RCNN: The results come from a model trained for 3 hours.
→ Time of setting all boundary boxes = 20 seconds.

- Boundary boxes false (false positive) = 20
- Boundary boxes reasonable correct = 56
- Actual amount of boundary boxes = 60

→ Accuracy = 56/60 = 0.933 --> 93% Accuracy
YOLO: Test results come from a model trained for 1 hour.
→ Time of setting all boundary boxes = 4 seconds.

- Boundary boxes false (false positive) = 5
- Boundary boxes reasonable correct = 48
- Actual amount of boundary boxes = 60

→ Accuracy = 48/60 = 0.8 --> 80% Accuracy

These results are interesting. They vary in accuracy and in speed.

About the time used for object detection. The faster-RCNN is way faster than its predecessors that used spartial filtering. The faster-RCNN solved that problem by having a second network predicting on the region proposals .. RCNNs uses region proposal network. By handling the proposals faster-RCNNs are able to do 5 FPS. This makes it possible to use it on live video. As you can very clearly see is that my implementation of the Detectron2 faster-RCNN did 50 images in 20 seconds. This was including outputing them visually. This is on average 2.5 images per second (2,5 FPS). Earlier RCNNs also made very many region proposals (2000). Faster-RCNN does 300 which of course can be seen in the speed of the object detector. As a last point of the reason of its much faster capabilities than the predicessors is the use of anchor boxes. The anchor boxes is a predifined set of boxes of different sizes. By using these the faster-RCNN can evaluate all the predicted boxes at once. This speeds it up AND helps detecting multiple objects. All of these points show the faster-RCNN has been a good object detector.

But, because of the need for speed it has been developed another variant. The new YOLO variant that can be 8 times faster than faster-RCNN. Yolo splits a image into a grid of SxS and only looks once at the areas in the grid. It makes it potential blistering fast. It is possible to make 45 FPS which makes it very relevant to use on a video. My results reflect the YOLOs possibilities. It did the 50 test images in 4 seconds. This is 25 images per seconds and including visualising them. This is because it only looks once per area / grid cell. For each of them it returnes a probability vector. When its done with one cell it goes to the next never to look back at the one it just did. So in ONE forward pass it can do the prediction. This is the reason it is so so fast.

Because of the the pages restraint I cant explain all the layers in depth and visualise them. But I think much of the reason of speed is explained (for both of them).

About the accuracy of the object detection. My results where good despite short training time. When reading the paper of: https://assets.researchsquare.com/files/rs-668895/v1_covered.pdf?c=1631875157 its clear that both YOLO and Faster-RCNN can reach good results and mostry the same results when trained enough. Though Faster-RCNN seems to be a bit better for accuracy, the YOLO is still not far behind. YOLO has one dissadvantage that it is bad at recognising small objects. Just as in my test it did bad at finding the small tattoos. This is because of the spartial contraints of the algorithm.

I have not mentioned the RCNN´s disadvantages. Thats because its a solid object detector. It detects the objects. Before it was the best but not anymore. So its dissadvantage is being way slower than YOLO. The need for speed of object detectors makes it not as good as YOLO in many scenarios.