

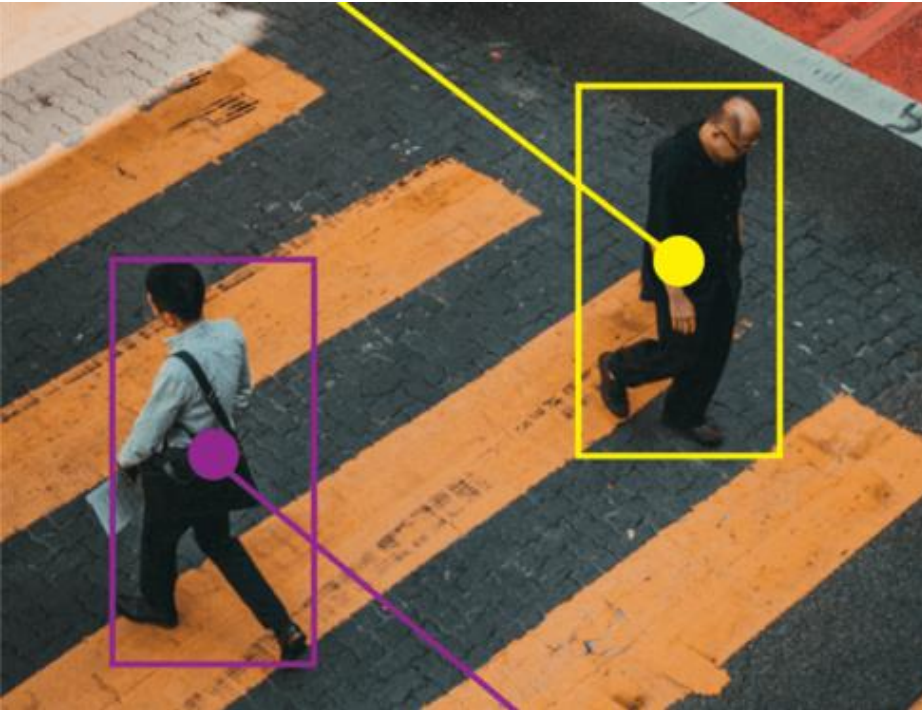
Project CV 2:

Person Tracking

Kelompok CV A & C

- Amar Ma'ruf
- Bayuzen Ahmad
- Benedictus Dikha Arianda
- Dika Mahendra
- Elsa Nurul Hidayah
- Fariz Rachman Hadi
- Fatah Abdul Jalil
- Haris Raharjo Putro
- Kio Sato
- M. Andhika Dwiki Nugraha
- Mahdia Aliyya Nuha Kiswanto
- Michael Andrian
- Moch Ardhi Kurniawan
- Muhammad Bramansyah
- Muhammad Fatih Zuhdy

Background: Person Tracking



Person tracking dalam Computer Vision adalah kemampuan untuk melacak orang-orang dalam gambar atau video secara otomatis. Ini bermanfaat untuk banyak hal, seperti keamanan di tempat umum, olahraga, dan teknologi kendaraan tanpa pengemudi.

Misalnya di keamanan, teknologi ini membantu mengawasi area publik atau gedung dengan melacak orang-orang. Di olahraga, bisa digunakan untuk melihat gerakan atlet. Dan di mobil tanpa pengemudi, teknologi ini membantu mobil untuk melihat dan menghindari orang yang berjalan kaki.

Kemampuan ini juga bisa membantu di bidang kesehatan dengan memantau gerakan dan aktivitas pasien. Dengan teknologi person tracking ini, komputer bisa lebih pintar dan membantu dalam berbagai situasi sehari-hari.

Objectives



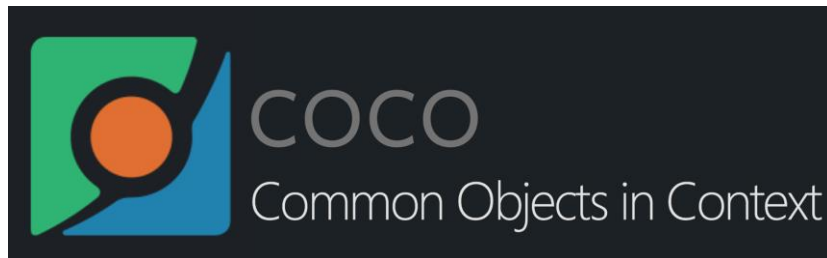
Person Tracking akan meminta kamu untuk mengembangkan sistem cerdas berbasis teknologi AI yang mampu melakukan pelacakan objek untuk kebutuhan smart city. Sistem ini akan mendeteksi dan melacak orang-orang yang ada di titik tertentu secara otomatis guna mendukung keamanan kota.

Timeline

Project Plan		Week 1							Week 2						
Project Description	Progression	17/11/23	18/11/23	19/11/23	20/11/23	21/11/23	22/11/23	23/11/23	24/11/23	25/11/23	26/11/23	27/11/23	28/11/23	29/11/23	30/11/23
Problem Understanding	100%														
Data Understanding	100%														
Exploratory Data Analysis	100%														
Preparation for Modeling	100%														
Data Training (YOLO)	100%														
Model Evaluation (YOLO)	100%														
Data Training (F-RCNN)	0%														
Model Evaluation (F-RCNN)	0%														
Presentation Preparation	80%														



Dataset: COCO



COCO Dataset in Kaggle

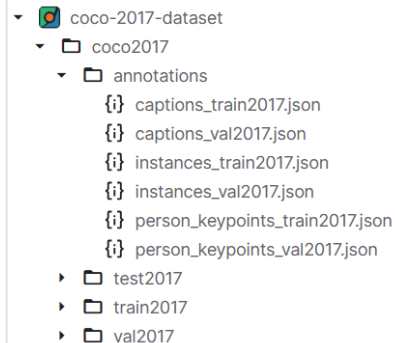
Image ID for Data Train: 118287

Image ID (Only Person) for Data Train: 64115

Image ID for Data Validation: 5000

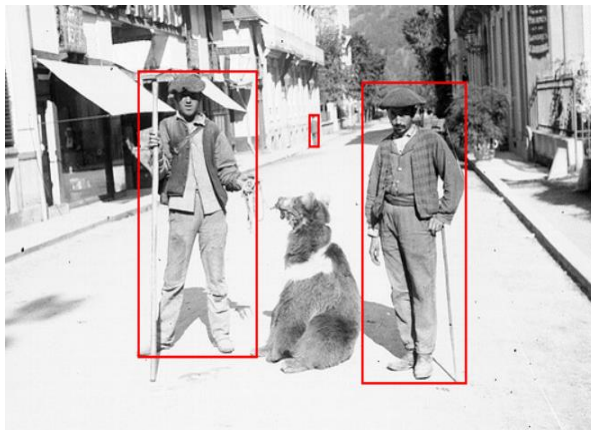
COCO Dataset from Official Website

COCO 2017 train/val browser (123,287 images, 886,284 instances).



Train	±3000 images
Val	±1000 images
Test	±500 images

Dataset: COCO



Modelling

Semenjak YOLOv5 - YOLOv8 dikerjakan oleh Ultralytics

<u>n</u>	nano size mode
<u>s</u>	small size model
<u>m</u>	medium size model
<u>l</u>	large size model
<u>x</u>	xtra large size model

Tidak ada perbedaan di antara kelima model dalam hal operasi yang digunakan, kecuali untuk layers dan parameternya,

YOLOv5:

YOLOv8:

YOLOv5s

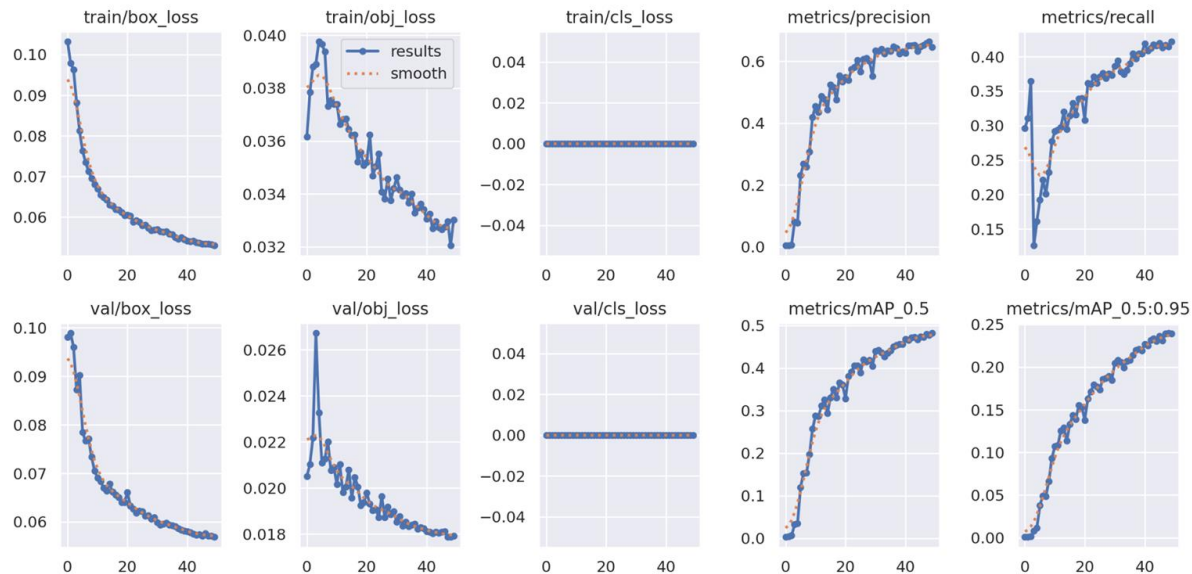
YOLOv8s

YOLOv5x

Model	size (pixels)	mAP ^{val} 50-95	mAP ^{val} 50	Speed CPU b1 (ms)	Speed V100 b1 (ms)	Speed V100 b32 (ms)	params (M)
YOLOv5n	640	28.0	45.7	45	6.3	0.6	1.9
YOLOv5s	640	37.4	56.8	98	6.4	0.9	7.2
YOLOv5m	640	45.4	64.1	224	8.2	1.7	21.2
YOLOv5l	640	49.0	67.3	430	10.1	2.7	46.5
YOLOv5x	640	50.7	68.9	766	12.1	4.8	86.7

Model	size (pixels)	mAP ^{val} 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)
YOLOv8n	640	37.3	80.4	0.99	3.2
YOLOv8s	640	44.9	128.4	1.20	11.2
YOLOv8m	640	50.2	234.7	1.83	25.9
YOLOv8l	640	52.9	375.2	2.39	43.7
YOLOv8x	640	53.9	479.1	3.53	68.2

Yolo5s



50 epochs completed in 0.662 hours.

Optimizer stripped from runs/train/yolov5s_results4/weights/last.pt, 14.8MB

Optimizer stripped from runs/train/yolov5s_results4/weights/best.pt, 14.8MB

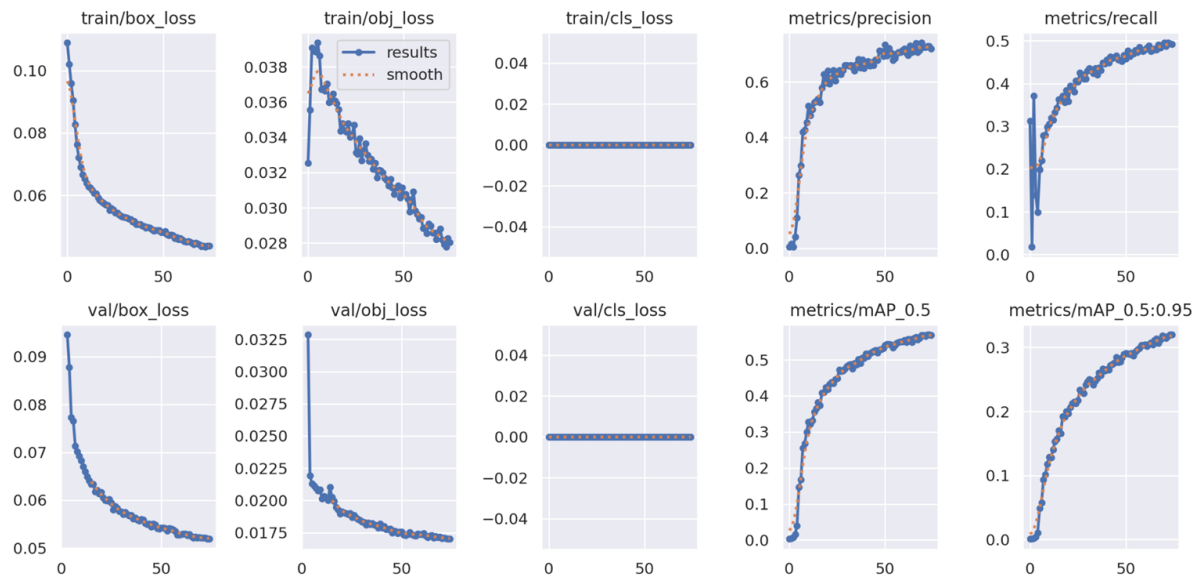
Validating runs/train/yolov5s_results4/weights/best.pt...

Fusing layers...

custom_YOLOv5s summary: 182 layers, 7246518 parameters, 0 gradients

Class	Images	Instances	P	R	mAP50	mAP50-95: 100% 16/16 [00:10<00:00, 1.47it/s]
all	1002	4021	0.664	0.415	0.477	0.24

Yolo5x



75 epochs completed in 2.610 hours.

Optimizer stripped from runs/train/yolov5x_results/weights/last.pt, 173.0MB

Optimizer stripped from runs/train/yolov5x_results/weights/best.pt, 173.0MB

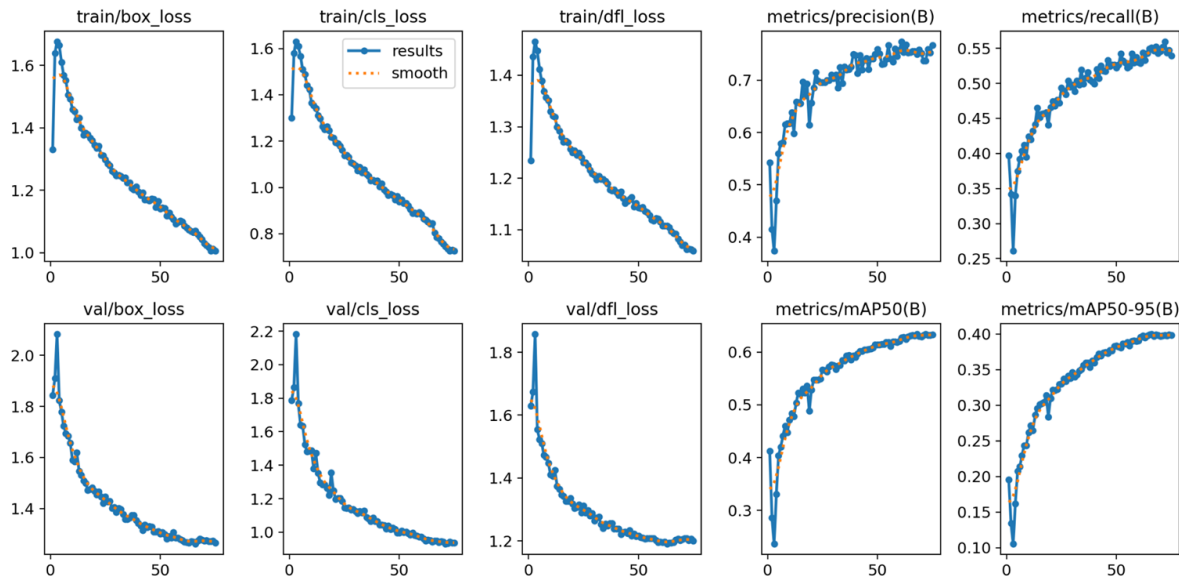
Validating runs/train/yolov5x_results/weights/best.pt...

Fusing layers...

custom_YOLOv5x summary: 322 layers, 86173414 parameters, 0 gradients, 203.8 GFLOPs

Class	Images	Instances	P	R	mAP50	mAP50-95: 100%	21/21 [00:16<00:00, 1.29it/s]
all	1002	4021	0.719	0.492	0.568	0.32	

Yolov8s



75 epochs completed in 1.772 hours.

Optimizer stripped from runs/detect/train/weights/last.pt, 22.5MB

Optimizer stripped from runs/detect/train/weights/best.pt, 22.5MB

Validating runs/detect/train/weights/best.pt...

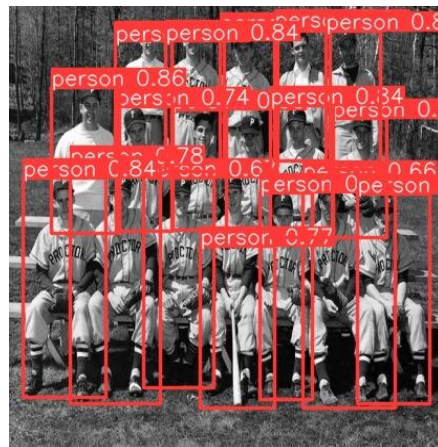
Ultralytics YOLOv8.0.220 🚀 Python-3.10.12 torch-2.1.0+cu118 CUDA:0 (Tesla T4, 15102MiB)

Model summary (fused): 168 layers, 11125971 parameters, 0 gradients, 28.4 GFLOPs

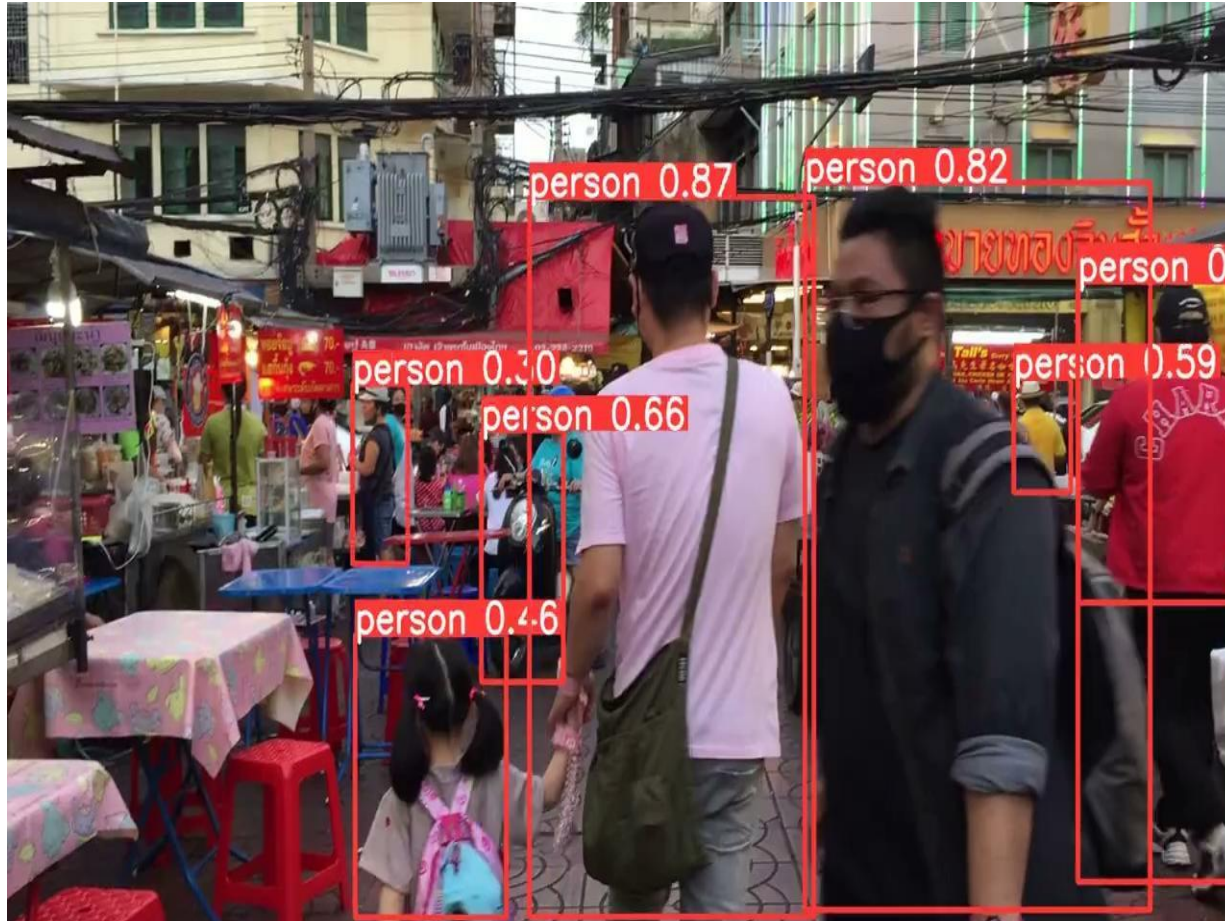
Class	Images	Instances	Box(P	R	mAP50	mAP50-95): 100% 32/32 [00:12<00:00, 2.57it/s]
all	1002	4021	0.755	0.552	0.632	0.4

Speed: 0.1ms preprocess, 1.9ms inference, 0.0ms loss, 1.5ms postprocess per image

Results Yolov8s



Results Yolov8s: Video



Results YOLOv8s: Video



Results Yolov8s: Video



Conclusions

- Untuk pemilihan tipe model yang telah kami buat, yaitu s, dan x, tipe s lama trainingnya lebih cepat dibandingkan tipe x, tetapi untuk hasil mAP50 nilainya lebih bagus tipe x.
- Untuk pemilihan versi, v5 lebih cepat dalam proses training, dibanding v8, tetapi untuk hasil mAP50 lebih bagus untuk v8
- Untuk epoch, makin besar epoch nilai mAP50 yang didapatkan juga makin baik, tetapi akan tradeoff dengan waktu trainingnya
- Model yang kami gunakan untuk video tracking adalah YOLOv8s, dengan hasil mAP50 sebesar 0.632, dan epoch 75. Kedepannya apabila memungkinkan bisa ditambahkan nilai epochnya, sehingga nilai mAP50 yang didapatkan bisa lebih baik
- Model yang kami gunakan tidak menggunakan data augmentasi, dikarenakan ketika kami coba menggunakan augmentasi resource yg kami gunakan (google colab) tidak mampu untuk memprosesnya

