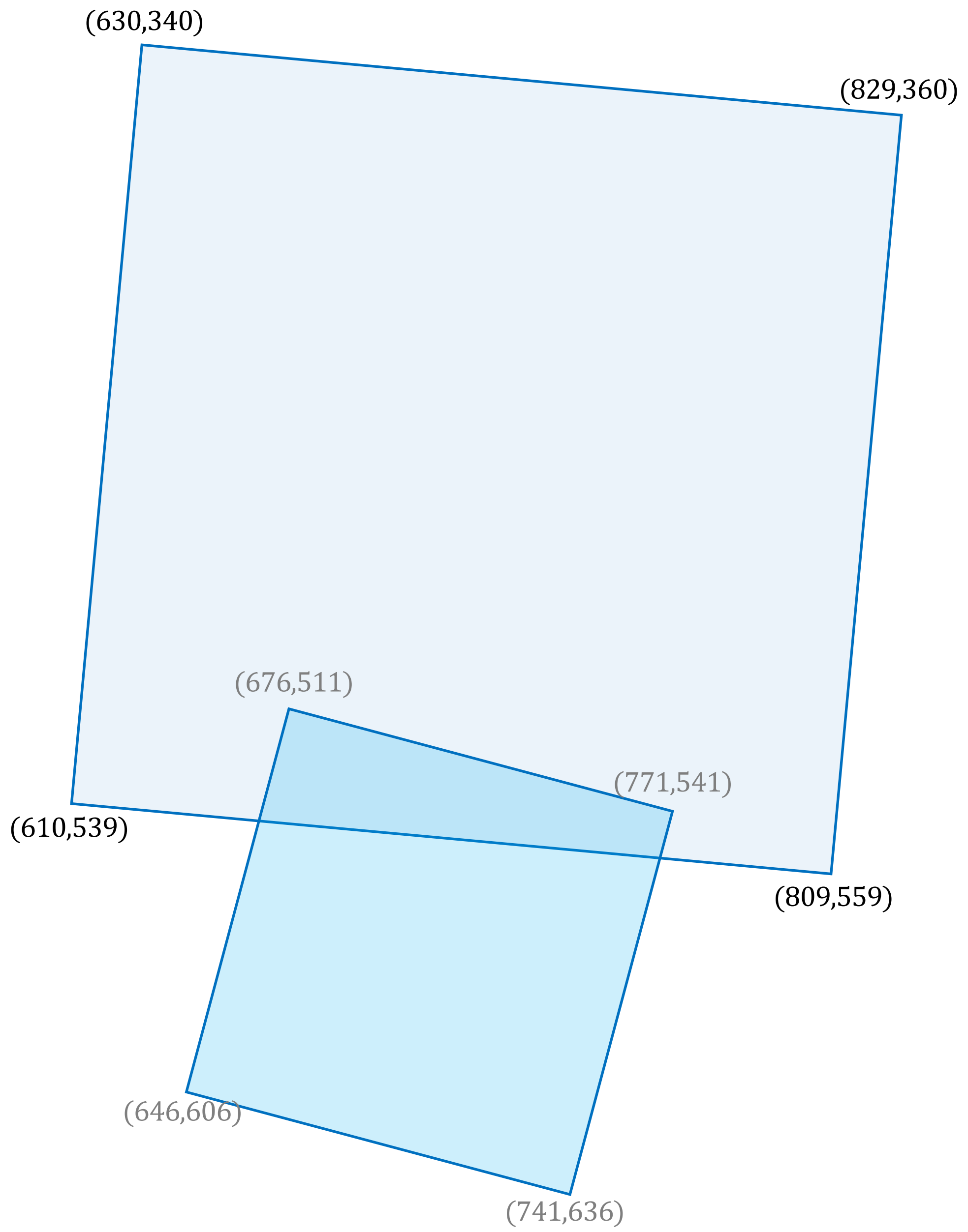
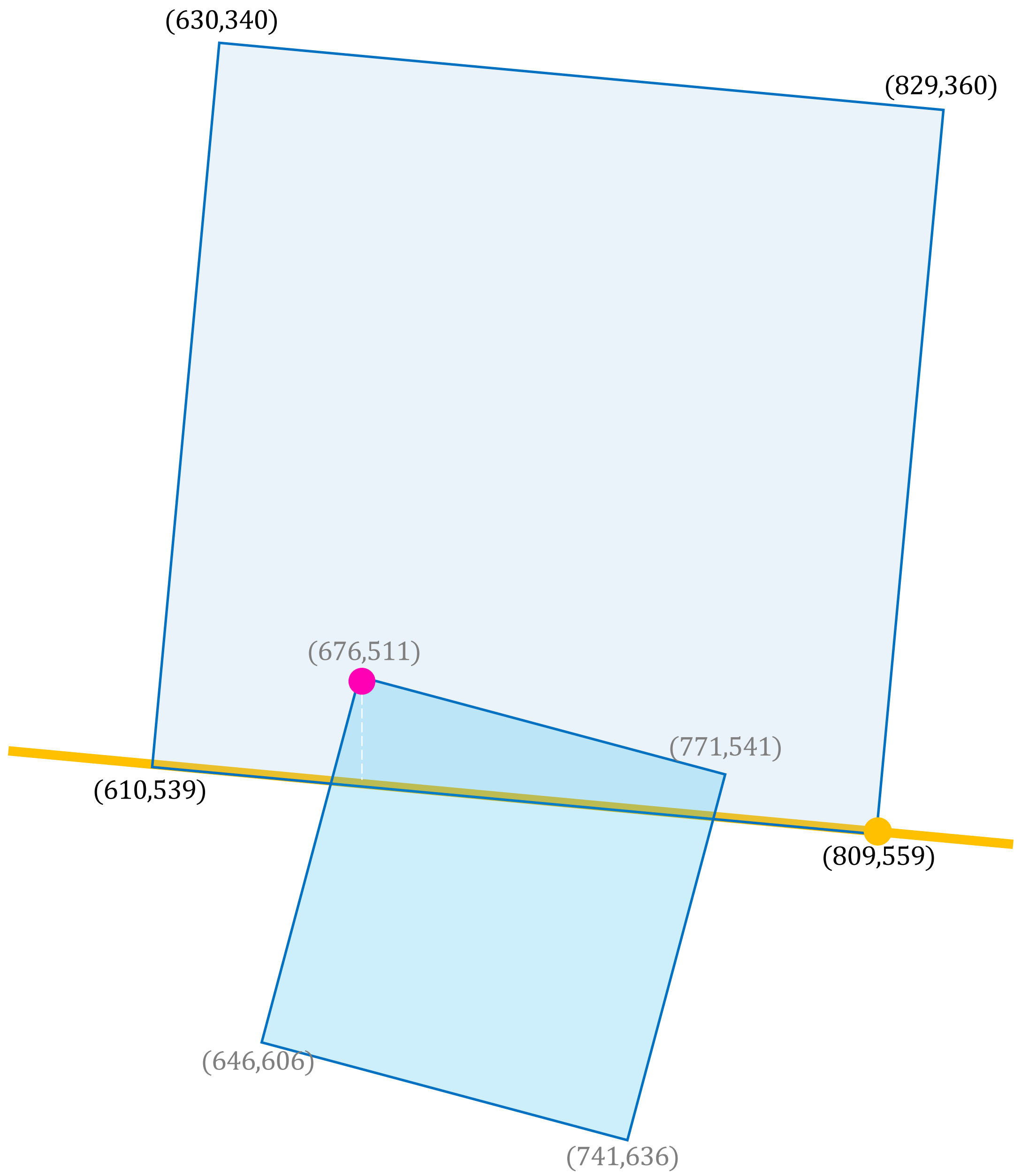
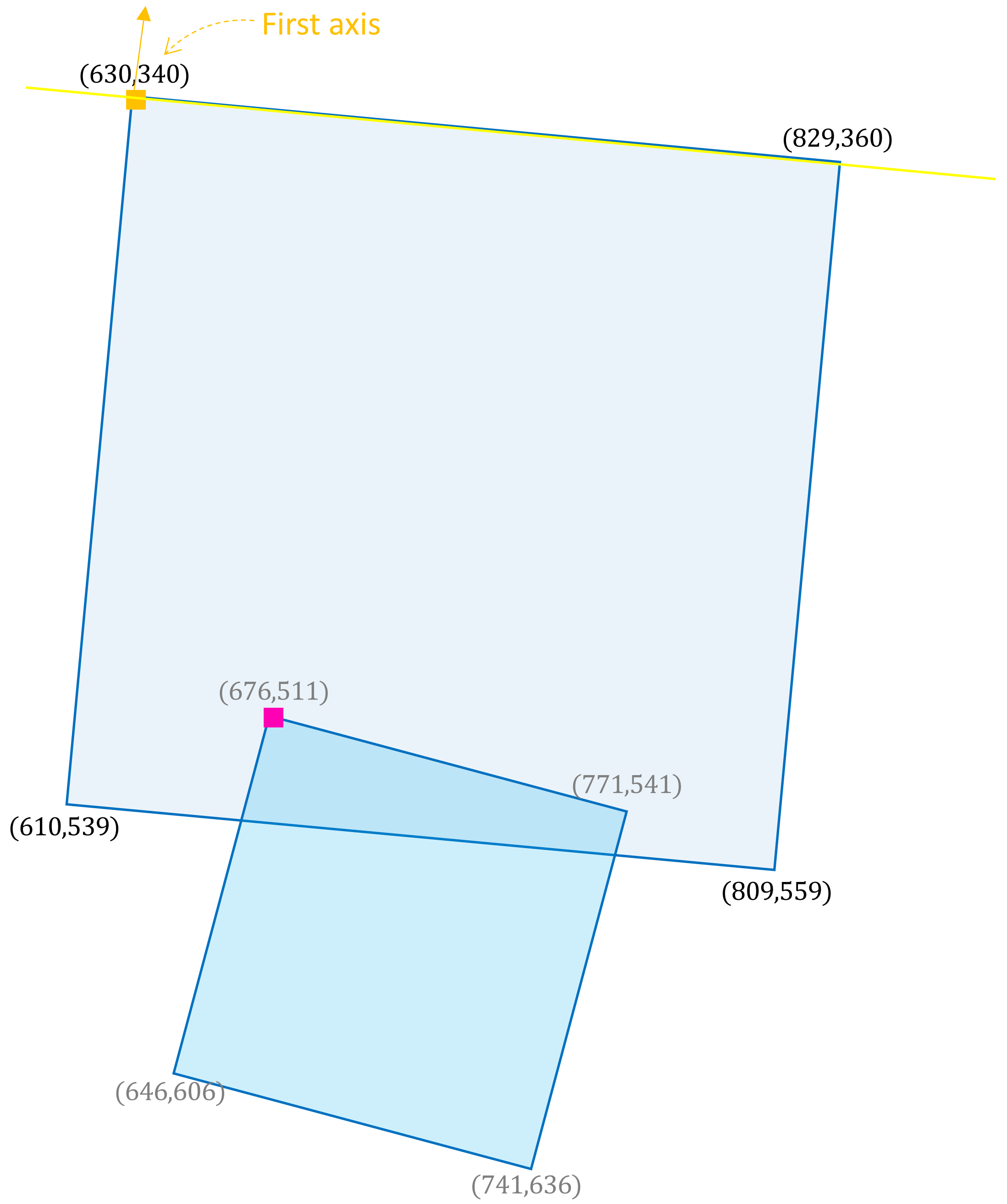


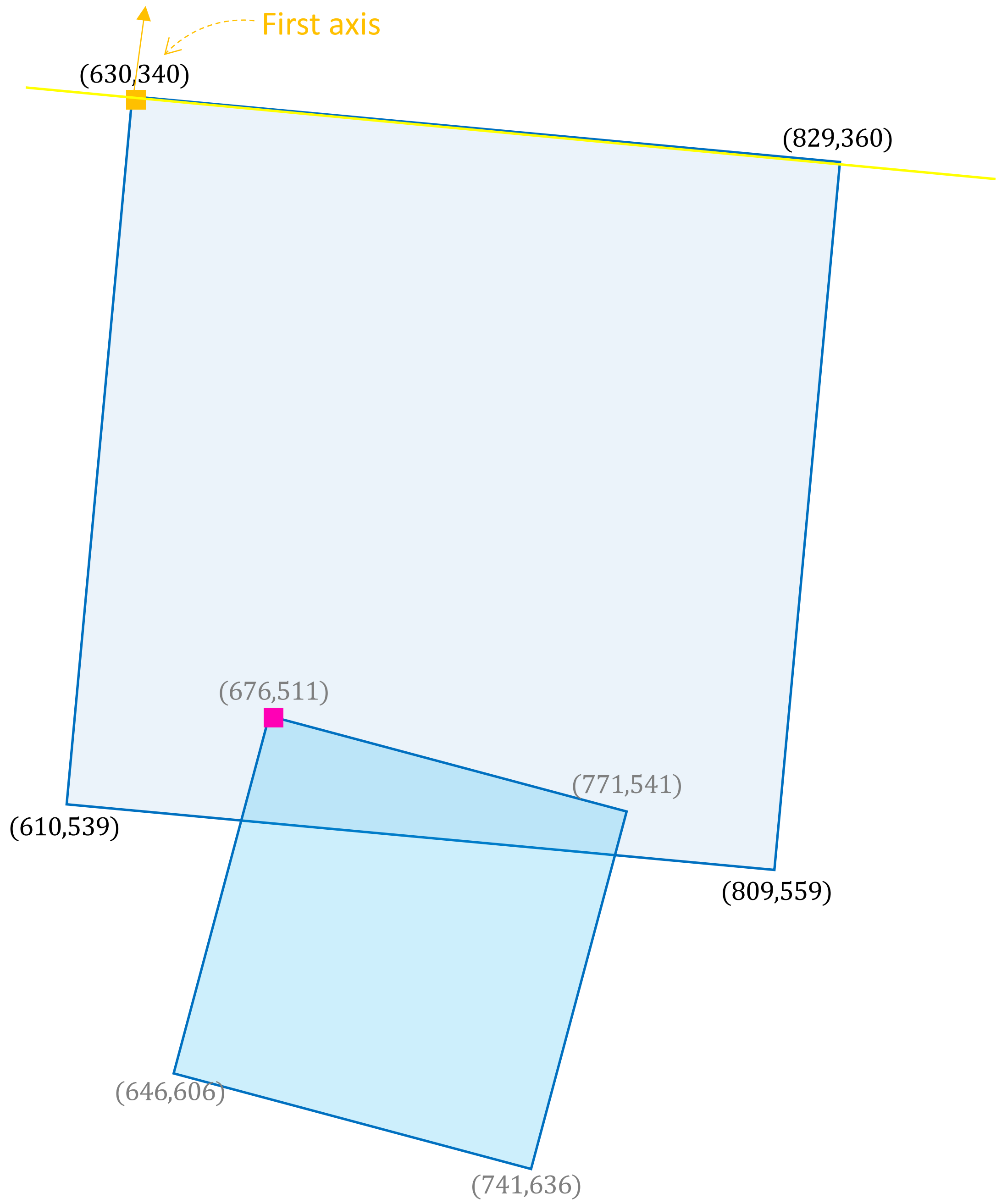
- Let’s try to visualize, step by step, how the algorithm to find the **axis of minimum separation** works.
- We’ll loop **all** the axis of the big box; and for each axis, we’ll loop and test with all the vertices of the small box.
- We want the **axis** of **minimum separation**, and for that we need to also compute the **point** of maximum penetration.



- Let's try to visualize, step by step, how the algorithm to find the **axis of minimum separation** works.
- We'll loop **all** the axis of the big box; and for each axis, we'll loop and test with all the vertices of the small box.
- We want the **axis** of **minimum separation**, and for that we need to also compute the **point** of maximum penetration.
- Intuitively, we know what they should be; but let's try to simulate how the computer executes the algorithm step by step.



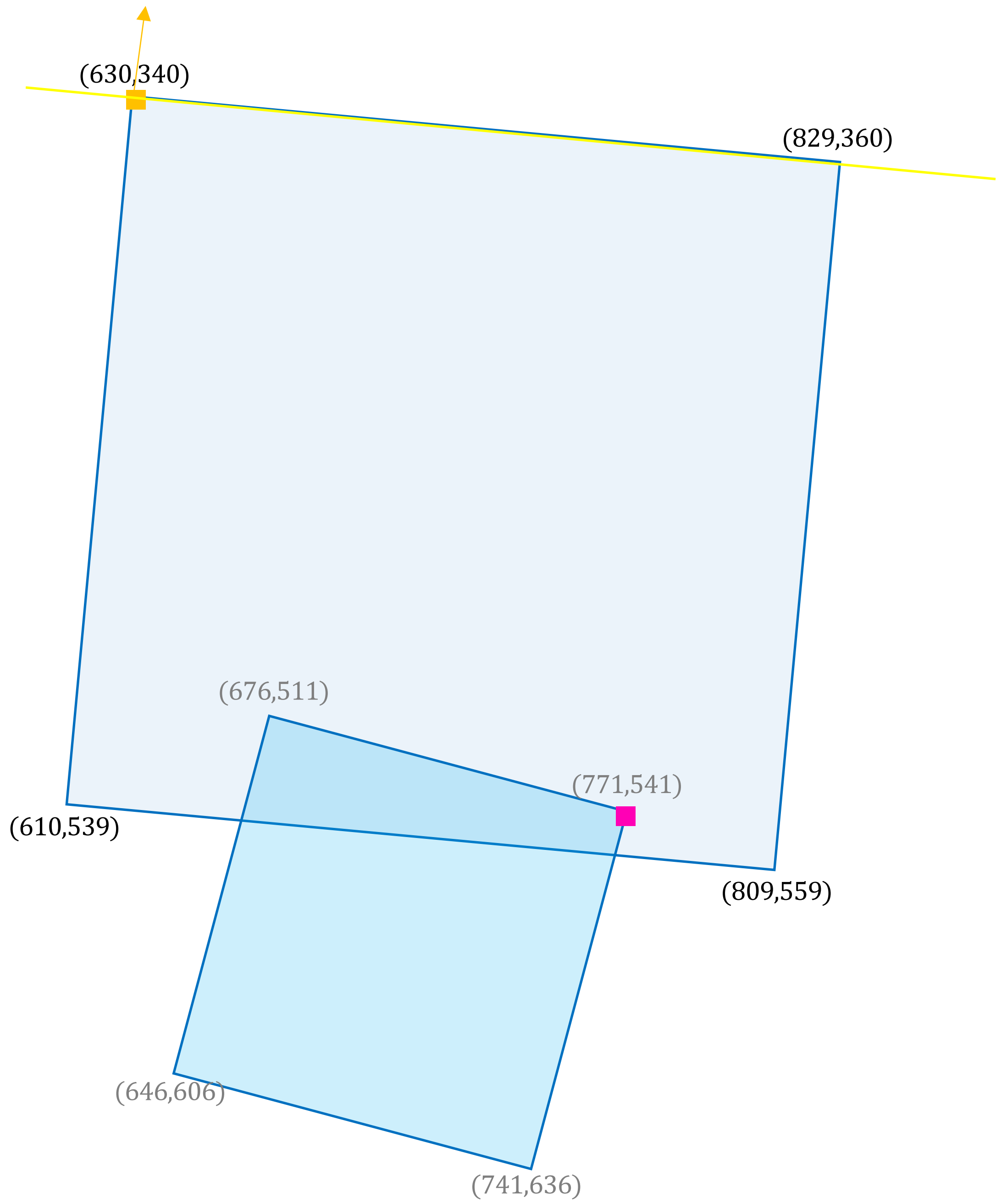
- We start with the **first vertex/axis** of the first object, and we proceed to loop all the vertices of the second object. We *project* them *(dot product)* and keep the **minimum** projection value.



proj: -165

minProj: -165

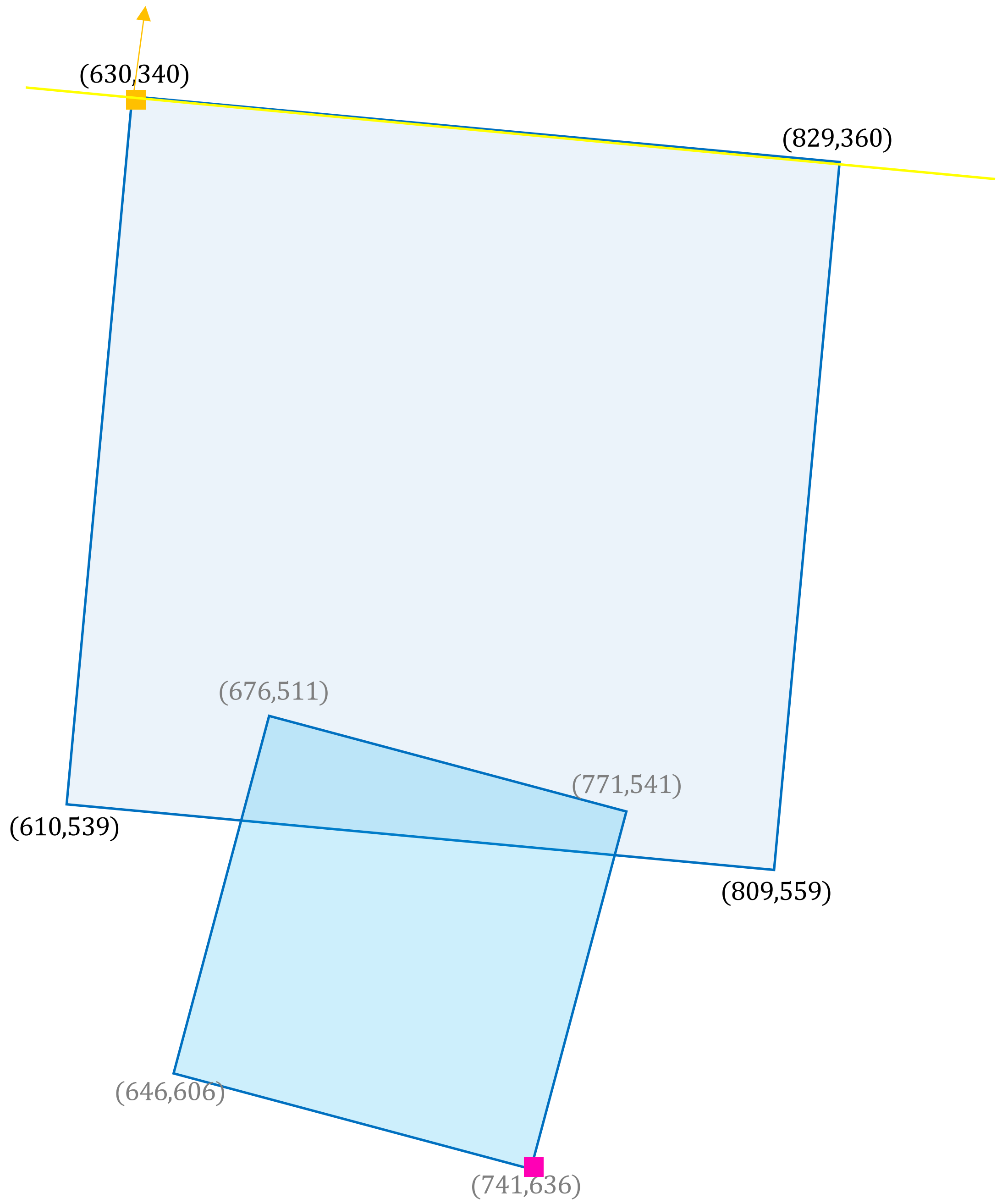
- We start with the **first vertex/axis** of the first object, and we proceed to loop all the vertices of the second object. We project them *(dot product)* and keep the **minimum** projection value.



proj: -185

minProj: -185

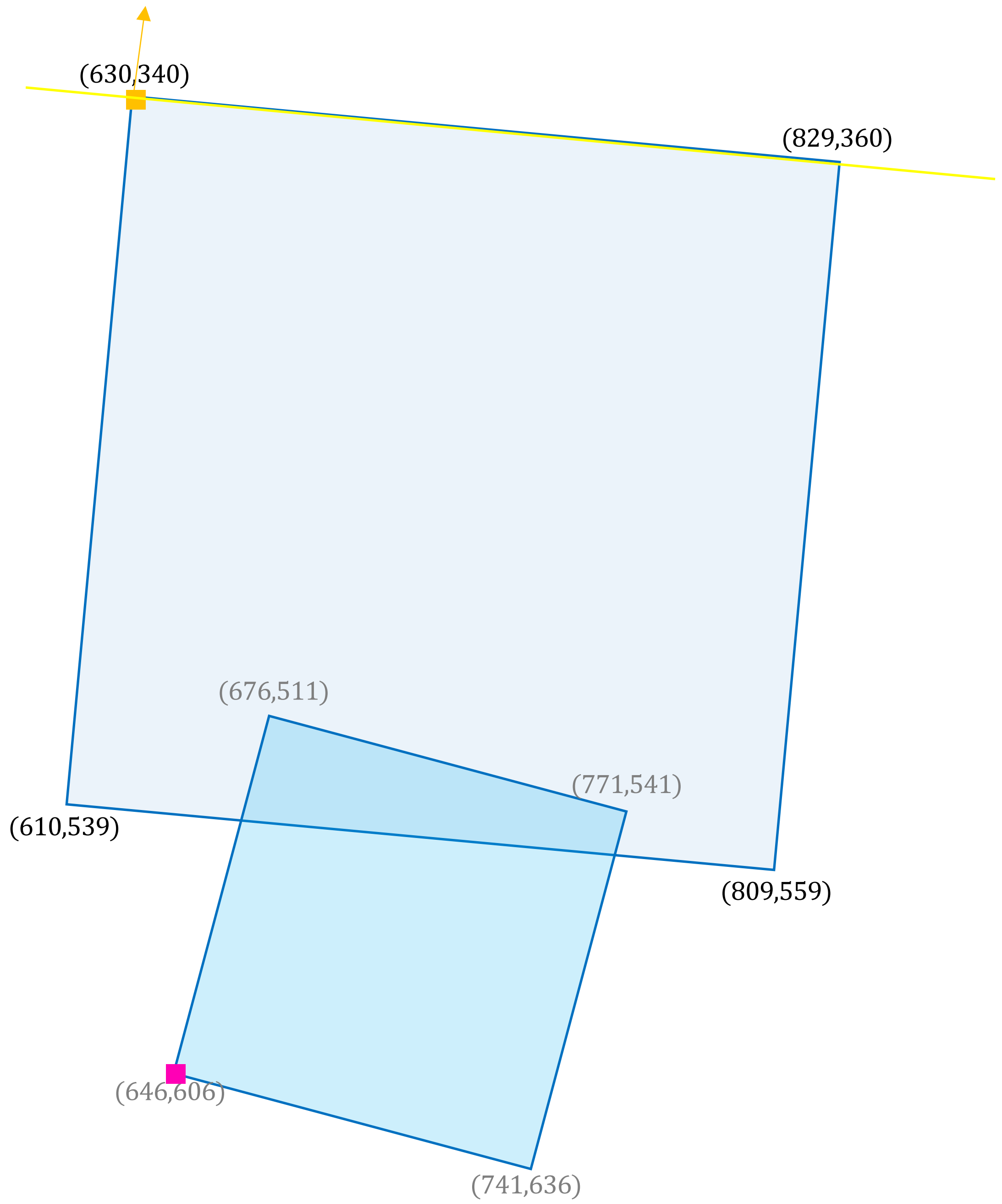
- Still on the **first vertex/axis** of the first object, we now proceed to test and project the next vertex of the second object. We keep the new minimum projection.



proj: -283

minProj: -283

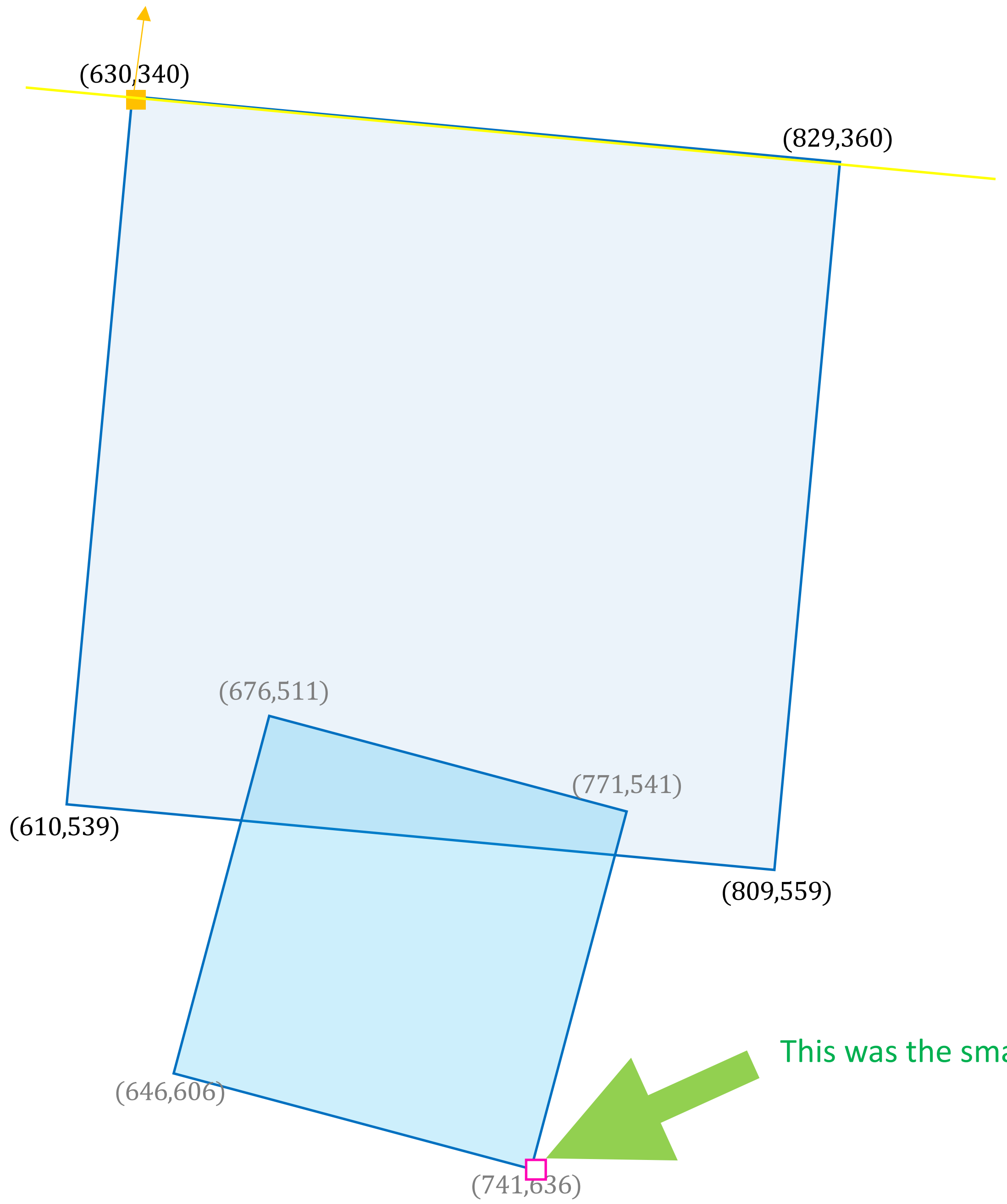
- And we do the same with all other vertices of the small box. We keep comparing them and keeping the smallest projection. In other words, we are trying to keep track of the **point** that is the **furthest back** from the **axis** of the first object that we are currently testing.



proj: -263

minProj: -283

- We proceed to the next vertex of the second small box, performing the same checks and keeping the minimum projection value. Pay attention to how the projection values are all negative so far, since they are **behind** our **axis**.

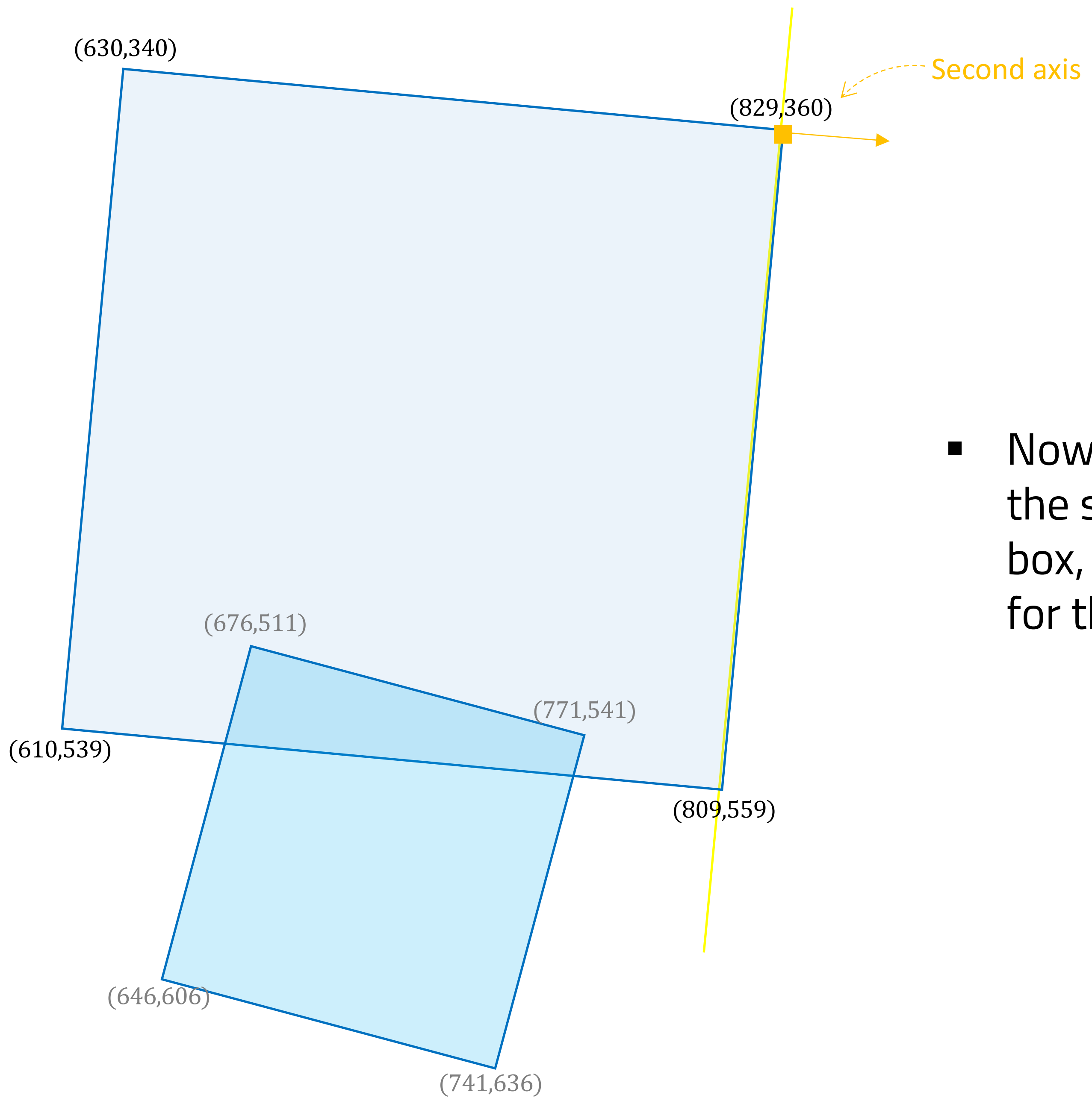


proj: -263

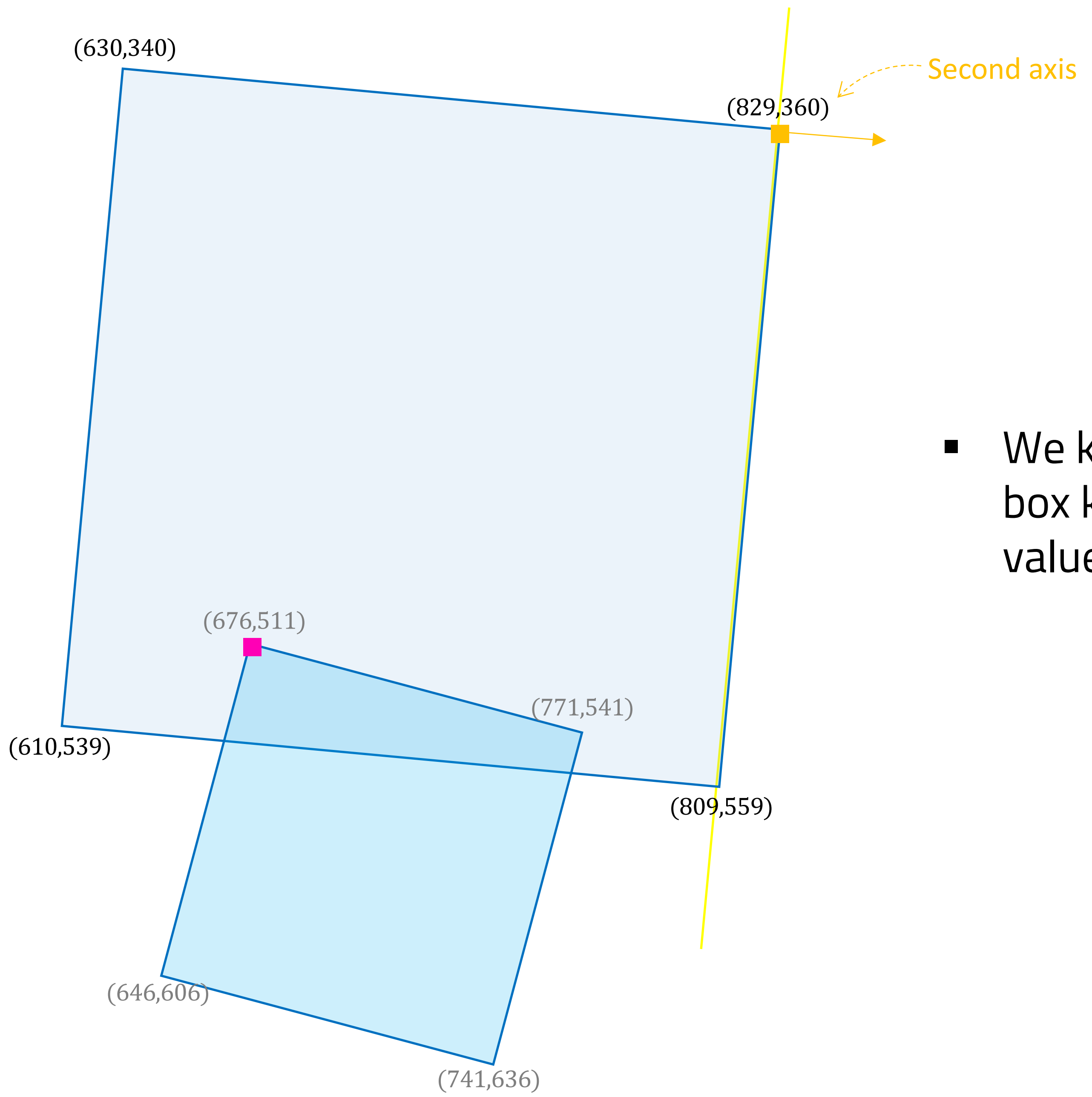
minProj: -283

- After looping all the vertices of the second small box, we found a **point** that has the minimum projection value. This is the point that is the furthest back from the **axis** that we are testing).

This was the smallest (most negative/furthest back) projection for the **first** point/axis



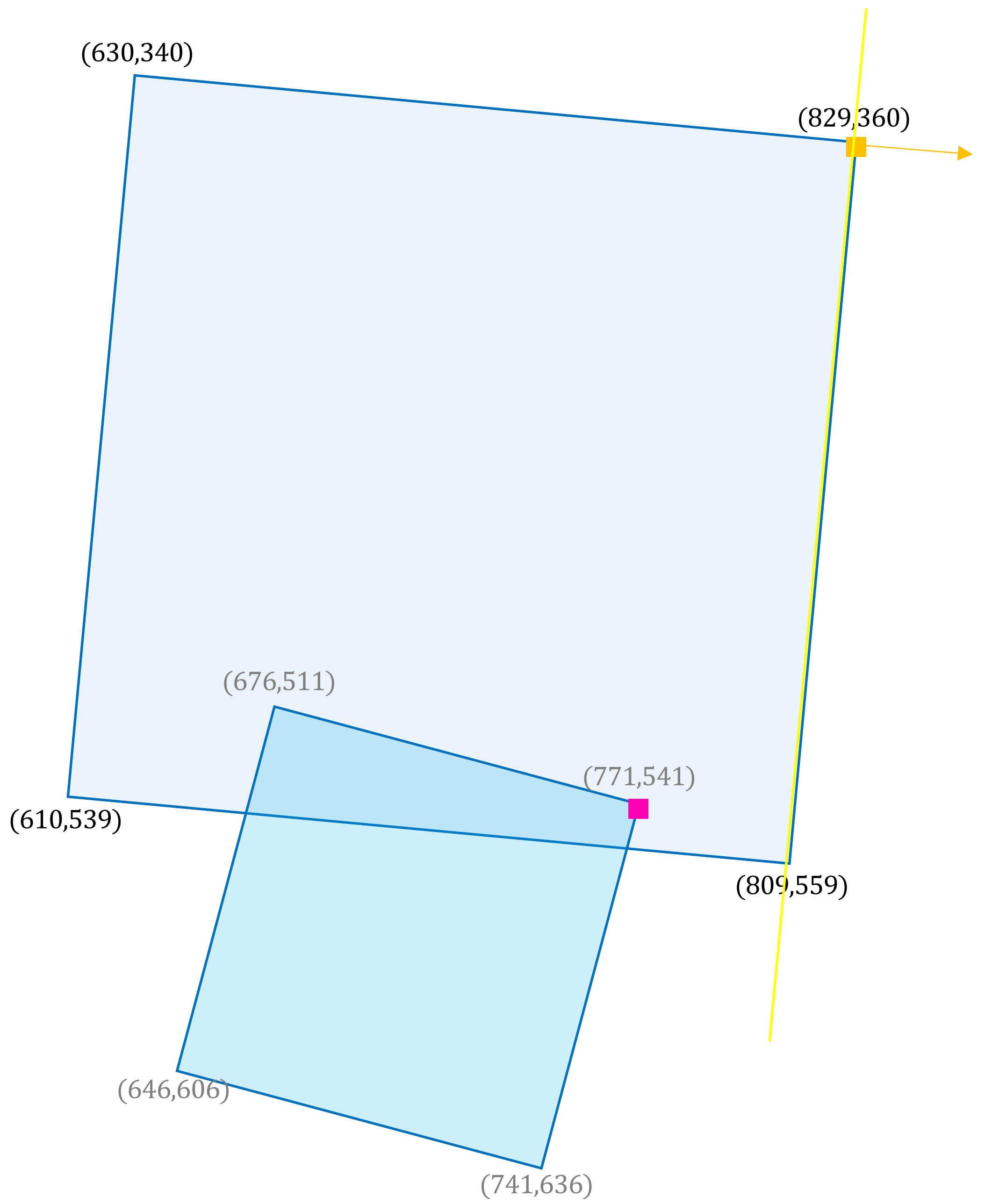
- Now that we finished testing all the vertices of the small box against the first axis of the first box, we need to move forward and test again for the **second axis**.



proj: -137

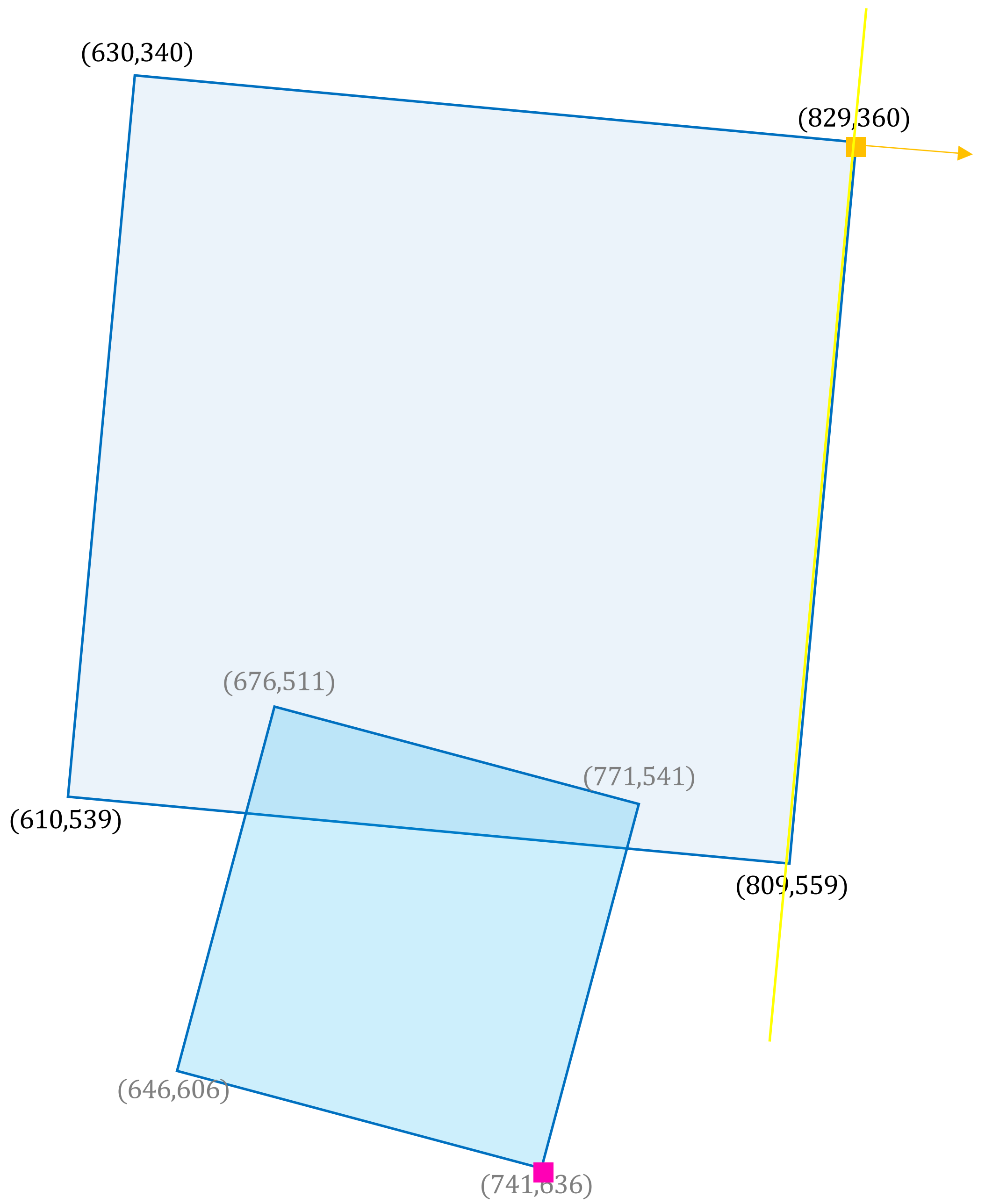
minProj: -137

- We keep looping all the vertices of the small box keeping track of the minimum projection value against the **second axis**.



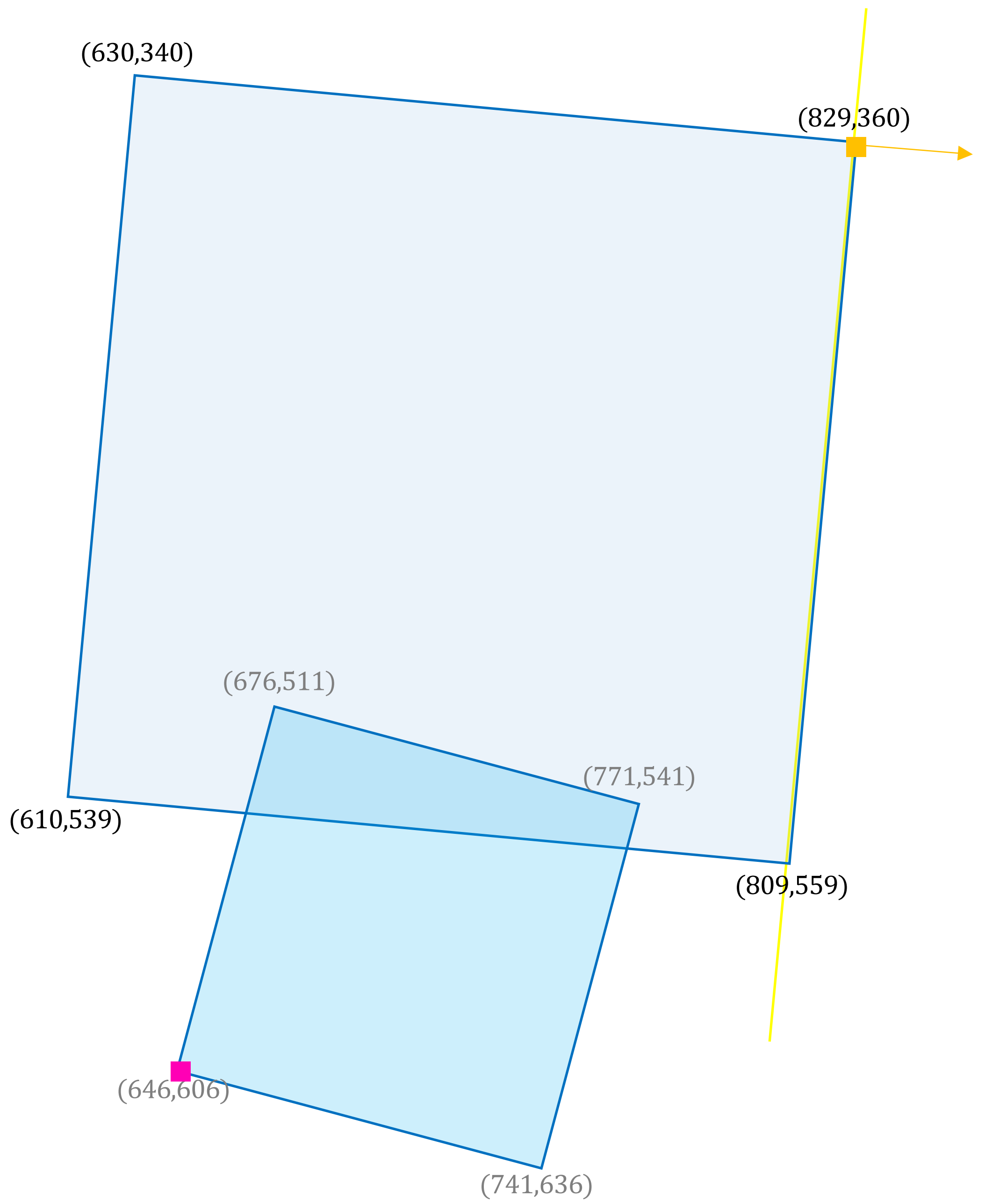
proj: -39

minProj: -137



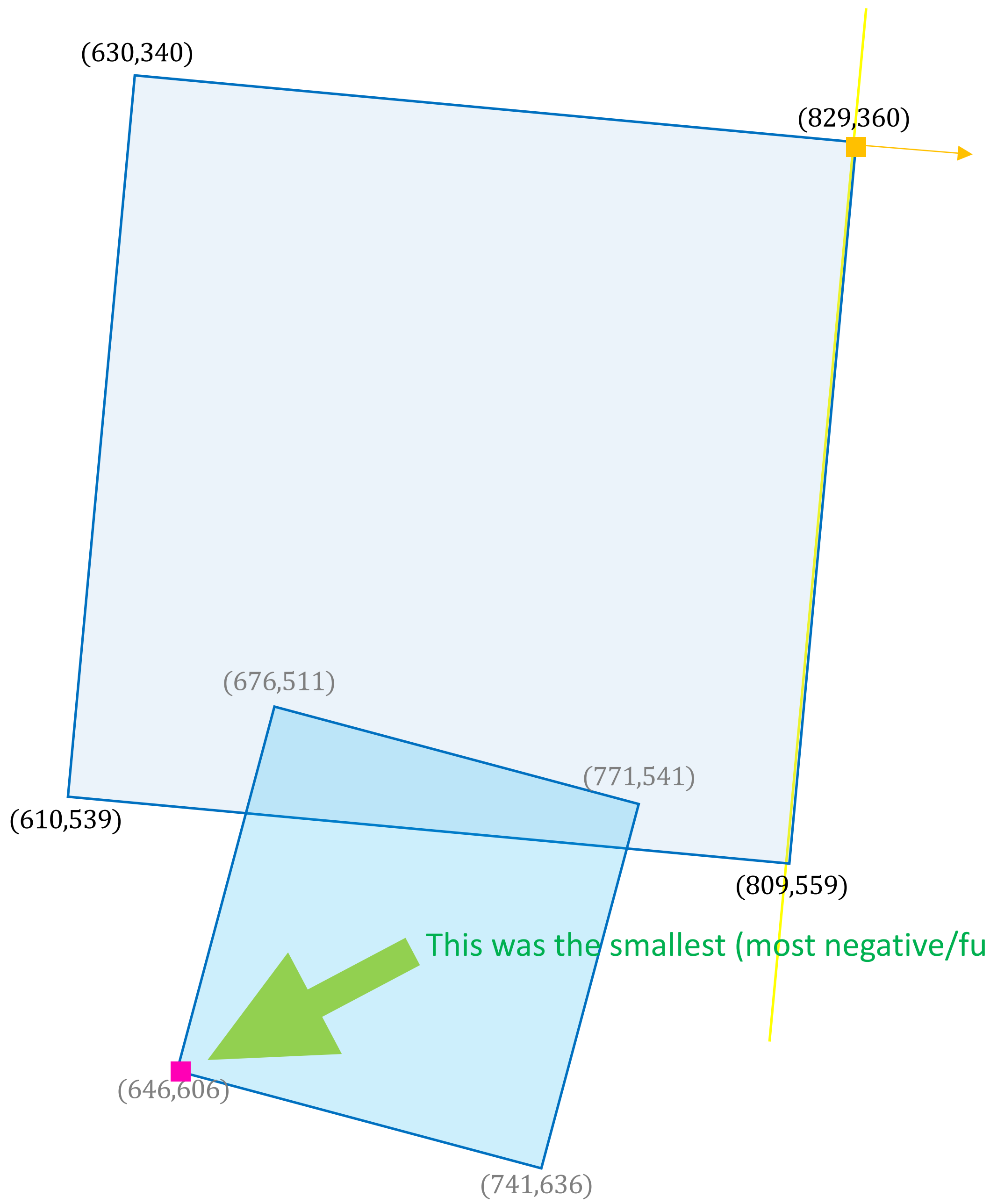
proj: -59

minProj: -137



proj: -157

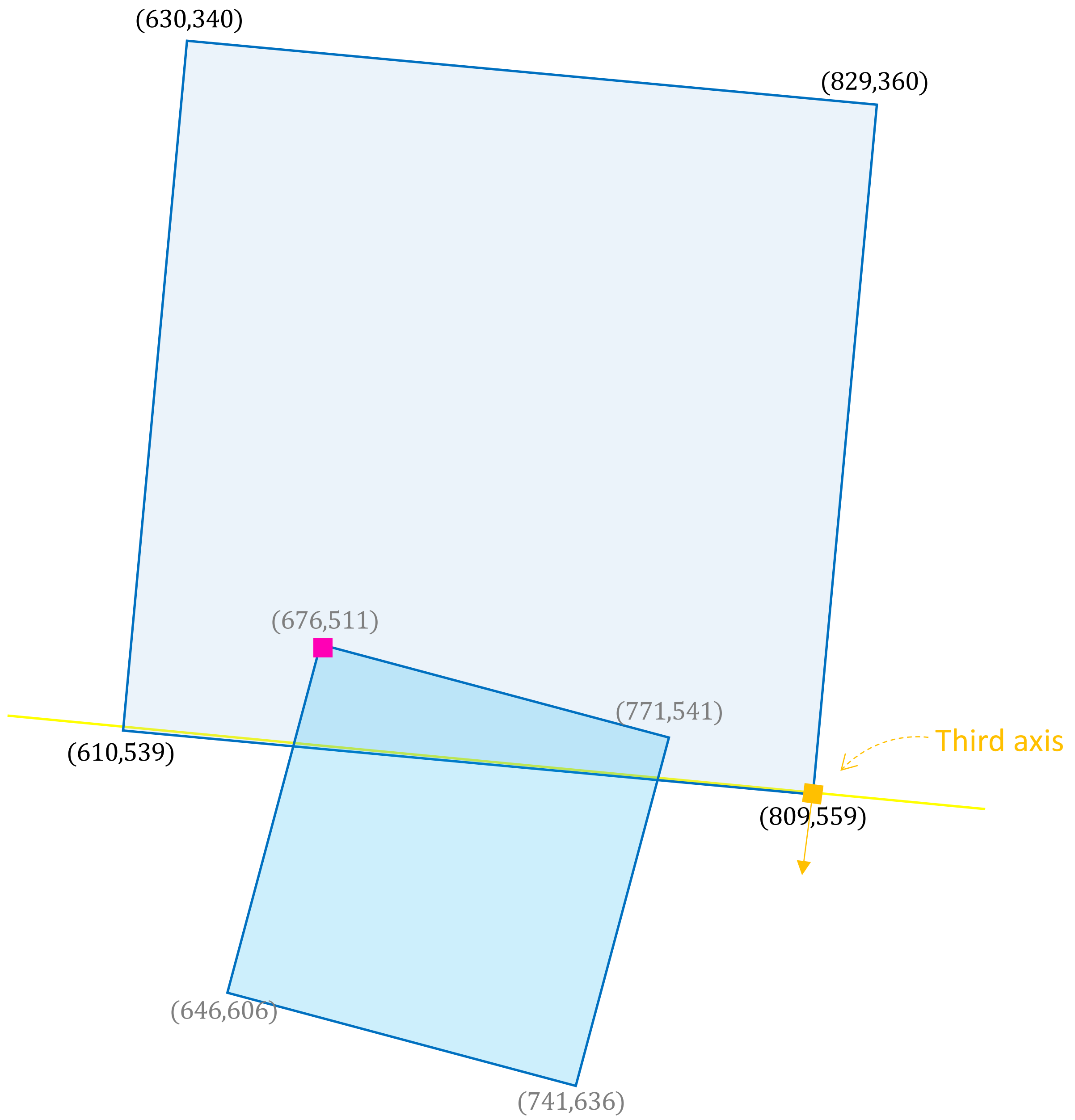
minProj: -157



proj: -157

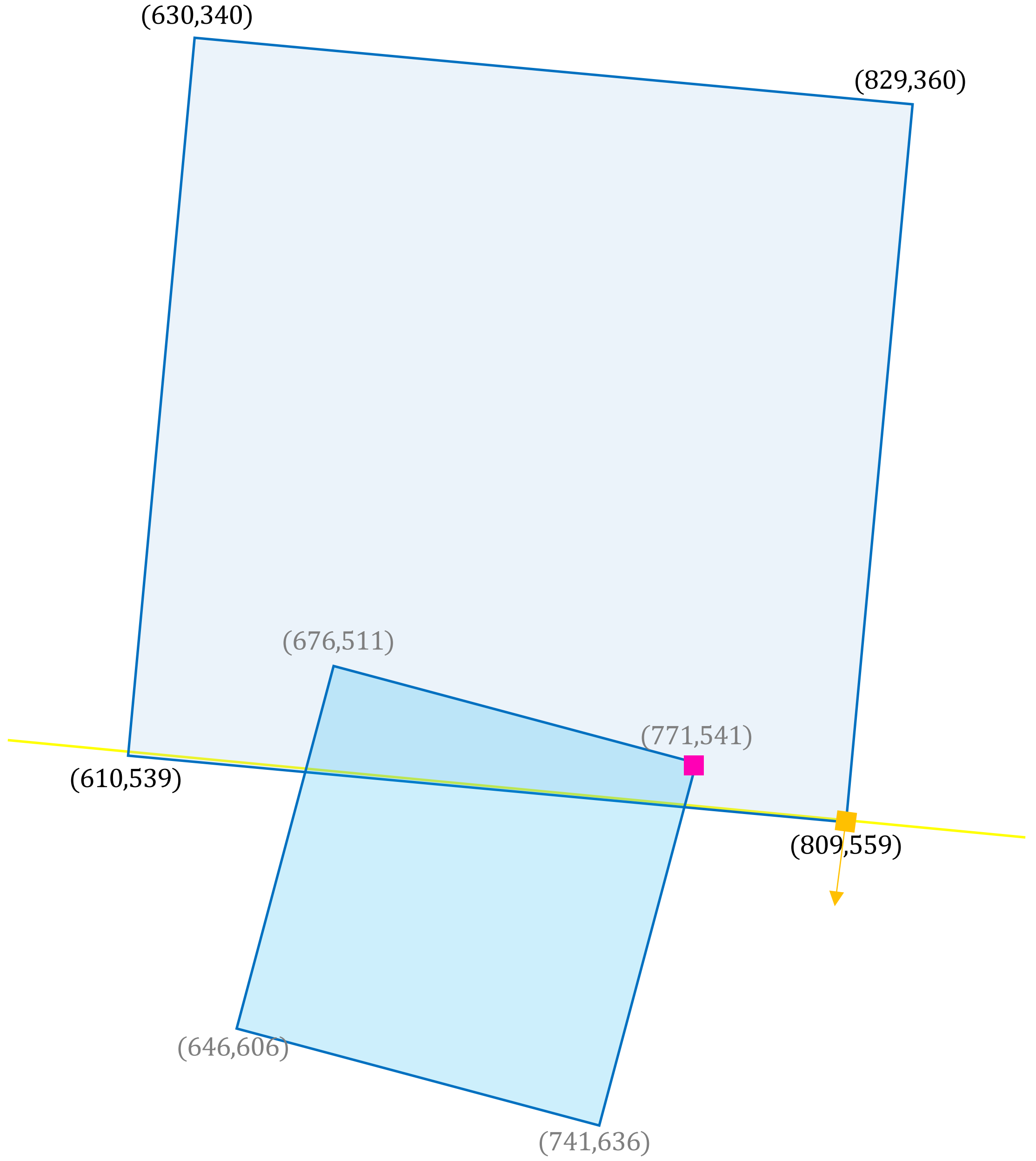
minProj: -157

This was the smallest (most negative/furthest back) projection for the **second** point/axis



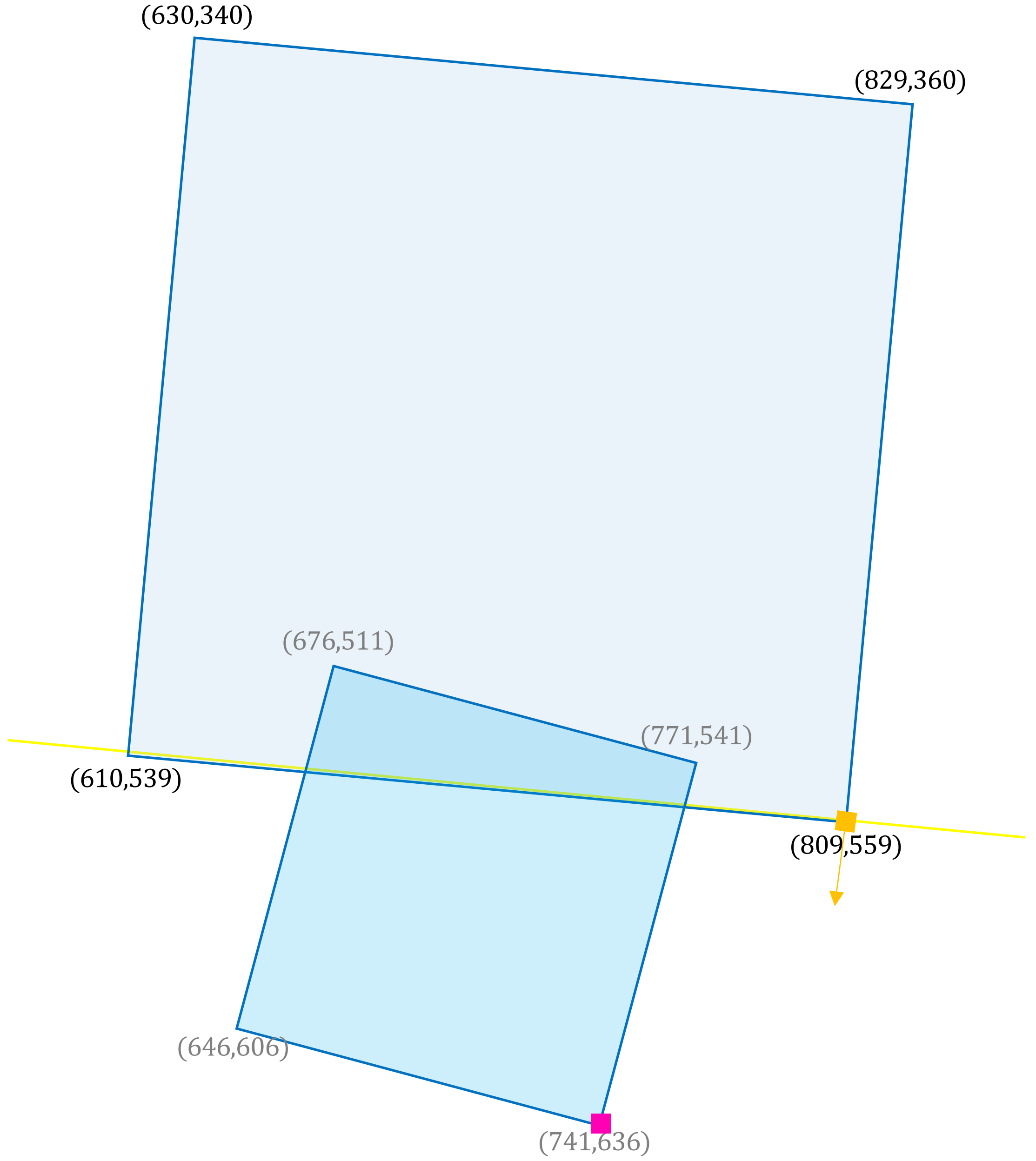
proj: -34

minProj: -34



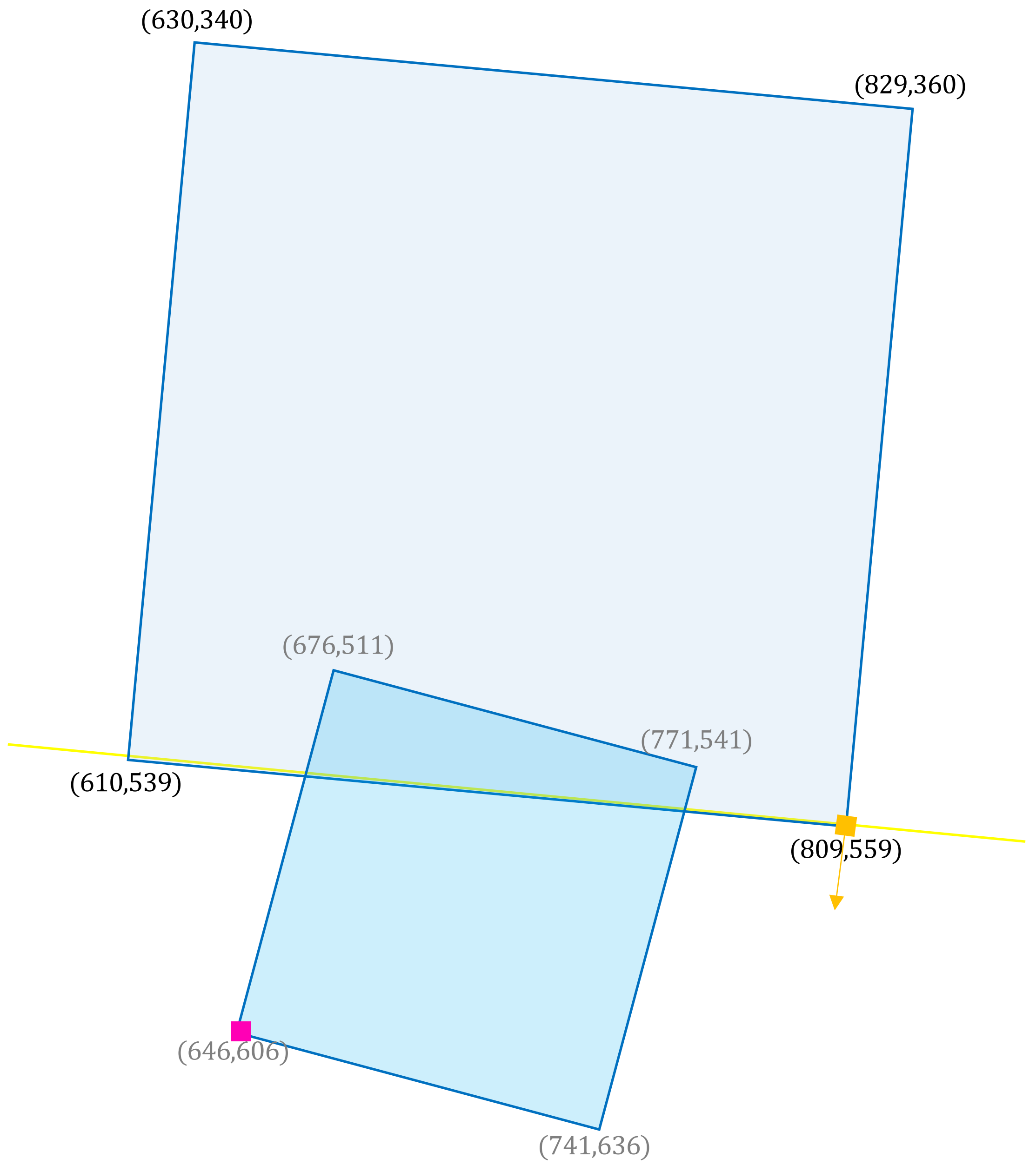
proj: -14

minProj: -34



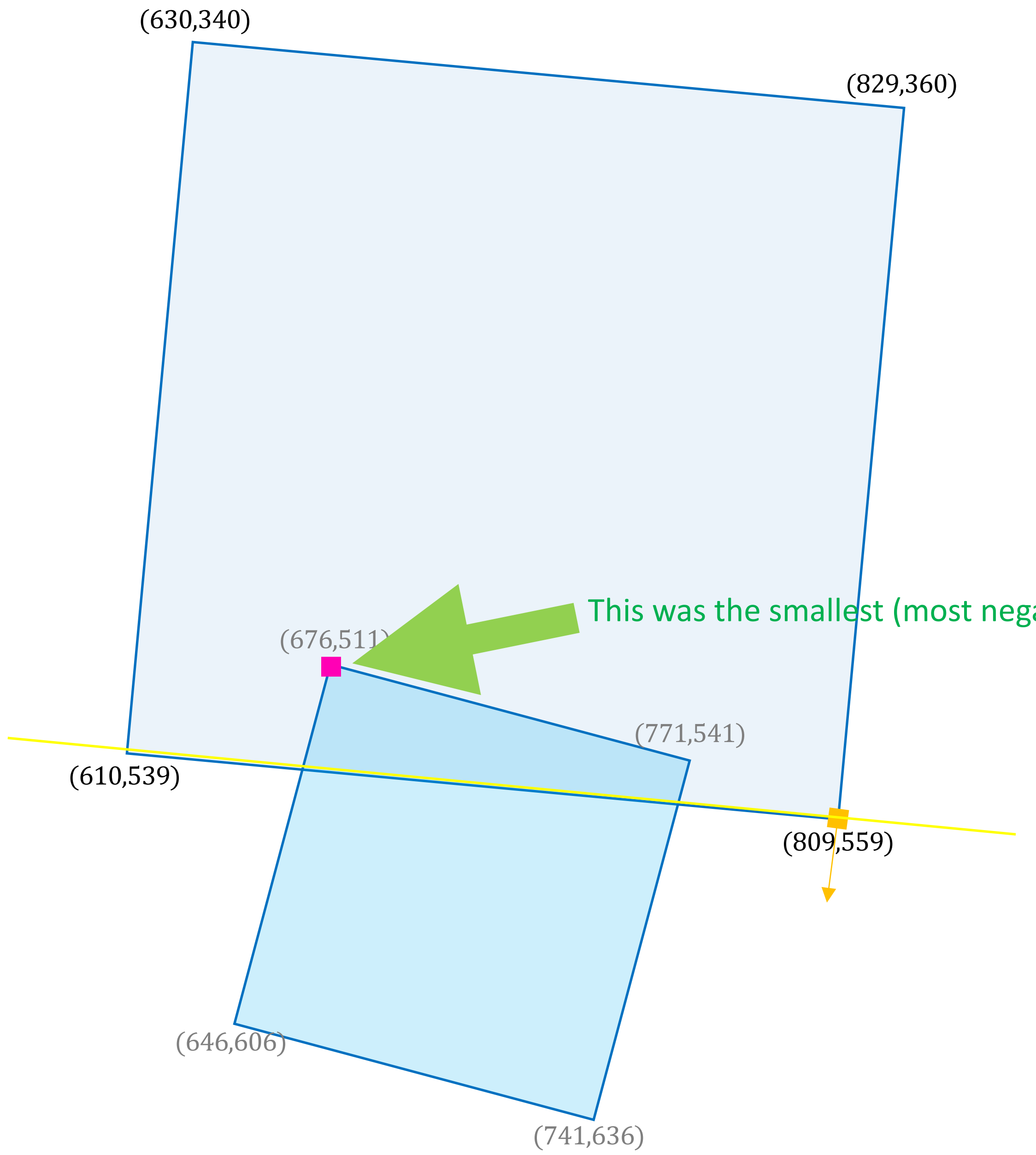
proj: 83

minProj: -34



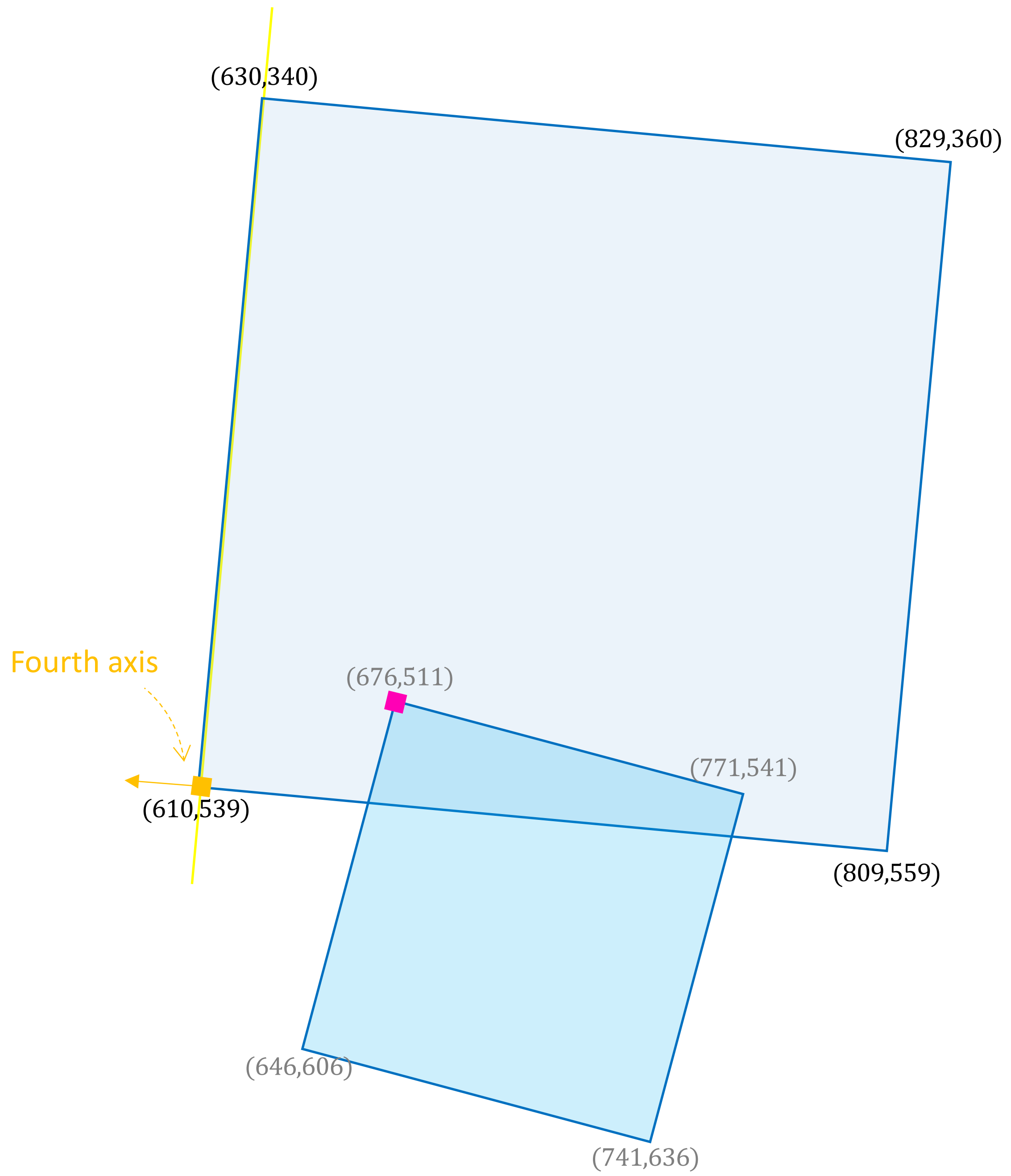
proj: 63

minProj: -34


(630,340)
(829,360)
(676,511)
(771,541)
(610,539)
(809,559)
(646,606)
(741,636)
This was the smallest (most negative/furthest back) projection for the **third** point/axis

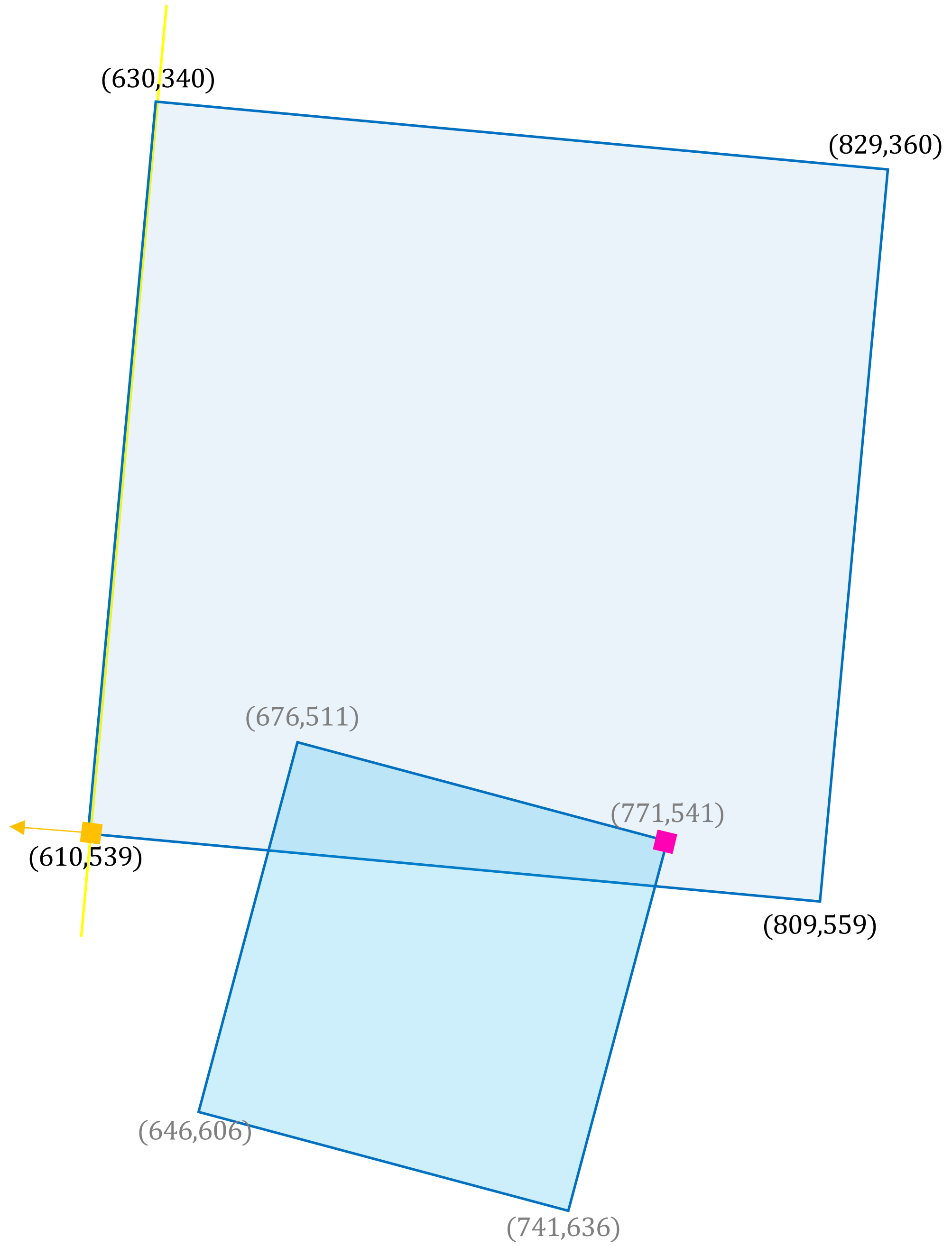
proj: 63

minProj: -34



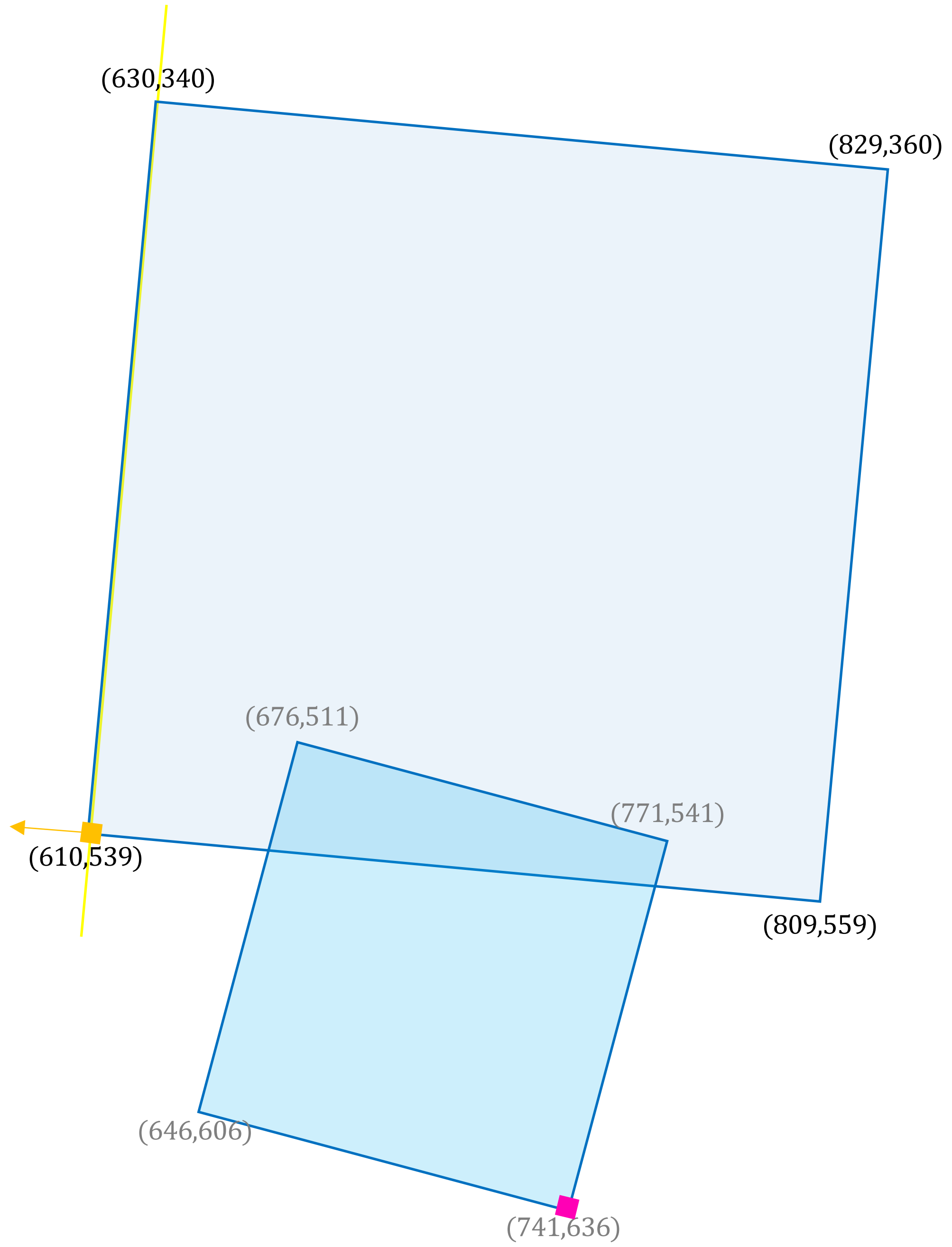
proj: -62

minProj: -62



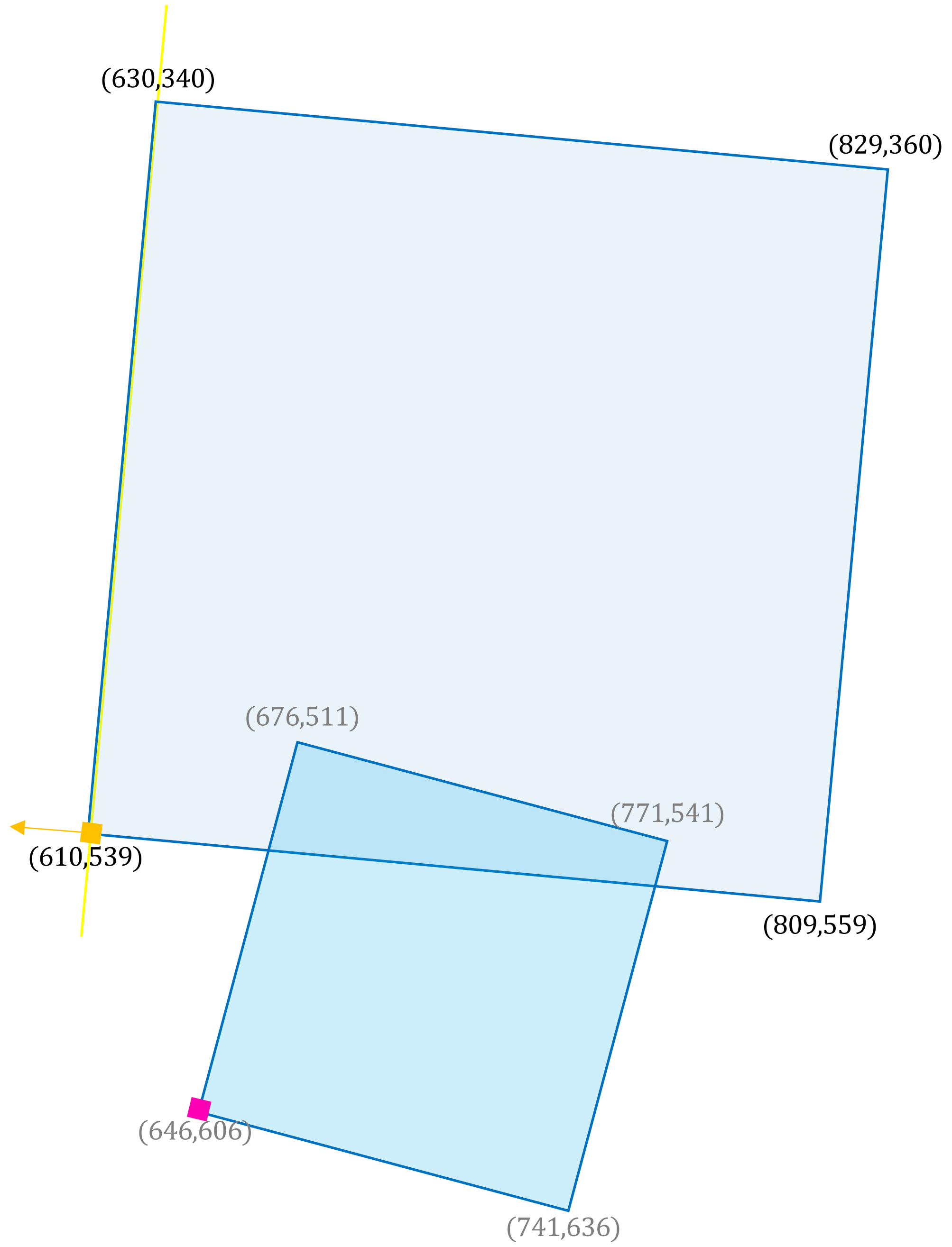
proj: -160

minProj: -160



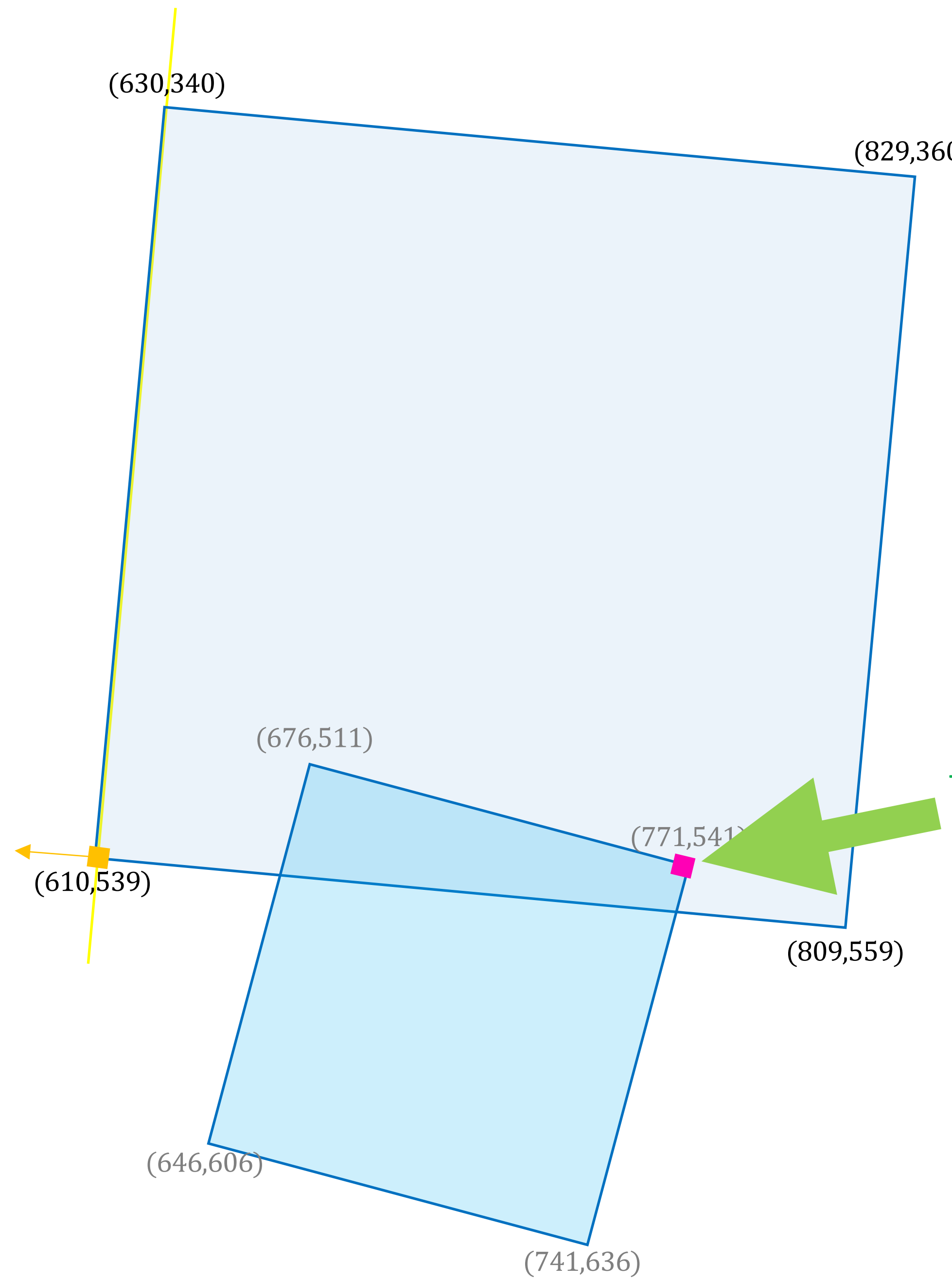
proj: -140

minProj: -160



proj: -42

minProj: -160



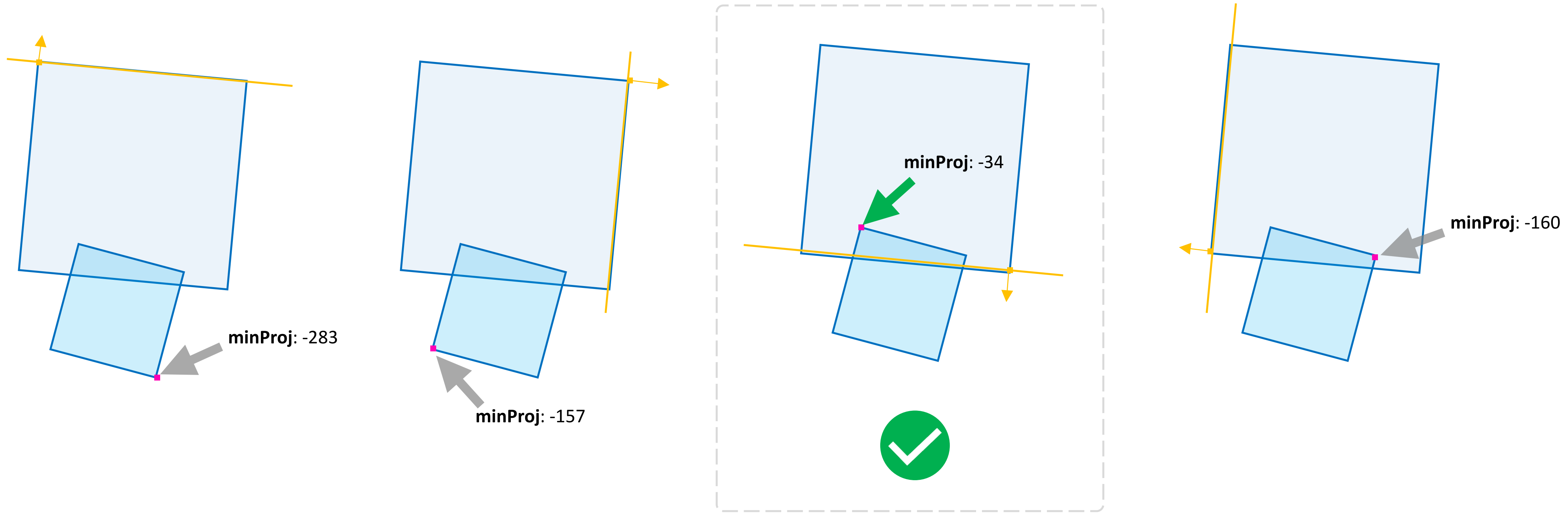
proj: -42

minProj: -160

This was the smallest (most negative/furthest back) projection for the **fourth** point/axis

Now, out of **all** those negative projection values, which one is our ***minimum axis*** of separation?

For that, we need to get the smallest distance (which means the *least* negative number).
That means we need to find the **maximum** value (since **-34** is *bigger* than **-283**, for example).


minProj: -283
minProj: -157
minProj: -34
minProj: -160