



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Da un secolo, oltre.



# Riconoscimento di oggetti in immagini termiche usando tecniche di augmentation

Object detection in thermal images using a combination of augmentation techniques

Candidato: **Borsi Leonardo**

Relatore: **Prof. Bertini Marco**

Anno Accademico **2023/2024**



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Da un secolo, oltre.

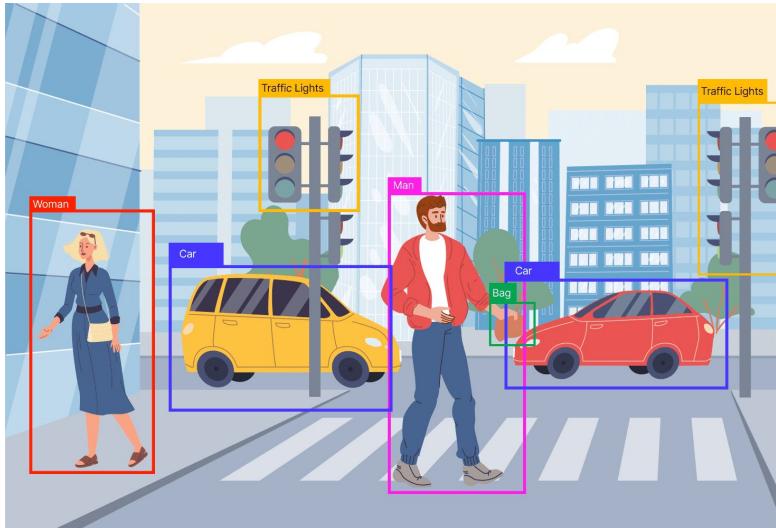


1

# Object Detection

# Object Detection

## Il problema dell'Object Detection



- Problema fondamentale della **Computer Vision**
- 2 obiettivi:
  - **Localizzazione**
  - **Classificazione**

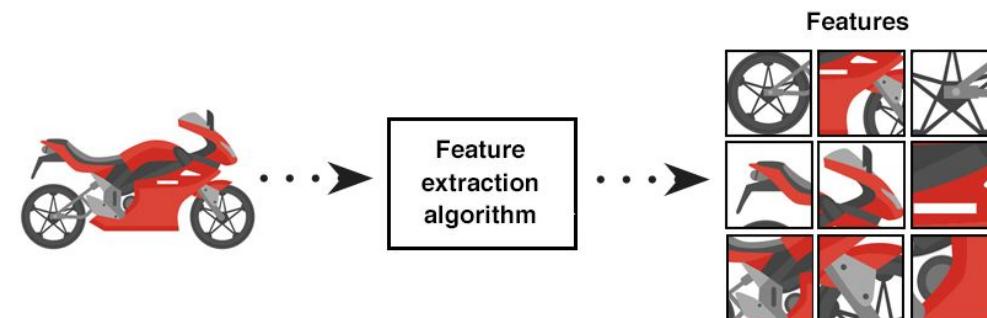
# Fasi dell'Object Detection

## Feature Extraction

Utilizzo delle **CNN** per identificare caratteristiche rilevanti delle immagini

Tipologie di Layer nelle CNN:

- **Convolutional Layer** - estrazione caratteristiche tramite kernel
- **Activation Layer** - funzione di attivazione
- **Pooling Layers** - riduzione dimensioni



# Fasi dell'Object Detection

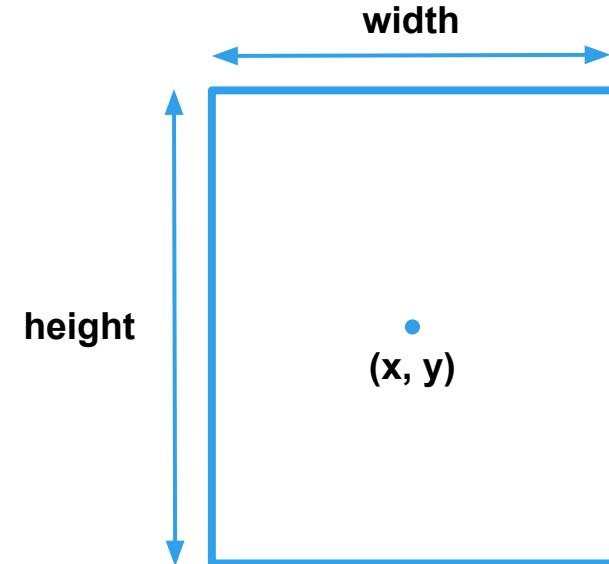
## Bounding Box Prediction e Classification

Modelli basati su **Region Proposals**

- Selective Search o RPN

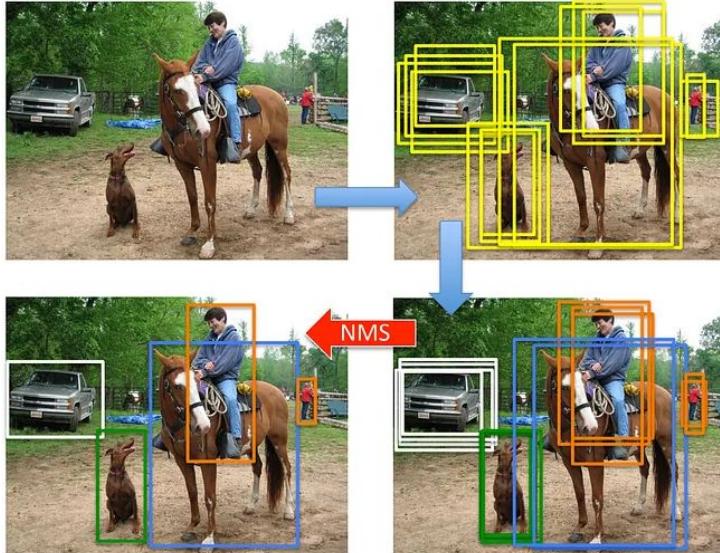
Modelli **Single-Shot**

- Suddivisione in Griglia
- Objectness Score



# Fasi dell'Object Detection

## Non-Maximum Suppression



Rimuove proposte sovrapposte ad altre di punteggio più alto, mantenendo solo quelle più significative



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Da un secolo, oltre.



HR EXCELLENCE IN RESEARCH

2

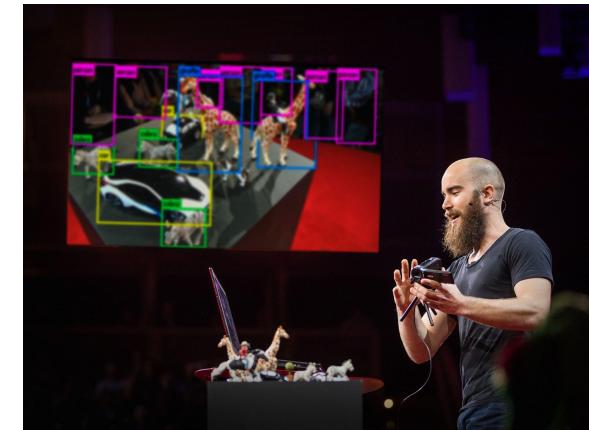
YOLO

## You Only Look Once

Ideato da Joseph Redmon, Santosh Divvala, Ross Girshick  
e Ali Farhadi nel 2016

Principi di design:

- **Approccio Globale** all'Immagine
- Efficienza Computazionale
- Semplicità e Generalizzazione



Da un secolo, oltre.

HR EXCELLENCE IN RESEARCH

## Modelli utilizzati

- YOLOv8
- YOLO-World
- RT-DETR

## Caratteristiche

Ottava versione del modello YOLO progettata per:

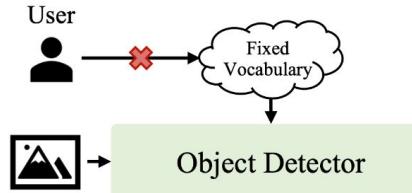
- Massimizzare la velocità di inferenza
- Offrire una **configurazione ed un utilizzo semplificati**
- Essere altamente efficiente su una vasta gamma di hardware,  
in particolare **dispositivi con risorse limitate**

## Architettura

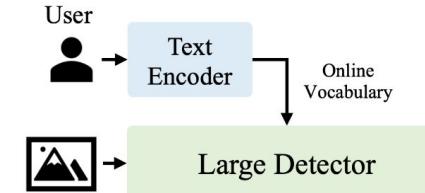
- **Backbone** - responsabile della Feature Extraction
- **Neck** - elabora le feature per il blocco successivo
- **Head** - genera le bounding box ed i punteggi di confidenza, classificando gli elementi

## Open-Vocabulary Object Detection

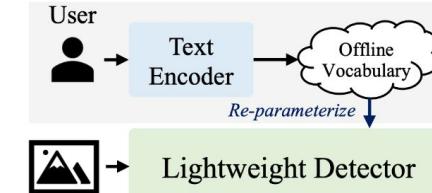
Consente di specificare dinamicamente le classi, permettendo di adattare il modello a nuovi domini o compiti specifici che non erano originariamente parte dei dati di addestramento



(a) Traditional Object Detector



(b) Previous Open-Vocabulary Detector



(c) YOLO-World

## Architettura

- **YOLOv8 Backbone** - estraie le feature visive
- **Text Encoder** - trasforma il linguaggio naturale in embedding testuali
- **RepVL-PAN** - combina feature visive ed embedding testuali tramite:
  - T-CSP Layer, il quale potenzia le feature tramite le indicazioni degli embedding
  - Image-Pooling Attention, che incorpora il contesto visivo globale negli embedding testuali

## Real-Time e Vision Transformers

Sviluppato specificatamente per l'Object Detection in tempo reale

Utilizza i **Vision Transformers**:

- Architettura dei Transformer applicata alle immagini
- Capacità di catturare relazioni a lungo raggio tra diverse parti dell'immagine
- Comprensione contestuale più ricca rispetto ai modelli CNN tradizionali

## Architettura

- **Efficient Hybrid Encoder** - sfrutta i ViT per acquisire una comprensione globale delle immagini
- **IoU Aware Query Selection** - modulo che seleziona le feature in base all'indice di sovrapposizione



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Da un secolo, oltre.



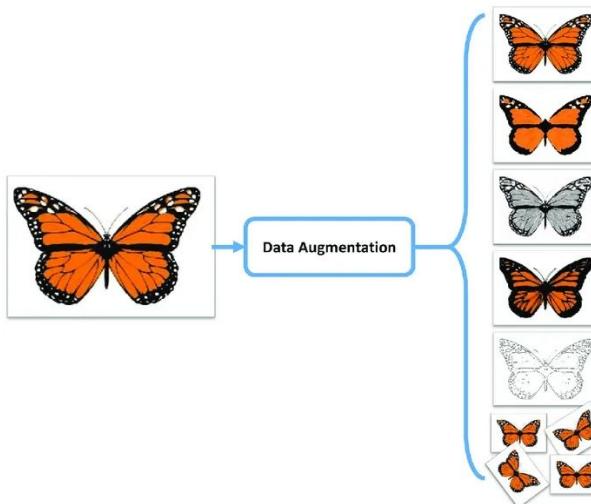
HR EXCELLENCE IN RESEARCH

# Data Augmentation

3

# Data Augmentation

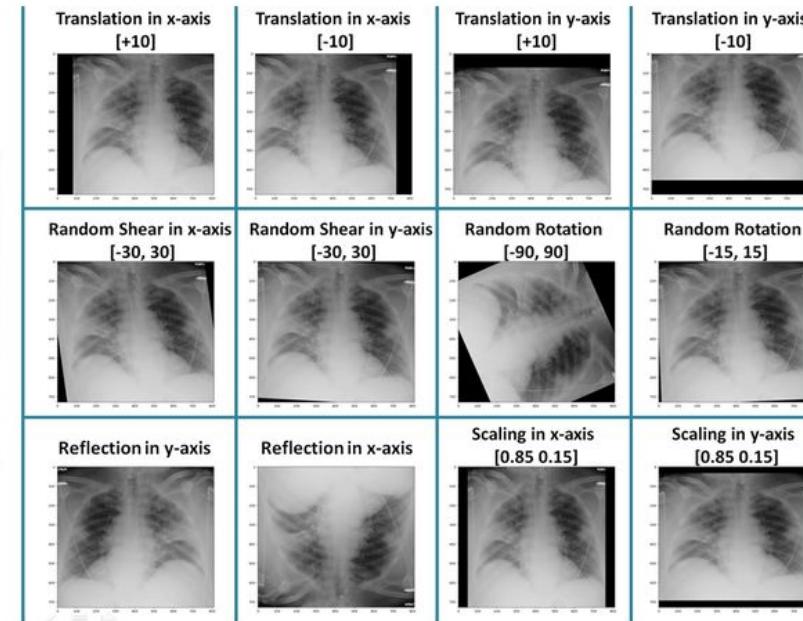
## Definizione e Scopo



- Tecnica per incrementare la quantità e la varietà dei dati di addestramento
- Migliora la generalizzazione tramite:
  - Aumento della varietà dei dati
  - Riduzione dell'overfitting
  - Bilanciamento dei dataset

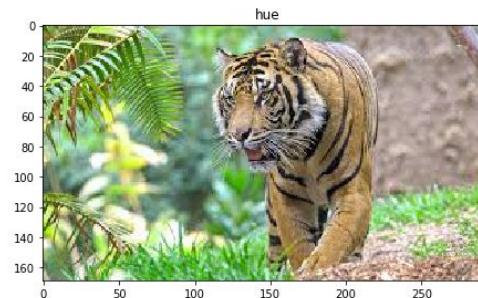
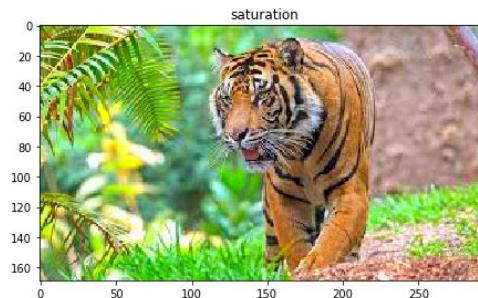
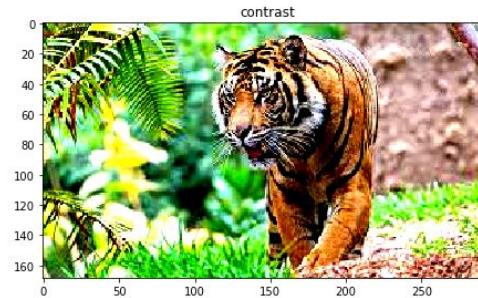
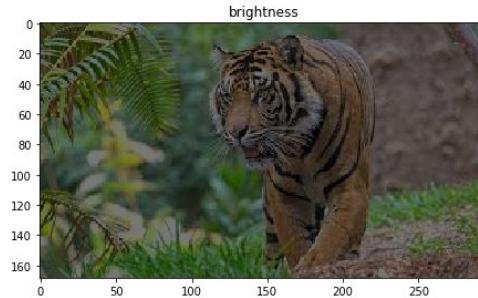
# Tecniche Base di Data Augmentation

## Trasformazioni Geometriche



# Tecniche Base di Data Augmentation

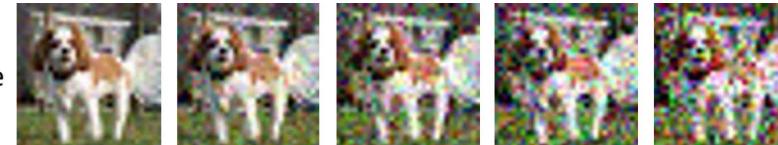
## Modifiche dei Colori



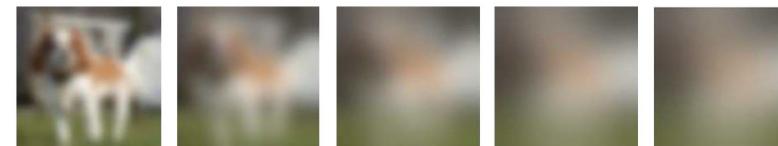
# Tecniche Base di Data Augmentation

## Distorsioni e Rumore

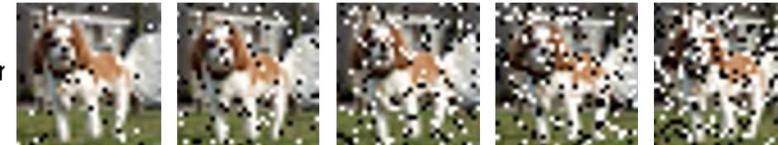
Gaussian Noise



Gaussian Blur



Salt and Pepper

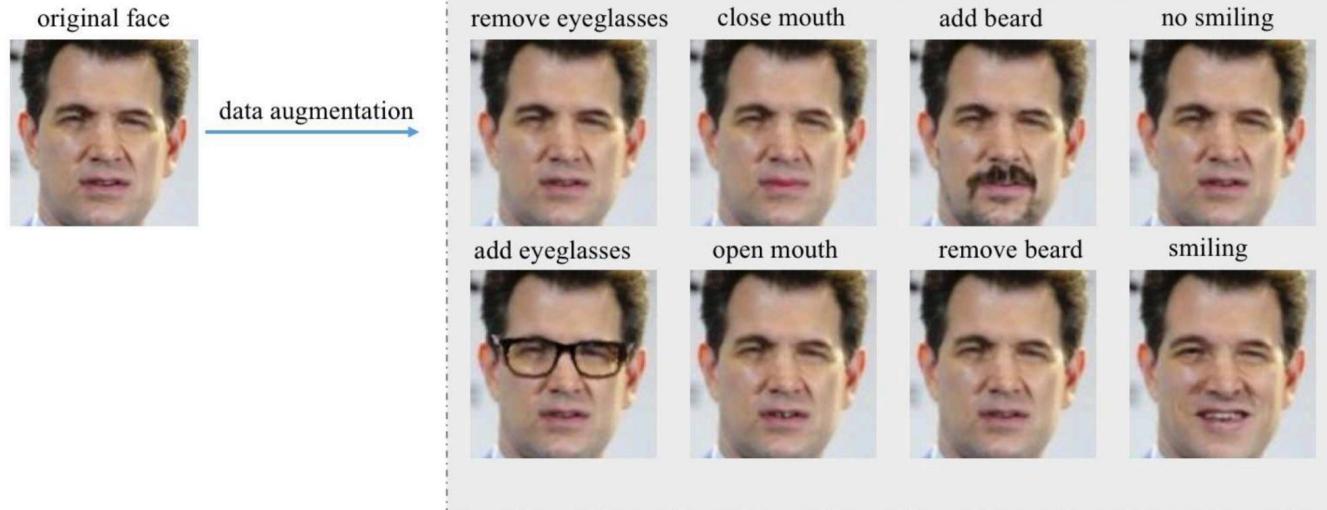


Motion Blur



# Tecniche Avanzate di Data Augmentation

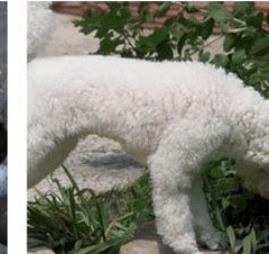
## Augmentation tramite GAN



# Tecniche Avanzate di Data Augmentation

## Mixup, Cutout e Cutmix

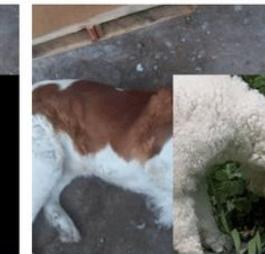
Original samples



Mixup



Cutout



Cutmix



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Da un secolo, oltre.



4

# Sperimentazione e Risultati

# Preparazione del dataset

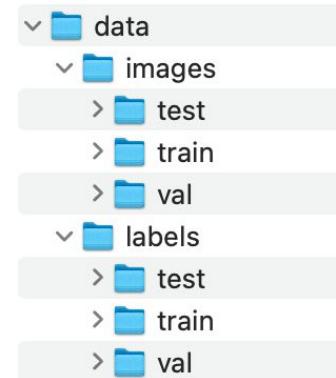
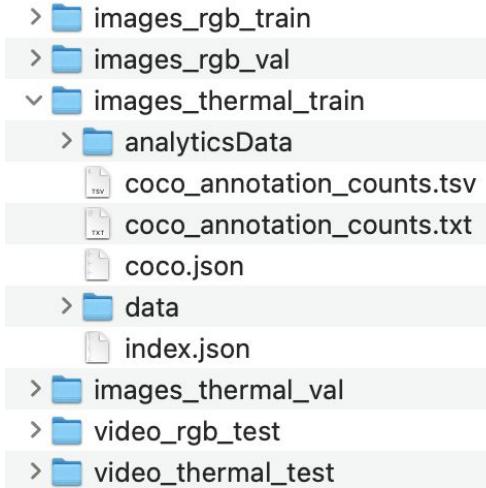
## Dataset termico FLIR

- Fornito da Teledyne FLIR
- Immagini catturate per il 60% di giorno ed il 40% di notte (40%) su strade e autostrade nell'area di Santa Barbara (California) da Novembre a Maggio, con varie condizioni meteorologiche.
- Immagini acquisite tramite termocamera ad infrarossi Tau2

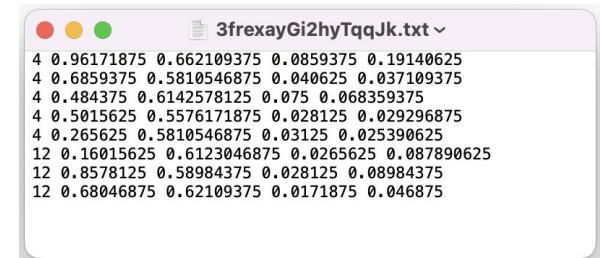


# Preparazione del dataset

## Estrazione e Formattazione



- Immagini “thermal”
- Annotazioni da formato MSCOCO/Conservator a formato YOLO



A screenshot of a terminal window showing a text file named '3frexayGi2hyTqqJk.txt'. The file contains YOLO formatted bounding box annotations for thermal images, including class ID, center coordinates, width, height, and confidence scores.

```
3frexayGi2hyTqqJk.txt ~
4 0.96171875 0.662109375 0.0859375 0.19140625
4 0.6859375 0.5810546875 0.040625 0.037109375
4 0.484375 0.6142578125 0.075 0.068359375
4 0.5015625 0.5576171875 0.028125 0.029296875
4 0.265625 0.5810546875 0.03125 0.025390625
12 0.16015625 0.6123046875 0.0265625 0.087890625
12 0.8578125 0.58984375 0.028125 0.08984375
12 0.68046875 0.62109375 0.0171875 0.046875
```

## Filtraggio delle classi

# Preparazione del dataset

TRAIN DATA			VAL DATA			TEST DATA		
Class	Count	%	Class	Count	%	Class	Count	%
animal	8	0,00%	animal	0	0,00%	animal	0	0,00%
bike	7237	4,11%	bike	170	1,01%	bike	113	0,18%
bird	1	0,00%	bird	0	0,00%	bird	0	0,00%
bus	2245	1,28%	bus	179	1,07%	bus	0	0,00%
car	73623	41,82%	car	7133	42,49%	car	30517	48,86%
dog	4	0,00%	dog	0	0,00%	dog	25	0,04%
face	752	0,43%	face	73	0,43%	face	142	0,23%
hydrant	1095	0,62%	hydrant	94	0,56%	hydrant	277	0,44%
license plate	270	0,15%	license plate	17	0,10%	license plate	0	0,00%
light	16198	9,20%	light	2005	11,94%	light	6758	10,82%
motor	1116	0,63%	motor	55	0,33%	motor	3314	5,31%
other vehicle	1373	0,78%	other vehicle	63	0,38%	other vehicle	696	1,11%
person	44527	25,29%	person	4309	25,67%	person	11242	18,00%
rider	5951	3,38%	rider	161	0,96%	rider	1081	1,73%
scooter	15	0,01%	scooter	0	0,00%	scooter	0	0,00%
sign	20770	11,80%	sign	2472	14,73%	sign	5660	9,06%
skateboard	29	0,02%	skateboard	3	0,02%	skateboard	0	0,00%
stroller	15	0,01%	stroller	6	0,04%	stroller	0	0,00%
train	5	0,00%	train	0	0,00%	train	0	0,00%
truck	829	0,47%	truck	46	0,27%	truck	2634	4,22%
<b>Total</b>	<b>176063</b>		<b>Total</b>	<b>16786</b>		<b>Total</b>	<b>62459</b>	
<b>Filtered</b>	<b>4439</b>		<b>Filtered</b>	<b>372</b>		<b>Filtered</b>	<b>444</b>	
<b>Total Filtered</b>	<b>171624</b>		<b>Total Filtered</b>	<b>16414</b>		<b>Total Filtered</b>	<b>62015</b>	

# Preparazione del dataset

## Nuovo dataset filtrato

Dataset ***filtered-data***:

- ***train*** subset: 10742 esempi
- ***val*** subset: 1144 esempi
- ***test*** subset: 3749 esempi

# Modelli utilizzati

Da un secolo, oltre.

## Modelli di Ultralytics

- **yolov8n**: modello YOLOv8 “nano”
- **yolov8s-world**: modello YOLOv8 “small-world” (YOLO-World)
- **rtdetr-l**: modello RT-DETR “large”

## Metodo e Metriche per la valutazione

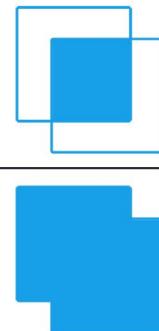
### Script Python:

```
1 from ultralytics import YOLO, YOLOWorld, RTDETR
2
3 model = YOLO('yolov8n.pt')
4 metrics = model.val(data='test-config.yaml')
5
6 print("map50-95: ", metrics.box.map)
7 print("map50: ", metrics.box.map50)
8 print("map75: ", metrics.box.map75)
9 print("maps: ", metrics.box.maps)
```

- Average Precision
- Intersection over Union (IoU)



- mAP50-95
- mAP50
- mAP75

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


# Modelli utilizzati

Da un secolo, oltre.

## Test dei modelli non addestrati

yolov8n	
mAP50-95	0,0092
mAP50	0,0160
mAP75	0,0079
AP per class	
bike	0,0621
car	0,0000
light	0,0092
motor	0,0158
other vehicle	0,0000
person	0,0000
rider	0,0000
sign	0,0092
truck	0,0000

yolov8s-world	
mAP50-95	0,0086
mAP50	0,0153
mAP75	0,0086
AP per class	
bike	0,0537
car	0,0000
light	0,0086
motor	0,0109
other vehicle	0,0000
person	0,0000
rider	0,0000
sign	0,0086
truck	0,0000

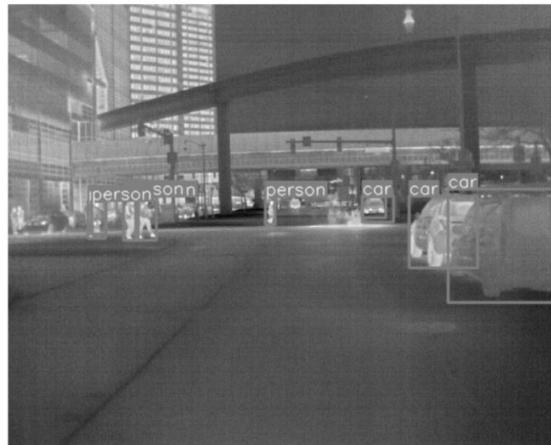
rtdetr-l	
mAP50-95	0,0026
mAP50	0,0066
mAP75	0,0017
AP per class	
bike	0,0000
car	0,0000
light	0,0026
motor	0,0236
other vehicle	0,0000
person	0,0000
rider	0,0000
sign	0,0026
truck	0,0001

mAP50-95 on COCO	
yolov8n	0,373
yolov8s-world	0,374
rtdetr-l	0,53

# Modelli utilizzati

Da un secolo, oltre.

## Detections effettuate dai modelli non addestrati



yolov8n



yolov8s-world



rtdetr-l

# Addestramento su dataset filtrato

## Metodo per l'addestramento

Script Python:

```
1 from ultralytics import YOLO, YOLOWorld, RTDETR
2
3 model = YOLO('yolov8n.pt')
4 results = model.train(data='train-config.yaml', device=0, epochs=50, batch=8, verbose=True)
```

Parametri:

- data
- device
- epochs
- batch
- verbose

Augmentations:

- Adjustment of Hue
- Adjustment of Saturation
- Adjustment of Brightness
- Translation
- Scaling
- Horizontal Flipping
- Mixup
- Cutout
- Crop

# Addestramento su dataset filtrato

## Test dei modelli addestrati su filtered-data

yolov8n	
Dataset	filtered-data
Epochs	50
Batch	8
mAP50-95	0,1421
mAP50	0,2617
mAP75	0,1344
AP per class	bike
	car
	light
	motor
	other vehicle
	person
	rider
	sign
	truck

yolov8s-world	
Dataset	filtered-data
Epochs	50
Batch	8
mAP50-95	0,1689
mAP50	0,3063
mAP75	0,1617
AP per class	bike
	car
	light
	motor
	other vehicle
	person
	rider
	sign
	truck

rtdetr-l	
Dataset	filtered-data
Epochs	50
Batch	8
mAP50-95	0,1820
mAP50	0,3348
mAP75	0,1682
AP per class	bike
	car
	light
	motor
	other vehicle
	person
	rider
	sign
	truck

# Addestramento su dataset filtrato

Confronto risultati dei modelli addestrati su filtered-data con i precedenti

yolov8n			
Dataset: filtered-data		Epochs: 50	Batch: 8
mAP50-90	0,1421	+1531,43%	vs untrained
mAP50	0,2617	+1439,76%	vs untrained
mAP75	0,1344	+1595,20%	vs untrained

yolov8s-world			
Dataset: filtered-data		Epochs: 50	Batch: 8
mAP50-90	0,1689	+1852,79%	vs untrained
mAP50	0,3063	+1897,01%	vs untrained
mAP75	0,1617	+1777,70%	vs untrained

rtdetr-l			
Dataset: filtered-data		Epochs: 50	Batch: 8
mAP50-90	0,1820	+6787,38%	vs untrained
mAP50	0,3348	+3967,83%	vs untrained
mAP75	0,1682	+9982,10%	vs untrained

# Addestramento su dataset filtrato

Detections effettuate dai modelli addestrati su filtered-data



yolov8n



yolov8s-world



rtdetr-l

# Addestramento su dataset aumentato

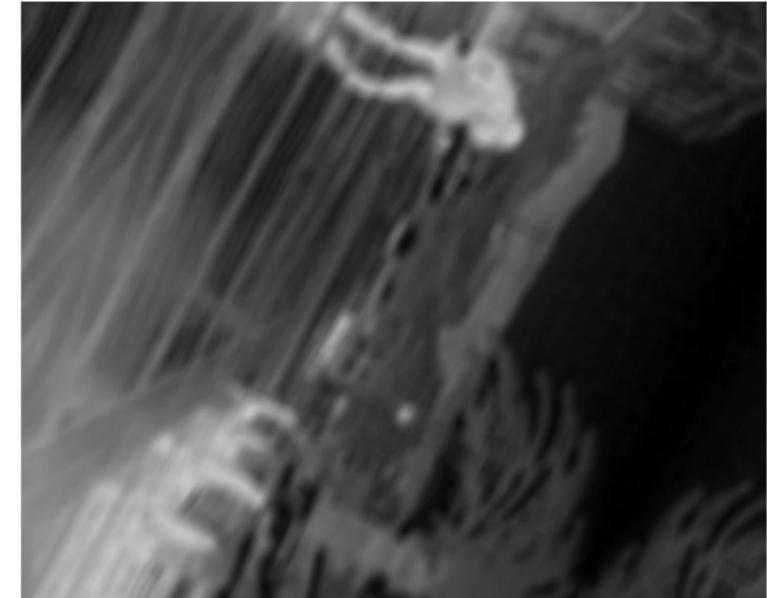
## Prima Data Augmentation

- Albumentations
- Script Python che utilizza il metodo ***transform***
- Dataset ***augmented-data-1*** con 21484 esempi nel ***train*** subset

```
transform = A.Compose([
    A.Blur(blur_limit=7, p=0.5),
    A.RandomBrightnessContrast(p=0.5),
    A.RandomResizedCrop(size=(dataset_height, dataset_width), scale=(0.2, 1), p=1),
    A.Rotate(limit=(-180,180), p=1),
    A.Sharpen(p=0.5),
    A.VerticalFlip(p=0.5)
], bbox_params=A.BboxParams(format='yolo', label_fields=['class_labels']))
```

# Addestramento su dataset aumentato

## Prima Data Augmentation



# Addestramento su dataset aumentato

## Test dei modelli addestrati su augmented-data-1

yolov8n	
Dataset	augmented-data-1
Epochs	50
Batch	8
mAP50-95	0,1378
mAP50	0,2539
mAP75	0,1277
AP per class	bike
	car
	light
	motor
	other vehicle
	person
	rider
	sign
	truck

yolov8s-world	
Dataset	augmented-data-1
Epochs	50
Batch	8
mAP50-95	0,1693
mAP50	0,3055
mAP75	0,1606
AP per class	bike
	car
	light
	motor
	other vehicle
	person
	rider
	sign
	truck

rtdetr-l	
Dataset	augmented-data-1
Epochs	50
Batch	8
mAP50-95	0,1384
mAP50	0,2684
mAP75	0,1228
AP per class	bike
	car
	light
	motor
	other vehicle
	person
	rider
	sign
	truck

# Addestramento su dataset aumentato

Confronto risultati dei modelli addestrati su augmented-data-1 con i precedenti

yolov8n			
Dataset: augmented-data-1		Epochs: 50	Batch: 8
mAP50-90	0,1378	+1393,09% -3,03%	vs untrained vs filtered-data training
mAP50	0,2539	+1482,54% -3,00%	vs untrained vs filtered-data training
mAP75	0,1277	+1510,81% -4,98%	vs untrained vs filtered-data training

yolov8s-world			
Dataset: augmented-data-1		Epochs: 50	Batch: 8
mAP50-90	0,1693	+1857,28% +0,23%	vs untrained vs filtered-data training
mAP50	0,3055	+1897,01% -0,25%	vs untrained vs filtered-data training
mAP75	0,1606	+1765,05% -0,67%	vs untrained vs filtered-data training

rtdetr-l			
Dataset: augmented-data-1		Epochs: 50	Batch: 8
mAP50-90	0,1384	+5137,44% -23,96%	vs untrained vs filtered-data training
mAP50	0,2684	+3967,83% -19,81%	vs untrained vs filtered-data training
mAP75	0,1228	+7259,16% -27,01%	vs untrained vs filtered-data training

# Addestramento su dataset aumentato

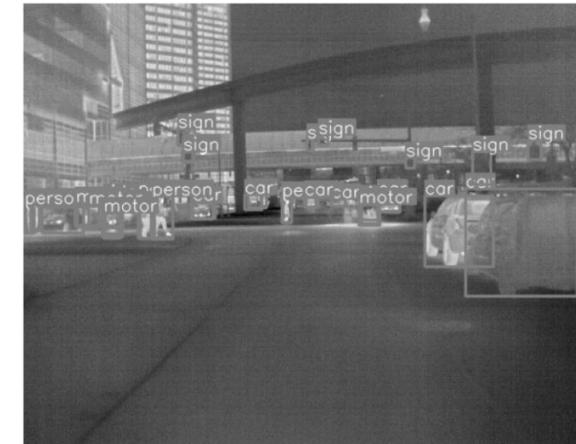
Detections effettuate dai modelli addestrati su augmented-data-1



yolov8n



yolov8s-world



rtdetr-l

# Addestramento su dataset aumentato

## Seconda Data Augmentation

- Trasformazioni più adeguate alle immagini termiche
- Dataset ***augmented-data-2***

```
transform = A.Compose([
    A.Blur(blur_limit=3, p=0.5),
    A.CLAHE(clip_limit=2, p=0.5),
    A.GaussianBlur(blur_limit=(3, 7), p=0.5),
    A.GaussNoise(var_limit=(10.0, 50.0), p=0.5),
    A.MultiplicativeNoise(multiplier=(0.9, 1.1), p=0.5),
    A.RandomBrightnessContrast(brightness_limit=0.2, contrast_limit=0.2, p=0.5),
    A.Rotate(limit=10, p=0.5),
    A.Sharpen(alpha=(0.2, 0.5), lightness=(0.5, 1.0), p=0.5),
    A.ShiftScaleRotate(shift_limit=0.0625, scale_limit=0.1, rotate_limit=10, p=0.5)
], bbox_params=A.BboxParams(format='yolo', label_fields=['class_labels']))
```

# Addestramento su dataset aumentato

## Seconda Data Augmentation



# Addestramento su dataset aumentato

## Test dei modelli addestrati su augmented-data-2

yolov8n		yolov8s-world		rtdetr-l		
Dataset	augmented-data-2	Dataset	augmented-data-2 <th>Dataset</th> <td>augmented-data-2</td>	Dataset	augmented-data-2	
Epochs	50	Epochs	50	Epochs	50	
Batch	8	Batch	8	Batch	8	
mAP50-95	0,1526	mAP50-95	0,1707	mAP50-95	0,1599	
mAP50	0,2826	mAP50	0,3110	mAP50	0,3065	
mAP75	0,1431	mAP75	0,1608	mAP75	0,1462	
AP per class	bike	0,0026	bike	0,0014	bike	0,0004
	car	0,4715	car	0,5166	car	0,4672
	light	0,0503	light	0,0832	light	0,1147
	motor	0,1097	motor	0,1251	motor	0,1318
	other vehicle	0,0964	other vehicle	0,0983	other vehicle	0,0366
	person	0,4417	person	0,48072	person	0,44891
	rider	0,0082	rider	0,0177	rider	0,0219
	sign	0,0715	sign	0,0871	sign	0,1077
	truck	0,1215	truck	0,1265	truck	0,1103

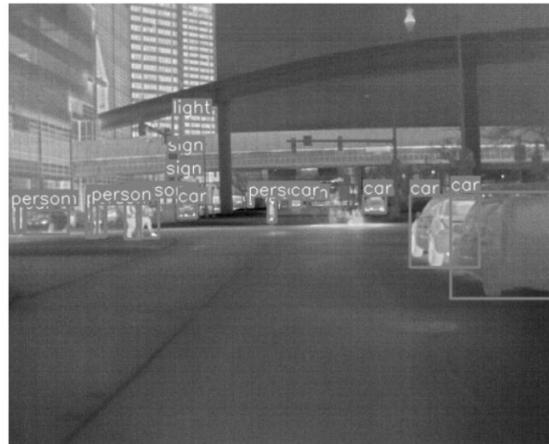
# Addestramento su dataset aumentato

Confronto risultati dei modelli addestrati su augmented-data-2 con i precedenti

yolov8n			yolov8s-world			rtdetr-l		
Dataset: augmented-data-2		Epochs: 50	Dataset: augmented-data-2		Epochs: 50	Dataset: augmented-data-2		Epochs: 50
mAP50-90	0,1526	+1553,22% vs untrained	+1,09% vs filtered-data training	+1874,16% vs untrained	+1,09% vs filtered-data training	+5951,82% vs untrained	-12,13% vs filtered-data training	+15,55% vs augmented-data-1 training
mAP50	0,2826	+1661,49% vs untrained	+7,97% vs filtered-data training	+1,52% vs untrained	+1,52% vs filtered-data training	+4544,11% vs untrained	-8,46% vs filtered-data training	+14,17% vs augmented-data-1 training
mAP75	0,1431	+1704,28% vs untrained	+6,43% vs filtered-data training	+1,78% vs augmented-data-1 training	+1767,76% vs untrained	+8661,82% vs untrained	-13,10% vs filtered-data training	+19,06% vs augmented-data-1 training
		+10,72% vs augmented-data-1 training		+0,53%				
				+0,15%				

# Addestramento su dataset aumentato

Detections effettuate dai modelli addestrati su augmented-data-2



yolov8n



yolov8s-world



rtdetr-l

# Addestramento su dataset aumentato

Test dei modelli addestrati su augmented-data-2 per 100 epochs

yolov8n		
Dataset	augmented-data-2	
Epochs	100	
Batch	8	
mAP50-95	mAP50-95	0,1618
mAP50	mAP50	0,3092
mAP75	mAP75	0,1594
AP per class	bike	0,0059
	car	0,4849
	light	0,0705
	motor	0,1312
	other vehicle	0,0953
	person	0,4657
	rider	0,0138
	sign	0,0748
	truck	0,1139

yolov8s-world		
Dataset	augmented-data-2	
Epochs	100	
Batch	8	
mAP50-95	mAP50-95	0,1732
mAP50	mAP50	0,3176
mAP75	mAP75	0,1647
AP per class	bike	0,0009
	car	0,5170
	light	0,0882
	motor	0,1358
	other vehicle	0,1093
	person	0,4807
	rider	0,0162
	sign	0,0829
	truck	0,1276

rtdetr-l		
Dataset	augmented-data-2	
Epochs	100	
Batch	8	
mAP50-95	mAP50-95	0,1428
mAP50	mAP50	0,2879
mAP75	mAP75	0,1247
AP per class	bike	0,0062
	car	0,4119
	light	0,0791
	motor	0,1449
	other vehicle	0,0286
	person	0,3918
	rider	0,0238
	sign	0,0848
	truck	0,1140

# Addestramento su dataset aumentato

Confronto risultati dei modelli addestrati su augmented-data-2 per 100 epochs con i precedenti

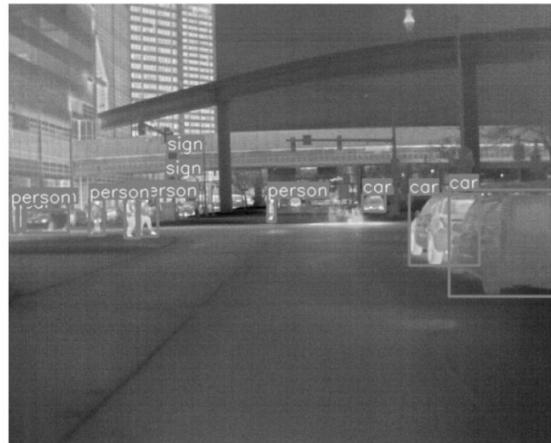
yolov8n			
Dataset: augmented-data-2		Epochs: 100	Batch: 8
mAP50-90	0,1618	+1653,15%	vs untrained
		+13,88%	vs filtered-data training
		+17,42%	vs augmented-data-1 training
		+6,04%	vs augmented-data-2 training (50)
mAP50	0,3092	+1826,99%	vs untrained
		+18,12%	vs filtered-data training
		+21,76%	vs augmented-data-1 training
		+9,40%	vs augmented-data-2 training (50)
mAP75	0,1594	+1909,98%	vs untrained
		+18,57%	vs filtered-data training
		+24,78%	vs augmented-data-1 training
		+11,40%	vs augmented-data-2 training (50)

yolov8s-world			
Dataset: augmented-data-2		Epochs: 100	Batch: 8
mAP50-90	0,1732	+1902,75%	vs untrained
		+2,56%	vs filtered-data training
		+2,32%	vs augmented-data-1 training
		+1,45%	vs augmented-data-2 training (50)
mAP50	0,3176	+1975,52%	vs untrained
		+3,67%	vs filtered-data training
		+3,93%	vs augmented-data-1 training
		+2,11%	vs augmented-data-2 training (50)
mAP75	0,1647	+1812,50%	vs untrained
		+1,85%	vs filtered-data training
		+2,54%	vs augmented-data-1 training
		+2,40%	vs augmented-data-2 training (50)

rtdetr-l			
Dataset: augmented-data-2		Epochs: 100	Batch: 8
mAP50-90	0,1428	+5303,50%	vs untrained
		-21,54%	vs filtered-data training
		+3,17%	vs augmented-data-1 training
		-10,71%	vs augmented-data-2 training (50)
mAP50	0,2879	+4263,09%	vs untrained
		-13,99%	vs filtered-data training
		+7,26%	vs augmented-data-1 training
		-6,05%	vs augmented-data-2 training (50)
mAP75	0,1247	+7377,41%	vs untrained
		-25,83%	vs filtered-data training
		+1,61%	vs augmented-data-1 training
		-14,66%	vs augmented-data-2 training (50)

# Addestramento su dataset aumentato

Detections effettuate dai modelli addestrati su augmented-data-2 per 100 epochs



yolov8n



yolov8s-world



rtdetr-l

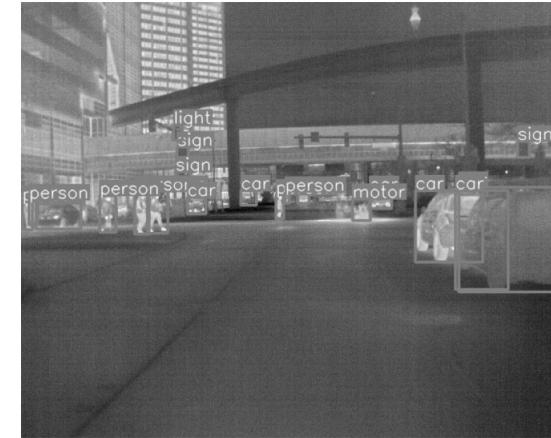
# Conclusioni

Da un secolo, oltre.

## Modello ottimale

yolov8l		
Dataset	augmented-data-2	
Epochs	100	
Batch	8	
mAP50-95	0,2076	+28,25%
mAP50	0,3555	+14,99%
mAP75	0,1966	+23,35%
AP per class	bike	0,0082
	car	0,5714
	light	0,1128
	motor	0,1495
	other vehicle	0,1212
	person	0,5351
	rider	0,0374
	sign	0,1188
	truck	0,2138
		+38,56%
		+17,82%
		+60,03%
		+13,97%
		+27,22%
		+14,92%
		+171,23%
		+58,70%
		+87,62%

- Performance migliori di YOLOv8
- Addestramento e test della versione “Large” **yolov8l**



yolov8l



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Da un secolo, oltre.



HR EXCELLENCE IN RESEARCH

# Grazie per l'attenzione