

M2 MIASHS

Année universitaire : 2024 / 2025

Projet AAA

Enseignant référent :

Salvatore Anzalone

Dominique Archambault

## Poubelle Connectée



Sofiane Lhiyat

UNIVERSITÉ  
**PARIS8**  
VINCENNES-SAINT-DENIS



# Sommaire

<b>1</b>	Introduction	<b>5</b>	Algorithme d'apprentissage
<b>2</b>	Problématique	<b>6</b>	Développement logiciel
<b>3</b>	Composants et architecture du système	<b>7</b>	Tests et validations
<b>4</b>	Flux de données et communications	<b>8</b>	Conclusion et perspectives

# Introduction

Contexte :

- Croissance importante du tri des déchets
- Difficultés pour les personnes malvoyantes ou ayant des troubles sensoriels
- Conséquences du non tri des déchets ?



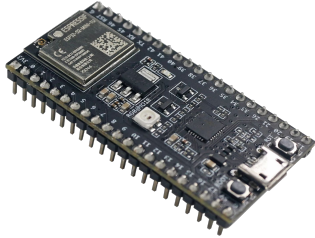
# Problématique

- Statistiques alarmantes :
  - 40% des déchets ménagers mal triés
  - Seulement 50% sont recyclés
  - 70% des personnes malvoyantes et non voyantes ont des difficultés pour trier
  - Taux d'erreur pour ces personnes de 60%
  - Impact économique et environnemental (40M€ / an pour les collectivités)

## **Comment faire une poubelle qui aide au tri des déchets ?**

Solution : Poubelle intelligente combinant vision par ordinateur, tri automatique et application mobile accessible

# Composants et architecture du système



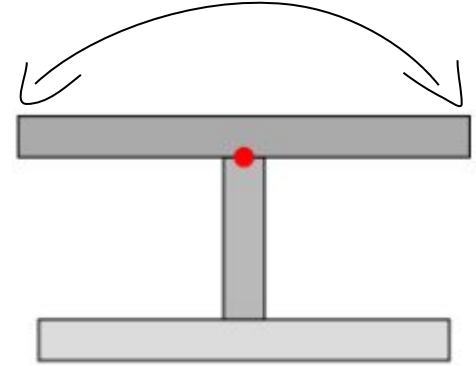
ESP32-S2



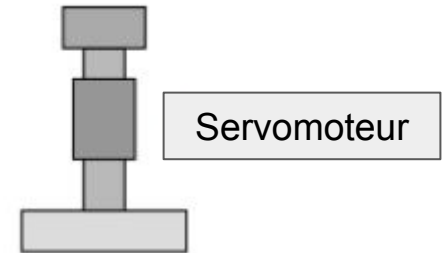
Servomoteur MG996R



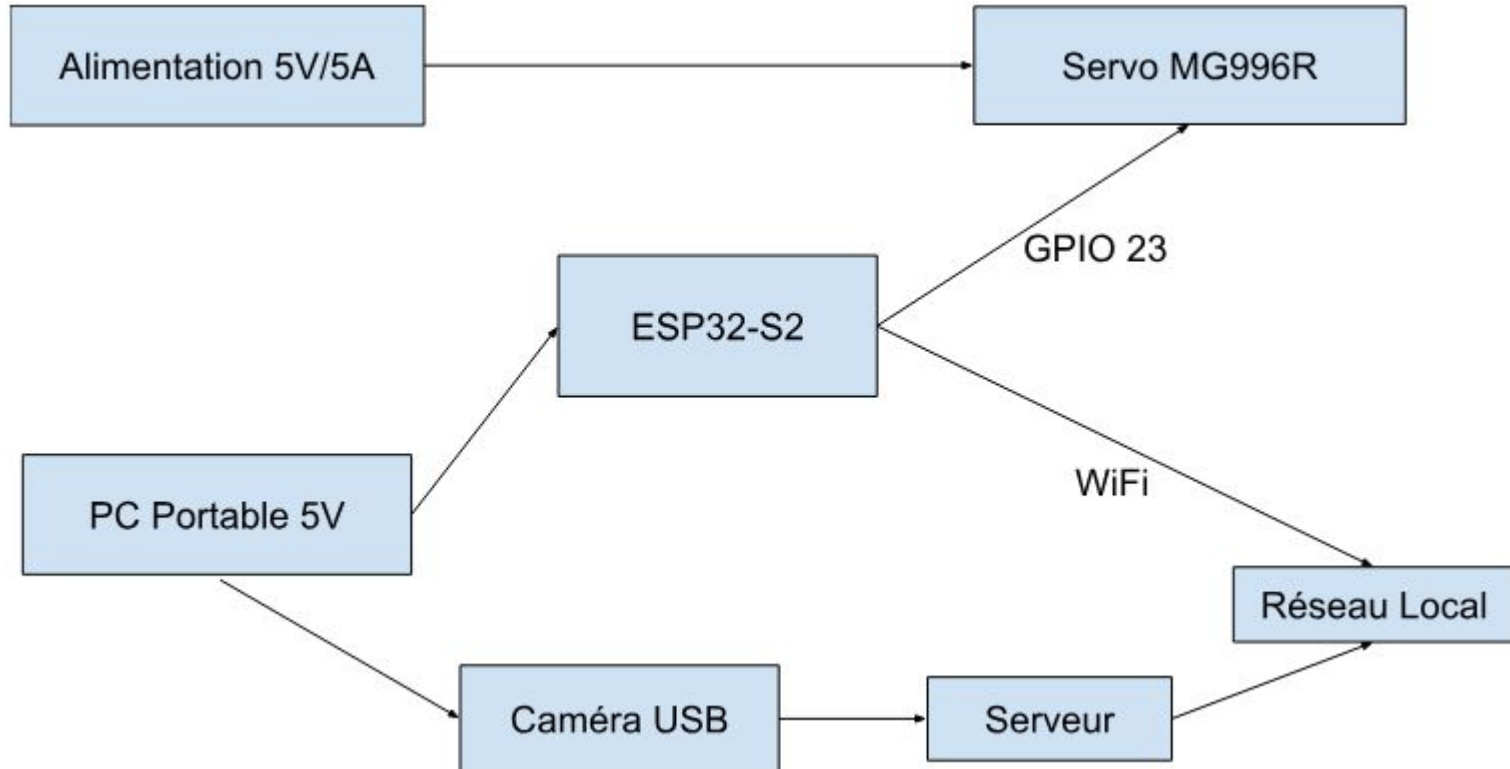
Caméra USB



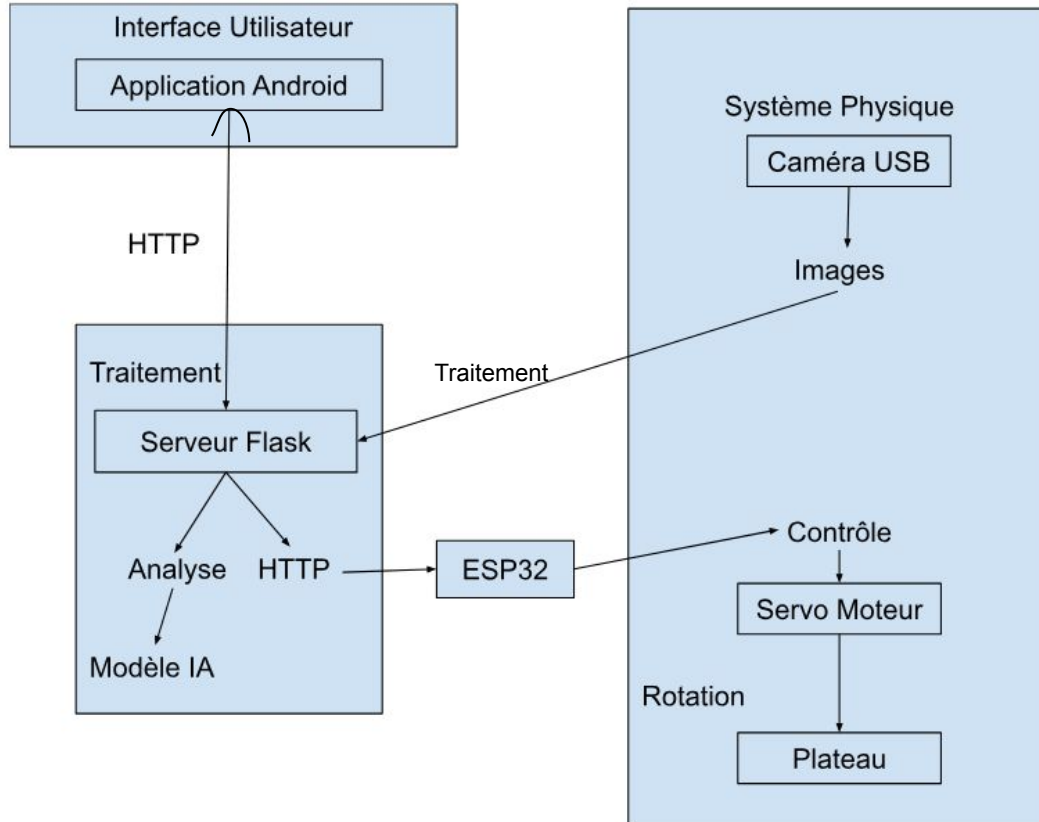
Mécanisme rotatif



# Composants et architecture du système



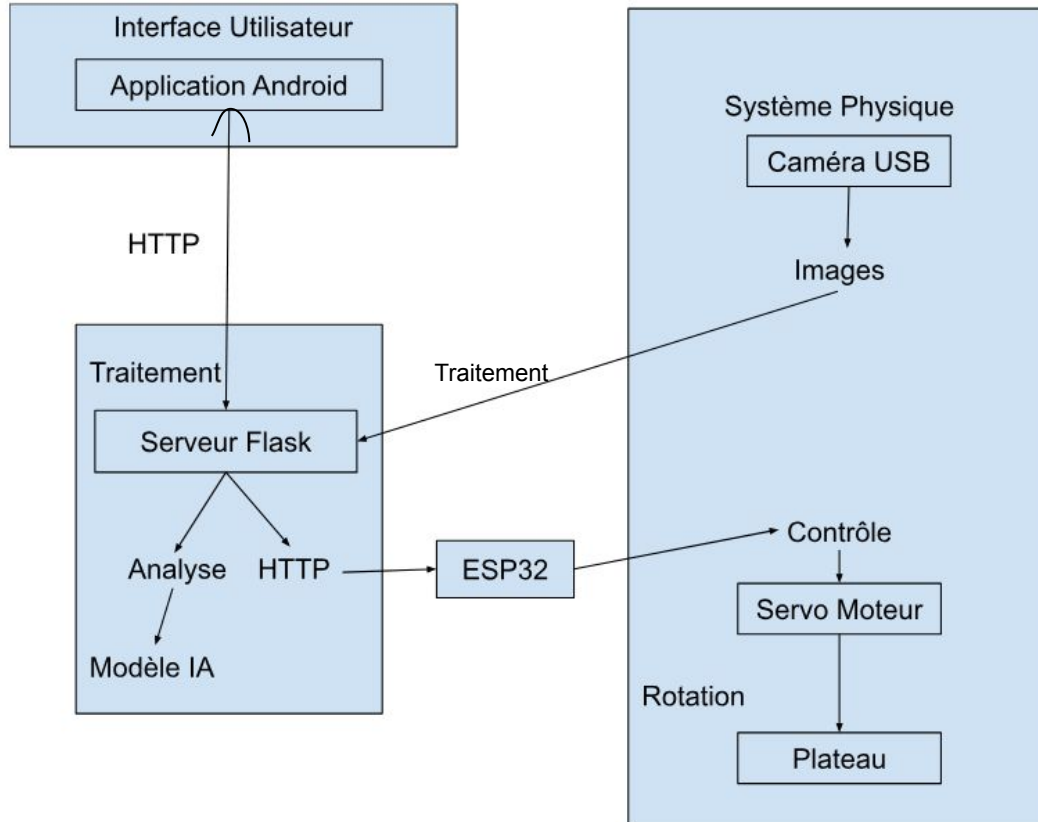
# Flux de données et communications



1ère étape :

Phase de capture d'image  
avec traitement

# Flux de données et communications

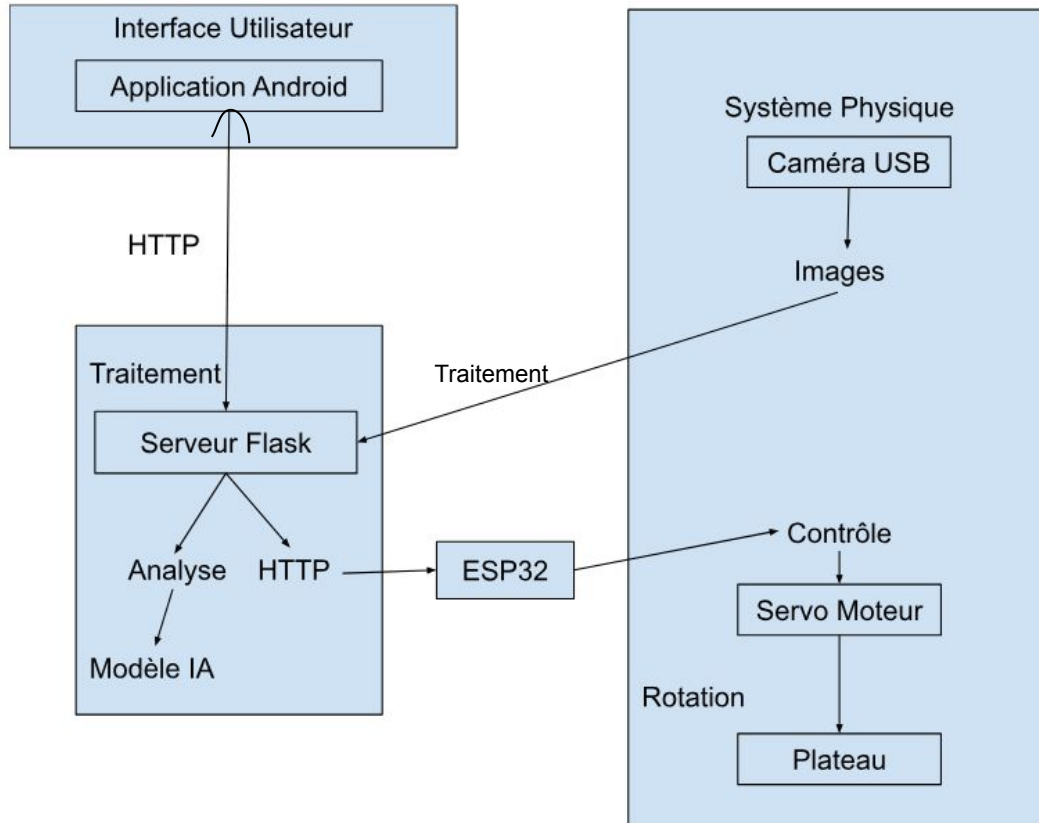


2ème étape :

Analyse via le modèle  
ResNet18



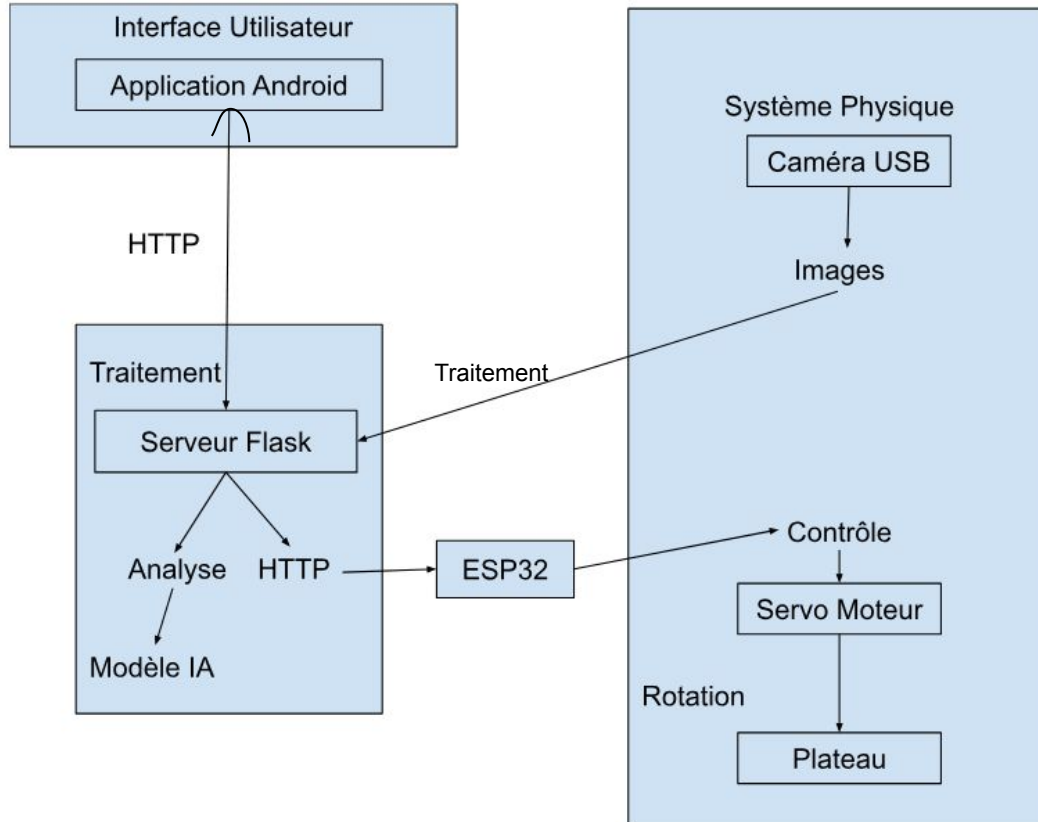
# Flux de données et communications



3ème étape :

Validation utilisateur via  
l'application Android

# Flux de données et communications



4ème étape :

Action physique sur le plateau

# Flux de données et communications

## Types de communications

Application - Serveur Flask	Serveur Flask - ESP32	Caméra - Serveur Flask
HTTP-REST Deux endpoints : <ul style="list-style-type: none"><li>• /predict</li><li>• /feedback</li></ul>	HTTP Trois endpoints : <ul style="list-style-type: none"><li>• /servo/right</li><li>• /servo/left</li><li>• /servo/center</li></ul>	Connexion USB OpenCV

# Flux de données et communications

## Serveur Flask :

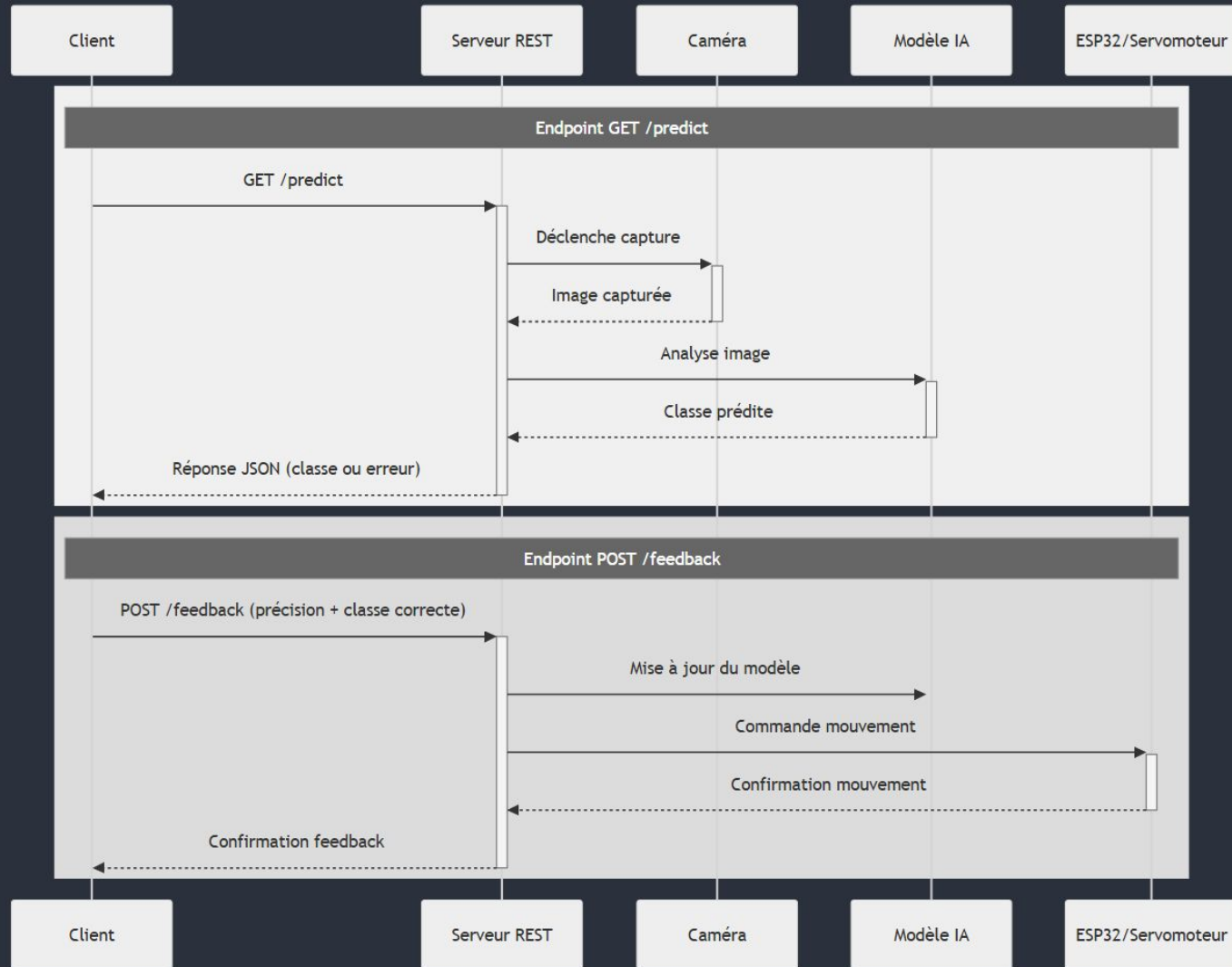
### Caractéristiques :

- Serveur web
- HTTP
- Traitement des requêtes  
GET, POST, PUT, DELETE
- Gestion des routes

### Endpoints :

- GET /predict
- POST /feedback
- GET /servo/{action}





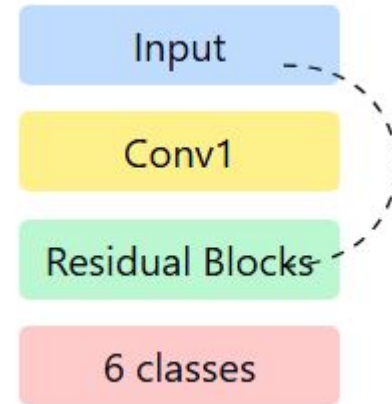
## API REST (Representational State Transfer)

# Algorithme d'apprentissage

## Architecture ResNet 18

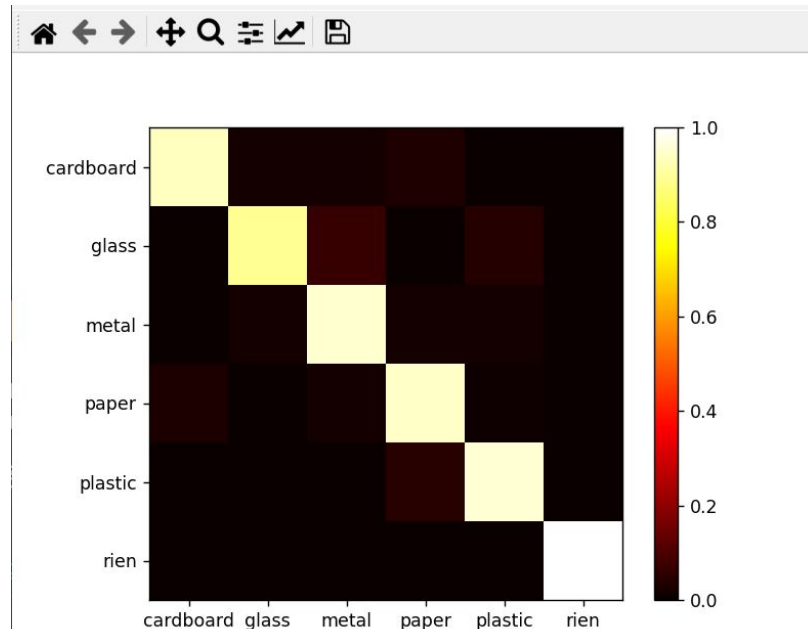
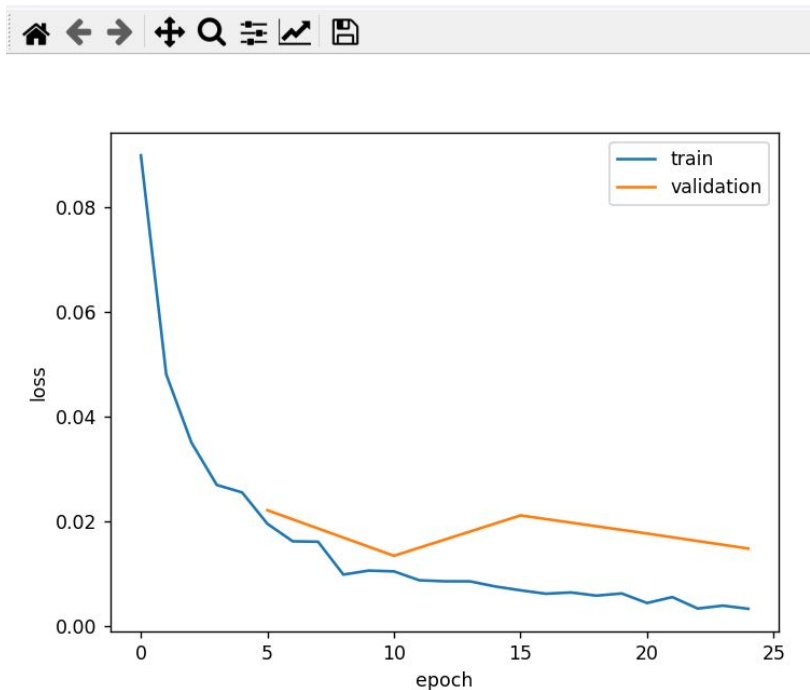
- Réseau de neurones convolutionnels
- 18 couches de convolution
- Connexions résiduelles
- Batch Normalization
- Utilisation d'une dataset de 500 images par classes
- Création d'un modèle entraîné

Structure :



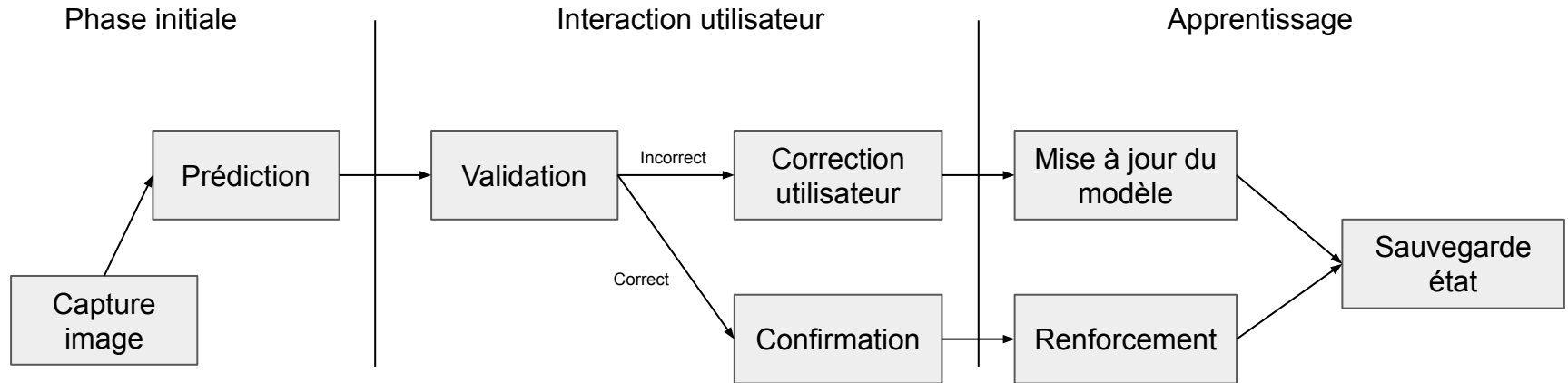
# Algorithme d'apprentissage

## Classification d'images



# Algorithme d'apprentissage

## Apprentissage continu



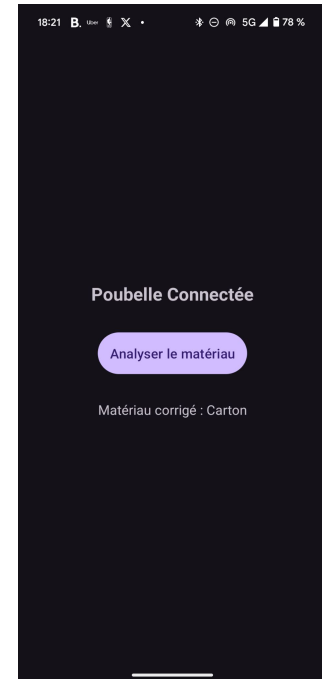
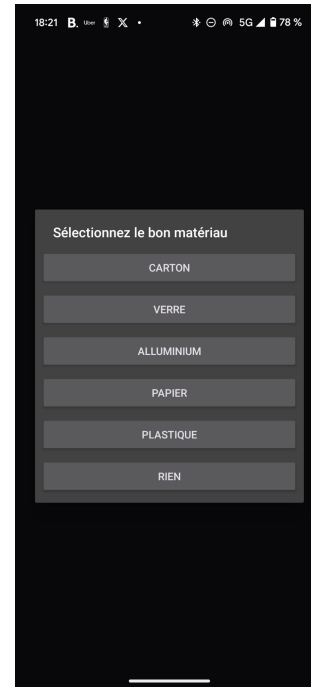
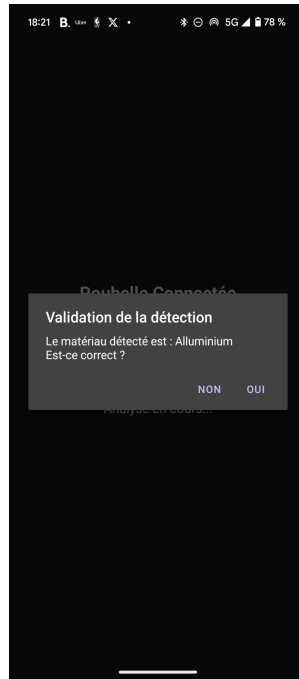


# Développement logiciel Android

- Architecture MVVM
- Communication API REST
- Composant clé : Retrofit
- Gestion d'état



# Développement logiciel Android



# Tests et validations

<p>Tests Unitaires :</p> <ul style="list-style-type: none"><li>• Tests API REST</li><li>• Tests traitement d'images</li><li>• Tests application android</li></ul>	<p>Tests d'intégrations :</p> <ul style="list-style-type: none"><li>• Communication Android-Serveur Flask</li><li>• Contrôle ESP32</li><li>• Chaîne complète de traitement</li></ul>
<p>Validation modèle :</p> <ul style="list-style-type: none"><li>• Précision : 95,26%</li><li>• F1-Score : 0.94</li></ul>	<p>Performance :</p> <ul style="list-style-type: none"><li>• Temps réponse = 3s</li><li>• Fiabilité des détections &gt; 70%</li><li>• Tests longue durée</li></ul>

# Conclusion et perspectives

## Résultats clés :

- Augmentation des réponses des prédictions
- Interface accessible
- Apprentissage continu efficace

## Limites :

- Pas de maquette
- Sous réseau WiFi
- 2 compartiments seulement

## Perspectives :

- Déploiement Cloud
- Intégration d'une mémoire
- Enlevez le feedback de l'utilisateur
- Retour vocaux

Merci de m'avoir écouer  
Avez-vous des questions ?