LICENSE PLATE DETECTION

Adrián, Pol, Nil, Xavi

CONTENTS

INTRODUCTION

2153 GYX

Detect the license plate

Extract the characters

CLASSICAL APPROACH: PREPROCESSING

Problems with morphological operators:

- Fine-grained solutions for each image.
- Not robust.

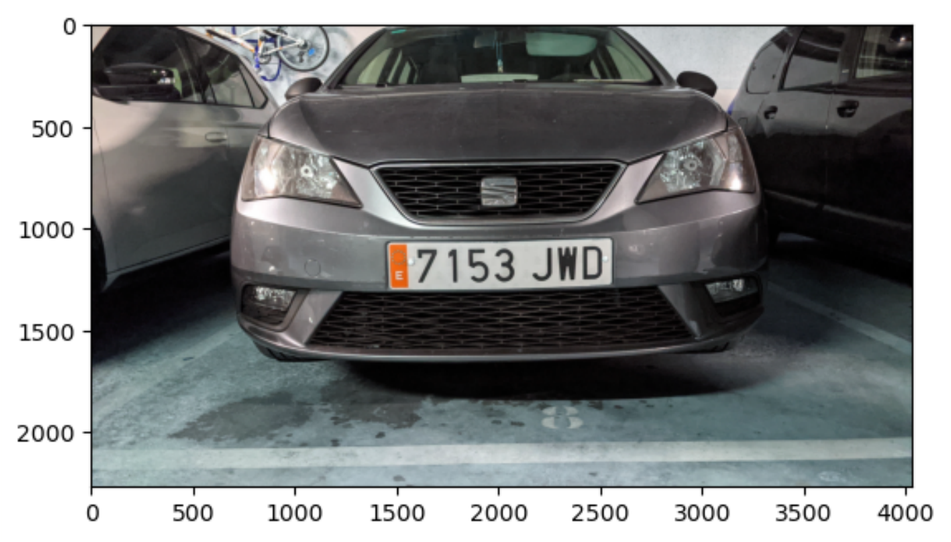
We decided to change the approach:

- Do the preprocess on a later stage of the pipeline.

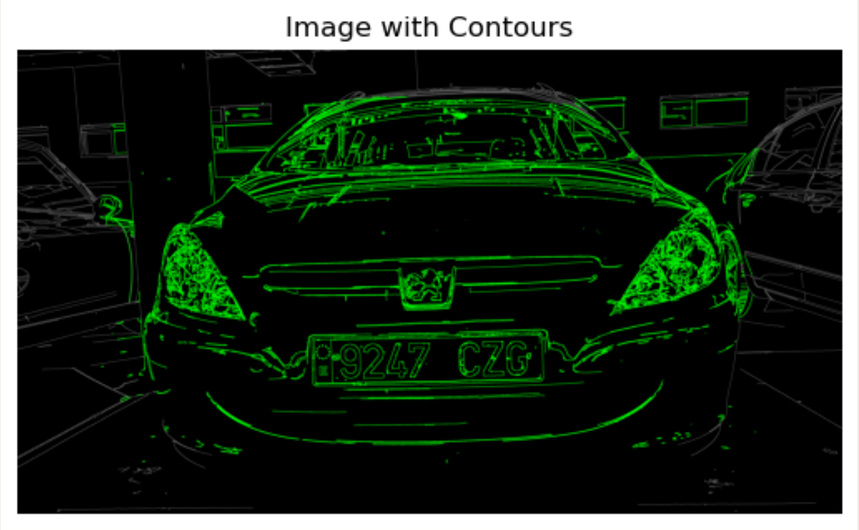
So the new process consisted in:

- Pass to gray scale Image.
- Use Canny to detect edges and extract boxes.
- If box didn't satisfy restrictions, omit detection. Else, run it through a classifier.

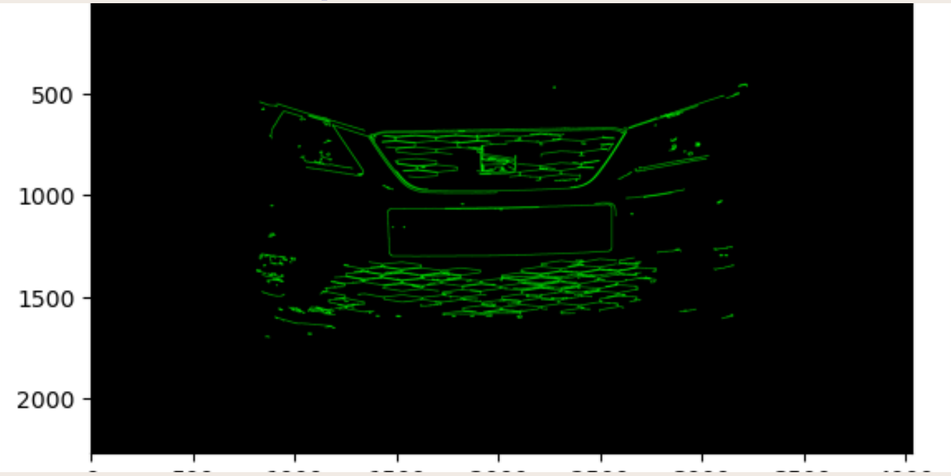
Input image



Output Canny



Filtered output

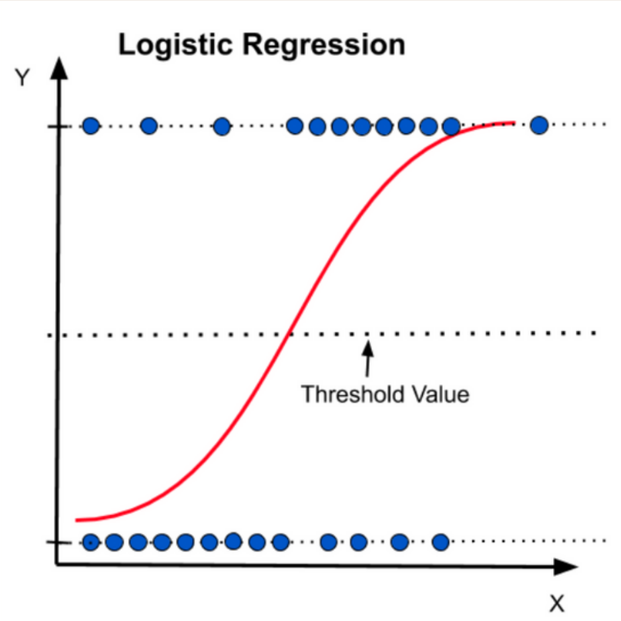


Logistic Regression Output

QUANTITATIVE RESULTS: CLASSIFICATION

Frontal images: **9 out of 15** images correctly detected

Lateral images: **5 out of 15** images correctly detected



Why were some images not detected?

- Logistic Regressor not assigning correctly the license plates labels to the crop images.
- Related to the quality of training data.

QUANTITATIVE RESULTS

OVERALL ACCURACY: **37’5%**
NUMBERS: **43’75%**
LETTERS: **43’75%**

FRONTAL IMAGES: **40%**
NUMBERS: **53’33%**
LETTERS: **46’67%**

LATERAL IMAGES: **35’29%**
NUMBERS: **35’29%**
LETTERS: **41’18%**

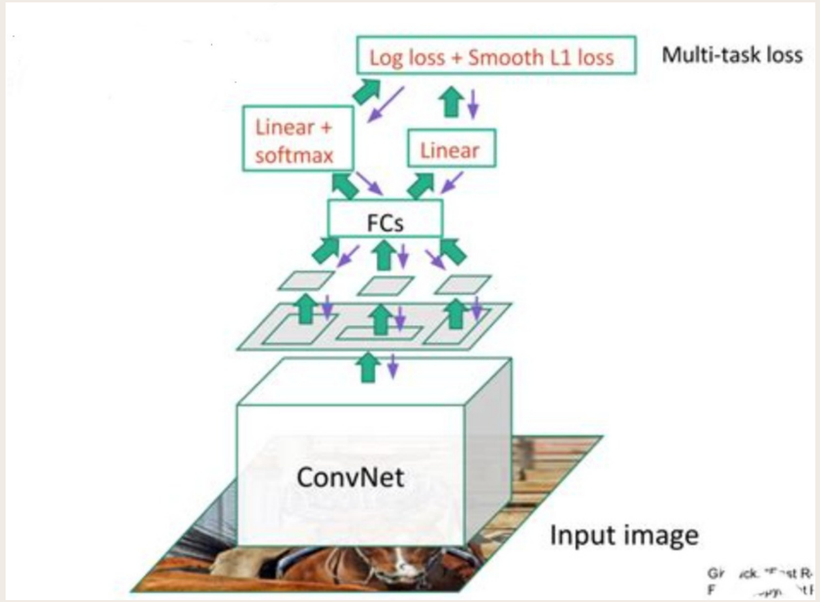
AVERAGE TIME OF COMPUTATION:
12’63 SECONDS

QUALITATIVE RESULTS

DEEP LEARNING APPROACH

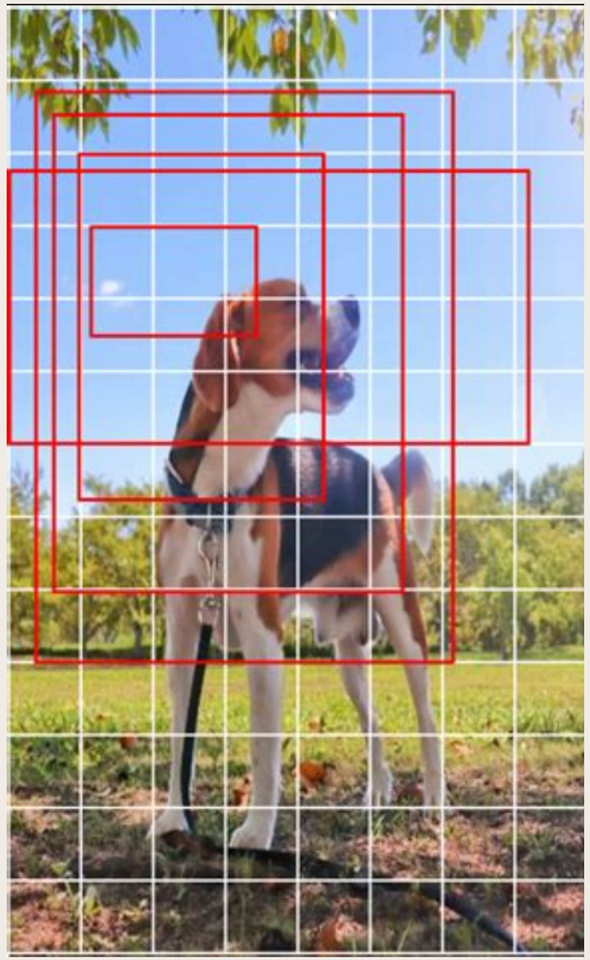
TWO-STAGE

FAST R-CNN



ONE-STAGE

YOLO



2 STAGE : RCNN

Data augmentation:

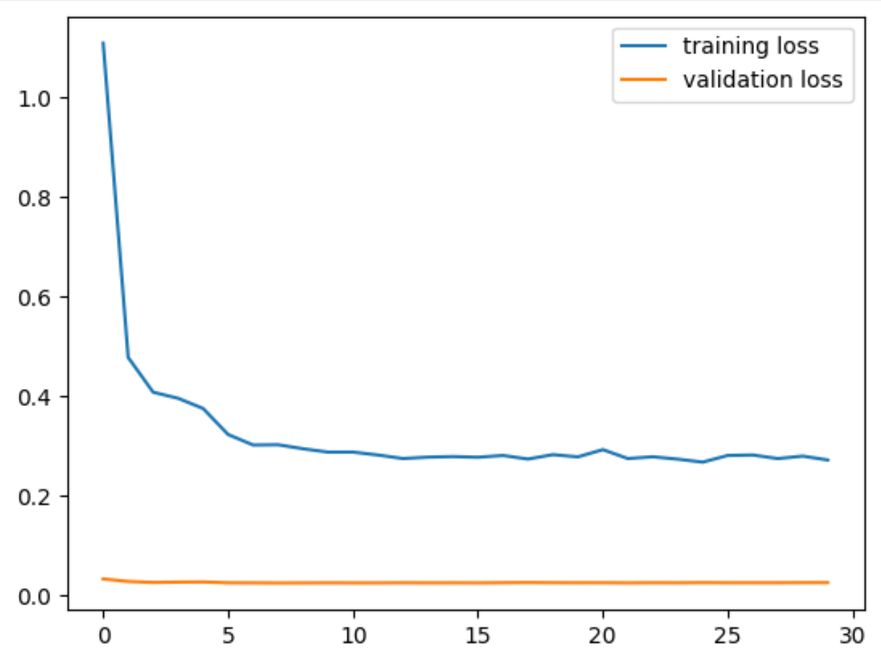
HORIZONTAL FLIP

VERTICAL FLIP

COLORJITTER

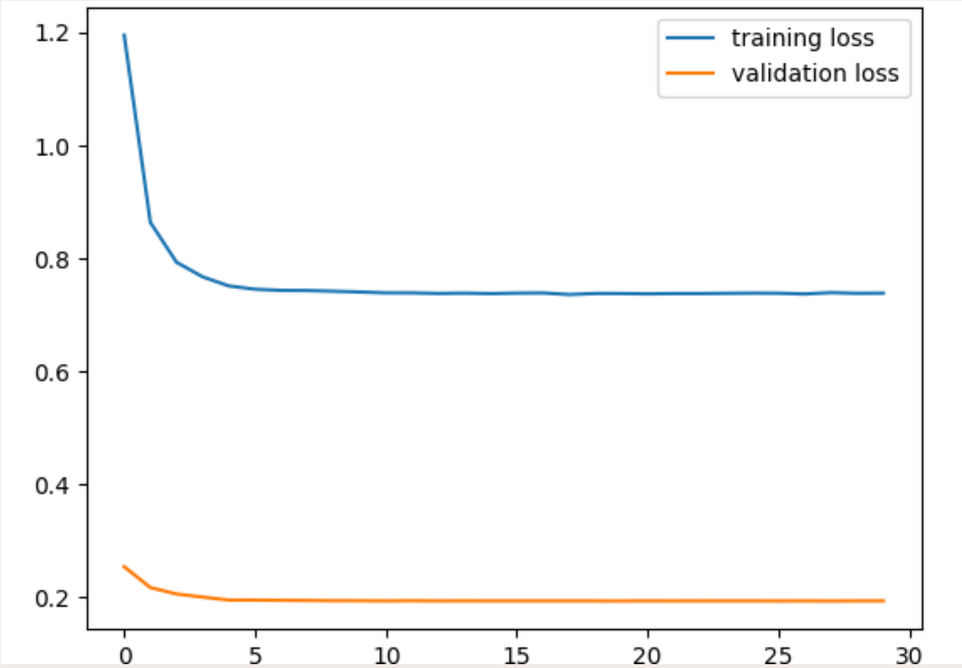
QUANTITATIVE RESULTS

PRETRAINED WEIGTS



MAP 50 - 85%
MAP 75 - 50%
MAP - 54%

TRAINED FROM SCRACTH



MAP 50 - 33%
MAP 75 - 1%
MAP - 11%

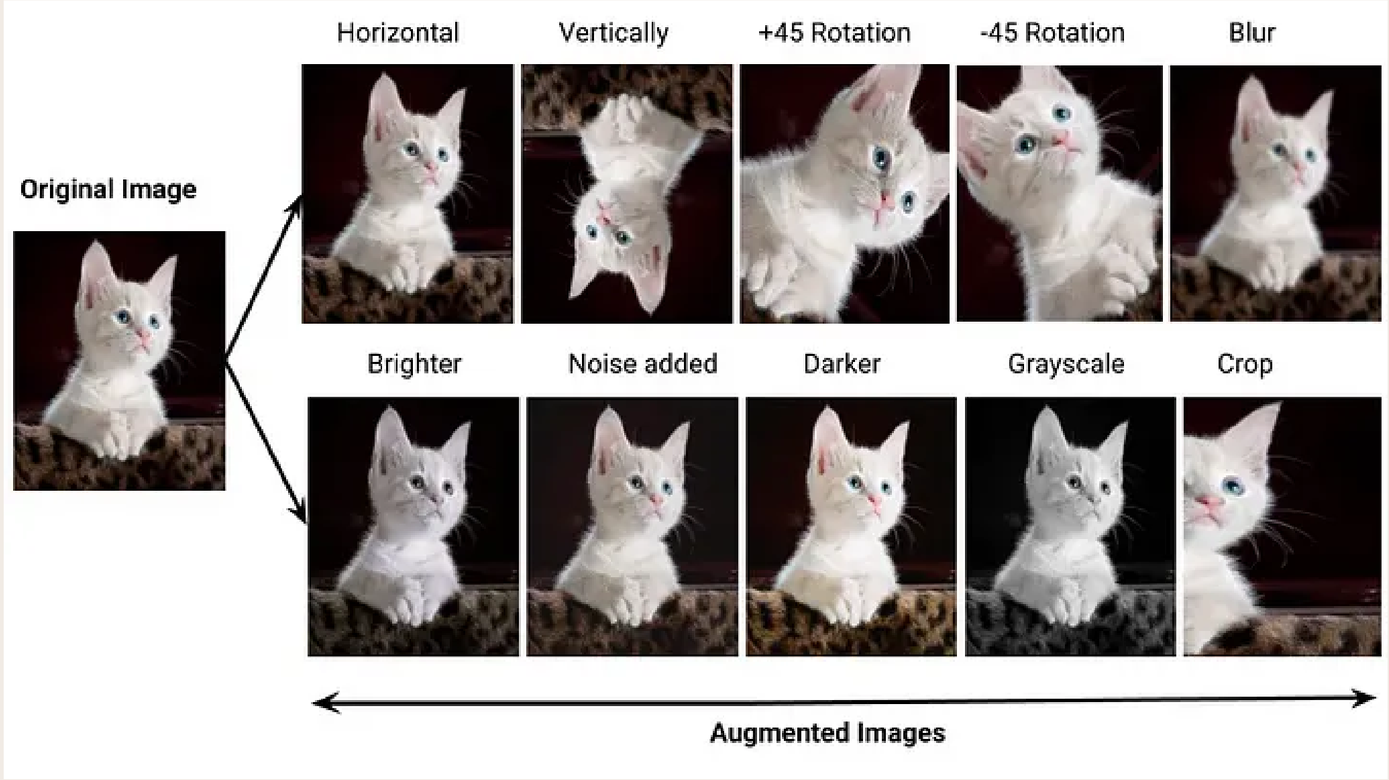
QUALITATIVE RESULTS

2 STAGE : RCNN

- **Epoch:** 30
- **GFLOPs**: 561.18 GFlops
- **Inference time**: 110 ms
- **Number of parameters**: 43.03 M

1 STAGE : YOLO V8

Data augmentation:



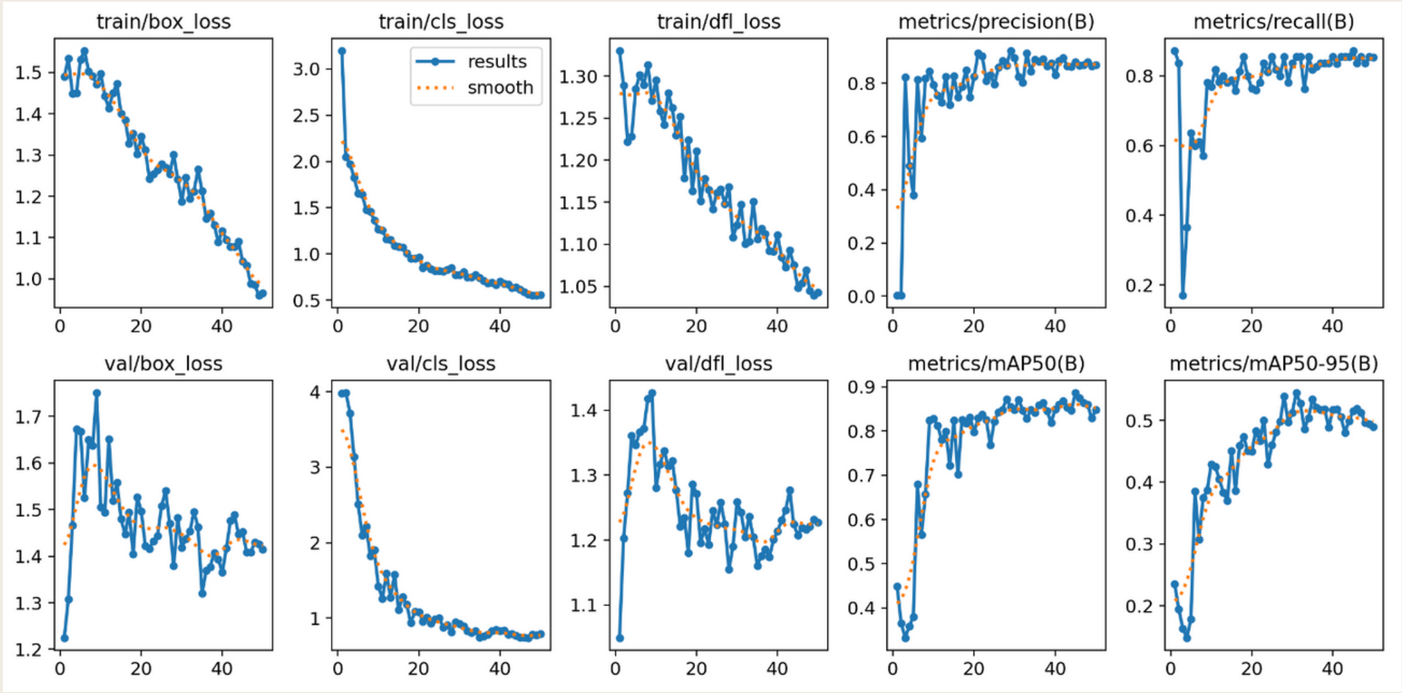
1 STAGE : YOLO V8

Quantitative results:

MAP-50: 93%

MAP-95: 53%

MAP: 49%



1 STAGE : YOLO V8

Qualitative results:

Labels

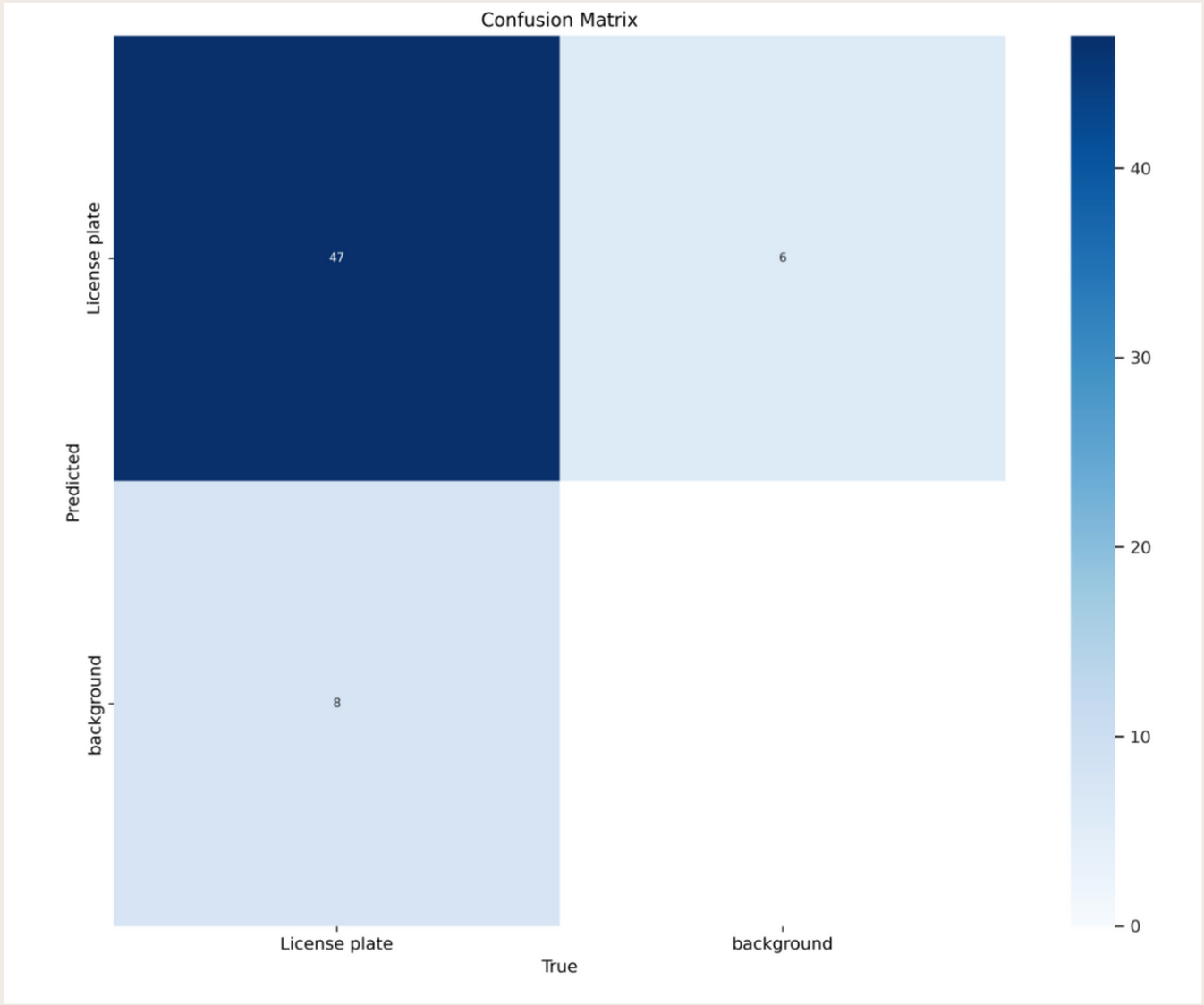
Predictions

1 STAGE : YOLO V8

Qualitative results:

- The detected license plate is the following: **1062FNT**
- The confidence of the interval is the following: **0.78**

1 STAGE : YOLO CONFUSION MATRIX



1 STAGE : YOLO V8

- **Epochs:** 50
- **GFLOPs**: 8.7
- **GMAOs:** 5.9
- **Inference time**: 30,35 ms
- **Number of parameters**: 3M

1-STAGE VS 2-STAGE

1-STAGE		2-STAGE
30ms	TIME PER IMAGE X3	110ms
3M	PARAMETERS X14	43M
8'7	GFLOPS X64	561
49%	ACCURACIES (MAP)	54%

CLASSICAL VS DEEP

CLASSICAL		DEEP
39,97 seconds	TIME TO TRAIN	**3 hours**
12,63 seconds	TIME TO INFER	**110 ms**
All features from images' patches	DATA NEEDED	**Images, plates' coordinates and labels**
37'5%	ACCURACIES	**54%**

CONCLUSSIONS

- **Single Forward Pass**: YOLO performs both object localization and classification in a single forward pass through the neural network.
- **Fewer Operations**: RCNN involves generating region proposals, which usually require additional computational steps.
- **Real-time Processing**: YOLO architectures are often optimized for real-time processing, and every design decision is made with computational efficiency in mind.
- **Unified Detection**: YOLO uses a single network to perform both classification and localization, reducing the overall computational footprint. In contrast, RCNN-based methods typically use separate networks or stages for these tasks.

THANKS FOR YOUR ATTENTION!