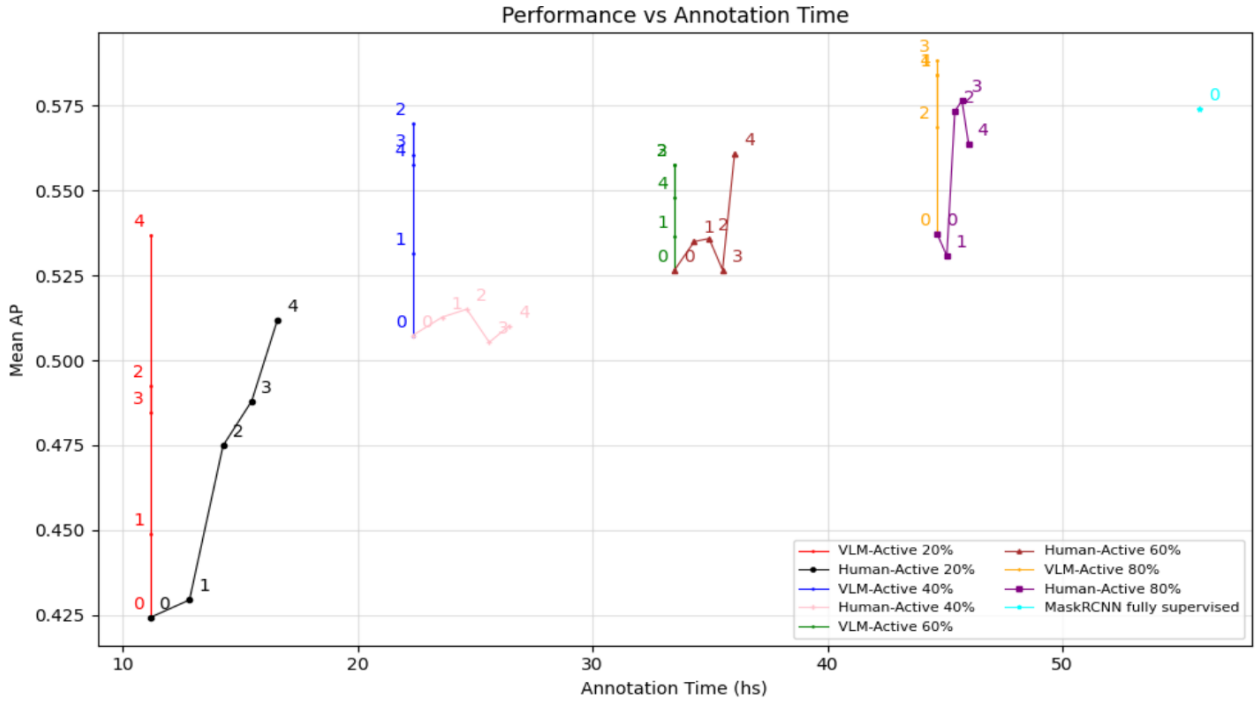

Run	Iteration 1		Iteration 2		Iteration 3		Iteration 4	
Init	+img	epochs	+img	epochs	+img	epochs	+img	epochs
0.2	-	7	-	-	-	-	-	-
0.4	-	8	-	-	-	-	-	-
0.6	-	8	-	-	-	-	-	-
0.8	-	8	-	-	-	-	-	-
1	-	9	-	-	-	-	-	-
Human Simulated								
0.2	295	8	252	7	145	7	67	7
0.4	246	7	99	7	44	8	31	7
0.6	244	7	76	9	70	7	24	7
0.8	90	9	52	9	22	7	32	8
VLM Active Learning								
0.2	301	7	215	7	169	7	73	8
0.4	387	10	180	11	129	11	71	11
0.6	305	9	76	8	56	6	0	9
0.8	167	9	42	8	42	7	20	9

Trend 1: the more data = more epochs until convergence/overfitting

Trend 2: Both the "human" and the VLM add less images with every iteration

What changed?

Training from skratch ---> (Pretrained=False)
Seed 42 – only NumPy – data split.
Learning rate=(0.005 + momentum=0.9, weight decay=0.0005)
Epochs scheduling:

- Overfitting is triggered when there is a 0.2 difference between the training mAP and the validation mAP.
- Adaptive epochs with data size:
  - Min epochs = 5
  - Base epoch=10
    - Train for: base epoch * [1 +(number of images training/1000)].
    - Patience: Stop if the VAL mAP has not improved in 5 epochs.
- Not ideal. The overfitting filter is a signal for overfitting, but should not stop training. The only advantage was that it helped shorten training time. (Useful considering the computational constraints and timely experiments)

Human Simulation Implementation:
An image is approved only when:

- Both prediction and gorund truth are empty
- Or all of the following conditions are met:
  - Every gt object has a matching predictoin of the same class
  - Each match has an IoU>=0.6
  - All predictions have confidence scores >0.5
  - No extra predictions exists(no FP)

Representative example of map per class evolution – VLM Active 0.4

	Book	Bird	Stop sign	Zebra
mAP(iter0)	0.045	0.469	0.829	0.615
mAP(iter5)	0.149	0.522	0.844	0.715