



الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne Démocratique et Populaire  
وزارة التعليم العالي والبحث العلمي  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
المدرسة الوطنية العليا للتكنولوجيات المتقدمة  
École Nationale Supérieure des Technologies Avancées  
Département : Génie Électrique Et Informatique Industrielle



## Mémoire de projet de fin d'études en vue de l'obtention du diplôme **Master**

Filière :

**Automatique**

Spécialité :

**Automatique et Informatique Industrielle**

Thème :

# **Application de la Vision Industrielle et de l'Intelligence Artificielle dans l'Inspection et l'Optimisation de la Production des Pneumatiques**

Réalisé par

**BENLOUNES Lounes  
LACHEKHAB Imad Eddine**

Soutenu le 22 / 06 / 2025

### **Les membres de Jury :**

|              |                           |               |            |
|--------------|---------------------------|---------------|------------|
| Président    | M. CHERIFI Tarek          | M.C.A         | ENSTA      |
| Promoteur    | M. DERMOUCHE Reda         | M.C.A         | ENSTA      |
| Co-Promoteur | M. HAMMADI Yassine        | Expert métier | IRIS Tyres |
| Examineur    | M. NAIT SLIMANI Boukhalfa | M.C.A         | ENSTA      |
| Examineur    | M. ZAOUCHE Mohamed        | M.C.A         | ENSTA      |

Année universitaire 2024/2025

# Application de la Vision Industrielle et de l'Intelligence Artificielle dans l'Inspection et l'Optimisation de la Production des Pneumatiques

BENLOUNES Lounes

LACHEKHAB Imad Eddine

DERMOUCHE Reda

*Département Génie Electrique et  
Informatique Industrielle  
École Nationale Supérieure des  
Technologies Avancées  
Alger, Algérie  
l\_benlounes@ensta.edu.dz*

*Département Génie Electrique et  
Informatique Industrielle  
École Nationale Supérieure des  
Technologies Avancées  
Alger, Algérie  
i\_lachekhab@ensta.edu.dz*

*Département Génie Electrique et  
Informatique Industrielle  
École Nationale Supérieure des  
Technologies Avancées  
Alger, Algérie  
reda.dermouche@ensta.edu.dz*

---

## Abstract

Dans le cadre de la transformation numérique de l'industrie 4.0, le recours à des systèmes d'inspection automatisée basés sur l'intelligence artificielle représente un levier stratégique pour améliorer la qualité et la traçabilité des produits industriels. Ce travail propose le développement d'un système de vision industrielle intelligent, basé sur le modèle YOLOv8, pour l'inspection automatisée des pneus dans l'usine IRIS Tyres. L'objectif principal est d'assurer la détection en temps réel des défauts de surface ainsi que des inscriptions présentes sur le flanc du pneu après la phase de vulcanisation, en remplaçant l'inspection visuelle humaine, souvent sujette à des erreurs et à une variabilité de jugement. Le modèle a été entraîné sur une base de données que nous avons constituée à l'aide d'une caméra 4K à haute fréquence (60 fps), puis intégré dans un système de supervision industriel. Les résultats expérimentaux montrent une précision de 99,83 % et un mAP de 96,67 %, attestant des performances élevées et d'un fonctionnement temps réel satisfaisant. Ce système ouvre ainsi la voie à une intégration en ligne du contrôle qualité automatisé, contribuant à renforcer la fiabilité, la rapidité et la traçabilité dans le processus de production de la société IRIS Tyres.

**Keywords :** *Inspection automatisée ; YOLOv8 ; Vision industrielle ; Industrie 4.0 ; intelligence artificielle ; pneumatiques*

---

## I. Introduction

Dans le contexte de l'industrie 4.0, l'intégration de technologies intelligentes dans les chaînes de production s'impose comme une nécessité afin de garantir la qualité, la traçabilité et la performance. Parmi ces technologies, la vision industrielle couplée à l'intelligence artificielle joue aujourd'hui un rôle central dans la modernisation des systèmes de contrôle qualité. Elle permet notamment d'automatiser des tâches autrefois réalisées manuellement, tout en réduisant les erreurs, les coûts et les temps de traitement. Ce besoin d'évolution se fait particulièrement sentir dans l'industrie du pneumatique, notamment chez IRIS Tyres, où l'étape de contrôle qualité constitue une phase critique du processus de fabrication. Après la vulcanisation des pneus, chaque pneu est inspecté visuellement par un opérateur, chargé de détecter des défauts de surface tels que des fissures, des déformations, des bulles ou la présence de corps étrangers. L'opérateur scanne ensuite le

code QR du pneu et enregistre manuellement son évaluation dans un système de traçabilité. En fonction de cette inspection, le pneu est soit validé, soit classé comme défectueux (Scrap). Cependant, bien que cette procédure soit relativement efficace, elle présente plusieurs limites importantes. La subjectivité des jugements selon les opérateurs, la fatigue induite par la répétition des tâches, ainsi que le risque d'erreurs peuvent compromettre la fiabilité du processus. Par ailleurs, le temps de traitement reste relativement élevé et le système demeure fortement dépendant du facteur humain. Face à ces contraintes, les systèmes de vision industrielle basés sur le Deep Learning apparaissent comme une alternative innovante et prometteuse. Contrairement aux méthodes classiques de traitement d'image, ces approches sont capables d'apprendre automatiquement à partir de données annotées, d'extraire des caractéristiques complexes, et de traiter des images en temps réel avec une grande précision. Parmi les modèles les plus performants dans ce domaine, la famille YOLO (You Only Look Once) s'est imposée comme une référence grâce à sa rapidité d'exécution et sa capacité à détecter plusieurs objets en une seule passe [1]. La version YOLOv8 améliore encore les performances des versions précédentes en intégrant une architecture optimisée, une meilleure gestion des petits objets, ainsi que des modules d'attention multi-échelle et des techniques d'optimisation avancées [1]. L'efficacité des réseaux de neurones profonds dans l'inspection visuelle a été démontrée dans de nombreux contextes industriels, notamment pour la détection de défauts sur des pales d'éoliennes, des surfaces métalliques, ou encore des ponts. Néanmoins, la détection de défauts sur des surfaces aussi complexes que celles des pneus, dont les rainures peuvent être confondues avec des fissures, reste un défi technique de taille. C'est précisément dans ce type d'application que l'approche YOLOv8 se révèle particulièrement pertinente, grâce à sa capacité à analyser des textures complexes et variées [1]. Ainsi, l'objectif de ce travail est de concevoir un système de vision industrielle intelligent basé sur YOLOv8, destiné à la détection automatique des défauts sur les pneus dans l'environnement réel de l'usine IRIS Tyres. Ce système vise à automatiser l'inspection, à réduire les erreurs humaines, à améliorer la cadence de production, ainsi qu'à renforcer la fiabilité et la traçabilité des décisions. Cet article présente les approches de Deep Learning pour la détection de défauts sur les pneus, la comparaison des méthodes, le choix du modèle YOLOv8, son architecture, ainsi que la constitution et l'exploitation du dataset. Il aborde ensuite la stratégie d'entraînement, les résultats expérimentaux en environnement industriel et les perspectives d'intégration du système développé.

## II. Approches de Deep Learning pour la détection de défauts sur les pneus

La détection d'objets repose largement sur l'utilisation d'architectures avancées de deep learning, notamment les réseaux de neurones convolutifs (CNN). Grâce à un entraînement approfondi sur des jeux de données annotés à grande échelle, ces modèles acquièrent la capacité de reconnaître des motifs complexes et des caractéristiques distinctives associées aux objets dans des images [2].

Un détecteur d'objets se compose généralement de trois étapes principales (figure 1) :

a) **Prétraitement des données :**

Les images sont redimensionnées, nettoyées de leur bruit, normalisées, et annotées avec des boîtes englobantes. Cette étape garantit la compatibilité et l'efficacité du traitement par le modèle.

b) **Modèle de détection :**

C'est le cœur du système, généralement un CNN structuré en plusieurs sous-composants :

- **Backbone** : extrait les caractéristiques brutes à partir de l'image d'entrée.
- **Neck** : fusionne et affine les cartes de caractéristiques issues de différentes couches du backbone afin de capter à la fois des détails fins (objets petits) et des structures globales (objets grands).
- **Tête de détection (Detection Head)** : prédit les boîtes englobantes et les classes des objets présents à partir des cartes de caractéristiques raffinées.

c) **Post-traitement** :

Élimine les prédictions redondantes via des techniques comme le Non-Maximum Suppression (NMS), et affine les résultats pour obtenir des détections précises et exploitables.

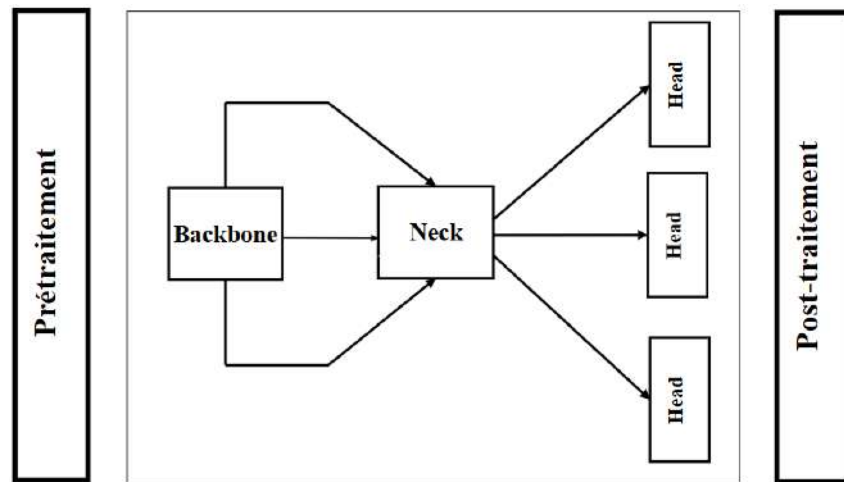


FIGURE 1. Étape du détecteur d'objets

Plusieurs méthodes de détection d'objets basées sur l'apprentissage profond ont été appliquées dans le domaine de l'inspection visuelle industrielle, notamment pour la détection des défauts de surface sur les pneus. Ces approches combinent une grande précision avec une capacité d'exécution en temps réel, ce qui est essentiel pour les systèmes d'inspection automatisés [1]. Les principales approches incluent :

- **R-CNN (Region-based Convolutional Neural Network)** : L'une des premières méthodes de détection d'objets basées sur le deep learning. Elle applique un CNN sur des régions d'intérêt générées par une recherche sélective. Bien qu'elle ait posé les bases des approches modernes, cette méthode est aujourd'hui jugée trop lente pour des cas d'usage industriels nécessitant une inspection rapide, comme celle des pneus [2].
- **Faster R-CNN** : Développé par Shaoqing Ren et al. en 2015, Faster R-CNN est un détecteur d'objets en deux étapes. Il commence par générer des propositions de régions à l'aide d'un réseau de propositions de régions (RPN), puis les classe et affine à l'aide d'un réseau de neurones convolutif (CNN). Bien qu'il offre une précision élevée dans les tâches de détection de défauts, son coût computationnel important le rend moins adapté aux environnements industriels en temps réel [3].
- **SSD (Single Shot MultiBox Detector)** : SSD est un détecteur à une seule étape qui effectue la localisation et la classification des objets en un seul passage dans le réseau. Son bon compromis entre vitesse et précision en fait une option viable pour la détection de défauts de surface en temps réel sur les pneus. Il est particulièrement performant pour les défauts de taille importante et bien définis [4].

- **YOLO (You Only Look Once)** : YOLO est un détecteur rapide et efficace à une seule étape, qui prédit directement les boîtes englobantes et les probabilités de classes à partir de l'image complète en un seul passage. Ses dernières versions, telles que YOLOv8, offrent une précision accrue et sont optimisées pour les applications industrielles. YOLO est particulièrement adapté à la détection de défauts de surface sur les pneus en temps réel grâce à sa faible latence et à sa grande vitesse d'inférence [5].

## II.1. Comparaison entre les approches

Le [tableau 1](#) présente une comparaison des principales approches de détection de défauts par deep learning, selon plusieurs critères techniques tels que l'architecture, la vitesse et la précision. L'approche YOLO a été retenue pour la suite de ce travail en raison de ses performances optimales dans un contexte industriel.

**TABLE 1.** Comparaison entre différentes méthodes de détection d'objets [1]

| Aspect                           | Faster R-CNN                                                        | R-CNN                                                                                 | YOLO                                                                                       | SSD                   |
|----------------------------------|---------------------------------------------------------------------|---------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|-----------------------|
| <b>Architecture</b>              | Deux étapes : Réseau de propositions de régions (RPN) + Classifieur | Multi-étapes : Propositions de régions + Extraction de caractéristiques + Classifieur | Une seule étape : prédiction directe des boîtes englobantes et des probabilités de classes | Une seule étape       |
| <b>Vitesse</b>                   | Plus lent que YOLO et SSD                                           | Plus lent que YOLO et SSD                                                             | Très rapide                                                                                | Moins rapide que YOLO |
| <b>Précision</b>                 | Élevée                                                              | Élevée                                                                                | Élevée                                                                                     | Élevée                |
| <b>Performance en temps réel</b> | Non                                                                 | Non                                                                                   | Oui                                                                                        | Oui                   |
| <b>Complexité d'entraînement</b> | Modérée                                                             | Élevée                                                                                | Modérée                                                                                    | Modérée               |
| <b>Flexibilité</b>               | Modérée                                                             | Faible                                                                                | Élevée                                                                                     | Modérée               |
| <b>Nombre de paramètres</b>      | Élevé                                                               | Élevé                                                                                 | Modéré                                                                                     | Modéré                |
| <b>Robustesse</b>                | Élevée                                                              | Élevée                                                                                | Élevée                                                                                     | Élevée                |

## II.2. Critères du choix de YOLOv8 par rapport aux autres versions de YOLO

Le choix de YOLOv8 pour ce projet repose sur une évaluation comparative approfondie des différentes versions de la famille YOLO. En prenant en compte des critères industriels tels que la précision, la vitesse, la légèreté et la facilité d'intégration, YOLOv8 se distingue comme la version la plus adaptée à un système de vision industrielle en temps réel pour la détection de défauts sur les pneus [1].

a) **Rapidité en temps réel :**

YOLOv8 affiche une vitesse d'inférence parmi les plus élevées de toutes les versions de YOLO. Contrairement aux versions plus anciennes comme YOLOv3 ou YOLOv4, qui sont plus lentes, et même YOLOv7, qui reste ancré dans une structure à ancres plus lourde, YOLOv8 utilise une architecture allégée permettant une exécution fluide sur des chaînes de production à cadence élevée, sans interruption du processus industriel.

b) **Haute précision :**

Le modèle YOLOv8 intègre des innovations comme les blocs C2f et une tête de détection *anchor-free*, qui améliorent la précision globale et la détection de défauts subtils ou irréguliers, typiques des anomalies de surface sur les pneus. À titre comparatif, YOLOv3 et YOLOv5 offrent une précision correcte, mais restent en retrait par rapport aux optimisations de YOLOv8, surtout dans les cas de faible contraste ou de formes déformées.

c) **Légèreté et efficacité :**

La légèreté du modèle est un facteur clé pour son intégration sur des systèmes embarqués, comme des caméras industrielles ou des PC industriels sans GPU haut de gamme. YOLOv8 a été conçu pour offrir un excellent rapport performance/consommation, surpassant YOLOv4 et YOLOv7 en termes d'efficacité énergétique et de mémoire utilisée, tout en conservant une précision supérieure.

d) **Facilité d'intégration :**

YOLOv8 propose une API claire, une interface Python conviviale et un support natif pour l'export vers des formats industriels comme ONNX, TensorRT ou CoreML. Cela le rend particulièrement pratique pour une intégration directe avec des systèmes SCADA, IHM ou des automates via OPC UA, ce qui est essentiel dans le contexte de l'usine IRIS Tyres.

e) **Support communautaire et évolutivité :**

YOLOv8 bénéficie d'une communauté active et en croissance, portée par l'organisation Ultralytics. Cette dynamique assure des mises à jour régulières, des correctifs rapides et une documentation riche, facilitant le déploiement, la maintenance et l'évolution du système d'inspection.

### III. Présentation de l'architecture YOLOv8

YOLOv8 est une architecture (figure 2) de détection d'objets de dernière génération développée par Ultralytics. Son design modulaire intègre des innovations telles que les blocs C2f et le module SPPF, favorisant un apprentissage efficace et une exécution optimisée pour les applications en temps réel. Le modèle se structure en quatre parties principales : Input, Backbone, Neck et Head [5].

a) **Backbone :**

Le Backbone de YOLOv8 constitue une évolution optimisée du traditionnel CSP-Darknet, repensée pour renforcer l'extraction des caractéristiques tout en réduisant la complexité computationnelle et le poids du modèle. Il repose sur l'intégration de blocs C2f (Cross Stage Partial avec deux convolutions et transition rapide), une version améliorée des blocs CSP classiques. Ces blocs facilitent la propagation efficace du gradient et réduisent le coût de calcul global [5]. Le Backbone de YOLOv8 comprend notamment :

- **Blocs C2f optimisés :** ils combinent des convolutions profondes légères avec une architecture à connexions partielles, améliorant la diversité des caractéristiques extraites tout en maintenant une excellente efficacité computationnelle.

- **Modules modernes de normalisation et d'activation** : garantissant une meilleure stabilité et rapidité lors de l'apprentissage du réseau.
- **Module SPPF (Spatial Pyramid Pooling – Fast)** : placé en fin de Backbone, ce module allégé permet de capturer efficacement des informations spatiales à différentes échelles, sans alourdir le modèle, en regroupant les caractéristiques issues de plusieurs résolutions.

b) **Neck** :

Le Neck de YOLOv8 joue un rôle essentiel dans la fusion et la mise en correspondance des caractéristiques extraites par le Backbone à différentes échelles, afin d'améliorer la détection d'objets de tailles variées. YOLOv8 repose sur une architecture de type PANet (Path Aggregation Network) optimisée, qui permet une circulation bidirectionnelle efficace de l'information entre les différentes couches de caractéristiques [5].

Principales caractéristiques du Neck de YOLOv8 :

- **Fusion multi-échelle efficace** : le Neck combine les caractéristiques issues du Backbone à plusieurs résolutions, ce qui permet de conserver à la fois les détails fins (utiles pour les petits objets) et les informations globales (représentatives des objets plus volumineux).
- **Modules de convolution légers et optimisés** : ils assurent un traitement rapide tout en maintenant une richesse d'information élevée.
- **Structure modulaire** : elle facilite l'intégration de blocs C2f, renforçant ainsi la capacité de représentation du modèle.
- **Traitement bidirectionnel (top-down et bottom-up)** : cette approche améliore la robustesse des caractéristiques avant la phase de détection finale.

c) **Head** :

Le Head de YOLOv8 assure la prédiction des boîtes englobantes et des classes grâce à une architecture découplée, dans laquelle la classification et la régression sont traitées séparément. Il adopte une approche anchor-free et repose sur une stratégie dynamique d'assignation des labels, ce qui permet d'améliorer la qualité de l'apprentissage, la précision des détections, tout en maintenant une exécution rapide et efficace [5].

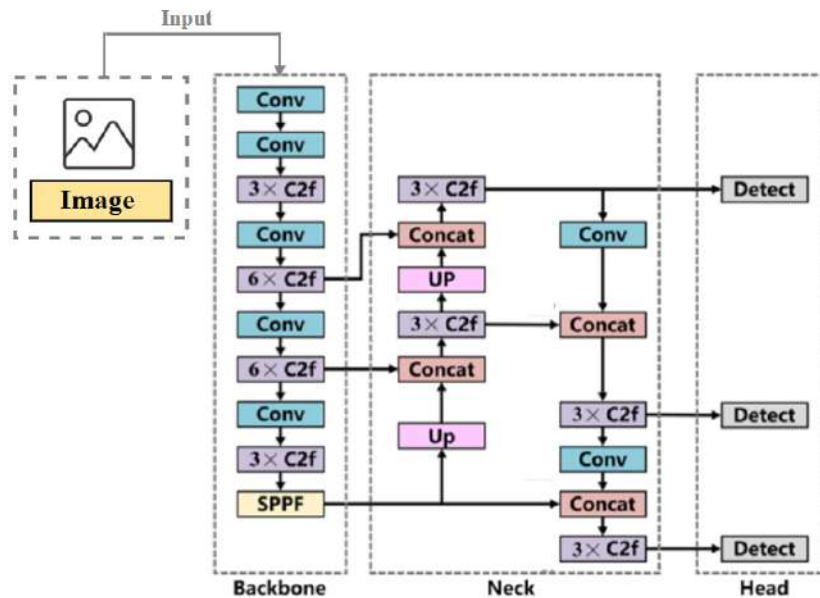


FIGURE 2. Architecture YOLOv8

## IV. Fonction de Perte et Optimisation de YOLOv8

L'efficacité de l'architecture YOLOv8 repose non seulement sur sa structure modulaire optimisée (Backbone, Neck et Head), mais aussi sur sa fonction de perte soigneusement conçue, qui guide l'apprentissage du modèle en équilibrant plusieurs objectifs .

### IV.1. Fonction de perte globale

La perte totale combine trois composantes pondérées : localisation, classification et confiance [6].

$$\mathcal{L} = \lambda_{\text{loc}} \cdot \mathcal{L}_{\text{loc}} + \lambda_{\text{cls}} \cdot \mathcal{L}_{\text{cls}} + \lambda_{\text{obj}} \cdot \mathcal{L}_{\text{obj}} \quad (1)$$

#### a) Perte de localisation (CIoU)

Mesure l'écart entre la boîte prédite et la boîte réelle, en tenant compte de la position, de la taille et du ratio.

$$\mathcal{L}_{\text{loc}} = 1 - \text{IoU} + \frac{\rho^2(\mathbf{b}, \mathbf{b}^{gt})}{c^2} + \alpha v \quad (2)$$

où :  $\rho$  : distance entre les centres des boîtes,  $c$  : diagonale de l'enveloppe minimale,  $\alpha, v$  : pénalités sur le ratio, IoU : Intersection over Union.

#### b) Perte de classification

Compare la probabilité prédite et l'étiquette réelle via l'entropie croisée.

$$\mathcal{L}_{\text{cls}} = - \sum_{c=1}^C y_c \log(\hat{y}_c) \quad (3)$$

où :  $y_c$  : étiquette réelle,  $\hat{y}_c$  : probabilité prédite.

#### c) Perte de confiance (objet)

Évalue si un objet est correctement détecté ou non.

$$\mathcal{L}_{\text{obj}} = - \left( y \cdot \log(\hat{C}) + (1 - y) \cdot \log(1 - \hat{C}) \right) \quad (4)$$

où :  $y$  : présence réelle de l'objet (1 ou 0),  $\hat{C}$  : score de confiance prédit.

## V. Production du Dataset

Dans le cadre de cette étude, aucune base de données existante ne répondait précisément aux exigences spécifiques de l'inspection visuelle automatisée des défauts sur pneus. Il a donc été nécessaire de constituer une base de données personnalisée de haute qualité, spécifiquement adaptée aux conditions industrielles de l'usine IRIS Tyres. Ce dataset couvre une grande diversité de pneus produits sur le site, incluant différentes tailles et modèles, afin de représenter fidèlement l'ensemble des cas rencontrés en production. Les images ont été capturées à l'aide d'une caméra 4K fonctionnant à 60 images par seconde, garantissant une résolution et une précision suffisantes pour la détection fine des défauts.

## V.1. Stratégie de Répartition des Données pour l'Entraînement de YOLO

Dans ce travail, la répartition des données a été soigneusement appliquée afin d'assurer une bonne généralisation du modèle YOLO. Trois ensembles distincts ont été constitués :

- **70 % pour l'entraînement** : afin d'apprendre les caractéristiques pertinentes liées à la détection et à la localisation des objets.
- **20 % pour la validation** : utilisé durant l'apprentissage pour ajuster les hyperparamètres et limiter le surapprentissage.
- **10 % pour les tests** : destiné à l'évaluation finale du modèle sur des données inédites, à l'aide de métriques telles que la précision.

## VI. Expérimentations et discussions

Les expérimentations ont été menées sur une station de travail (figure 4) dotée d'un processeur AMD Ryzen 5 8645HS et d'une carte graphique NVIDIA GeForce RTX 4050. Le système d'exploitation utilisé est Windows 11, et l'environnement de développement repose sur Visual Studio Code. L'entraînement et l'évaluation du modèle YOLOv8 ont été réalisés en utilisant la version 3.12.3 de Python, choisie pour sa stabilité et sa compatibilité avec les bibliothèques d'intelligence artificielle les plus récentes. Les données exploitées ont été acquises à l'aide d'une caméra 4K (figure 3) capable d'enregistrer à 60 images par seconde, garantissant une qualité d'image élevée, essentielle pour la détection précise des défauts de surface sur les pneus.



FIGURE 3. Caméra EMEET S600 4K



FIGURE 4. Station de travail

### VI.1. Procédure expérimentale et paramètres d'entraînement

Les images collectées en résolution 4K présentent une grande richesse en détails, ce qui favorise une meilleure qualité d'apprentissage pour la détection des défauts. Afin d'entraîner efficacement le modèle, le taux d'apprentissage a été 0,001, et le nombre d'itérations 1293. Comme le montre la (figure 5), le modèle atteint une précision de 99,8265 % avec une perte réduite à 0,1735 (figure 6), ce qui indique une convergence stable et efficace du processus d'apprentissage. Cette précision élevée s'explique en partie par le nombre restreint de classes (trois au total : fissure SW2, gamme et dimension), ce qui simplifie la tâche de classification et contribue à une meilleure qualité d'entraînement.

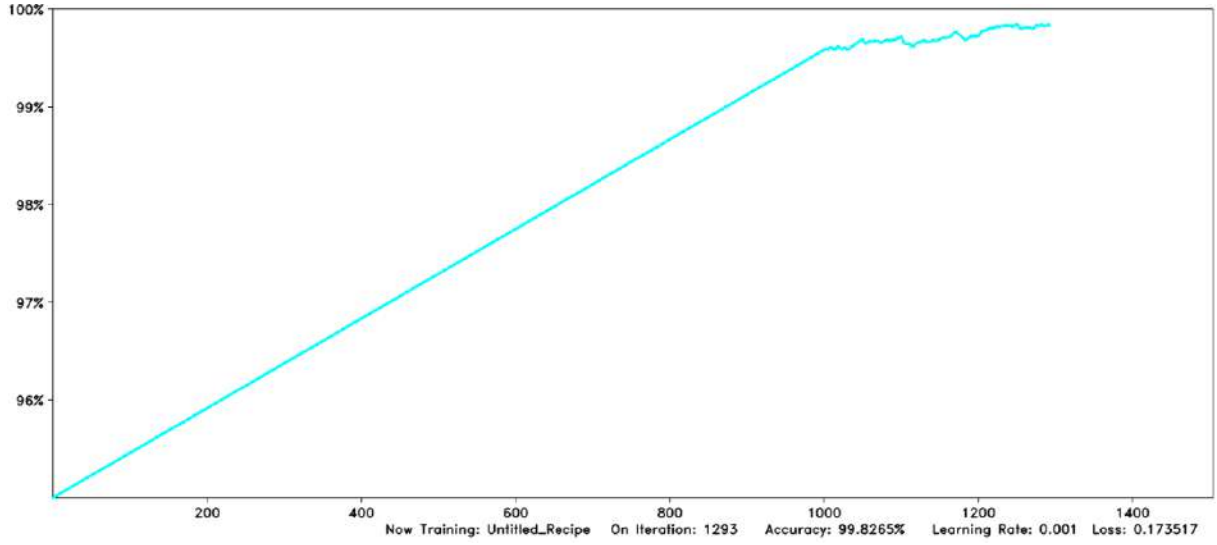


FIGURE 5. Courbe de précision de YOLOv8

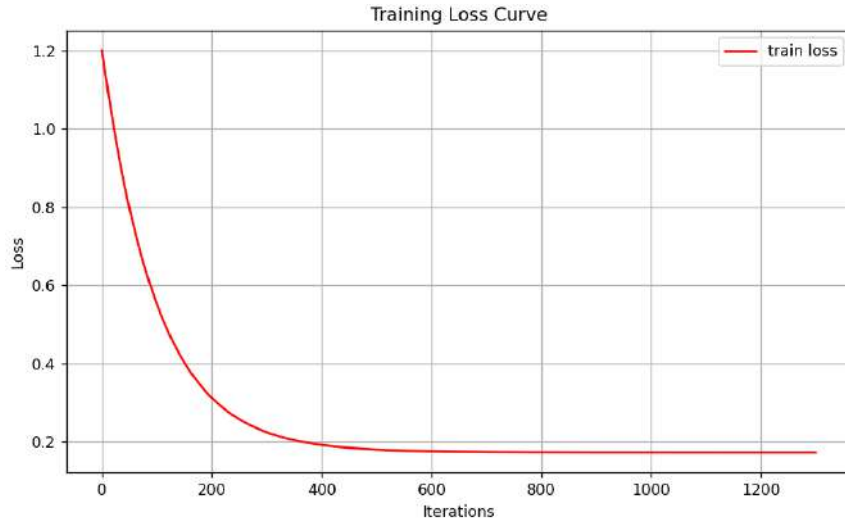


FIGURE 6. Courbe de perte d'entraînement de YOLOv8

## VI.2. Critères et Métriques d'Évaluation du Modèle

L'évaluation du modèle repose sur plusieurs métriques : la moyenne de la précision moyenne (mAP), la précision moyenne (AP), la précision (P) et le rappel (R). Les expressions de la précision et du rappel sont données par les équations (5) et (6) [7] :

$$P = \frac{TP}{TP + FP} \quad (5)$$

$$R = \frac{TP}{TP + FN} \quad (6)$$

où :

- **TP (True Positives)** désigne le nombre de boîtes englobantes correctement détectées,
- **FP (False Positives)** correspond au nombre de prédictions incorrectes considérées à tort comme positives,

- **FN (False Negatives)** représente le nombre de cibles réelles non détectées.

La précision moyenne (AP) mesure la performance du modèle pour une classe spécifique. La moyenne de la précision moyenne (mAP) est obtenue en calculant la moyenne des AP sur l'ensemble des classes, avec  $k$  représentant le nombre total de classes [7].

Les équations (7) et (8) définissent respectivement AP et mAP :

$$AP = \int_0^1 p(r) dr \quad (7)$$

$$mAP = \frac{1}{k} \sum_{i=1}^k AP_i \quad (8)$$

La figure 7 montre que le modèle YOLOv8 atteint un mAP de 96,67 %, confirmant la performance du système d'inspection automatique.

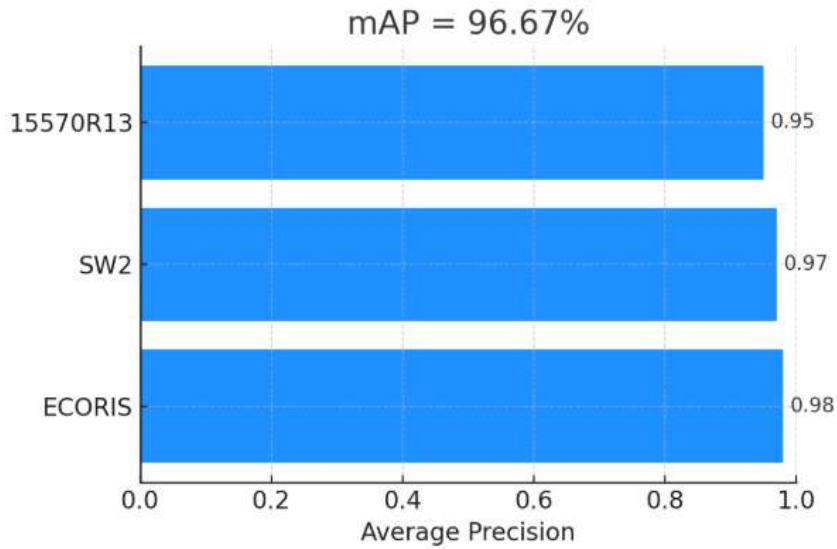


**FIGURE 7.** mAP obtenu pour la détection des défauts sur les pneus

### VI.3. Visualisation des résultats de l'inspection automatisée par YOLOv8

Les figures ci-après présentent des exemples concrets de l'inspection automatisée de pneus à l'aide du modèle YOLOv8 entraîné.

Les images ci-dessous illustrent la capacité du modèle à détecter et lire automatiquement les inscriptions présentes sur le flanc du pneu, telles que la gamme (figure 8) et les dimensions (figure 9).



**FIGURE 8.** Détection automatique de la gamme



**FIGURE 9.** Détection automatique des dimensions

La figure 8 suivante met en évidence la détection précise des zones fissurées, avec une identification claire du type et de l'emplacement des défauts .



**FIGURE 10.** Détection automatique des fissures (SW2)

#### **VI.4. Environnement de capture et de test des données : scènes réelles en milieu industriel**

Cette section décrit le contexte industriel authentique dans lequel les images ont été collectées pour la constitution de la base de données. Les prises de vue ont été réalisées directement sur la chaîne de production de pneus de l'usine IRIS Tyres, en particulier à la sortie de la presse de vulcanisation. L'objectif était de garantir la représentativité des données en reproduisant fidèlement les conditions réelles d'exploitation, assurant ainsi la robustesse du modèle de détection. Les photographies issues de ces scènes réelles illustrent la diversité des situations rencontrées, tant en termes d'angles de prise de vue que de types de défauts visibles sur les pneus. Cette base de données a ensuite été utilisée à la fois pour l'entraînement et pour le test du modèle YOLOv8.



**FIGURE 11.** Scène réelle de détection de défauts sur pneu

## VII. Développements futurs envisagés

Le système de détection des défauts sur les pneus présente un fort potentiel d'amélioration, notamment en vue de son adaptation complète aux exigences spécifiques de l'usine IRIS Tyres. L'objectif à terme consiste à intégrer l'ensemble des défauts représentés dans la figure 12, tout en assurant une interopérabilité optimale avec les équipements déjà en place au sein de la chaîne de production. Cette intégration vise à garantir une inspection automatisée fiable, rapide et ininterrompue.

Le fonctionnement global anticipé repose sur une séquence d'opérations automatisées. Une fois le pneu éjecté de la presse de vulcanisation, il est transféré par un convoyeur motorisé jusqu'à une position prédéfinie, détectée par un capteur de position. À ce point, une caméra 4K à haute résolution effectue une analyse de la surface supérieure du pneu, celui-ci étant maintenu à l'arrêt pendant une durée d'environ cinq secondes.

Le dispositif prend ensuite en charge le retournement automatisé du pneu, permettant l'exposition de sa surface inférieure. Une seconde caméra, positionnée en conséquence, procède à une inspection complémentaire. Les données issues des deux inspections sont transmises à un automate programmable industriel (API), qui exécute les actions appropriées en fonction des défauts déjà évalués.

La présence d'un défaut déclenche l'activation d'un vérin pneumatique, chargé d'éjecter le pneu vers un convoyeur d'évacuation dédié aux produits non conformes. En l'absence de défaut, le pneu est dirigé vers la machine ZF, où des essais physiques sont effectués, notamment des tests d'équilibrage et de mesure de rigidité.

L'ensemble du système est placé sous la supervision d'une interface homme-machine (IHM) assurant la visualisation en temps réel de l'état des inspections, l'affichage structuré des alertes et des statistiques relatives aux défauts identifiés, ainsi que le suivi des paramètres de fonctionnement de l'installation.

Les résultats des inspections sont systématiquement enregistrés dans une base de données centralisée, consultable depuis l'IHM. Cette architecture permet une traçabilité complète des produits inspectés et facilite considérablement l'analyse continue de la qualité au sein du processus de production.

| IRIS Tyres                         |                                                         | QUALITY MANAGEMENT<br>Tire inspection defect code |                |                |                   |                 |              | code: IT02-SP04-PO1<br>Version: 03<br>Date: 25/01/2022 |         |      |
|------------------------------------|---------------------------------------------------------|---------------------------------------------------|----------------|----------------|-------------------|-----------------|--------------|--------------------------------------------------------|---------|------|
| The main point of the tire failure |                                                         | Location of the tire failure                      |                |                |                   |                 |              | PA                                                     | BODYPLY | Belt |
|                                    |                                                         | Tread<br>TR                                       | Shoulder<br>SH | Sidewall<br>SW | Rim Cushion<br>RC | Bead area<br>BA | Inside<br>IL | IL-SW                                                  | BP      | BE   |
| 1                                  | Bareness, Lightness or Air entrapment                   | TR1                                               | SH1            | SW1            | RC1 - RC2         | BA1             | IL1          |                                                        |         |      |
| 2                                  | Split, Crack, hole or laminating                        | TR2                                               | SH2            | SW2            |                   | BA2             | IL2          |                                                        |         |      |
| 3                                  | Blister                                                 | TR3                                               |                | SW 3           | RC3               |                 |              |                                                        |         |      |
| 4                                  | Airtrap                                                 | TR4                                               | SH4            | SW4            | RC4               | BA4             | IL3 - IL4    |                                                        |         | BE4  |
| 6                                  | Open joint / open splice                                | TR6                                               | SH6            | SW6            | RC6               |                 | IL6          | PA6                                                    | BP6     | BE6  |
| 7                                  | Big splice                                              | TR7                                               | SH7            | SW7            | RC7               |                 | IL7          |                                                        | BP7     | BE7  |
| 10                                 | Visible cord(Cord fabric visible on the surface)        | TR10                                              | SH10           | SW10           | RC10              | BA10            | IL10         |                                                        |         |      |
| 11                                 | Cord (bodyply / Belt ) wrinkled                         | TR11                                              | SH11           | SW11           | RC11              |                 | IL11         |                                                        | BP11    | BE11 |
| 12                                 | Component out of center                                 | TR12 = SH12                                       |                | SW12           | RC12              | BA12            | IL12         |                                                        |         | BE12 |
| 13                                 | Wrong component                                         | TR13 = SH13                                       |                | SW13           | RC13              | BA13            | IL13         |                                                        |         | BE13 |
| 14                                 | Flash, Rubberflash                                      | TR14                                              | SH14 = SW14    |                | RC14              | BA14            |              |                                                        |         |      |
| 15                                 | Foreign matter or bulge in the tread or in the compound | TR15                                              | SH15           | SW15           | RC15              | BA15            | IL15         |                                                        |         |      |
| 16                                 | Foreign matter or bulge in the construction             | TR16                                              | SH16           | SW16           |                   |                 |              |                                                        |         | BE16 |
| 17                                 | Dirty mould                                             | TR17                                              | SH17           | SW17           | RC17 = BA17       |                 |              |                                                        |         |      |
| 18                                 | Under cured tire                                        | TR18                                              | SH18           | SW18           | RC18 = BA18       |                 |              |                                                        |         |      |
| 19                                 | Damaged tire / Transport damage(Damage after curing)    | TR19                                              | SH19           | SW19           | RC19 = BA19       |                 | IL19         |                                                        |         |      |
| 20                                 | Bodyply / Innerliner joints to close (nearby)           |                                                   |                |                |                   |                 |              |                                                        | BP20    |      |
| 21                                 | Defect not included in the failure codes                | TR21                                              | SH21           | SW21           | RC21              | BA21            | IL21         |                                                        |         | BE21 |

FIGURE 12. Types de défauts des pneus et leurs codes

## VIII. Conclusion

Ce travail a présenté une approche complète et performante pour l'inspection automatisée des défauts de surface sur les pneus, en s'appuyant sur le modèle de vision par ordinateur de dernière génération YOLOv8. Grâce à l'acquisition d'images en très haute résolution dans des conditions industrielles réelles et à un entraînement rigoureux du modèle, le système a démontré des performances remarquables, atteignant une précision de 99,83 % et un mAP de 96,67 %. Ces résultats confirment la robustesse et la fiabilité de l'algorithme pour une mise en œuvre concrète dans un environnement de production exigeant. L'intégration de cette solution au sein de l'usine IRIS Tyres ouvre la voie à une automatisation renforcée du contrôle qualité, générant des gains significatifs en termes de rapidité, de traçabilité et de réduction des erreurs humaines. À terme, le système pourra être étendu afin de couvrir l'ensemble des défauts détectables sur les pneus, tout en assurant une interopérabilité totale avec les équipements existants de la chaîne de production. Ainsi, cette démarche s'inscrit pleinement dans la dynamique de l'industrie 4.0, en apportant une réponse concrète et innovante aux enjeux de qualité, de performance et de compétitivité dans le secteur de la fabrication des pneumatiques.

## Références

- [1] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, "A Comprehensive Review of YOLO Architectures in Computer Vision : From YOLOv1 to YOLOv8 and YOLO NAS," *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1680–1716, 2023. [Online]. Available : <https://doi.org/10.3390/make5040083>
- [2] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation (R-CNN)," *Proc. CVPR*, pp. 580–587, 2014. [Online]. Available : [https://www.cv-foundation.org/openaccess/content\\_cvpr\\_2014/papers/Girshick\\_Rich\\_Feature\\_Hierarchies\\_2014\\_CVPR\\_paper.pdf](https://www.cv-foundation.org/openaccess/content_cvpr_2014/papers/Girshick_Rich_Feature_Hierarchies_2014_CVPR_paper.pdf)
- [3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN : Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017. [Online]. Available : <https://arxiv.org/pdf/1506.01497.pdf>
- [4] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD : Single Shot MultiBox Detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 21–37. [Online]. Available : <https://www.cs.unc.edu/~wliu/papers/ssd.pdf>
- [5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once : Unified, Real-Time Object Detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779–788. [Online]. Available : [https://www.cv-foundation.org/openaccess/content\\_cvpr\\_2016/papers/Redmon\\_You\\_Only\\_Look\\_CVPR\\_2016\\_paper.pdf](https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Redmon_You_Only_Look_CVPR_2016_paper.pdf)
- [6] Y. Li, Y. Wang, D. Sui, and M. Guo, "Dense Buddha head object detection and counting YOLOv8 network based on multi-scale attention and data augmentation fusion," *npj Heritage Science*, vol. 13, art. no. 23, Feb. 2025. [Online]. Available : <https://www.nature.com/articles/s40494-025-01587-0>
- [7] H. Lou, X. Duan, J. Guo, H. Liu, J. Gu, L. Bi, and H. Chen, "DC YOLOv8 : Small Size Object Detection Algorithm Based on Camera Sensor," *Electronics*, vol. 12, no. 10, art. 2323, May 2023. [Online]. Available : <https://doi.org/10.3390/electronics12102323>