

## Rapport de stage de fin d'année

Filière : Smart information and communication technologies engineering

---

### Développement d'une application de vérification des standards de sécurité des opérateurs

---

Réalisé par :

Khadija GOUAGHOU

Encadré par :

M. Omar SOUISSI

INPT

M. Samir TETOUANI

Laboratoire en Ingénierie Numérique Avancée, ESITH



AGENCE NATIONALE DE RÉGLEMENTATION DES TÉLÉCOMMUNICATIONS  
INSTITUT NATIONAL DES POSTES ET TÉLÉCOMMUNICATIONS

# Abstract

As part of the development of its industry 4.0 smart factory project, the Research and Development Laboratory of Advanced Numerical Engineering (LINA) at the Superior School of Textile and Apparel (ESITH) wishes to implement an application for verifying the adherence to security standards by operators in the factory (e.g., wearing gloves, glasses, aprons, etc.) and monitoring their presence in the factory by analyzing images captured by the cameras. The application will utilize open-source tools to recognize the safety equipment worn by operators.

This report mainly focuses on presenting the architecture of the application, with a particular emphasis on the development of the machine learning models that will handle image recognition and computer vision. Subsequently, we will test this application in the laboratory's factory using video data from the cameras.

Key words : Monitoring, Computer vision, Image recognition, Face detection, Face recognition, YOLOv10, FaceNet,...

# Table of contents

1	General project context
1.1.	Presentation of LINA
1.2.	Presentation of the project
1.3.	Introduction to the R&D center at ESITH
2	.....
2.1.	.....
2.2.	....
2.3.	...
3	....
4	...
5	

# List of figures

- 1.1 LINA logo
- 1.2 Fabric cutting machine that requires the use of iron gloves
- 1.3 Iron cut-proof gloves
- 2.1 YOLOv10 AP compared to other state-of-the-art detectors
- 2.2 Google Open Images v7 'Human face' class with bounding boxes for detection tasks
- 2.3 OIDv4\_ToolKit for downloading images from 'Human face' OID v4 class
- 2.4 Yolo annotations

# Chapter 1

## General Project Context

### Introduction

In this chapter, we aim to provide a comprehensive understanding of the project and internship experience by establishing the context. We begin with an introduction to LINA, the host organization, followed by an in-depth exploration of the project. This includes a detailed description of the project's scope and an outline of the various tasks accomplished during the internship.

## 1.1. Presentation of LINA :



Figure 1.1 : LINA logo

LINA (Laboratoire en Ingénierie Numérique Avancée) or the Laboratory of Advanced Numerical Engineering is a Research & Development laboratory founded in the Superior School of Textile and Apparel (ESITH) in Casablanca, Morocco with the mission to promote applied research in the domain of advanced numerics, develop students skills in advanced numerics research and contribute to the evolution of study programs.

It focuses mainly on the digital transformation of 4.0 industry, the development of smart captors and textile (IoT) and on digital factories/digital twins.

## 1.2. Presentation of the project :

This project aims to develop an application that uses computer vision techniques to monitor and enforce safety standards among operators in industrial settings. By analyzing real-time video footage captured by cameras positioned strategically in the factory, the application will detect and verify if operators are wearing essential safety equipment such as gloves, goggles, aprons, badges, and other prescribed items according to their role in the factory while also monitoring their presence in the factory (Arrival, breaks, etc.).

Key components of the application include leveraging open-source tools and libraries specialized in image recognition and object detection. These tools will enable the system to accurately identify and classify safety gear worn by operators. The application will employ machine learning algorithms trained on datasets to improve the accuracy and reliability of detection.

The ultimate goal is to enhance workplace safety by automating the monitoring process, ensuring compliance with safety protocols without relying solely on manual inspections. This approach not only improves operational efficiency but also reduces the risk of accidents and promotes a safer working environment for all personnel involved.

## 1.3. Introduction to the Research & Development center at ESITH:

We began our internship with a visit to the Research & Development Department at ESITH on Monday, July 1st, 2024. Professor Samir TETOUANI guided me (Khadija GOUAGHOU), along with my colleague Mohamed Elaouan and our supervisor, Dr. Omar Souissi, through the factory where we will be testing our application at the conclusion of our two-month internship at LINA. Accompanied by ESITH staff, Professor TETOUANI showcased the

work environment and demonstrated the safety gear operators wear when operating specific machinery.



Figure 1.2 : Fabric cutting machine that requires the use of iron gloves



Figure 1.3 : Iron cut-proof gloves

# Chapter 2

## Factory operators attendance monitoring system

### Introduction

In this chapter, we will present a Factory Operators Attendance Monitoring System that utilizes advanced face detection and recognition models. By incorporating recent developments in deep learning, the system aims to improve workforce management through accurate and efficient operator identification, facilitating real-time attendance tracking with reliable facial recognition technology.



## 2.1. Face Detection :

### 2.1.1. YOLOv10 :

YOLOv10, released in May 2024 and built on the Ultralytics Python package by researchers at Tsinghua University, introduces a new approach to real-time object detection, addressing both the post-processing and model architecture deficiencies found in previous YOLO versions. By eliminating non-maximum suppression (NMS) and optimizing various model components, YOLOv10 achieves state-of-the-art performance with significantly reduced computational overhead. Extensive experiments demonstrate its superior accuracy-latency trade-offs across multiple model scales.

**Non-maximum suppression (NMS) :** A post-processing technique used in object detection to eliminate duplicate detections and select the most relevant detected objects. This helps reduce false positives and the computational complexity of a detection algorithm. It works by comparing the confidence scores of the proposed bounding boxes and eliminating the ones that overlap significantly with a higher-scoring bounding box.

The YOLOv10 model achieves a higher mean Average Precision (mAP) compared to earlier YOLO models such as YOLOv9, YOLOv8, and YOLOv7 when benchmarked against the MS COCO dataset.

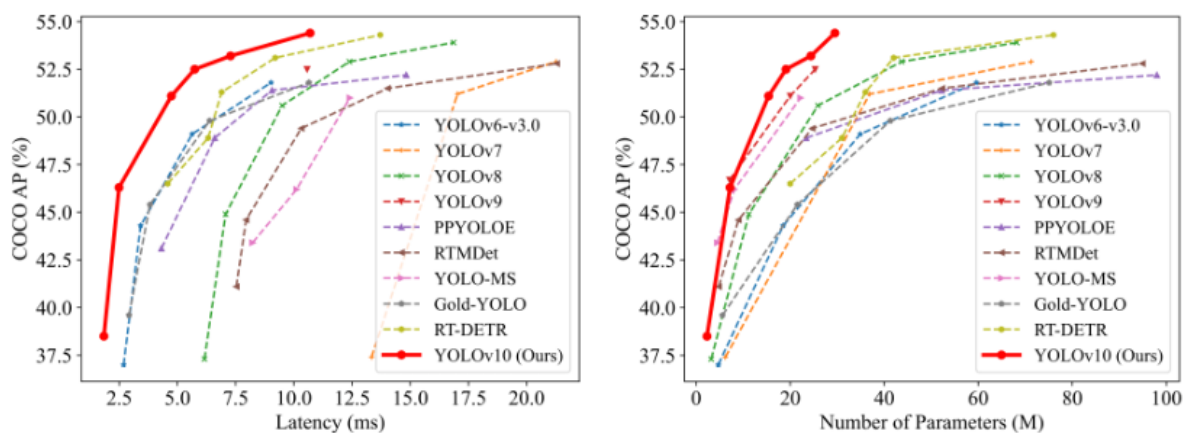


Figure 2.1 :YOLOv10 AP compared to other state-of-the-art detectors

#### 2.1.1.1. Architecture :

The architecture of YOLOv10 builds upon the strengths of previous YOLO models while introducing several key innovations. It consists of the following components:

1. **Backbone:** Responsible for feature extraction, the backbone in YOLOv10 uses an enhanced version of CSPNet (Cross Stage Partial Network) to improve gradient flow and reduce computational redundancy.
2. **Neck:** The neck is designed to aggregate features from different scales and passes them to the head. It includes PAN (Path Aggregation Network) layers for effective multiscale feature fusion.

3. One-to-Many Head: Generates multiple predictions per object during training to provide rich supervisory signals and improve learning accuracy.
4. One-to-One Head: Generates a single best prediction per object during inference to eliminate the need for NMS, thereby reducing latency and improving efficiency.

#### 2.1.1.2. Key Features :

**NMS-Free Training:** Utilizes consistent dual assignments to eliminate the need for NMS, reducing inference latency.

**Holistic Model Design:** Comprehensive optimization of various components from both efficiency and accuracy perspectives, including lightweight classification heads, spatial-channel decoupled down sampling, and rank-guided block design.

**Enhanced Model Capabilities:** Incorporates large-kernel convolutions and partial self-attention modules to improve performance without significant computational cost.

#### 2.1.1.3. Methodology :

**Consistent Dual Assignments for NMS-Free Training :**

YOLOv10 employs dual label assignments, combining one-to-many and one-to-one strategies during training to ensure rich supervision and efficient end-to-end deployment. The consistent matching metric aligns the supervision between both strategies, enhancing the quality of predictions during inference.

**Holistic Efficiency-Accuracy Driven Model Design :**

##### 1. Efficiency Enhancements

**Lightweight Classification Head:** Reduces the computational overhead of the classification head by using depth-wise separable convolutions.

**Spatial-Channel Decoupled Down sampling:** Decouples spatial reduction and channel modulation to minimize information loss and computational cost.

**Rank-Guided Block Design:** Adapts block design based on intrinsic stage redundancy, ensuring optimal parameter utilization.

##### 2. Accuracy Enhancements

**Large-Kernel Convolution:** Enlarges the receptive field to enhance feature extraction capability.

**Partial Self-Attention (PSA):** Incorporates self-attention modules to improve global representation learning with minimal overhead.

### 2.1.2. Open Images dataset V7 :

Open Images V7, released on October 25, 2022 is a versatile and expansive dataset championed by Google. Aimed at propelling research in the realm of computer vision, it boasts a vast collection of images annotated with a plethora of data, including image-level labels, object bounding boxes, object segmentation masks, visual relationships, and localized narratives.

Open Images is a dataset of ~9M images annotated with image-level labels, object bounding boxes, object segmentation masks, visual relationships, and localized narratives:

It contains a total of 16M bounding boxes for 600 object classes on 1.9M images, making it the largest existing dataset with object location annotations. The boxes have been largely manually drawn by professional annotators to ensure accuracy and consistency. The images are very diverse and often contain complex scenes with several objects (8.3 per image on average).

Open Images also offers visual relationship annotations, indicating pairs of objects in particular relations (e.g. "woman playing guitar", "beer on table"), object properties (e.g. "table is wooden"), and human actions (e.g. "woman is jumping"). In total it has 3.3M annotations from 1,466 distinct relationship triplets.

In V5 we added segmentation masks for 2.8M object instances in 350 classes. Segmentation masks mark the outline of objects, which characterizes their spatial extent to a much higher level of detail.

In V6 we added 675k localized narratives: multimodal descriptions of images consisting of synchronized voice, text, and mouse traces over the objects being described.

In the latest version v7 we added 66.4M point-level labels over 1.4M images, covering 5,827 classes. These labels provide sparse pixel-level localization and are suitable for zero/few-shot semantic segmentation training and evaluation.

Finally, the dataset is annotated with 61.4M image-level labels spanning 20,638 classes.

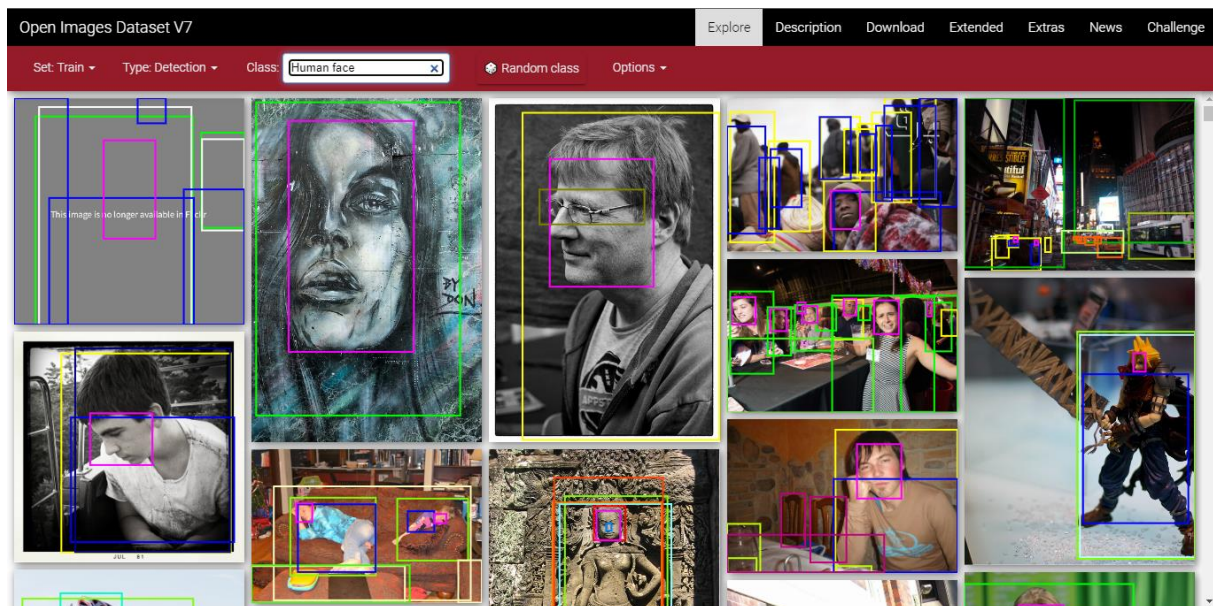


Figure 2.2 : Google Open Images v7 'Human face' class with bounding boxes for detection tasks

### 2.1.3.YOLOv10 training for face detection :

We downloaded 10,000 images from the OpenImages V7 dataset by cloning a Git repository that facilitates bulk downloading of one or many classes of the dataset. By leveraging the repository's scripts, we could efficiently obtain the images and their corresponding csv annotations.

```
(base) C:\Program Files\OIDv4_ToolKit>python main.py downloader --classes "Human face" --type_csv train --limit 10000

OIDv4
Downloader

[INFO] | Downloading Human face.
[ERROR] | Missing the class-descriptions-boxable.csv file.
[DOWNLOAD] | Do you want to download the missing file? [Y/n] y
...145%, 0 MB, 3201 KB/s, 0 seconds passed
[DOWNLOAD] | File class-descriptions-boxable.csv downloaded into OID\csv_folder\class-descriptions-boxable.csv.
[ERROR] | Missing the train-annotations-bbox.csv file.
[DOWNLOAD] | Do you want to download the missing file? [Y/n] y
...1%, 18 MB, 4780 KB/s, 3 seconds passed

100%|
[INFO] | Done!
[INFO] | Creating labels for Human face of train.
[INFO] | Labels creation completed.

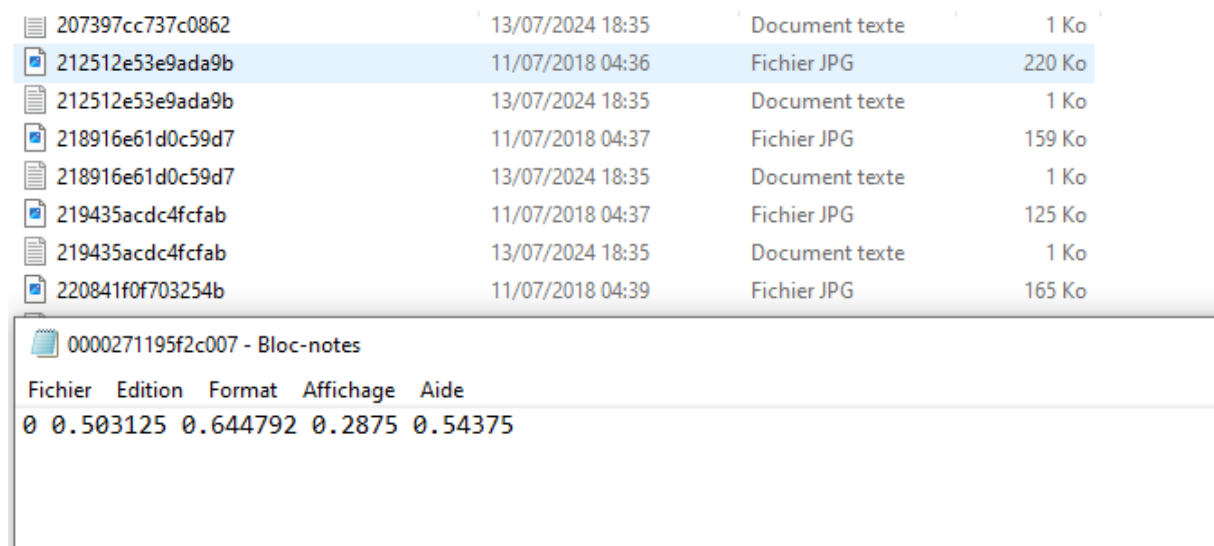
(base) C:\Program Files\OIDv4_ToolKit>
(base) C:\Program Files\OIDv4_ToolKit>
(base) C:\Program Files\OIDv4_ToolKit>
```

Figure 2.3 : OIDv4\_ToolKit for downloading images from 'Human face' OID v4 class

#### 2.1.3.1. Covertng csv annotations to yolo format :

To convert CSV annotations to YOLO format, we used a script to parse the CSV files and extract relevant information such as class number, center X, center Y, width and height.

By normalizing these coordinates and restructuring the data, we generated text files for each image. These files, formatted for YOLO, contain class numbers and normalized bounding box values, facilitating seamless integration with YOLO training pipelines. This process ensures compatibility and enhances the efficiency of training object detection models.



The image shows a file explorer window with a list of files and a detailed view of a selected file. The file list includes:

Icon	File Name	Date	Type	Size
Document icon	207397cc737c0862	13/07/2024 18:35	Document texte	1 Ko
Image icon	212512e53e9ada9b	11/07/2018 04:36	Fichier JPG	220 Ko
Document icon	212512e53e9ada9b	13/07/2024 18:35	Document texte	1 Ko
Image icon	218916e61d0c59d7	11/07/2018 04:37	Fichier JPG	159 Ko
Document icon	218916e61d0c59d7	13/07/2024 18:35	Document texte	1 Ko
Image icon	219435acdc4fcfab	11/07/2018 04:37	Fichier JPG	125 Ko
Document icon	219435acdc4fcfab	13/07/2024 18:35	Document texte	1 Ko
Image icon	220841f0f703254b	11/07/2018 04:39	Fichier JPG	165 Ko

Below the file list, a detailed view of a file named "0000271195f2c007 - Bloc-notes" is shown. It displays a table with the following data:

Fichier	Edition	Format	Affichage	Aide
0	0.503125	0.644792	0.2875	0.54375

Figure 2.4 : Yolo annotations

#### 2.1.3.2. Training YOLOv10 :

We used 8,000 images for training the model and allocated the remaining 2,000 images for testing and validation. This process was orchestrated using a YAML file, ensuring a structured and efficient workflow.

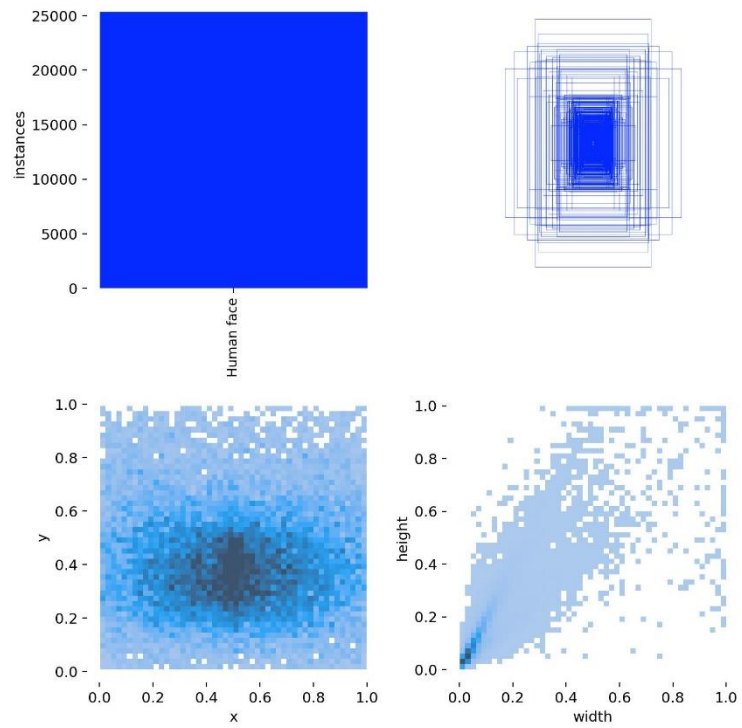


Figure 2.5 : Distribution of instances and bounding boxes throughout the dataset