

# IoT Application System Project

**Presented by : Robin LANDRAUD**

**Laurent HSIA  
Jelle GOOSSENS  
Nico LAVIGNE  
Renan DUBOIS**

# Summary

- Introduction
- Presentation of Project
- Brainstorming
- Team Distribution
- Technical Aspects
- Challenges
- Conclusion





# Intro



First we were  
thinking about 3  
different choices.

01.

Project License plate:

"Project License Plate" harnesses cutting-edge artificial intelligence to swiftly and precisely recognize and process license plate data, enhancing security and automation in traffic management systems.

02.


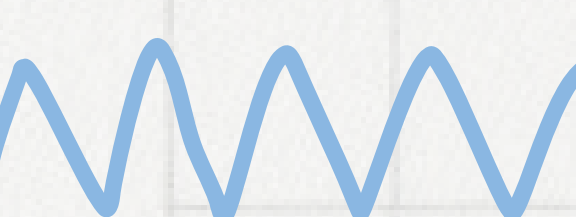
Project Document scanner:

The aim of the student project "Document Scanner" is to use AI for quick and straightforward document scanning, simplifying the process of digitizing information.

03.

Hand gesture detection:

The hand gesture detection project utilizes AI to recognize and interpret hand movements, enabling interaction with devices through gestures.



# Presentation of Project

License Plate Detection and Recognition system using a Raspberry Pi and a camera.

This project combines computer vision, machine learning, and IoT technologies to create a powerful solution for automatic license plate recognition.





# Brainstorming

## Idea :

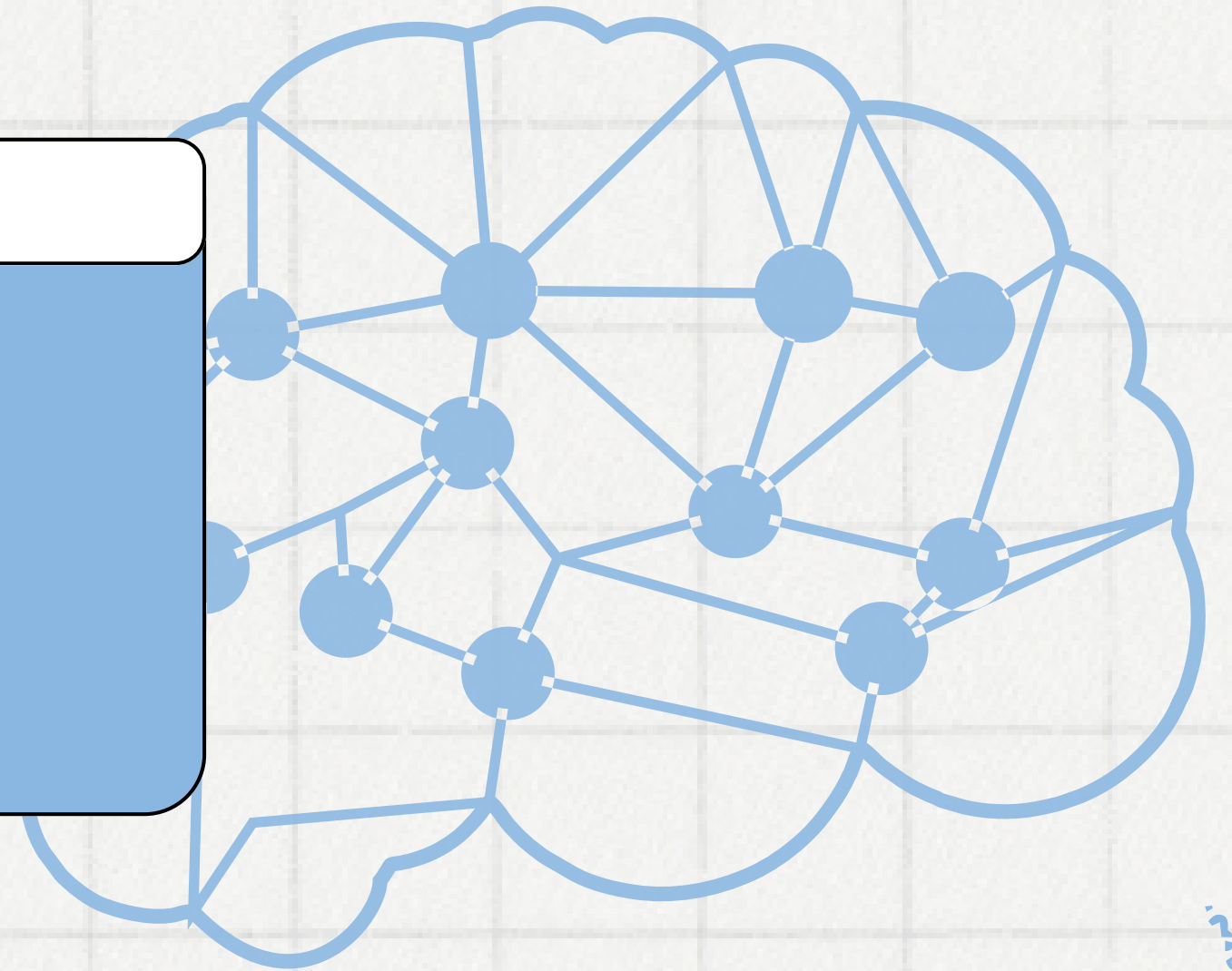
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.


## Libraries:

- Numpy
- OpenCV
- Text Processing
- Libusb
- FFmpeg

## ML/AI algorithms :


- YOLO (You Only Look Once)
- Text processing
- Tesseract OCR:
  - Tesseract is an open-source OCR (Optical Character Recognition) engine developed by Google. While not specifically designed for license plate recognition, you can fine-tune it to recognize license plates.
- ANPR (Automatic Number Plate Recognition) Systems





**Jelle Goossens**


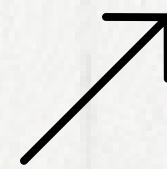
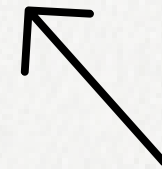
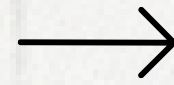
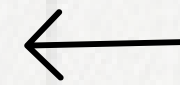
- **Persenter**
- **Surveyor**



**Renan Dubois**

- **Team Leader**
- **Facilitator**
- **Presenter**

# Team Distribution




**Robin Landraud**

- **Developer**
- **Surveyor**



**Laurent Hsia**

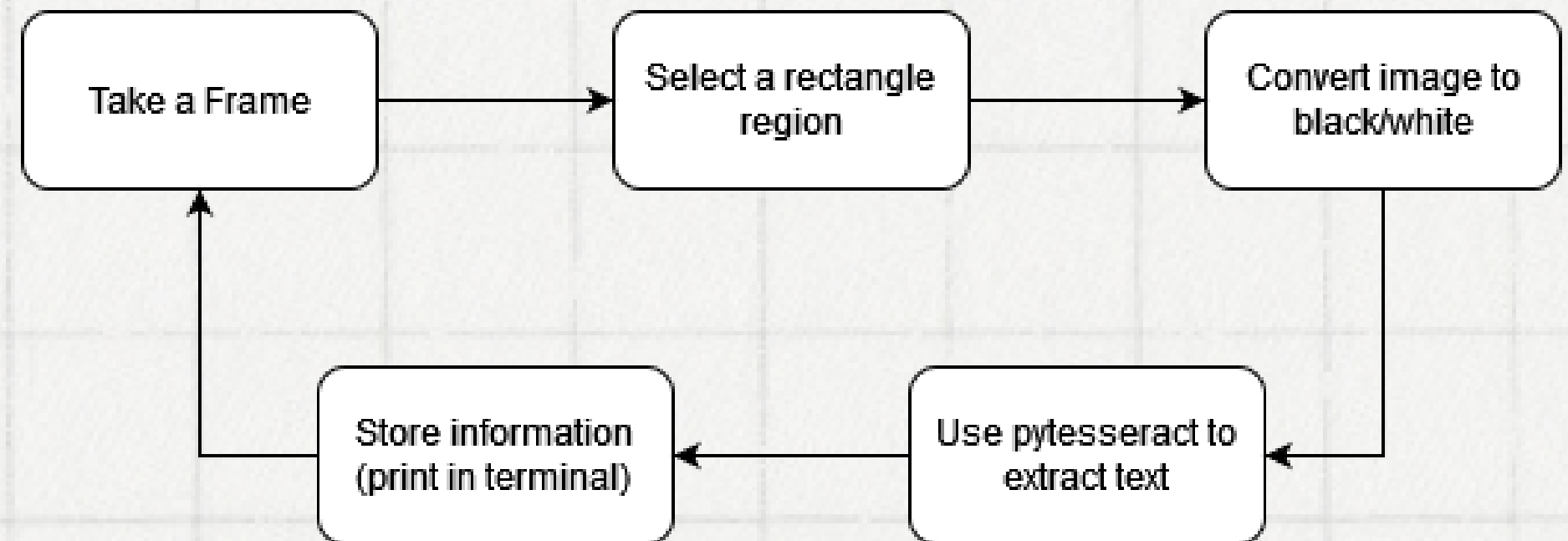
- **Recorder**
- **Developer**



**Nicolas Lavigne**

- **Devloper**

# Technical Aspect



This program uses a camera to capture video. Looking for license plates in the video by recognizing certain patterns and shapes using OpenCV. When it finds a plate, it draws a box around it. Then tries to read the characters on the plate using another tool called Pytesseract. Finally, it shows the video with the recognized plate highlighted and the characters they contain are printed out .

# Challenges & Solutions

## Challenges :

- Varied Conditions:
  - Weather and lighting conditions affecting image quality.
- Processing Power:
  - Raspberry Pi's limited processing capabilities.
- Security and Privacy Concerns:
  - Handling sensitive information from license plates raises security and privacy concerns.

## Solutions :

- Image Pre-processing:
  - Robust pre-processing techniques for varying conditions.
- Optimization:
  - Algorithmic optimizations for efficient resource utilization.
- Encryption :
  - Store and handle data securely, and adhere to privacy regulations.





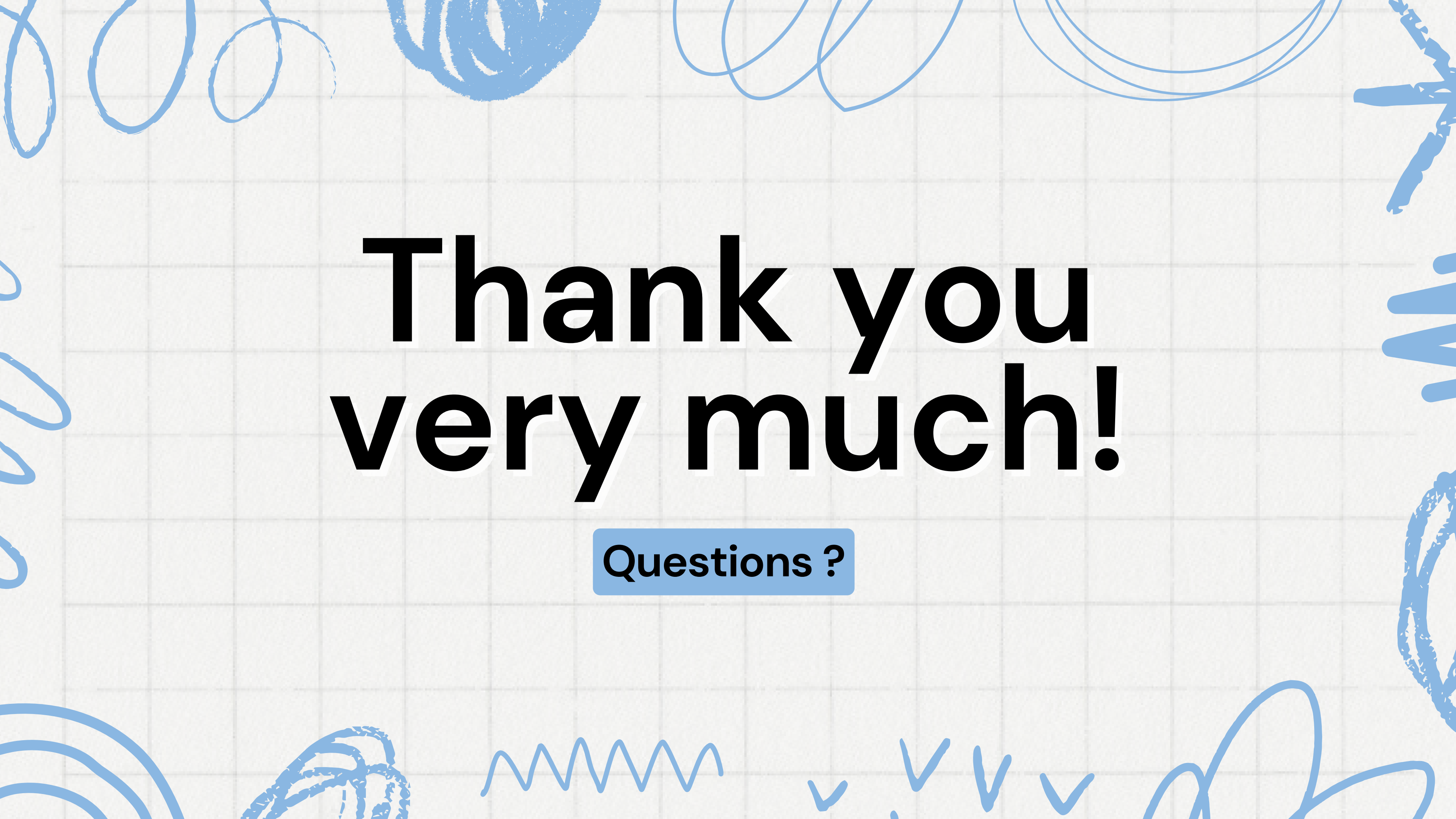
# Conclusion

In conclusion our licence plate detector system, powered by Raspberry Pi was a good introduction to the vision based IOT world. We discovered that it was a good adaptable solution with real-world applications in security & parking management.

We end up having a 60% of success when running our algorithm on the Raspberry Pi.

We also discovered that with more time our algorithm would have been easy to improve.

By using more filtering on the image to get more readable licence plate or with a bigger database to get information on multiples country's licence plate.

The background is a light gray grid. It is decorated with various hand-drawn blue doodles, including loops, swirls, a star-like shape, a zigzag line, and several checkmarks.

# **Thank you very much!**

**Questions ?**