

Project 2

# Person Tracking

**Vision Plus**

# Meet The Team

We are proud to introduce the dedicated and talented team behind our successful projects.



**Teguh Imanto**

**Contribution:**  
Yolov5 (v5s & v5x)

 Teguh  
Imanto



**Christian B  
Kuswandi**

**Contribution:**  
Faster RCNN

 Christian Bernard Kuswandi



**Robert Rasidy**

**Contribution:**  
Yolov8

 Robert Rasidy

# Background & Problem Statement

## Background

- Object tracking adalah bagian penting dari sistem pengawasan video di smart city.
- Teknologi ini membantu dalam keamanan publik, pengawasan lalu lintas, dan analisis perilaku.
- Dengan kemajuan AI, pelacakan objek kini dapat dilakukan secara otomatis dan real-time.

## Problem Statement

- Bagaimana membangun sistem yang mampu mendeteksi dan melacak manusia secara otomatis dalam sebuah video?
- Tantangan utama:
  1. Pencahayaan
  2. tumpang tindih objek
  3. kecepatan pemrosesan

# Objectives & Scope



## Objectives

- Mengimplementasikan dua algoritma object detection: YOLO dan Faster R-CNN.
- Melakukan evaluasi dan perbandingan performa keduanya.



## Scope

- Fokus pada pelacakan objek manusia.
- Tidak mencakup pelacakan objek non-manusia
- Tidak mencakup real-time deployment atau integrasi dengan sistem produksi.

# Data Set



Sumber DataSet berasal dari :

**COCO Dataset (Common Objects in Context)**  
[https://docs.voxel51.com/dataset\\_zoo/datasets.html#coco-2017](https://docs.voxel51.com/dataset_zoo/datasets.html#coco-2017)

## Format Data

- 330K gambar > 1.5 juta anotasi objek.
- seluruh data difilter, diambil satu class person saja.

## Sample Images

Person



# Sample Data Set

No.	Model	Total Gambar	%Train	%Valid	%Test	Train	Valid	Test
1	Faster RCNN	5000	80%	10%	10%	4000	500	500
3	YOLO v5s & v5x	5000	80%	10%	10%	4000	500	500
4	YOLO v8m	17401	75%	20%	5%	15210	1431	760

# Model Development

## Faster RCNN

No.	Model	pretrained	Epoch	Batch	Image Size	Learning Rate	Momentum	Weight Decay	Optimizer	GPU
1.	ResNet-50	True	5	4		0.005		0.0005	SGD	CPU

## YOLO

No.	Model	pretrained	Epoch	Batch	Image Size	Learning Rate	Momentum	Weight Decay	Optimizer	GPU
1.	YOLO V5s	True	100	16	640x640	lr0 : 0.01 lrf : 0.0001	0.937	0.0005	AdamW	NVIDIA L40s
2.	YOLO V5x	True	50	8	640x640	lr0 : 0.01 lrf : 0.0001	0.937	0.0005	AdamW	NVIDIA L40s
3.	YOLO V8m	True	5	16	416x416	lr0 : 0.002 lrf : 0.0001	0.9	0.0005	AdamW	Tesla T4

# Yolo Augmentation

## Faster RCNN

No.	Model	Horizontal Flip	Random Brightness Contrast	Affine	CLAHE	Motion Blur	Hue Saturation	Resize
1	ResNet-50	0.5	0.2	scale: (0.9–1.1), translate: 5%, rotate: $\pm 10^\circ$	0.2	0.1	0.2	640, 640

## YOLO

No.	Model	Mosaic	Translate	Scale	fliplr	hsv_h (hue)	hsv_s (saturation)	hsv_v (value)
1	YOLO V5s	1.0	0.1	0.5	0.5	0.015	0.7	0.4
2	YOLO V8m	1.0	0.1	0.5	0.5	0.015	0.015	0.4

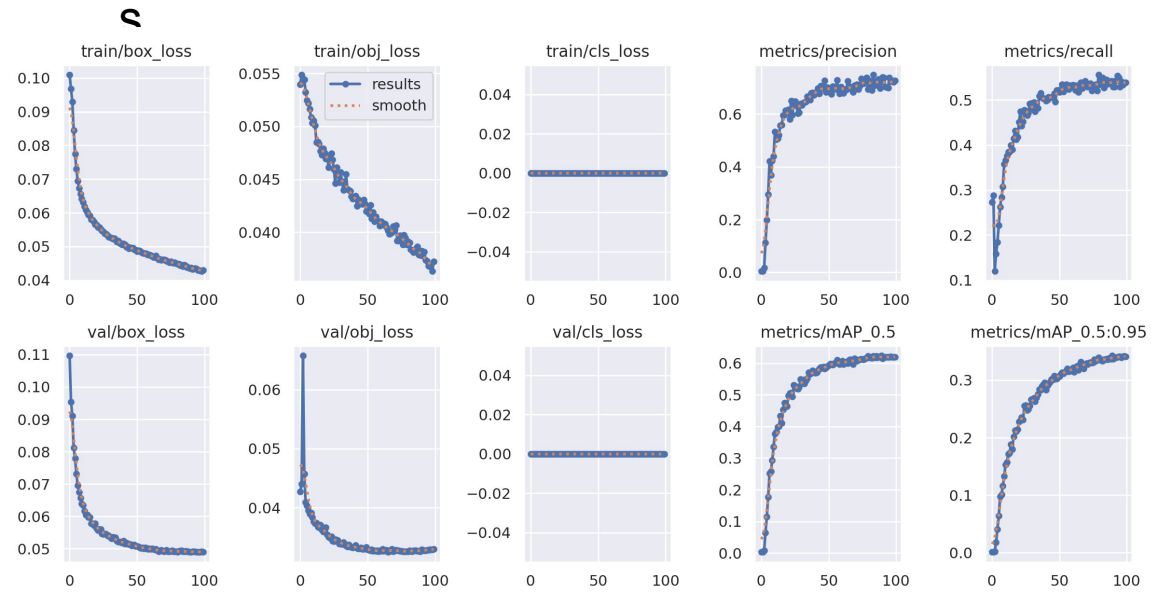


# Model Evaluation & Comparison

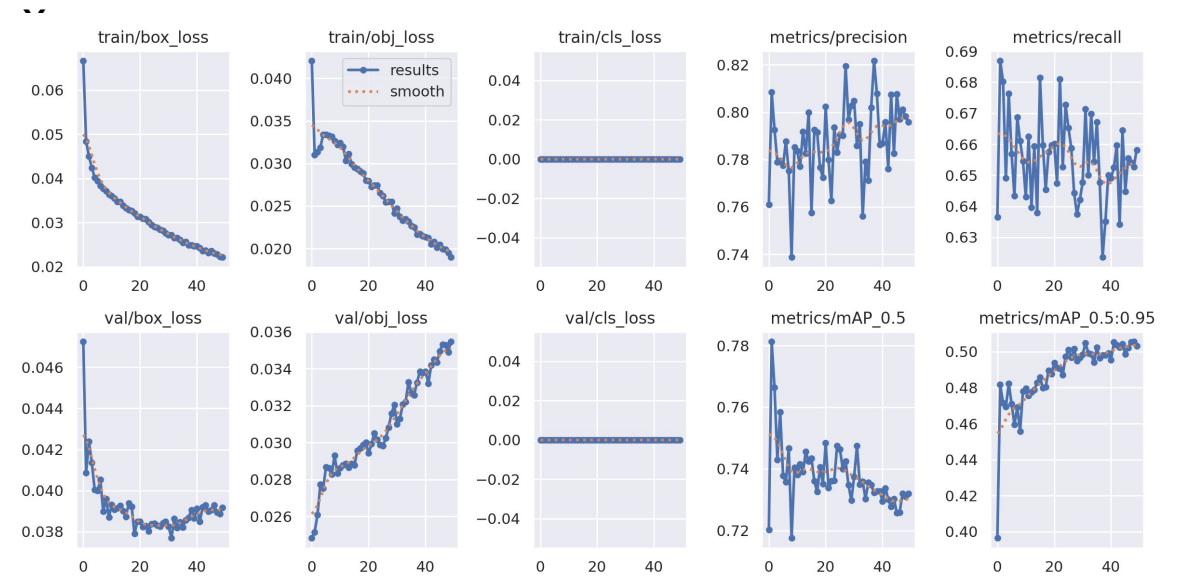
Model	Training (4000)				Validation (500)				
	Precision	Recall	mAP (50)	mAP (50-95)	Precision	Recall	mAP (50)	mAP (50-95)	Inference Time (ms)
Faster R-CNN ResNet50	0.5335	0.7743	0.4131	0.1998					
YOLO v5s	0.72603	0.53933	0.61933	0.34143	0.725	0.541	0.624	0.343	1.3 ms
YOLO v5x	0.79593	0.65817	0.73206	0.50321	0.786	0.673	0.747	0.506	4.8 ms
YOLO v8m	0.768	0.549	0.615	0.353					

# Grafik Result YOLOv5(Train)

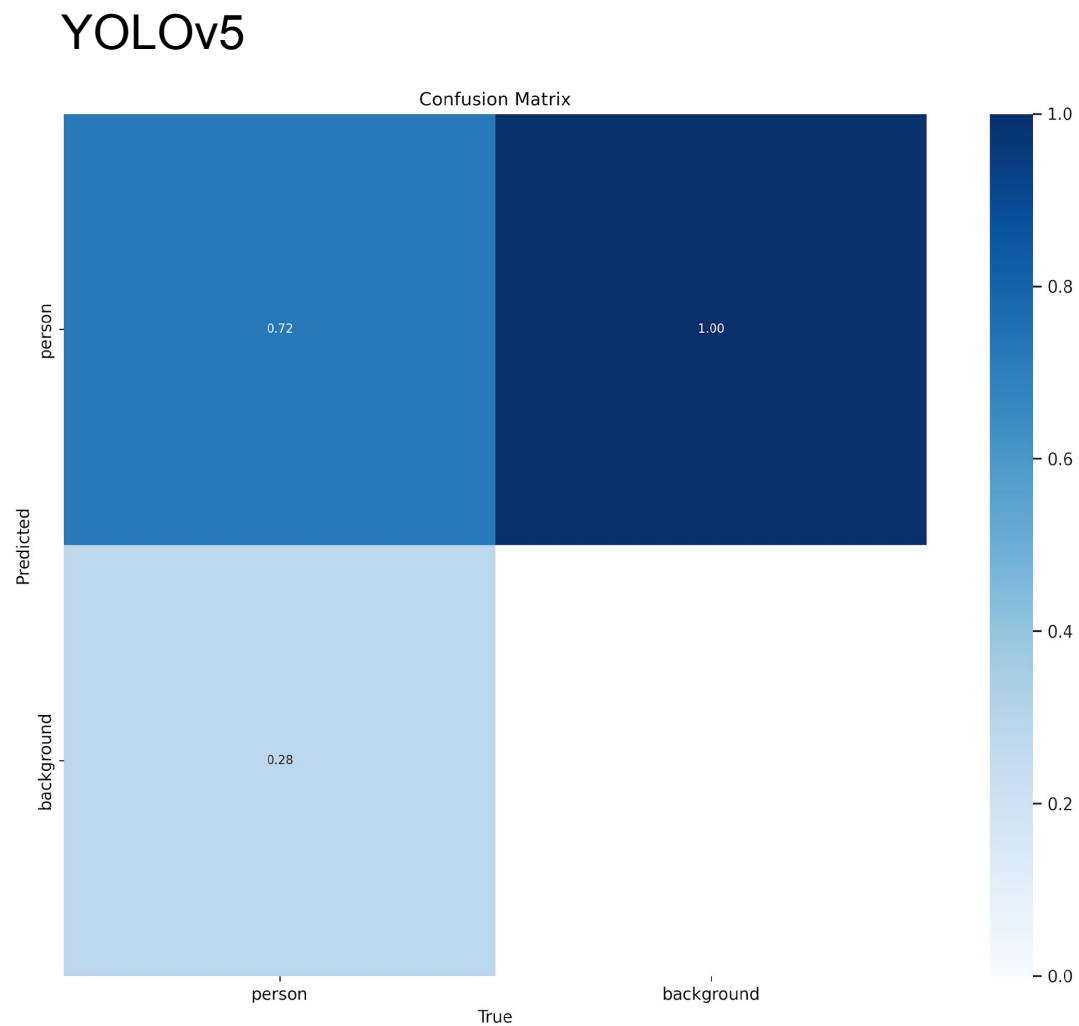
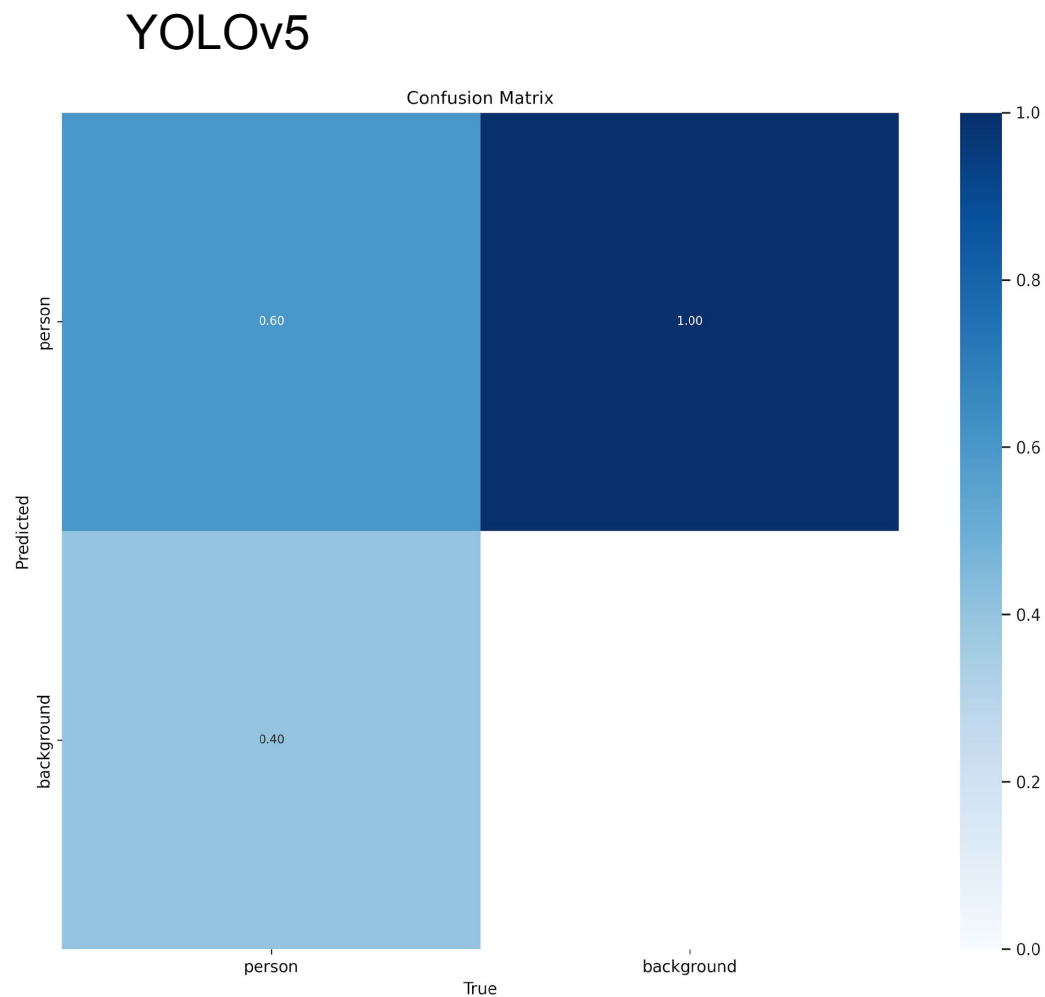
YOLOv5



YOLOv5



# Confusion Matrix YOLOv5

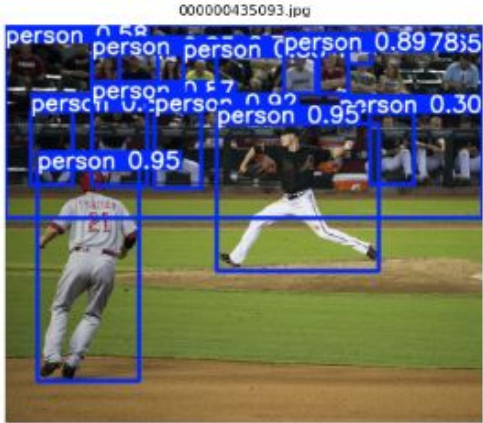


Inference Images (500) sample 10 - YOLO5s





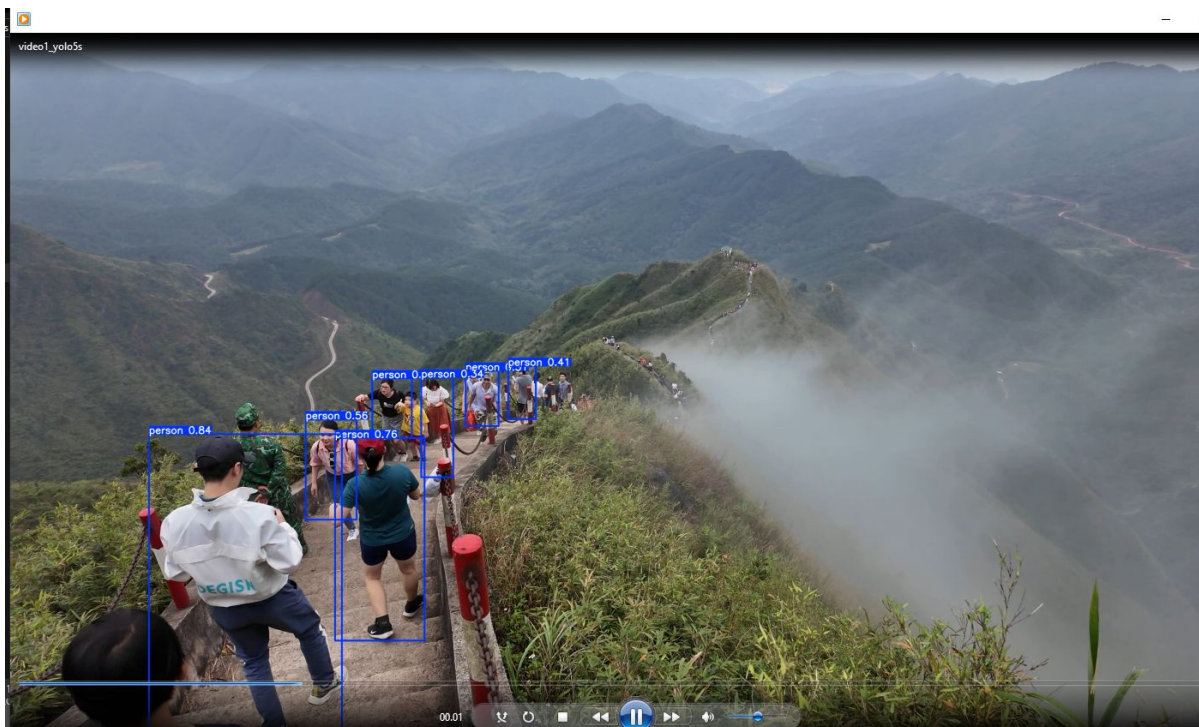
Inference Images (500) sample 10 - YOLO5x





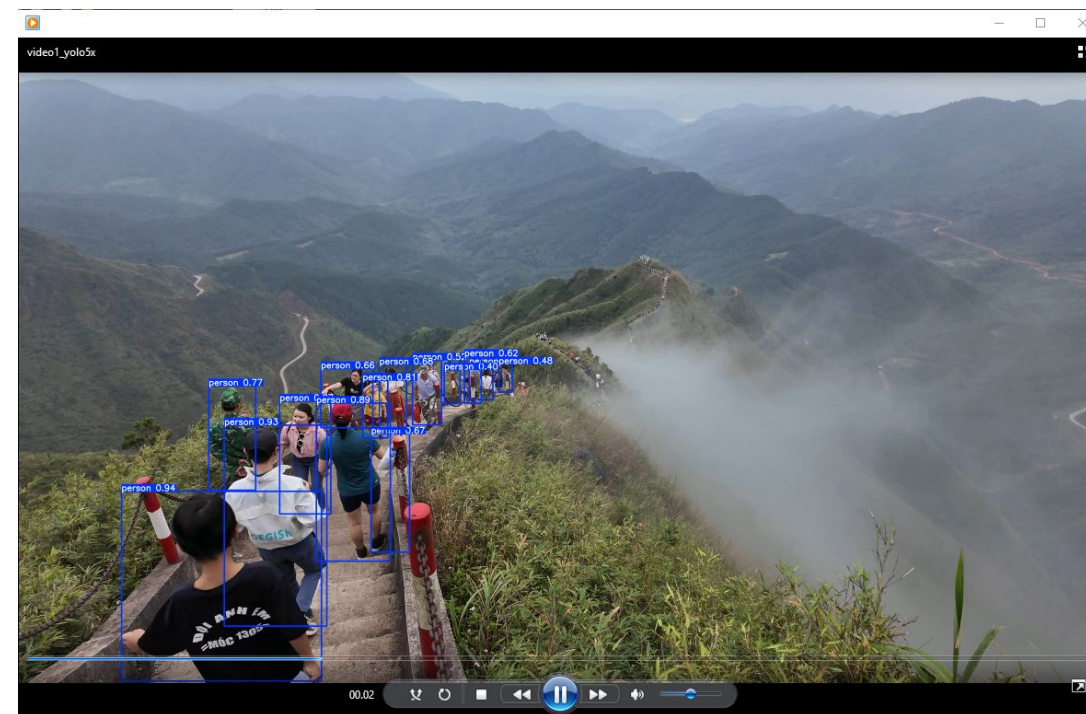
# Video Inference

video1\_yolo5s



[https://drive.google.com/open?id=18hsQgdbeUjdXSU\\_\\_dPq2neQLx66R1Qp9&usp=drive\\_fs](https://drive.google.com/open?id=18hsQgdbeUjdXSU__dPq2neQLx66R1Qp9&usp=drive_fs)

video1\_yolo5x



[https://drive.google.com/open?id=11kBEG61SnltMDn-xvXM5SOyLhpssLDtb&usp=drive\\_fs](https://drive.google.com/open?id=11kBEG61SnltMDn-xvXM5SOyLhpssLDtb&usp=drive_fs)

# Conclusion

- YOLOv8m memberikan performa terbaik secara keseluruhan, dengan precision tertinggi (0.768) dan mAP@50 tertinggi (0.615), menunjukkan akurasi deteksi yang sangat baik.
- YOLOv5s memiliki keunggulan pada mAP@50–95 tertinggi (0.391) serta inference time tercepat (1.3 ms), cocok untuk aplikasi real-time dengan kebutuhan akurasi tinggi di berbagai tingkat IoU.
- Faster R-CNN ResNet50 unggul dalam recall (0.7743), namun memiliki precision dan mAP yang rendah serta inference time paling lambat (97.9 ms), membuatnya kurang ideal untuk aplikasi real-time.

# Future Development

- Penambahan Proses Augmentasi: Dalam upaya untuk terus meningkatkan kemampuan model deteksi objek, salah satu langkah utama yang perlu diambil adalah memperluas data dan memperdalam proses augmentasi data.
- Ex, hyperparameter tuning, epoch dll Ganti model .dll



# Contact Us

**Teguh Imanto**



Teguh  
Imanto

**Christian B**



swandi  
Christian Bernard Kuswandi

**Robert Rasidy**



Robert Rasidy