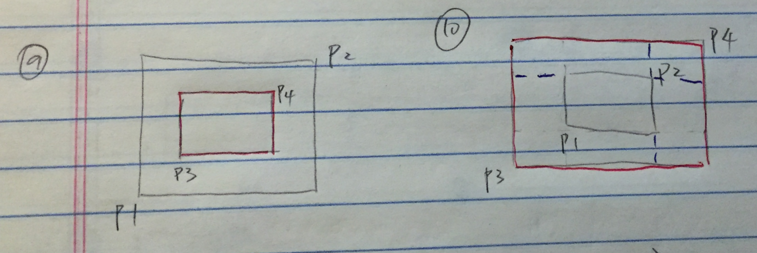
3 2 rec: (P3, (P1.x, P4.y)), ((P3.x, P2.y), P4)

4 2 rec: (P3, (P1.x, P4.y)), (P3, (P4.x, P1.y))



5 (P3, (P4.x, P1.y))　　7 (P3, (P1.x, P4.y))

6 ((P3.x, P2.y), P4)　　8 ((P2.x, P3.y), P4)



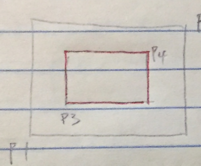
9 X　　10 4 recs: ((P3.x, P2.y), P4)

((P2.x, P3.y), P4)

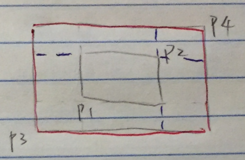
⑤ (P3, (P4.x, P1.y)) ⑦ (P3, (P1.x, P4.y))

⑥ ((P3.x, P2.y), P4) ⑧ ((P2.x, P3.y), P4)

⑨



⑩



⑨ X ⑩ 4 recs: ((P3.x, P2.y), P4)

((P2.x, P3.y), P4)

(P3, (P4.x, P1.y))

(P3, (P1.x, P4.y))

⑤ P1.x < P3.x and P2.x > P4.x and P1.y > P3.y and P2.y > P4.y

⑥ P1.x < P3.x and P2.x > P4.x and P1.y < P3.y and P2.y < P4.y

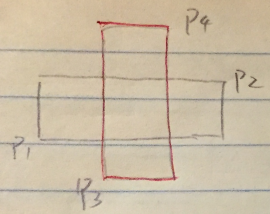
⑦ P1.x > P3.x and P2.x > P4.x and P1.y < P3.y and P2.y > P4.y

⑧ P1.x < P3.x and P2.x < P4.x and P1.y < P3.y and P2.y > P4.y

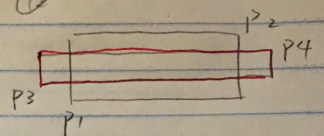
⑨ P1.x < P3.x and P2.x > P4.x and P1.y < P3.y and P2.y > P4.y

⑩ P1.x > P3.x and P2.x < P4.x and P1.y > P3.y and P2.y < P4.y

⑪



⑫



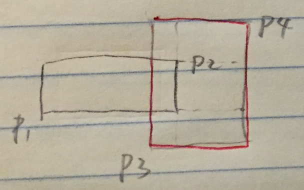
⑪ 2 recs: (P3, (P4.x, P1.y)), ((P3.x, P2.y), P4)

⑫ 2 recs: (P3, (P1.x, P4.y)), ((P2.x, P3.y), P4)

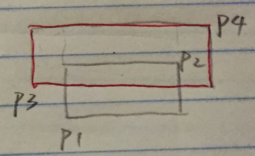
~~⑪ P1.x < P3.x and P2.x > P4.x and P1.y > P3.y and P2.y < P4.y~~

⑫ P1.x > P3.x and P2.x < P4.x and P1.y < P3.y and P2.y > P4.y

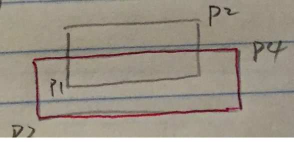
⑬



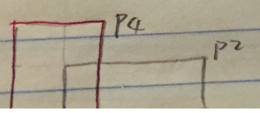
⑭



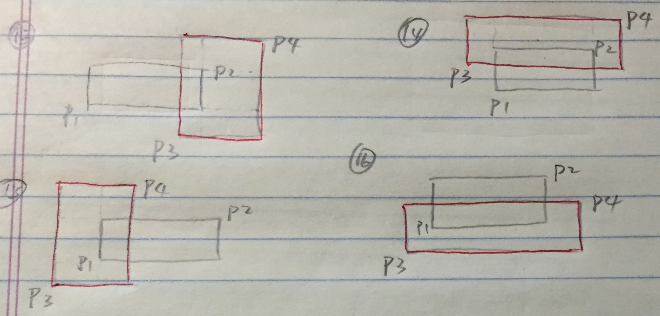
⑯



⑮



② P1.x > P3.x and P2.x < P4.x and P1.y < P3.y and P2.y > P4.y



⑬ P1.x < P3.x and P2.x < P4.x and P1 y > P3.y and P2.y < P4.y

3 recs: ((P3.x, P2.y), P4), (P3, (P4.x, P1.y)), ((P2.x, P3.y), P4)

⑭ P1.x > P3.x and P2.x < P4.x and P1.y < P3.y and P2.y < P4.y

3 recs: (P3, (P1.x, P4.y)), ((P3.x, P2.y), P4), ((P2.x, P3.y), P4)

⑮ P1.x > P3.x and P2.x > P4.x and P1.y > P3.y and P2.y < P4.y

3 recs: ((P3.x, P2.y), P4), (P3, (P1.x, P4.y)), (P3, (P4.x, P1.y))

⑯ P1.x > P3.x and P2.x < P4.x and P1.y > P3.y and P2.y > P4.y

3 recs: (P3, (P1.x, P4.y)), (P3, (P4.x, P1.y)), ((P2.x, P3.y), P4)