License Plate Detection and Character Recognition using Deep Learning and Font Evaluation

Zahra E.Vargoorani

A Thesis

In

The Department

Of

Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Science (Computer Science) at

Concordia University

Montreal, Quebec, Canada

August 2024

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: Zahra Ebrahimi Vargoorani

Entitled: License Plate Detection and Character Recognition using Deep Learning and Font Evaluation

and submitted in partial fulfillment of the requirements for the degree of

Master of Science (Computer Science)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Chair
Dr. Joey Paquet

_____ Examiner
Dr. Tse-Hsun (Peter) Chen

_____ Examiner
Dr. Yang Wang

_____ Supervisor
Dr. C. Y. Suen

Approved by _____
Dr. Juergen Rilling Chair of Department or Graduate Program Director

_____
Dr. Mourad Debbabi Dean Gina Cody School of Engineering

Date August 27, 2024

ABSTRACT

License Plate Detection and Character Recognition using Deep Learning and Font Evaluation

Zahra Ebrahimi Vargoorani

License plate detection and character recognition pose challenges due to environmental sensitivity, such as lighting, dust, and the impact of the chosen font type on recognition tasks. Automatic License Plate Detection and Recognition (ALPR) are crucial in practical applications such as traffic control and parking, vehicle tracking, toll collection, and law enforcement. While much research has been done using image processing and machine learning algorithms, deep learning methods need further exploration due to their recent advances in reliable performance in various scenarios. Moreover, current proposals are limited to specific regions and dataset applicability.

This study has a dual focus: firstly, we suggest utilizing a Deep Learning technique, specifically using Faster R-CNN for the license plate detection task and a CNN-RNN model with CTC loss, and a MobileNet V3 backbone for recognition task. We also utilized You Only Look Once (YOLO) for license plate detection and recognition tasks. Secondly, we aim to assess font features within the LP context. This work uses Brazilian dataset and datasets from two different provinces in Canada and two different states in the United States of America, including Ontario, Quebec, California, and New York State. We suggest employing an adaptive algorithm based on Faster R-CNN and CTC network along with YOLO, fine-tuned with optimized parameters to improve its effectiveness using two different approaches, including domain generalization. Alongside presenting the recall ratio findings, this study will perform a thorough error analysis to gain insights into the nature of false positives. The proposed model demonstrated a commendable recall ratio of 94% using a single YOLO network.

Specific fonts pose readability challenges for humans, while others present difficulties for computer systems regarding recognition. In this study, we provide five sets of outcomes for font assessment: results about font anatomy and those related to the recognition of commercial products. The font anatomy analysis focuses on five specific fonts: Driver Gothic, Dreadnought, California Clarendon, Zurich Extra Condensed, and Mandatory. Additionally, we assess the impact of these fonts in the context of a dataset made of five different license plates using a commercial product, OpenALPR. The font anatomy findings unveil significant confusion cases and quality features associated with chosen fonts.

ACKNOWLEDGEMENTS

My sincere gratitude goes to my supervisor Prof. C.Y.Suen. I would not have been able to finish this work without his valuable advice, assistance, and encouragement. I am very grateful for the outstanding supervision I have received.

I would like to thank the examination committee members, Drs., for their efforts and valuable feedback.

DEDICATION

I am deeply indebted to my parents, Mr Mohsen Ebrahimi Vargoorani and Mrs Leila Samadi Tari for their continuous encouragement, support and patience during the years I was away from home. My deep appreciation goes to my sisters who have all been a source of support.

Finally, my deepest gratitude and appreciation goes to my small family; to my dear husband Mr Mohammad Al-Anjari, for his love, support, and patience which has enabled me to finish my graduate study.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

Automatic License plate recognition system (ALPR) with high accuracy is challenging due to environmental factors such as light conditions, rain, and dust. Moreover, other factors make the detection task harder, like the car's speed, pictures taken from different angles, and low-quality and low-resolution pictures. Additionally, the selected font style on the License Plate (LP) significantly influences the recognition task. License Plate Detection is crucial for different applications, including but not limited to traffic and parking control, enhancing security, and law enforcement. Although many techniques have been applied to license plate detection tasks, including traditional image processing techniques and machine learning algorithms, there is always room for improvement and proposing a method with higher accuracy. Achieving Automatic License Plate Detection and Recognition (ALPR) involves completing distinct phases. Hence, this study takes a dual approach. Firstly, we propose utilizing Deep Learning (DL) techniques in license plate detection and character recognition using two solutions. Secondly, we suggest evaluating font characteristics within the license plate detection and character recognition task. Subsequent sections will delve into detailed explanations of license plates, license plate detection and character recognition, fonts, and font evaluation.

I. License Plates

A license plate, often a metal rectangle, is essential for vehicle identification. Mounted on a vehicle's rear, front, or both sides, it aids in easy recognition and registration. Possessing a license plate is a legal requirement for any vehicle to operate on public streets, playing a pivotal role in maintaining regulatory standards, ensuring safety, and enhancing the transportation system's effectiveness. Globally, designated official bodies are responsible for overseeing vehicle registration and creating license plate designs. This framework is pivotal for adhering to regulations, facilitating accurate documentation, and confirming identities on public thoroughfares.

Additionally, each jurisdiction employs varied background colours, graphics, and fonts on license plates to impart uniqueness and enhance recognition. This distinctive aesthetic serves as a means

of identification and functions as a representation of the jurisdiction. As such, the intentional design of license plates becomes a critical element of regional identity, contributing to practical recognition and a sense of local character on the roadways.

The significance of license plates (LPs) becomes evident in critical situations such as car theft, drug trafficking, speeding, running red lights, or parking offences, emergency alerts, border security and control, and managing parking lots. In such scenarios, deploying a license plate detection system with high accuracy plays a pivotal role in promptly and efficiently identifying the vehicle and its owner. Recognizing the need for enhanced detection and recognition, jurisdictions establish and enforce unique designs for license plates. This contributes to improved efficiency in emergency responses and ensures the proper functioning of automatic license plate recognition systems.

The LP design outlines the following aspects:

- Alphanumeric Characters: License plates contain a combination of letters and numbers. These alphanumeric characters serve as a unique identifier for each vehicle. The format and arrangement of characters can vary by jurisdiction.
- Colour Scheme: License plates often have a specific colour scheme that is associated with the issuing jurisdiction. Different colours may be used for different types of vehicles.
- Jurisdictional Information: License plates typically display information about the issuing jurisdiction, such as the state or country. This helps identify where the vehicle is registered.
- Material: License plates are commonly made of materials like aluminum or plastic. The choice of material can impact the plate's durability and resistance to weather conditions.
- Reflectivity: Many license plates have reflective coatings to enhance visibility, especially during low-light conditions. This improves the plate's readability for law enforcement and automated license plate recognition (ALPR) systems.

- Size and shape: License plates come in various sizes and shapes. The dimensions are often standardized within a jurisdiction to ensure uniformity and compatibility with vehicle mounting spaces.
- Font and typeface: The style, size, and readability of the font used for alphanumeric characters are essential design considerations. Clear and legible fonts are crucial for easy identification.

The choice of typeface plays a pivotal role in identification. Historically, typefaces were specifically selected to ensure clear and accurate readability by the unaided human eye, particularly in natural daylight and from a considerable distance. This deliberate selection aimed to optimize visibility and recognition, emphasizing the importance of legibility for effective identification under typical daytime conditions.

This work considers five datasets. The first dataset is from California/United States, the second is from New York State, and the third and fourth are from Ontario and Quebec. The fifth dataset is Brazilian[5] dataset.

1. California License Plates

In the state of California, United States, license plates for private vehicles typically adhere to the format of three letters followed by three numbers, such as ABC 123 or one digit followed by three letters and then three digits, such as 1 ABC 234 also people can customize the combination of letters and numbers and choose their pattern. California license plates often include the state name ("California") and a distinctive slogan, "SESQUICENTENNIAL – 150 YEARS," for some license plates since this slogan was added only three years ago. The designs may also incorporate additional elements that reflect the state's rich cultural and natural diversity. Figure 1 depicts an LP in the state of California.

The size of California license plates for regular vehicles is standardized, but the exact dimensions may vary slightly. The size is generally around 12 inches by 6 inches in width for cars and 7 inches in height by 4 inches in width for motorcycles. The background colour of California license plates is dark blue on white with a red state name graphic and slogan printed in red at the bottom. The typeface used in the LPs is California Clarendon. This dataset uses a combination of white, yellow, and black colours since these pictures are taken from different time periods.

Figure 1. Sample California State License Plate

2. New York State License Plates

License plates for private vehicles in New York adhere to a pattern commonly comprised of three letters followed by a dash and concluded with three numbers (e.g., ABC-1234). The license plates proudly display the state name, "New York," and incorporate the motivational slogan "Excelsior," signifying "ever upward" in Latin. Figure 2 provides a representation of a New York state license plate.

The size of New York State license plates for cars is typically around 12 inches in height by 6 inches in width. These plates are known for their clean and recognizable aesthetic, with a prevalent background colour of white, contributing to their appealing look.

Figure 2. Sample New York State License Plate

A Zurich Extra Condensed font is employed in the design of these license plates, ensuring a modern and legible appearance. Notably, the dataset from which these details are derived may include colour variations, incorporating a combination of white and yellow hues due to the diverse periods from which the images originate. This amalgamation of colours adds a layer of historical context to the visual representation of New York State license plates.

3. Ontario License Plates

Ontario has witnessed diverse license plate designs throughout history, each crafted to incorporate distinctive and visually striking elements. In Ontario, the alphanumeric format usually adheres to a structured pattern, such as ABCD 123, with variations contingent on the specific series. Also, residents have the option to have their personalized license plate.

These license plates often bear the symbolic slogan "Yours to Discover," serving as a captivating reflection of the province's enduring appeal. Maintaining adherence to standard North American dimensions, these license plates typically measure approximately 12 inches in height by 6 inches in width, with a prevailing background colour of white complemented by contrasting lettering, contributing to their distinctive and easily recognizable visual identity. The typeface used in the Ontario license plate is Dreadnought. Figure 3 showcases a representative sample of the latest license plates designed for Ontario vehicles.

Figure 3. Sample Ontario License Plate

4. Quebec License Plates

Throughout the years, Quebec has embraced numerous license plate designs, each reflecting distinct visual elements. Notably, the diversity extends to various categories of license plates, each designated for specific vehicle types and purposes. A significant recent advancement involves the approval of personalized license plates, commonly referred to as vanity plates. This allows vehicle owners to incorporate a personalized and unique element into their license plates.

Quebec license plates for private passenger cars have followed the A12-BCD alphanumeric format since 2010, featuring the enduring slogan "Je me souviens," translating to "I remember." The standard size adheres to typical North American dimensions, approximately 12 inches in

height by 6 inches in width, with a specific background colour of white and blue lettering. The typeface used in the Quebec license plate is Driver Gothic.

Figure 4 showcases a representative sample of the latest license plates designed for Quebec commercial vehicles. This exemplifies the province's commitment to tailoring license plate designs to suit different vehicle categories, maintaining a dynamic and visually appealing approach to vehicle identification.

Figure 4. Sample Quebec License Plate

5. Brazilian License Plates

In Brazil, license plates (LP) follow the format ABC.1234. Once assigned to a vehicle, this combination cannot be transferred to another vehicle. Each plate features a metallic band displaying the state abbreviation (e.g., RJ for Rio de Janeiro) and the municipality's name positioned above the plate number.

The dimensions of license plates differ between cars and motorcycles. Car plates measure 400 mm x 130 mm, whereas motorcycle plates are 200 mm x 170 mm. Additionally, the color of the plates varies according to the vehicle category. For instance, private vehicles have gray plates, while public vehicles, such as buses and taxis, use red plates. The typeface employed for these plates is called Mandatory.

Brazil introduced a new license plate design starting in 2018, with the expectation that all vehicles will adopt this new design by 2023. Despite this transition, the dataset referenced here still uses the old license plate design for all vehicles. Figure 5 showcases a representative sample of the latest license plates designed for Brazilian vehicles.

Figure 5. Sample Brazilian License Plate

II. License Plate Detection and Character Recognition

Recently, numerous researchers have utilized deep neural networks to tackle various challenges spanning diverse domains such as image and natural language processing. Among these tasks, object detection is a significant area within image processing where researchers leverage deep learning methodologies to address issues. However, recently, a new version of the model called YOLO[1], which stands for "You Only Look Once," has shown promising results in contrast to conventional object detection systems, which typically examine a predefined grid of cells or suggest regions of interest, YOLO takes a distinctive approach.

The effectiveness of deep learning algorithms is significantly dependent on the calibre and variety of the training dataset. In this study, we utilized the License Plate dataset provided by the CENPARMI at Concordia University in Montreal. This dataset focuses explicitly on license plates in complex scenes and comprises many images from Quebec, Ontario, New York, and California. The photos were captured using a mobile phone camera and were collected and annotated by the authors of this research. We also used Brazilian dataset[5] to compare the results.

For our study, a comprehensive dataset of license plates from cars was carefully labelled with the DINO (DIstillation with NO labels) approach, a self-supervised learning technique that utilizes Vision Transformers (ViT) to train models without labelling data.

We employed two solutions to conduct our experiments on the license plate detection and character recognition tasks. In our first solution, to separate the license plate from the image, we employed the Faster R-CNN[2] (Region-based Convolutional Neural Network) model, renowned for its exceptional performance in object detection tasks. This model is built upon the powerful

ResNet-101 (Residual Network) architecture, effectively capturing intricate features and hierarchies in visual data. The Faster R-CNN model provides a unified framework for accurate object detection by seamlessly integrating region proposal generation and object classification. After detecting license plates using Faster-RCNN[2], we utilized Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) with Connectionist Temporal Classification (CTC)[3] Loss to recognize the letters on the plates. This model significantly enhances the accuracy and efficiency of automatic license plate recognition systems. The system is adept at handling complex scenarios by leveraging CNNs for spatial data processing and RNNs for sequence prediction. Including CTC Loss[3] helps align input sequences with their corresponding outputs, marking a substantial improvement over traditional methods. This advancement broadens the applications of such systems in areas like autonomous driving and surveillance, making them more effective in automation and intelligent monitoring.

In our second solution, we implemented the latest version of YOLO[1], YOLOv8, to detect license plates and recognize letters. This model is renowned for its real-time performance and efficient computational processing; YOLOv8 is particularly advantageous for license plate recognition tasks. Its lightweight architecture balances speed and accuracy, making it suitable for deployment on resource-constrained devices. With its ability to generalize across various scenarios and ease of integration into existing systems, YOLOv8 stands out as a compelling choice for developers and researchers in the field of license plate recognition.

To assess the outcomes derived from the experiments outlined earlier, we suggest employing the subsequent metrics for evaluation:

1. Recall ratio: The recall ratio, pivotal in classification tasks, is a fundamental measure for assessing the model's ability to identify pertinent objects within a given dataset accurately. It quantifies the ratio of true positives, denoting correctly identified instances, to the combined total of true positives and false negatives, representing instances the model missed. Consequently, a higher recall value indicates the model's efficacy in capturing a more substantial portion of relevant objects in the dataset, thereby emphasizing its proficiency in object detection and classification tasks. This metric holds particular significance in scenarios where comprehensive coverage of

relevant objects is crucial for the success of the classification process, such as in medical diagnosis or security surveillance applications.

2. Error analysis: it is crucial for understanding false positives (FP)[8], instances where the model incorrectly identifies objects. FP represents the count of erroneously detected objects. Conducting error analysis provides insights into common sources of errors and facilitates targeted improvements in model performance, ultimately enhancing accuracy and reliability.

III. Fonts

A font is a cohesive collection of text characters, including letters, numbers, and symbols, defined by unique style and size characteristics. At the same time, a typeface represents a family of fonts sharing common design traits. Each font's identity is shaped by various attributes, such as style, weight, width, and decorative elements, alongside the specific designer or foundry behind its creation. Conversely, a typeface consists of glyphs for individual letters, numbers, and symbols designed to convey text with both readability and aesthetic appeal. Some typefaces are crafted for specialized uses, like mathematical notation or music scores, illustrating the role of typography in enhancing the visual presentation and interpretation of text. This intricate relationship between font and typeface highlights the significance of typography in communication, allowing for a nuanced expression beyond the written word itself. Table 1 lists some of the most important characteristics and their definition [6]. Figure 6 shows an example of these features and how they relate to sample letters. The figure is extracted from [7]. Selecting the right font is essential for ensuring a system can accurately and efficiently interpret text, boosting its reliability and overall performance. Thus, the font choice goes beyond just visual appeal and becomes a crucial element in the effectiveness and success of computer-based text recognition systems. This decision impacts the accuracy with which these systems process information, highlighting the importance of font selection in the operational functionality of such applications.

Table 1. Font Characteristics [6]

Characteristics	Definition
Weight	Thickness of the character outlines relative to their height
Slope	Used to represent italic type or oblique type to emphasise important words
Width	Represents the character's width or stretch
Serif and sans serif	A small line attached to the end of a stroke in characters. Fonts can be either serif or sans serif
Baseline	The optical line on which type sits
Mean line	An optical line created by the eye moving across the top of a set of lowercase letters
Ascender	The stem of a lowercase letter which extends past the mean line
Descender	The stem of a lowercase letter which extends below the baseline
x-height	The height of the lowercase letters without descenders or ascenders
Cap line	An optical line created by the eye moving across the top of a set of uppercase letters
Bowl	Fully or modified rounded forms found in letters such as C
Counter	Negative space fully or partially enclosed by the stroke of a letter form like in B
Tail	Short downward strokes
Crossbar	Horizontal stroke connecting two parts of a letterform like in H
Apex	Juncture of a stem like in letters A and M
Spur	A projection smaller than a serif, that reinforces the point at the end of curved stroke. It can be found in G

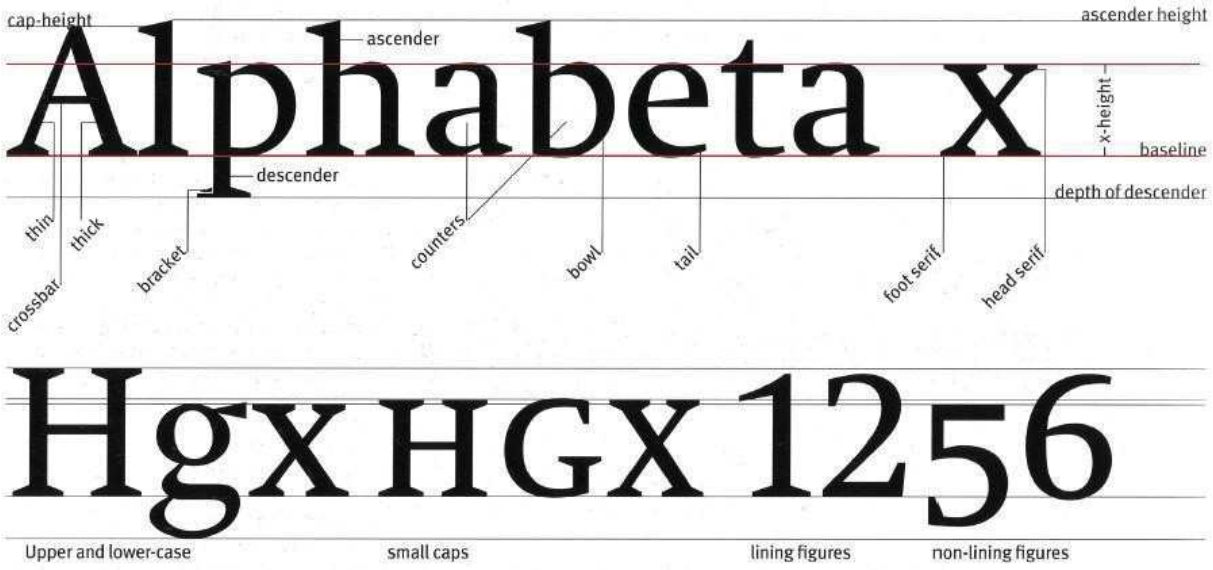

Figure 6. Example of Font Characteristics [7]

IV. Digital Font Evaluation

Automatic license plate systems are based on detecting the license plate and recognizing the letters. In the detection phase, the system identifies the presence of a license plate within an image and in the recognition phase, segmented characters are analyzed and interpreted into readable text. It's important to acknowledge that even advanced segmentation algorithms sometimes fail to isolate letters and digits perfectly from their segments, leading to recognition errors. The font choice plays a critical role in the success of license plate recognition. A well-chosen font can significantly enhance the system's ability to recognize license plate numbers accurately by ensuring each character remains legible and intact despite potential segmentation flaws.

This work proposes the following contributions:

1. Study the characteristics of 5 different fonts: Driver Gothic, Dreadnought, California Clarendon, Zurich Extra Condensed, and Mandatory. Confusing cases will be considered for each font separately.

2. Analyze the impact of different fonts in specific contexts using two datasets: the CENPARMI dataset for Driver Gothic, Dreadnought, California Clarendon, and Zurich Extra Condensed fonts, and the UFPR-ALPR[5] dataset for the Mandatory font.

We assessed our work by comparing our results with a commercial product called OpenALPR [4]. OpenALPR is a software solution designed to identify license plates from videos and photos of vehicles. This software comes in both open-source and commercial versions, with the commercial variant featuring advanced capabilities and support suitable for enterprise use.
From the results we have gathered, we propose a series of recommendations regarding suitable font characteristics that could improve real-time recognition accuracy.

V. Thesis Structure

The remainder of this thesis is structured as follows: Chapter 2 reviews some related works regarding ALPR systems. Chapter 3 describes the different models that have been proposed. Chapter 4 details and analyzes the results acquired in this project. Chapter 5 font analysis and results. Chapter 6 concludes the study and proposes directions for future research.

Chapter 2. Literature Review

Vehicle license plate recognition is a critical topic in automatic transportation systems due to its importance in creating intelligent and safe cities. Researchers focus on various techniques to achieve high detection ratios, broadly categorized into machine learning and deep learning approaches. License plate recognition typically involves detecting the vehicle, locating the license plate, and recognizing characters. However, challenges like varying license plate designs, diverse angles, unpredictable fonts, and image quality make these steps difficult. While technology has made plates more legible, limitations persist due to controlled conditions like fixed illumination and monochrome backgrounds. This chapter reviews license plate detection techniques using machine learning in section (I) and deep learning techniques in section (II). Section (III) presents some related work for font evaluation.

I. Machine Learning Based Techniques

In this section, we delve into various machine learning (ML) methods for detecting license plates. The work in [10] used the K-means algorithm to recognize license plate characters. They improved the traditional K-means method by introducing an automatic cluster number determination system using scale-invariant feature transform (SIFT) key points.

A 6-layer cascaded classifier was employed for license plate localization, utilizing global edge detection and local Haar-like features. Various image processing techniques, such as binarization, vertical edge detection, horizontal and vertical projections, and a modified K-means segmentation algorithm, were also implemented to refine character segmentation.

The researchers demonstrated the effectiveness of their methods by achieving an impressive 94.03% accuracy rate on a dataset of 578 images containing 3502 characters from Chinese license plates. This was accomplished by combining their innovative methods with the Tesseract Optical Character Recognition (OCR) software, enabling them to convert image text into machine-readable characters with exceptional precision.

However, it's worth noting that the methodologies required multiple preprocessing steps, which made the process time-consuming. The study demonstrates that advanced segmentation techniques, such as this improved K-means algorithm and layered classifiers, can enhance license plate recognition accuracy. Despite the high computational demand and the time-intensive nature of preprocessing, these results indicate the potential of sophisticated machine-learning approaches to improve recognition in practical applications.

In [11], the authors introduced a K-Nearest Neighbors (KNN) algorithm with pre-training steps to recognize numbers and letters on multi-style license plates. These included single-line and double-line plates with complex backgrounds and various character colours, explicitly focusing on Korean and U.S. plates.

The algorithm's performance was rigorously evaluated using a substantial dataset. A 50-minute video, featuring 138 vehicles with a diverse range of license plate styles, was analyzed. The system demonstrated exceptional results, achieving character recognition accuracy of over 99% and maintaining a processing time of less than 50 milliseconds per character.

Despite its impressive speed and accuracy, the system had a limitation: it could only effectively identify Korean and U.S. license plates.

In reference [12], the authors introduced a novel method for locating and recognizing text in natural scene images with complex backgrounds. Their multi-step approach began with identifying superimposed text regions using various character descriptor features, including bounding boxes, perimeter, Euler numbers, and horizontal crossings. Once these text regions were isolated, Support Vector Machine (SVM) classifiers were used to determine if the region contained letters.

Line segmentation was then conducted using horizontal profiles, and individual characters were separated through vertical profiles. The researchers utilized Optical Character Recognition (OCR) tools for accurate character recognition. The method achieved different accuracy rates depending on the algorithms used: 64.4% with the Otsu algorithm, 75.04% with AdaBoost, and 78.8% with SVM.

While this approach performed well even with complex backgrounds, it emphasized the need for a more specialized license plate character recognition system. Nevertheless, this technique demonstrated the potential to enhance character detection in visually challenging environments.

In reference [13], the authors employed a Histogram of Oriented Gradients (HOG) feature-based Support Vector Machine (SVM) classifier to detect and recognize license plates. The training data came from Google Images, providing a diverse set of clear and visible license plates for the model's training process. The methodology effectively extracted license plates with high precision using the HOG-SVM detector.

However, the approach needed more design considerations for more challenging real-world scenarios, potentially limiting its application in practical settings. Moreover, although the authors claimed that their classifier achieved high accuracy, they needed to provide more qualitative or quantitative data to verify this claim. More detailed results that raise questions regarding the classifier's performance across varied conditions need to be produced, emphasizing the need for further validation in different real-world scenarios.

In reference [14], the researchers developed a novel approach for detecting and recognizing Indian license plates using template matching. This method identified characters within an image by comparing them to predefined templates, incorporating English and Hindi characters to reflect India's multilingual landscape. Before applying template matching, the researchers used morphological and thresholding operations to enhance character clarity, improving recognition.

The license plate was flagged as invalid if any detected characters did not match the templates. This stringent approach yielded impressive results, achieving localization, segmentation, and recognition rates of 92%, 97%, and 98%, respectively.

One of the significant challenges in machine learning (ML)[15] involves feature extraction, which must occur before any model training begins. Selecting relevant features is crucial, dramatically influencing the model's performance. Proper feature selection can enhance a model's accuracy, efficiency, and generalization ability. Conversely, choosing irrelevant or noisy features can result in overfitting, higher computational costs, and subpar results.

II. Deep Learning Based Techniques

In this section, we delve into various deep learning techniques for detecting license plates.

In the study presented in [16], the researchers employed the Single Shot Detection (SSD) algorithm to detect license plates, replacing the typical base algorithm, the VGG Network, with the Residual Network (ResNet) to enhance detection performance. They chose ResNet due to its architecture, which prevents gradient vanishing problems despite being a deep network, thus allowing it to maintain high performance across multiple layers.

Their results demonstrated that ResNet substantially improved license plate detection, achieving an average accuracy of 85.5%, compared to the VGG Network's slightly lower 83.6%. This improvement illustrates how ResNet's design enables better detection outcomes by managing gradients effectively and maintaining strong learning capabilities in deep networks.

These enhancements in detection accuracy highlight the significance of continuously refining base algorithms to optimize computer vision tasks like license plate detection.

In reference [17], the researchers enhanced the FAST-YOLO network to detect both frontal vehicle views and their corresponding license plates. They customized the FAST-YOLO model to detect both cars and license plates simultaneously. For character detection and recognition, they adapted the YOLO architecture and implemented a heuristic strategy to refine their results. This modification improved the final detection and recognition outcomes.

The authors reported that their refined approach achieved a 63.18% accuracy rate for accurately detecting and recognizing license plates, compared to Sighthound, a competing method, which achieved only 55.47% accuracy.

These findings highlight the advantages of customizing neural network architectures and incorporating heuristic techniques to enhance detection precision. By refining both the detection and recognition phases, this approach demonstrated significant improvements in identifying vehicles and their license plates, emphasizing the value of continuous refinement and network customization in computer vision tasks.

In reference [18], the researchers used a Convolutional Neural Network (CNN) to extract relevant features, utilizing convolutional layers to identify the distinct visual patterns in license plates.

These features were then passed to a Recurrent Neural Network (RNN) with 36 hidden units, which sequenced the detected characters to accurately reconstruct the license plate information.

By combining CNNs' spatial pattern recognition with RNNs' sequential learning capabilities, they achieved a 76% accuracy rate for recognizing all characters on a license plate and a 95.1% accuracy rate per individual character. This approach showcases the complementary strengths of CNNs and RNNs: the CNN excels at capturing intricate visual details, while the RNN is adept at processing sequences of characters in the correct order.

In reference [19], the authors employed a Generative Adversarial Network (GAN) to enhance their system's accuracy by generating synthetic training images. This approach expanded their dataset, enabling them to train a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN). The pre-trained network was subsequently fine-tuned on their specific dataset for license plate recognition. Using synthetic images produced by the GAN, the researchers achieved an impressive 92.1% overall accuracy in license plate recognition and a remarkable 98% accuracy in recognizing individual characters.

Using a pre-trained network significantly improved the performance of the final model, illustrating the effectiveness of GANs in generating additional training data to address data scarcity and bolster machine learning systems. Their approach demonstrates how GANs can increase accuracy by supplying high-quality, varied data, resulting in more robust and reliable license plate recognition systems adaptable to real-world scenarios.

In reference [20], the researchers designed a model that integrates Convolutional Neural Networks (CNN) with Gated Recurrent Units (GRU). The CNN component was used to extract features from images, while the GRU sequenced characters for recognition without requiring segmentation. Suvarnam and colleagues trained a two-dimensional CNN model using resized input images. Unlike traditional sliding window methods, their algorithm enabled hidden layers to analyze features from the entire image.

Their study showed that this combination of CNN and GRU outperformed layout-matching methods and standalone CNN models, demonstrating its effectiveness for automatic license plate recognition (ALPR). The system achieved a 90% accuracy rate on a dataset of 5,000 training images. However, the researchers acknowledged that this accuracy needed improvement for

practical ALPR applications, suggesting that further refinement and additional data would be necessary to ensure reliable, high-precision recognition.

In the research detailed in [21], the authors proposed a system structured as a three-stage pipeline. The first stage involved training a basic CNN with five convolutional layers to detect vehicles by identifying regions with the highest Intersection-over-Union (IoU) overlap relative to the ground truth bounding box.

In the second stage, another CNN focused on accurately identifying candidate license plates from high-confidence vehicle regions detected by the initial CNN.

The final stage involved refining the bounding boxes using the distinctive edge features of license plates. Specifically, the borders of high-confidence bounding boxes were expanded by 30% on each side, and the Canny edge detection algorithm was applied to this enlarged area to pinpoint the precise contours of the license plates.

This detection system was evaluated using a dataset that included various vehicle types under diverse real-world traffic scenarios and achieved a recall rate of 90.51%. However, the authors noted that the three-stage methodology could result in longer processing times, potentially slowing the overall detection process.

[22] describes a distinct CNN-based method for license plate detection that features a three-phase process. The first phase, known as image partitioning, involves processing RGB images and segmenting them into subregions measuring 120 by 180 pixels each. This ensures that each subregion contains only one license plate while the entire plate fits within the specified dimensions. This careful partitioning reduces the possibility of overlapping plates, simplifying the identification process.

In the second phase, called region processing, each sub-region is analyzed by a Convolutional Neural Network (CNN), which assigns a score between 0 and 1. This score indicates the likelihood that a specific sub-region contains a license plate, providing a clear assessment of plate presence and guiding subsequent steps toward the most promising areas.

The final phase, result integration, involves reviewing the CNN scores to pinpoint regions of interest. This step determines which sub-regions will most likely contain license plates, refining the detection process and improving accuracy.

In [23], the researchers presented a license plate detection model designed to function effectively in challenging environments. In [24], the YOLO algorithm was adapted to detect license plates in complex scenarios. This study utilized two versions of YOLO: the original YOLO model and the more advanced YOLO9000 model. Both versions were modified to include fully connected layers that divided input images into 11x11 grids to improve detection accuracy.

Each model was trained on datasets of different sizes: 1500, 2861, 3161, and 1661 instances, but all models were tested on a consistent dataset for accurate comparison. The results indicated that the YOLO9000 model consistently outperformed the original YOLO model. The original model performed poorly because the training dataset needed to represent the real-world variations adequately. However, despite training on a smaller dataset than the second and third models, the fourth model delivered the highest precision results. This suggests that the dataset used in the fourth model was comprehensive enough to cover the critical aspects required for license plate detection.

III. Font Evaluation

For font evaluation, we present some of the works in the literature that look at different aspects to evaluate fonts and examine their impact on text readability.

The research in [25] investigates the role of fonts in digital publishing and display. Given the various fonts available in different visual styles, understanding their impact on daily digital usage is crucial. The study emphasizes that fonts play a significant role in letter identification, essential for human word recognition. Font characteristics such as spacing influence reading and comprehension by signalling pauses, grouping information, and revealing the complexities of the reading process.

The authors propose a new shape descriptor to recognize ancient characters, analyzing the graphic features of inscriptions. This descriptor is invariant to translation, rotation, and scaling. Experimental results confirmed its effectiveness in character classification.

The study also illustrates how typography provides valuable insights into the relationship between font legibility and design elements. The research highlights how typography influences

the overall reading experience by demonstrating that design characteristics directly affect readability.

The research presented in [26] describes a collaborative initiative between type design and science, creating a new typeface called Sitka. This typeface was designed to assess font legibility from the viewpoint of typographers. The authors stress that enhancing the recognizability of individual characters is vital for improving word readability. Therefore, they developed Sitka, a versatile typeface explicitly tailored for digital display. Sitka is a serif font in Roman, Italic, and Bold, supporting Latin, Greek, and Cyrillic characters.

The primary goal of this design is to accelerate evaluation studies and facilitate iterative design processes. Various letterforms were evaluated through several iterative stages, with the team analyzing results at each stage to refine the design. The research prioritizes legibility, emphasizing the importance of creating a readable typeface while conducting comprehensive legibility studies.

The findings revealed a trade-off associated with a large x-height. While larger sizes benefit neutral-height letters, they are less suitable for letters with ascenders and descenders. Furthermore, the study corroborated previous research, particularly the inherent challenges of narrow letters.

In [27], the researchers focused on designing legible fonts from a distance, explicitly identifying the ideal characteristics for signage fonts. They emphasized that the viewing distance significantly impacts the readability of more minor design elements and finer details. Their findings revealed that serifs on vertical stems could improve legibility at a distance. However, low-contrast sans-serif fonts also performed well in certain situations, particularly when halation occurred or light was projected from behind the letters.

The study generally found that open inner counters improved legibility at a distance. The researchers suggested increasing the x-height and broadening letter shapes to address this. Moreover, expanding the spacing between letters further enhanced readability from afar.

In [28], the authors explored using Personal Digital Assistants (PDAs) in specific Arabic communities and investigated the appropriate fonts for these devices. They evaluated 13

typefaces based on six different factors, with each font standardized to a fixed size. A normalization process was used for fonts of varying heights to maintain their aspect ratio.

Experiments were conducted to evaluate the legibility of both letters and words. The results of the phrase legibility tests were combined with those of the letter legibility assessments to form a comprehensive metric.

The findings indicated that the Almohanad font achieved the highest legibility score, reaching 79%. Due to their high legibility, the Script Hafs and Geeza Pro fonts are also recommended for e-books. Given their effective performance with smaller-sized characters, Almohanad, Geeza Pro, and Yakout Reg fonts are particularly suitable for devices with small screens, such as the iPad Mini.

Chapter 3. Proposed Model

Detecting license plates in complicated environments poses a significant challenge because they are vulnerable to various external factors such as rain, dust, shadows, and fluctuating lighting conditions. These elements can severely impact detection accuracy, creating additional hurdles for effective LPD. Furthermore, achieving reliable license plate detection becomes even more demanding in real-time systems, where the necessity for rapid and accurate recognition leaves little room for error or delays. These constraints underscore the difficulty of developing an LPD system that consistently performs under varied and unpredictable conditions.

The font style used on license plates plays a crucial role in the recognition phase of computer-based analyses. It significantly impacts recognition accuracy, making the selection of an appropriate font essential. Despite its importance and substantial influence on recognition outcomes, research focusing on evaluating fonts specifically for recognition purposes still needs to be completed in the literature. This gap suggests a need for more comprehensive studies to understand how different font styles affect recognition accuracy and to identify the most effective ones for automated systems.

Therefore, this study has two main objectives. First, it aims to apply Deep Learning (DL) techniques to license plate detection and recognition of letters. Second, it seeks to assess font characteristics within the context of license plates.

The main contributions of this work are:

1. Examine two approaches for detecting license plates and recognizing letters using Faster-RCNN[2], CTC Networks[3], and YOLO[1] deep neural network with domain generalization approach.
2. Conduct a qualitative assessment of the incorrect predictions made by the trained deep learning models.
3. Perform an analytical evaluation of five fonts commonly used on license plates and explore how typeface design can impact computer-based studies in this context.

The remainder of this chapter is structured as follows: Section I introduces the datasets utilized in this research, providing an overview of their composition and relevance. Section II addresses the formulation of the license plate detection problem and outlines the implementation of the deep learning models for this purpose. Section III explains the proposed detection model in detail, highlighting its unique features and how it improves upon previous approaches. Finally, Section IV presents the study conducted to evaluate various fonts used on license plates, discussing the methodology and findings about recognition performance.

I. Datasets

The characteristics and volume of the datasets utilized significantly influence deep learning methodologies. For this study, experiments were conducted using two distinct datasets:

1. The UFPR-ALPR dataset[29], curated by the Federal University of Paraná, encompasses a comprehensive collection of 4,500 license plate images sourced from vehicles, including motorcycles, cars, and public transportation within Paraná, Brazil. This dataset captures the diversity of real-world conditions by employing three distinct cameras: the GoPro Hero4 Silver, Huawei P9 Lite, and iPhone 7 Plus. It's structured to facilitate machine learning applications with a strategic division of the data—40% is designated for training purposes, 20% for validation, and the remaining 40% for testing, ensuring a balanced distribution across different segments. This dataset is particularly notable for its dynamic range of capture scenarios, mirroring real-world conditions. All images are obtained from vehicles in motion during daylight hours, which presents unique challenges for automatic recognition systems. The proximity of the camera to the license plates varies significantly, ranging from as close as 1 meter to as far as 10 meters. Furthermore, about 30% of the images contend with varying environmental elements such as intense sunlight, shadows, and dust. This variability is crucial for developing and testing the robustness of automatic license plate recognition technologies under a wide array of operational conditions. Figure 7 displays example images from this dataset, while Figure 8 illustrates the distribution of letters and digits within the UFPR-ALPR[29] dataset. Detailed counts for each letter and digit are provided in Table 2.

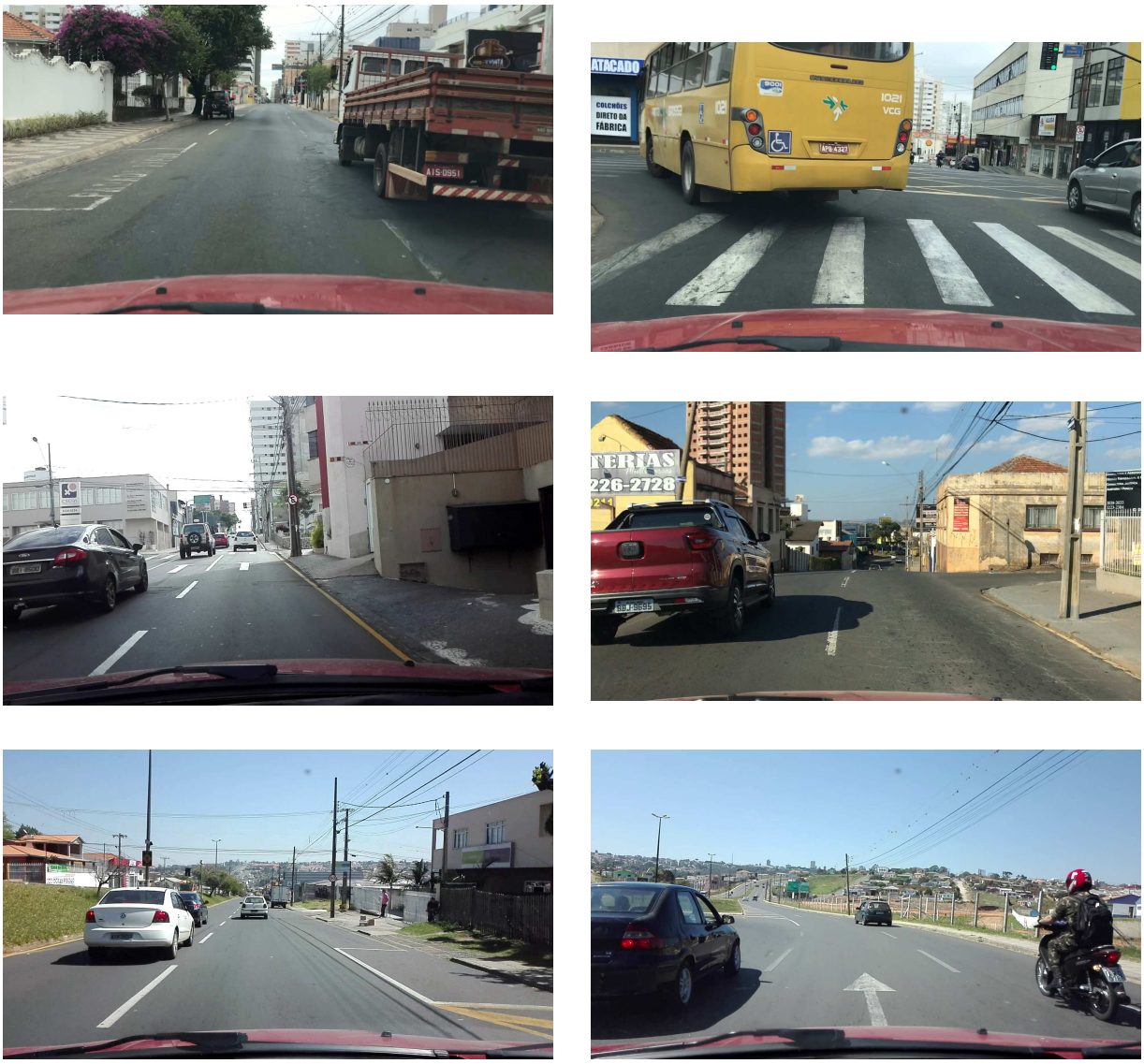

Figure 7. Example Images from UFPR-ALPR Dataset

2. The Centre for Pattern Recognition and Machine Intelligence (CENPARMI) has curated a dataset of license plates captured in complex environments, emphasizing the rigorous methodology behind its creation. This dataset includes 1600 images of license plates sourced from California and New York in the United States and Ontario and Quebec in Canada, all photographed using a mobile phone camera. The researcher of this thesis meticulously collected and annotated images. The CENPARMI License Plate dataset is particularly well-suited for training advanced deep learning models due to its diverse and challenging features. The images span a broad spectrum of lighting scenarios, ranging from harsh sunlight and

shadowy conditions to dimly lit parking lots, presenting a significant challenge for consistent license plate visibility. Several images feature numerous license plates, which complicates the task of accurately detecting and recognizing each plate. The distance between the camera and the license plates varies significantly, from as close as 1 meter to as far as 30 meters. This disparity introduces additional complexity in detecting and focusing on the plates. Around 20% of the images show license plates at an angle, complicating their detection and recognition due to their non-standard orientations.

Additionally, the dataset captures images from various types of vehicles, adding further intricacies such as different license plate sizes, vehicle distances, and a myriad of background colours that affect the imaging conditions. These elements pose substantial challenges for license plate detection (LPD) algorithms.

The goal is to develop deep learning models adept at managing the multifaceted nature of real-world license plate recognition by utilizing the comprehensive and varied CENPARMI License Plate dataset. The documentation provides a glimpse into the dataset, showcasing sample images and the distribution of letters and digits among the collected license plates. Figure 9 presents sample images from this dataset.

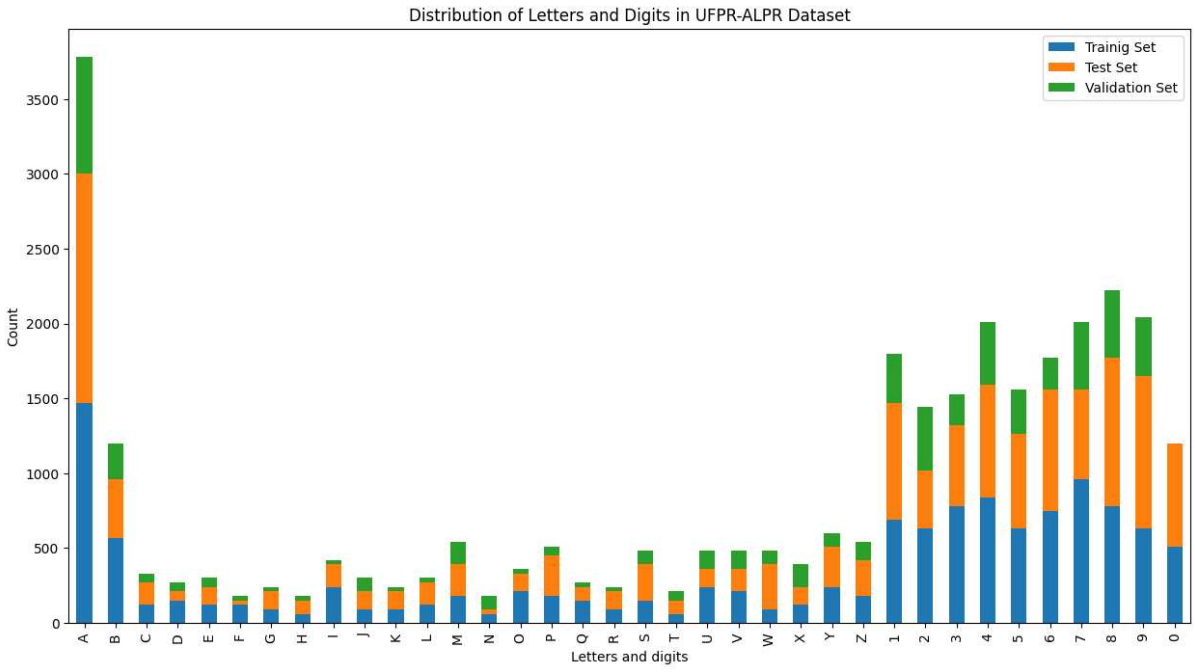

Figure 8. Letter and Digit Distributions in UFPR-ALPR

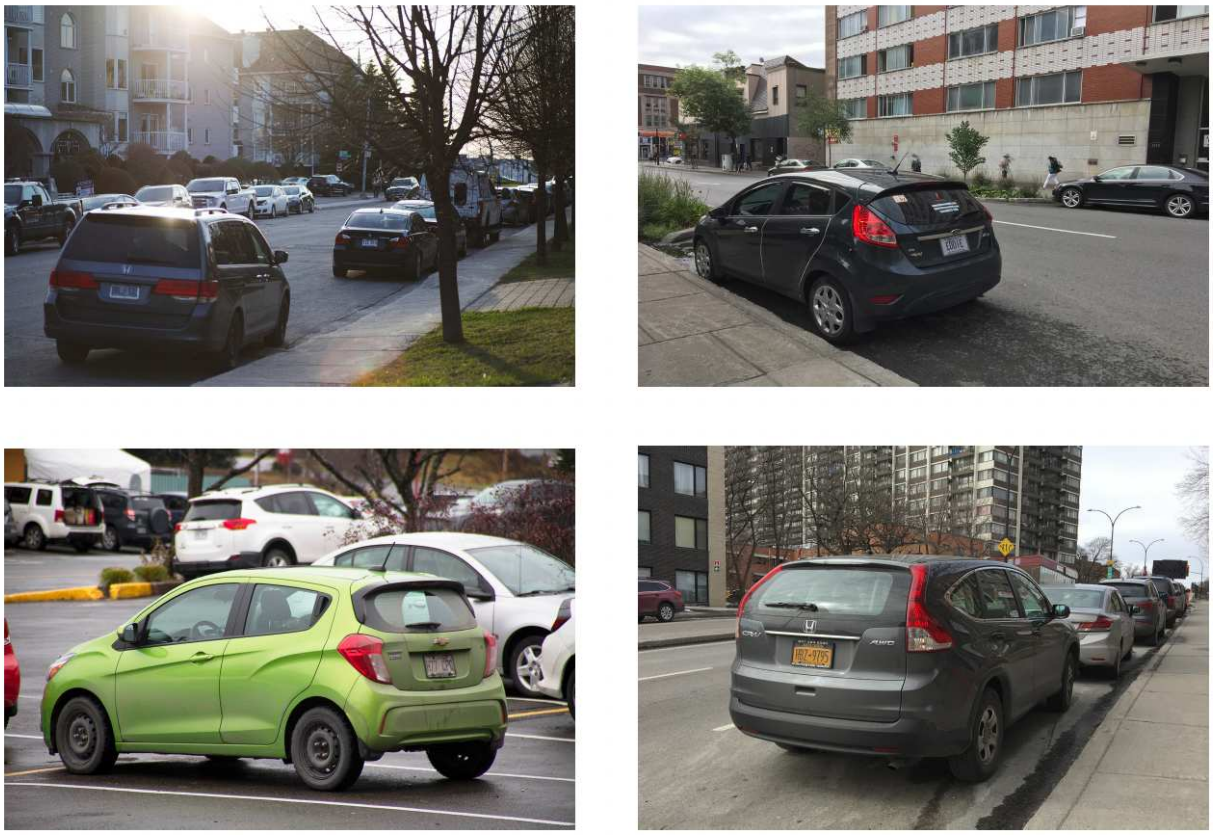

Figure 9. Example Images from CENPARMI Dataset

II. License Plate Detection and Character Recognition

We employed two solutions for license plate detection and character recognition. Our first solution utilized a Faster-RCNN[2] network to detect the license plate and separate it from the image. We used a CNN-RNN architecture with CTC[3] loss and a MobileNet V3[33] backbone to recognize the letters for the recognition task. In our second solution, we used YOLOv8 for detection and recognition tasks.

Detection Task:

We conducted two experiments for the detection task:

Experiment 1: We trained and tested the Faster-RCNN[2] model on the CENPARMI dataset.

Experiment 2: We trained and tested the YOLOv8 model on the CENPARMI dataset.

Table 2. Distribution of Letters and Digits in UFPR-ALPR Dataset

Letters/Digits	Training set	Test set	Validation set	Total
A	1470	1530	780	3780
B	570	390	240	1200
C	120	150	60	330
D	150	60	60	270
E	120	120	60	300
F	120	30	30	180
G	90	120	30	240
H	60	90	30	180
I	240	150	30	420
J	90	120	90	300
K	90	120	30	240
L	120	150	30	300
M	180	210	150	540
N	60	30	90	180
O	210	120	30	360
P	180	270	60	510
Q	150	90	30	270
R	90	120	30	240
S	150	240	90	480
T	60	90	60	210
U	240	120	120	480
V	210	150	120	480
W	90	300	90	480
X	120	120	150	390
Y	240	270	90	600
Z	180	240	120	540
1	690	780	330	1800
2	630	390	420	1440
3	780	540	420	1740
4	840	750	210	1800
5	630	630	420	1680
6	750	810	300	1860
7	960	600	210	1770
8	780	990	450	2220
9	630	1020	450	2100
0	510	690	390	1590

Recognition Task:

We conducted five experiments for the recognition task:

Experiment 1: We trained our YOLOv8 model on the CENPARMI dataset and tested it on the same dataset.

Experiment 2: We trained our YOLOv8 model on a publicly available dataset with different variations of license plates and tested it on the CENPARMI dataset. This experiment employed domain generalization technique to evaluate if our model could recognize and generalize to unseen typefaces. Our model was powerful enough to identify letters from typefaces not included in the training set.

Experiment 3: We trained our CNN-RNN architecture with CTC loss[3] on the CENPARMI dataset's train and test sets.

Experiment 4: We trained our YOLOv8 model on the train and test sets of the UFPR-ALPR[29] dataset.

Experiment 5: We trained our CNN-RNN architecture with CTC loss[3] using synthetic data as the training set and tested it on the test set. One reason for using synthetic data was that the UFPR-ALPR dataset needs more variations in license plates. It consists of limited tracks of license plates, and our model needed to see more data variations to converge correctly.

III. Faster R-CNN

Faster R-CNN is a highly influential computer vision model known for its effectiveness in object detection tasks. It represents an evolution from previous models by integrating a Region Proposal Network (RPN) that works with a detection network to quickly and efficiently propose regions of interest. This system allows Faster R-CNN to share computation across the entire image, dramatically speeding up the detection process and improving accuracy. The model operates in two main stages: first, the RPN generates region proposals[31], which are refined through the detection network, which classifies the objects and adjusts their bounding boxes; Faster R-CNN's ability to train end-to-end, its high accuracy, and the speed of processing makes it a popular choice for applications ranging from surveillance systems to autonomous vehicle navigation. Its

flexibility to handle different object sizes and robustness across varied scenarios further underscore its utility in advanced computer vision tasks.

IV. CNN+RNN Model with CTC Loss

A CNN-RNN architecture with CTC loss[3] and a MobileNet V3[33] backbone is a sophisticated framework for efficiently handling sequence prediction tasks such as speech and handwriting recognition. MobileNet V3[33] is the backbone, optimizing feature extraction with its lightweight design tailored for mobile devices, enhancing processing speed without compromising accuracy. The RNN component is adept at managing data sequences and capturing temporal dependencies necessary for continuous input streams. The CTC loss[3] function is integral to this architecture, enabling the model to align input sequences with their labels dynamically. It is essential when the timing between sequences and their corresponding labels varies. This combination makes the CNN-RNN with CTC loss[3] a robust choice for applications requiring real-time performance and high accuracy, including text recognition scenarios.

V. YOLO

YOLO (You Only Look Once) operates through a unified convolutional network that simultaneously processes multiple bounding boxes and computes the likelihood of object classes within those boxes. The process begins with YOLO examining the entire image, divided into a grid of S x S cells[30]. Within each cell, the model predicts B bounding boxes. Each box is defined by five specific elements: the coordinates (x, y) of the box's center, the box's height (h) and width (w) relative to the dimensions of the entire image, and a confidence score. This confidence score reflects the Intersection over the Union (IoU) metric, comparing the overlap between the predicted box and any actual (ground truth) box. Furthermore, each grid cell computes C conditional probabilities for object classes, indicating the likelihood of each class being present within the box.

During training, YOLO's objective is to optimize the prediction of bounding boxes around objects. It assigns responsibility to a specific bounding box predictor for each detected object.

This responsibility is determined based on which predictor currently shows the highest IoU with a ground truth box. This mechanism encourages a form of specialization among the predictors, enhancing the model's accuracy and efficiency in identifying and localizing objects within an image. Through this approach, YOLO accelerates the detection process and improves the precision of bounding box predictions.

This work uses YOLOv8 Nano, an enhanced version of the original YOLO.YOLOv8 Nano excels in fast, efficient object detection for edge devices, which is ideal for real-time applications. This compact model leverages an anchor-free detection system for improved accuracy across various object sizes and shapes, making it versatile for surveillance and augmented reality tasks. Integrated with PyTorch, it allows easy customization and deployment, balancing functionality with low computational demands.

VI. Proposed Detection Methodology using Faster R-CNN

The license plate detection task utilizes an improved Faster R-CNN[2] (Region-based Convolutional Neural Network) model. It is renowned for its robustness and exceptional detection accuracy under challenging conditions like varying lighting, occlusions, and diverse plate designs. This model is built on the ResNet-101 (Residual Network) architecture, praised for its ability to extract intricate features and hierarchical patterns in visual data. Pre-trained on ImageNet, the backbone network balances accuracy and computational efficiency while extracting high-level features from the input images. The Region Proposal Network (RPN)[9] generates region proposals likely to contain license plates, refining and filtering these proposals to reduce false positives. These proposals undergo ROI (Region of Interest) pooling and pass through fully connected layers for precise bounding box regression and classification. The strength of Faster R-CNN[2] lies in its innovative approach that combines region proposal with object classification into a cohesive framework, thereby enhancing the precision of object detection. Figure 10 illustrates a detailed schematic of the Faster R-CNN[2] architecture.

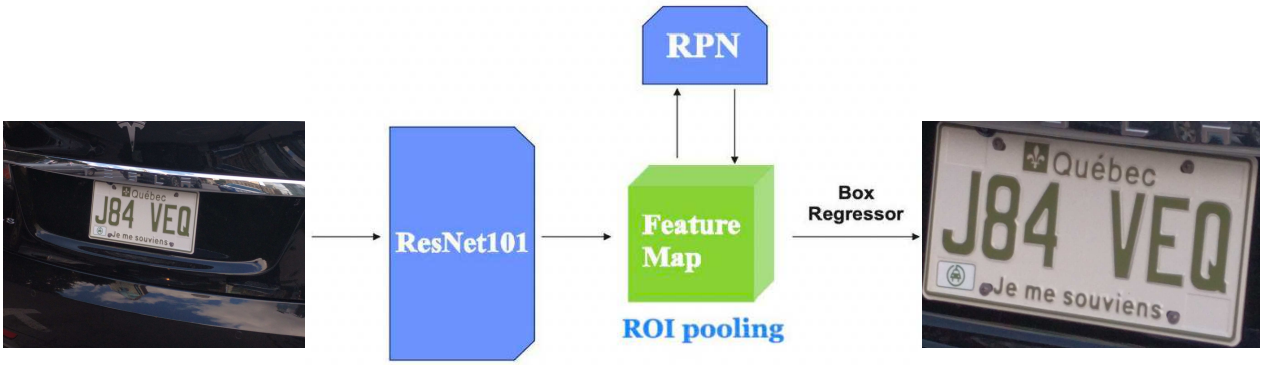

Figure 10. Faster R-CNN Architecture

VII. Proposed Recognition Methodology using CNN+RNN Model with CTC Loss

We developed a CNN-RNN model integrated with CTC loss[3] for the OCR task, optimized for sequence-based recognition tasks like reading characters on license plates. This configuration employs the MobileNetV3[33] architecture as the CNN feature extractor. It is chosen for its lightweight yet powerful design, ensuring efficient operation on edge devices while extracting spatial features from detected license plate images. The extracted features are processed by a bidirectional LSTM (Long Short-Term Memory) network, capturing contextual information across the character sequence. The CTC loss[3] function allows end-to-end training for sequence prediction without requiring pre-segmented training data, which is crucial for handling varying lengths and unaligned character sequences typical of license plates. This CNN-RNN framework enhances the precision of capturing fine details from tiny images. It ensures reliable recognition of character sequences, effectively managing the common challenge of varying sequence lengths in text recognition tasks. Figure 11 represents the architecture of this CNN-RNN model.

VIII.Proposed Detection and Recognition Methodology using YOLOv8

Our research stands out for its innovative use of domain generalization technique. We trained our YOLOv8 nano model on a training set and tested it on unseen data. What sets us apart is our use

of the DINO (DIstillation with NO labels) approach [34], a self-supervised learning technique that leverages Vision Transformers (ViT) to train models without labeling data, a novel and cutting-edge method in the field.

DINO is a powerful general-trained model that can detect many objects. It gets a prompt and detects the desired objects in the input image, promoting consistency in feature representation despite varied input conditions. DINO is effective for developing rich, meaningful feature representations that can be applied to various downstream tasks, thus reducing the reliance on extensive labeled datasets and making better use of available unlabeled data.

We used the YOLOv8 model to detect license plates and separate them from the image. Once the plates were isolated, we used the YOLOv8 model again for the character recognition task, effectively recognizing the individual characters on the plates. This two-step process of detection followed by recognition leverages the powerful feature extraction capabilities of YOLOv8 for both object detection and character recognition tasks.

In the YOLOv8 Nano model, there are a total of 22 convolution layers and six max pooling layers. The convolution layers predominantly use a 3x3 kernel with a stride of 1, which is typical for extracting complex spatial features without altering the dimensionality of the input. These layers are instrumental in detecting various features at different levels of abstraction, from bare edges and textures to more complex shapes and objects. Additionally, six layers utilize a smaller 1x1 kernel with a stride of 1, primarily used for adjusting the number of feature channels, thereby manipulating the depth of the feature maps without affecting their spatial dimensions.

The YOLOv8 Nano model is further enhanced by its six max-pooling layers, each using a 2x2 kernel with a stride of 2. These layers play a crucial role in reducing the spatial dimensions of the feature maps by half[32], thereby significantly lowering the computation required for subsequent layers. This strategic use of max pooling not only helps manage the computational load but also maintains effective feature extraction capabilities, a key feature of the YOLOv8 Nano model designed for real-time object detection in resource-constrained environments.

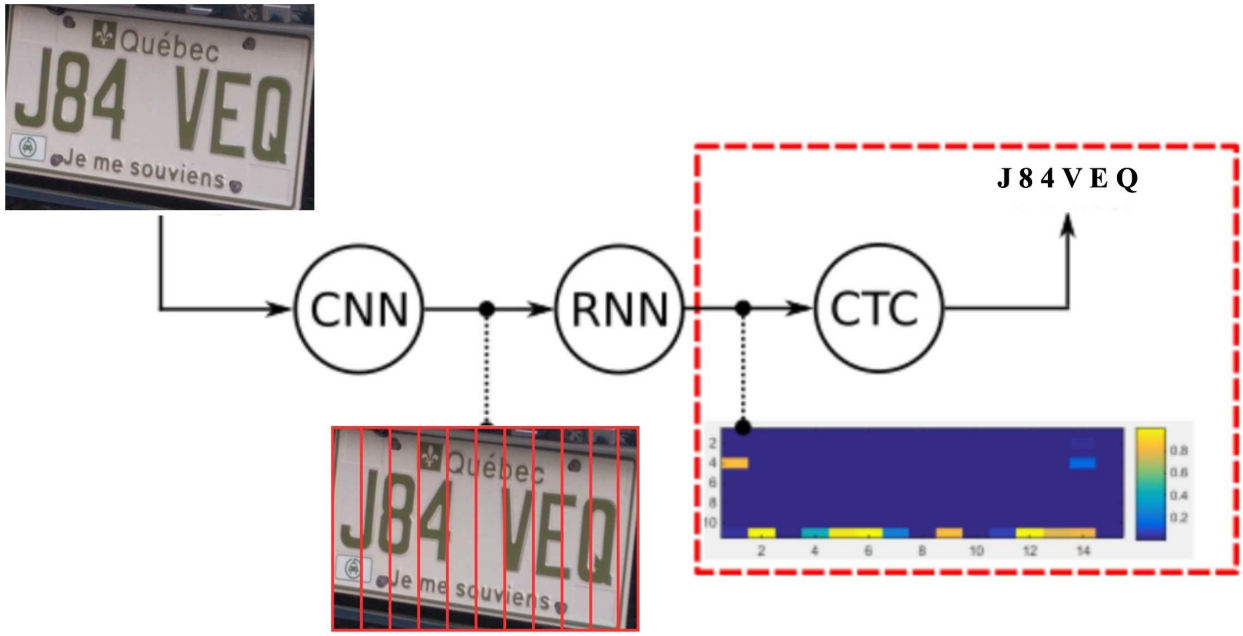

Figure 11. CNN+RNN Network Architecture

IX. Model Training and Optimization and Evaluation

The detection and recognition models underwent 100 iterations or epochs during the training phase to ensure optimal convergence and proficiency. The learning rate was set to 0.01, determining the step size at which the stochastic gradient descent optimizer adjusted the model's parameters during the back-propagation process. This choice of learning rate was informed by empirical observations and experimentation, balancing the need for stable convergence and fine-grained parameter updates. We utilized a batch size of 64. By carefully selecting this batch size, we aimed to balance computational efficiency and the model's ability to generalize to varying regions of interest within license plate images.

We evaluate our model using three key metrics: mean average precision, recall, and character error rate. Each offers a unique perspective on performance and accuracy.

Mean average precision (MAP) is employed to assess our model's accuracy in identifying and retrieving relevant items from a dataset. This metric averages the precision achieved at various levels of recall, providing a holistic view of the model's performance across different retrieval thresholds. Precision measures the proportion of relevant retrieved items, ensuring our model's effectiveness in delivering accurate results.

Recall determines the model's ability to capture all relevant instances within the dataset. This metric is critical when missing a relevant instance, as it can lead to significant consequences. By measuring recall, we ensure our model effectively identifies all potential positive cases, minimizing the risk of oversight.

Character error rate (CER) evaluates the model's precision at the character level. It is beneficial in text processing tasks, like optical character or handwriting recognition. This metric quantifies the percentage of incorrectly predicted characters, offering a granular insight into the model's accuracy in text interpretation.

Together, these metrics provide a comprehensive assessment of our model's capabilities, from overall accuracy in data retrieval (MAP) to thoroughness in identifying relevant items (recall) to precision in text-based tasks (CER). This multifaceted evaluation ensures that our model performs robustly across various dimensions of data processing.

X. Data Augmentation and Synthetic Data Generation

To enhance the recognition model's robustness, various data augmentation techniques were employed, including rotation, perspective transform, and color channel transform. Additionally, synthetic data was generated by combining plate backgrounds with cropped character images. These techniques significantly increased the dataset's variability, allowing the model to learn from various examples. The data augmentation and synthetic data generation contributed to improved model performance, particularly in handling challenging conditions such as occlusions, different lighting, and varied plate designs.

We used synthetic data in some experiments to investigate its impact on model performance. Applying these augmentation techniques exposed the model to a diverse set of scenarios during training, which is crucial for developing a resilient and generalizable recognition system. This strategic augmentation ensured that the model could effectively manage real-world variations and inconsistencies in license plate appearances, thus enhancing its overall accuracy and reliability in practical applications.

XI. Font Type Evaluation Aspects

Regarding ALPR systems, it's essential to recognize that even the most advanced segmentation algorithms might only sometimes result in clearly recognizable characters from the segmented portions. This issue highlights the necessity for each character to preserve its distinct integrity, allowing it to remain identifiable despite potential shortcomings in the cropping process. Consequently, selecting an appropriate font is crucial for successfully recognizing license plates, including those with general plate numbers that do not conform to regular expressions.

In light of these challenges, this work introduces several vital contributions:

1. This research investigates the characteristics of five distinct fonts: Driver Gothic, Dreadnought, California Clarendon, Zurich Extra Condensed, and Mandatory. The analysis of the characteristics will be comprehensive, with specific factors detailed in Table 3 of the study[35]. This methodical examination delineates the subtle differences and similarities that affect how easily each character can be distinguished from others, particularly in scenarios where clarity is paramount.
2. This study aims to assess the impact of selected fonts within specific contexts by analyzing five datasets: UFPR-ALPR[5], Ontario, Quebec, California, and New York State. The evaluation will utilize a commercial Automatic License Plate Recognition product: OpenALPR[4].
3. The dual result analysis in this study will encompass two key aspects. Firstly, the recall values will be determined, focusing on the accuracy of license plate recognition by treating the predicted license plate as either wholly correct or incorrect. Secondly, the analysis will delve into more detailed qualitative aspects by using confusion matrices for each character on each license plate across the datasets. These confusion matrices are crucial as they provide a detailed view of the recognizability of each character, highlighting specific areas where characters are most frequently misidentified or confused.
4. Recommendations for Optimal Font Characteristics: Drawing from the insights gained through our analysis, we intend to develop a series of recommendations that outline

optimal font characteristics designed to boost recognition accuracy in real-time applications. These recommendations will be grounded in the data derived from evaluating various fonts across different datasets and recognition systems. By pinpointing the specific attributes that contribute to adequate character distinction and readability, we aim to provide actionable guidelines that can be implemented to improve the performance of license plate recognition systems under diverse operational conditions.

Table 3. Font Type Evaluation Aspects[35]

Aspect	Explanation
Similar apex	Apex is defined as the juncture of a stem. If two glyphs have the apex at the same position and design, the probability of confusion will increase. For example, the top apex in letter A and digit 4.
Similar crossbar position	Crossbar is defined as the horizontal stroke connecting two parts of a letter form (e.g. H). if two glyphs have crossbars, then locating the crossbars at different positions would improve their recognizability.
Similar top counter	Counter is defined as the negative space fully or partially enclosed by the stroke of a letter form. If two glyphs have top counters that look identical, probability of confusion will increase.
Similar bottom counter	If two glyphs have bottom counters that look identical, probability of confusion will increase.
Similar bowl	Bowl is defined as the fully or modified rounded forms. If two glyphs have the bowl at the same position and design, the probability of confusion will increase
Identical spur	Spur is defined as a projection smaller than a serif, that reinforces the point at the end of curved stroke. If two glyphs have the spur at the same position and design, the probability of confusion will increase.
Identical bottom, or top horizontal stroke	Stroke is defined as any linear element that makes up a letterform. If two glyphs have the stroke at the same position and design, the probability of confusion will increase.
Similar diagonal stroke	If two glyphs have the diagonal stroke at the same position and design, the probability of confusion will increase (e.g., 5 and S).
Tail not clear	Tail is defined as a short downward stroke. If the tail is not clear or visible in context, it is more likely to be confused with other glyphs.

Chapter 4. LPD Results and Analysis

In this chapter, we will present the experimental results of our study. Additionally, we provide a detailed experimental analysis of our License Plate Recognition (LPR) system. This includes comparing our system's performance comprehensively against leading methods in the field and established commercial systems. The analysis aims to highlight the strengths and potential areas for improvement in our LPR system relative to current industry standards.

For the training process of detection and recognition models, we leveraged the power of the NVIDIA Tesla T4 GPU. The NVIDIA Tesla T4 has 16GB of GPU memory, offering substantial computational resources and parallel processing capabilities. This GPU enabled us to accelerate the training and optimization of our models, effectively handling large-scale datasets.

I. Evaluation Metrics

The performance of our models on the test set was evaluated using various metrics. For the object detection model, we used average precision (AP) at different IOU thresholds, areas, and maximum detections to gain insights into its capabilities. We employed recognition rate, character error rate (CER), and recall ratio as evaluation metrics for the recognition model. Additionally, confusion matrices were computed to analyze the types of errors made by the recognition model.

II. Detection Results of Faster-RCNN Model

The detection model maintained high precision at an IoU threshold of 0.75. The model's performance varied based on object size—the average precision needed to be computed for small objects, indicating a lack of reliable detection. However, for medium-sized objects, the model achieved a reliable average precision. The model excelled in detecting large objects, earning an impressive average precision. These results demonstrate the effectiveness of our Faster-RCNN[2] model in accurately detecting license plates in various scenarios, highlighting its potential for real-world license plate recognition applications. The results of the detection models for four datasets are shown in Table 4.

Dataset	AP50:95	AP75
Quebec	79.54%	94.16%
Ontario	80.37%	94.95%
New York	78%	92.14%
California	83.87%	95.72%

Table 4. AP50:95, and AP75 scores for various datasets

We found that California's detection results were the highest, while New York's were the lowest. License plates with a white background and dark letters are more detectable than those without a white background due to the higher contrast of colors. The higher contrast in color between the background and the letters significantly enhances the model's ability to identify and recognize the license plates, leading to better performance in detection tasks. This finding underscores the importance of color contrast in the effectiveness of license plate recognition systems. Figure 12 shows some samples of the detection results by the Faster-RCNN model.

Figure 12. Detection Sample Results of Faster-RCNN Model

III. Recognition Results of CNN+RNN Model with CTC Loss

The recognition model was evaluated on the UFPR-ALPR[29] dataset and our CENPARMI dataset. The results indicate a significant performance improvement when using the CENPARMI dataset, attributed to the extensive data augmentation and synthetic data generation techniques employed during training. The results are summarized in the Table 5.

The higher recognition rate and lower Character Error Rate (CER) on the CENPARMI dataset suggest that the model generalizes well to the varied conditions and plate designs encountered in real-world scenarios. This highlights the effectiveness of our advanced training techniques. This improvement is particularly noteworthy given the diversity of the CENPARMI dataset, which includes images from different states with varying plate designs and fonts, lighting conditions, and environmental factors.

One of our approach's key innovations is using a lightweight backbone inspired by MobileNetV3[33] for the recognition model. This design choice ensures the model can run efficiently on edge devices with limited computational resources. The lightweight architecture reduces the model's complexity and computational cost without compromising accuracy, making it suitable for real-time applications on devices such as smartphones, tablets, and embedded systems.

Model	Dataset	Recognition Rate	Character Error Rate(CER)	Recall
Our Model	UFPR-ALPR	84.2%	5.3%	92.1%
Our Model	CENPARMI	89.8%	4.5%	90.0%
OpenALPR	UFPR-ALPR	71.7%	15.5%	55.8%
OpenALPR	CENPARMI	65.5%	19.6%	80.2%

Table 5. Performance Metrics for Different Models and Datasets

IV. State-wise Recognition Performance of CNN+RNN Model with CTC Loss

The recognition model's performance was further analyzed using four different datasets separately from Canada and the United States. The results for each dataset are detailed in the Table 6.

The model demonstrated consistent performance across the different datasets, with slight variations attributed to differences in plate designs, font styles, and environmental conditions. California achieved the highest accuracy and the lowest Character Error Rate (CER), likely due to more precise and consistent plate designs. New York also performed exceptionally well, indicating that the model can handle diverse plate designs accurately.

Province/State	Recognition Rate	Character Error Rate(CER)	Recall
Quebec	91.6%	3.8%	91.5%
Ontario	91.6%	3.8%	91.5%
New York	80.1%	8.1%	80.6%
California	96%	2.3%	96.4%

Table 6. Performance Metrics for Different Provinces/States.

V. Detection Results of YOLOv8 Model

Figures 13 compare the F1-Confidence curves for the YOLOv8 model trained on the CENPARMI and UFPR-ALPR[29] datasets, highlighting notable differences and similarities in performance. For the CENPARMI dataset, the peak F1 score is 0.98 at a confidence threshold of 0.469, indicating optimal performance at a higher confidence level. The F1 score rises sharply and stabilizes at nearly 1.0, showing robustness and reliability across a broad range of confidence levels. This higher threshold suggests the model is more conservative and beneficial in scenarios where high precision is crucial.

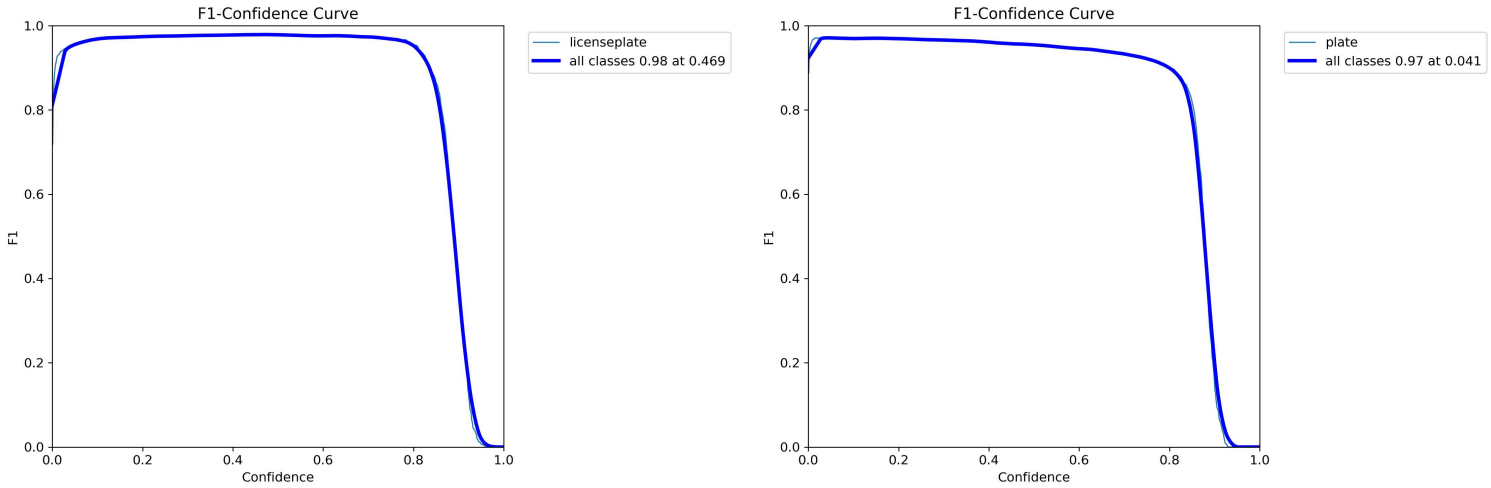

Figure 13. F1-Confidence Curve for the YOLOv8 Model Trained on the CENPARMI Dataset on the Left and UFPR-ALPR Dataset on the right

In contrast, the UFPR-ALPR[29] dataset shows a peak F1 score of 0.97 at a much lower confidence threshold of 0.041. The F1 score increases quickly and plateaus at nearly 1.0, but the optimal confidence threshold is narrower than for the CENPARMI dataset. This lower threshold implies the model is more liberal in making predictions, which is advantageous when recall is more critical.

Both models exhibit consistent detection performance across different license plate types, a testament to their robustness in handling diverse data. In summary, the model trained on the CENPARMI dataset achieves slightly higher peak performance at a higher confidence threshold. In contrast, the model trained on UFPR-ALPR[29] performs well at a lower threshold. As these results clearly demonstrate, the dataset's choice significantly influences the model's optimal confidence threshold, affecting its conservativeness or liberality in predictions.

The precision-confidence curves, the main focus of this evaluation, are compared in Figure 14 for the YOLOv8 model trained on the CENPARMI and UFPR-ALPR[29] datasets. The precision-confidence curve for the CENPARMI dataset, a standout in this analysis, exhibits excellent performance, with precision reaching 1.00 at a confidence threshold of 0.936. This high threshold indicates that the model maintains perfect precision across a broad range of higher confidence levels. The initial sharp rise in precision, followed by stabilization at 1.0, demonstrates high accuracy and very few false positives. The curve's shape suggests a robust model performing consistently across varying confidence levels.

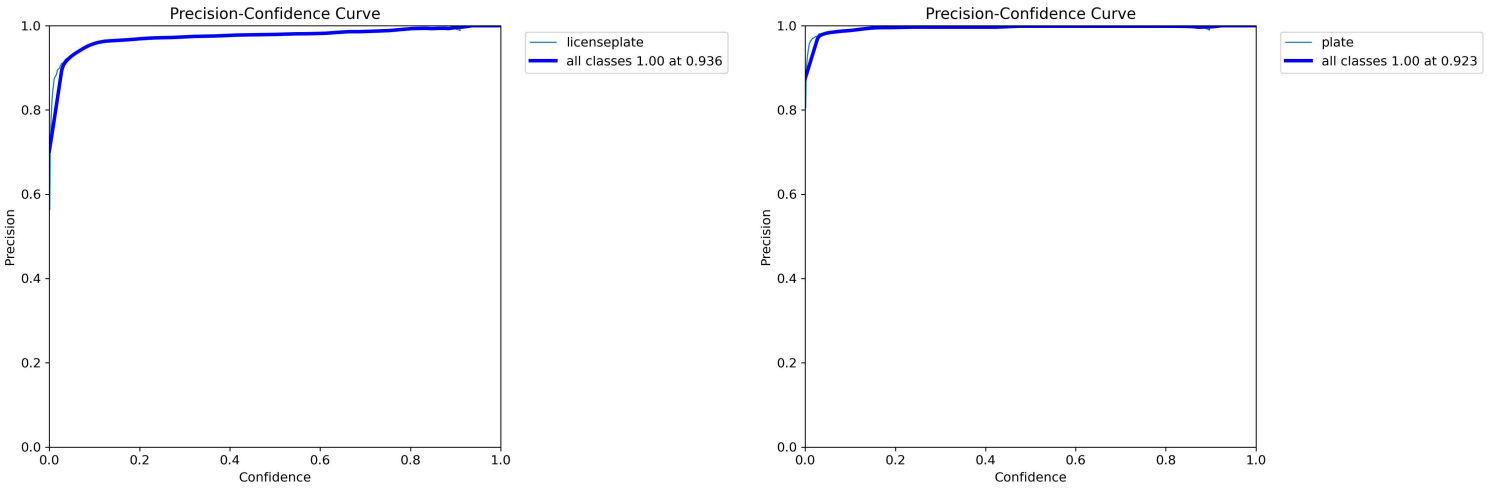

Figure 14. Precision-Confidence Curve for the YOLOv8 Model Trained on the CENPARMI Dataset on the Left and UFPR-ALPR Dataset on the right

Similarly, the precision-confidence curve for the UFPR-ALPR[29] dataset shows high performance, with precision also reaching 1.00 but at a slightly lower confidence threshold of 0.923. This indicates robust performance and high accuracy at a lower threshold than the CENPARMI dataset. The steep rise and stabilization at 1.0 further underline the model's effectiveness in minimizing false positives.

The key difference between the two datasets lies in the peak precision and corresponding confidence thresholds. The CENPARMI dataset achieves perfect precision at a higher threshold of 0.936, indicating a more conservative model. In contrast, the UFPR-ALPR[29] dataset achieves perfect precision at 0.923, suggesting a slightly more lenient model. Both models demonstrate a sharp increase in precision followed by stabilization at 1.0, with the UFPR-ALPR[29] dataset achieving high accuracy at lower thresholds.

In conclusion, the precision-confidence curves for the YOLOv8 model trained on both datasets demonstrate excellent performance, with both achieving perfect precision. The CENPARMI model reaches peak precision at a higher confidence threshold of 0.936, while the UFPR-ALPR[29] model does so at 0.923. Both models are highly accurate and robust, with minimal false positives, although the UFPR-ALPR[29] model is slightly more lenient in its predictions. This reaffirms the YOLOv8 model's overall excellent performance and its ability to adapt to different datasets.

Figure 15 compares the precision-recall curves for the YOLOv8 model trained on the CENPARMI and UFPR-ALPR[29] datasets. The precision-recall curve for the CENPARMI dataset shows excellent performance, with a mean average precision (mAP) of 0.989 at a recall

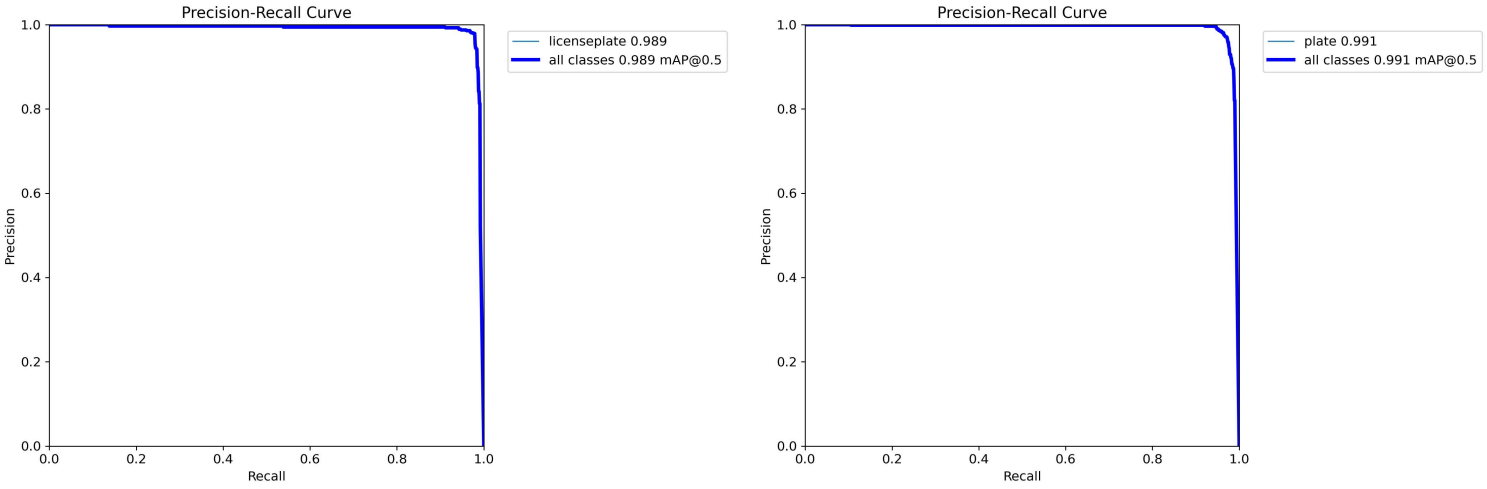

Figure 15. Precision-Recall Curve for the YOLOv8 Model Trained on the CENPARMI Dataset on the Left and UFPR-ALPR Dataset on the right

value of 0.5.The curve maintains high precision across a broad range of recall values, staying near 1.0 until recall approaches 1.0, where a slight drop-off occurs. This indicates that the model is highly effective in detecting license plates, achieving a balance between high precision and high recall. The stability of the curve across various recall levels suggests robust performance, with the model effectively minimizing false positives while maintaining high detection rates.

The YOLOv8 model applied to the UFPR-ALPR[29] dataset also demonstrates impressive performance, with a slightly higher mAP of 0.991 at a recall value of 0.5. Similar to its performance on the CENPARMI dataset, the model maintains precision close to 1.0 across most recall values, only dropping off slightly as recall approaches 1.0. This consistent performance across different datasets underscores the model's adaptability and versatility, providing reassurance about its ability to maintain high accuracy and minimize false positives.

When comparing the two datasets, the key differences are marginal. The model trained on the CENPARMI dataset achieves an mAP of 0.989, while the model trained on the UFPR-ALPR[29] dataset achieves a slightly higher mAP of 0.991. Both models maintain high precision across a wide range of recall values, indicating strong performance in both datasets. The slight drop in precision at higher recall values is minimal in both cases, demonstrating the models' robustness. The subtle edge in mAP for the UFPR-ALPR[29] dataset suggests marginally better overall performance in minimizing false positives while maintaining high recall.

Figure 16 compares the recall-confidence curves for the YOLOv8 model trained on the CENPARMI and UFPR-ALPR[29] datasets. The recall-confidence curve for the CENPARMI dataset demonstrates high performance, achieving a recall of 0.99 at a confidence threshold of 0.000. The curve maintains a high recall value near 1.0 across a broad range of confidence levels, indicating the model's effectiveness in capturing true positives with varying confidence levels. As confidence increases, recall remains high until it declines sharply around a threshold of 0.8, suggesting the model becomes more conservative, reducing recall as confidence increases.

Similarly, the recall-confidence curve for the YOLOv8 model trained on the UFPR-ALPR[29] dataset shows strong performance, with recall reaching 0.98 at a confidence threshold 0.000. The curve maintains high recall close to 1.0 across most confidence levels, similar to the CENPARMI dataset. The recall declines significantly at a confidence threshold of around 0.8, indicating the model also becomes more conservative as confidence increases.

Comparing the two datasets, the CENPARMI model achieves a slightly higher recall at low confidence thresholds (0.99) compared to the UFPR-ALPR[29] model (0.98). Both models maintain high recall values across a broad range of confidence levels and exhibit a sharp decline around a confidence threshold of 0.8, indicating increasing conservativeness. These results demonstrate that both models effectively detect license plates with high recall, although the CENPARMI model has a slight edge at lower confidence levels.

Figure 17 shows various performance metrics and loss values for the YOLOv8 model trained on the CENPARMI dataset. The train/box loss decreases steadily from around 1.0 to approximately 0.7, indicating improved bounding box predictions. The train/classification loss drops sharply from 2.0 to about 0.5, reflecting better class prediction accuracy. Similarly, the train/DFL loss falls from 1.1 to 0.9, showing enhanced distribution predictions. Precision during training stabilizes close to 1.0, demonstrating high accuracy with few false positives. Recall increases from about 0.4 to nearly 1.0, indicating the model captures most true positives with minimal misses. Validation metrics show parallel trends: box loss decreases from 1.2 to 0.8, classification loss from 3.0 to 0.5, and DFL loss from 1.4 to 1.0, highlighting consistent improvement on the validation set. The mAP at IoU 0.5 rises and stabilizes near 1.0, reflecting excellent precision and recall balance, while the mAP across IoU 0.5 to 0.95 steadily improves to around 0.8, indicating

robust performance across varying thresholds. Overall, the model demonstrates robust, consistent accuracy and detection quality enhancements throughout training and validation.

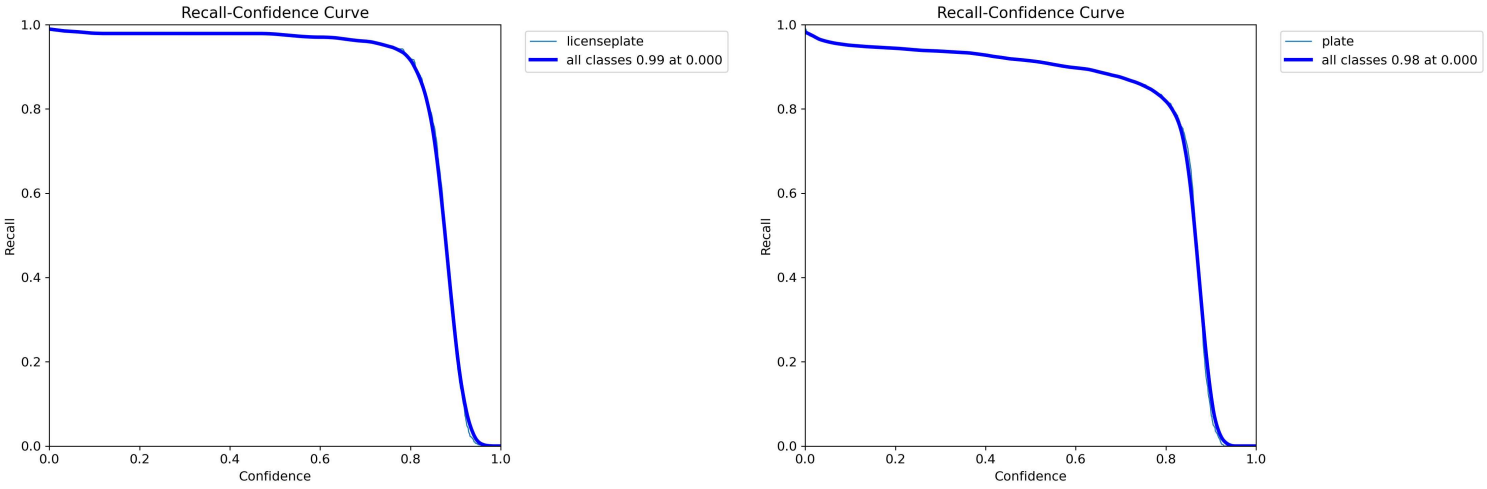

Figure 16. Recall-Confidence Curve for the YOLOv8 Model Trained on the CENPARMI Dataset on the Left and UFPR-ALPR Dataset on the right

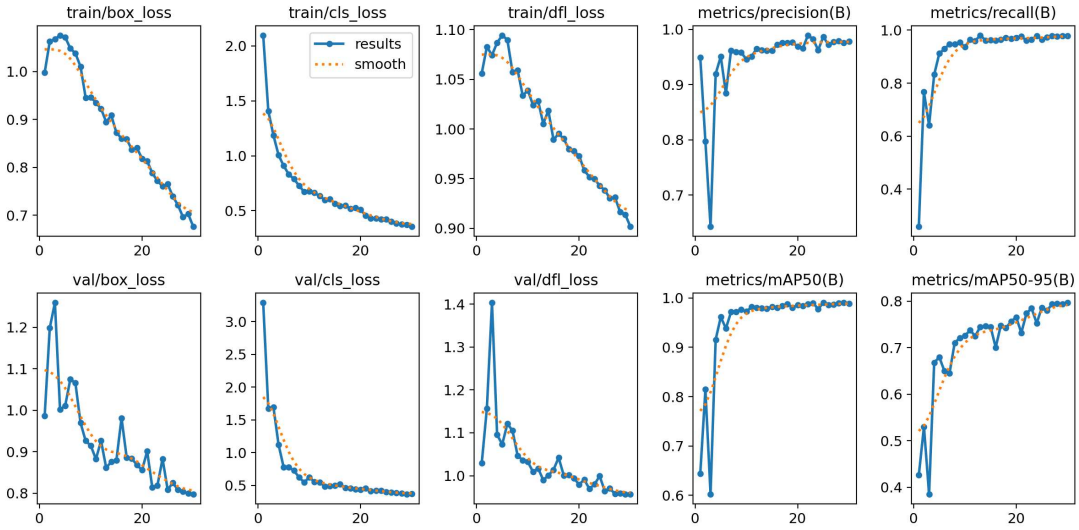

Figure 17. Various Performance Metrics and Loss Values for the YOLOv8 Model Trained on the CENPARMI Dataset

Figure 18 shows various performance metrics and loss values for the YOLOv8 model trained on the UFPR-ALPR[29] dataset. The train/box loss decreases steadily from around 1.6 to approximately 0.6, indicating improved bounding box predictions. The train/classification loss drops sharply from 3.0 to about 0.5, reflecting better class prediction accuracy. Similarly, the

train/DFL loss falls from 0.95 to 0.8, showing enhanced distribution predictions. Precision during training stabilizes around 0.96, demonstrating high accuracy with few false positives. Recall increases sharply from about 0.4 to nearly 1.0, indicating the model captures most true positives with minimal misses. Validation metrics show parallel trends: box loss decreases from 1.5 to 1.25, classification loss from 3.0 to 1.0, and DFL loss from 0.88 to 0.85, highlighting consistent improvement on the validation set. The mAP at IoU 0.5 rises and stabilizes near 0.98, reflecting excellent precision and recall balance, while the mAP across IoU 0.5 to 0.95 steadily improves to around 0.65, indicating robust performance across varying thresholds. Overall, the model demonstrates robust, consistent accuracy and detection quality enhancements throughout training and validation on the UFPR-ALPR[29] dataset.

Figures 19 and 20 show YOLOv8's detection sample results for the CENPARMI and UFPR-ALPR[29] datasets, respectively.

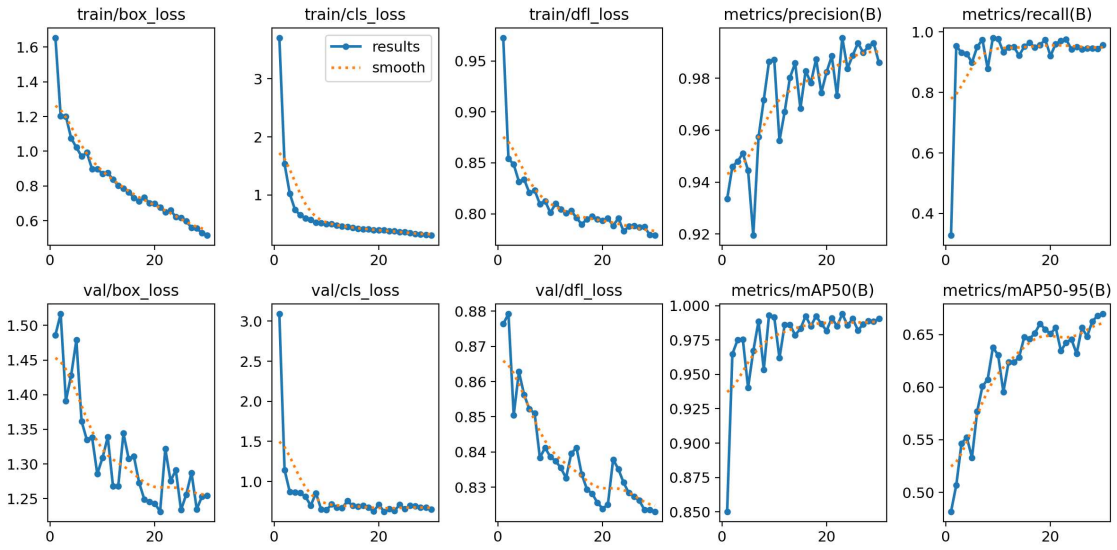

Figure 18. Various Performance Metrics and Loss Values for the YOLOv8 Model Trained on the UFPR-ALPR Dataset

Figure 19. Detection Sample Results of YOLOv8 Model on the CENPARMI Dataset

Figure 20. Detection Sample Results of YOLOv8 Model on the UFPR-ALPR Dataset

VI. Recognition Results of YOLOv8 Model

Figure 21 compares the F1-confidence curves for the character recognition task in license plates using the YOLOv8 model trained on the UFPR-ALPR[29] and CENPARMI datasets. For the UFPR-ALPR[29] dataset, the model achieves a peak F1 score of 0.94 at a confidence threshold of 0.412. The F1 score increases sharply up to this point, then stabilizes near 0.94 before gradually

declining as the confidence threshold approaches 1.0. This indicates optimal performance at a lower confidence level, achieving a good balance between precision and recall.

In contrast, the CENPARMI dataset's F1-confidence curve shows a peak F1 score of 0.93 at a higher confidence threshold of 0.579. The F1 score rises sharply and stabilizes near 0.93 before declining as the confidence threshold increases towards 1.0. This suggests the model requires a higher confidence level to achieve optimal performance, effectively balancing precision and recall. The key differences between the two datasets lie in the optimal confidence thresholds and peak F1 scores. The UFPR-ALPR[29] model achieves a slightly higher peak F1 score at a lower confidence threshold, indicating better performance with more liberal predictions. Both models show robust performance, maintaining high accuracy across various confidence levels. The

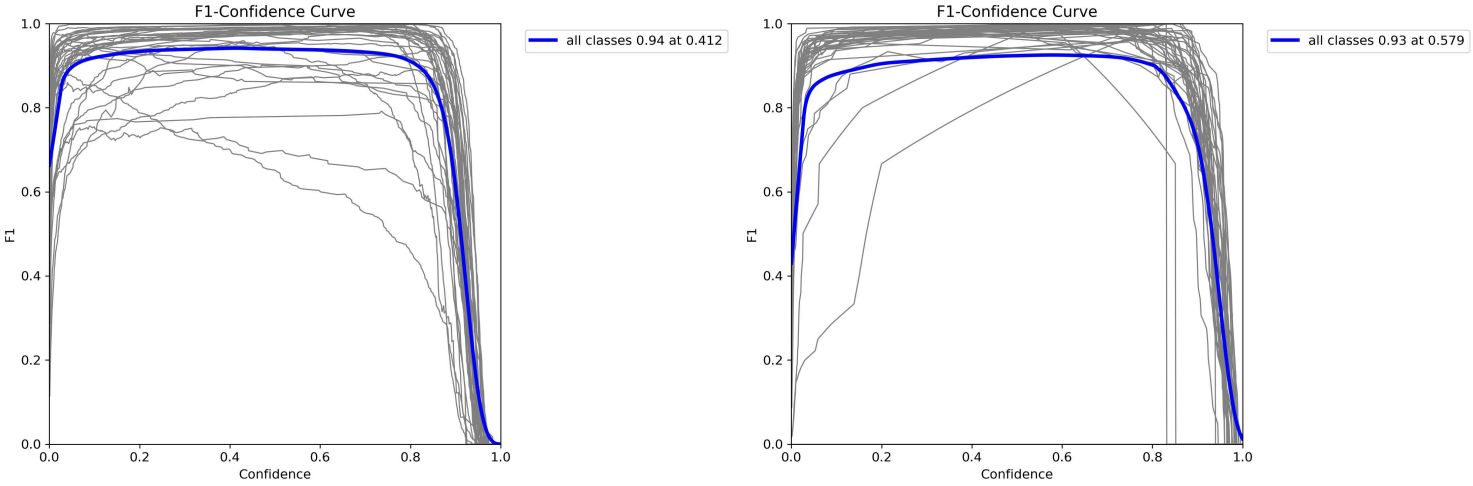

Figure 21. F1-Confidence Curve for the YOLOv8 Model Trained on the CENPARMI Dataset on the Right and UFPR-ALPR Dataset on the Left

UFPR-ALPR[29] model is more effective at lower confidence thresholds, while the CENPARMI model performs better at higher thresholds.

Figure 22 presents a robust comparison of the precision-confidence curves for the character recognition task in license plates using the YOLOv8 model trained on the UFPR-ALPR[29] and CENPARMI datasets. The precision-confidence curve for the UFPR-ALPR[29] dataset exhibits a sharp rise in precision, stabilizing near 1.0 as the confidence threshold approaches 1.0. This indicates that the model achieves perfect precision at high confidence levels with minimal false positives, a testament to its robust performance.

In contrast, the precision-confidence curve for the CENPARMI dataset exhibits a peak precision of 0.95 at a confidence threshold of 0.966. Like the UFPR-ALPR[29] dataset, the precision rises sharply and stabilizes near the peak value as the confidence threshold increases. However, the peak precision is slightly lower, indicating that while the model performs well, it does not consistently reach perfect precision.

The critical difference between the two datasets lies in their peak precision and corresponding confidence thresholds. The UFPR-ALPR[29] model achieves perfect precision at a slightly higher confidence threshold, indicating superior accuracy with high confidence. Both models maintain high precision across a wide range of confidence levels, but the UFPR-ALPR[29] model is more effective at achieving perfect precision.

It's important to note that both models demonstrate strong performance, with the UFPR-ALPR[29] dataset achieving a peak precision of 1.00 at a confidence threshold of 0.980, and the CENPARMI dataset reaching a peak precision of 0.95 at a confidence threshold of 0.966. This indicates that both models maintain high precision across varying confidence levels, with the UFPR-ALPR[29] model showing slightly better performance in achieving perfect precision.

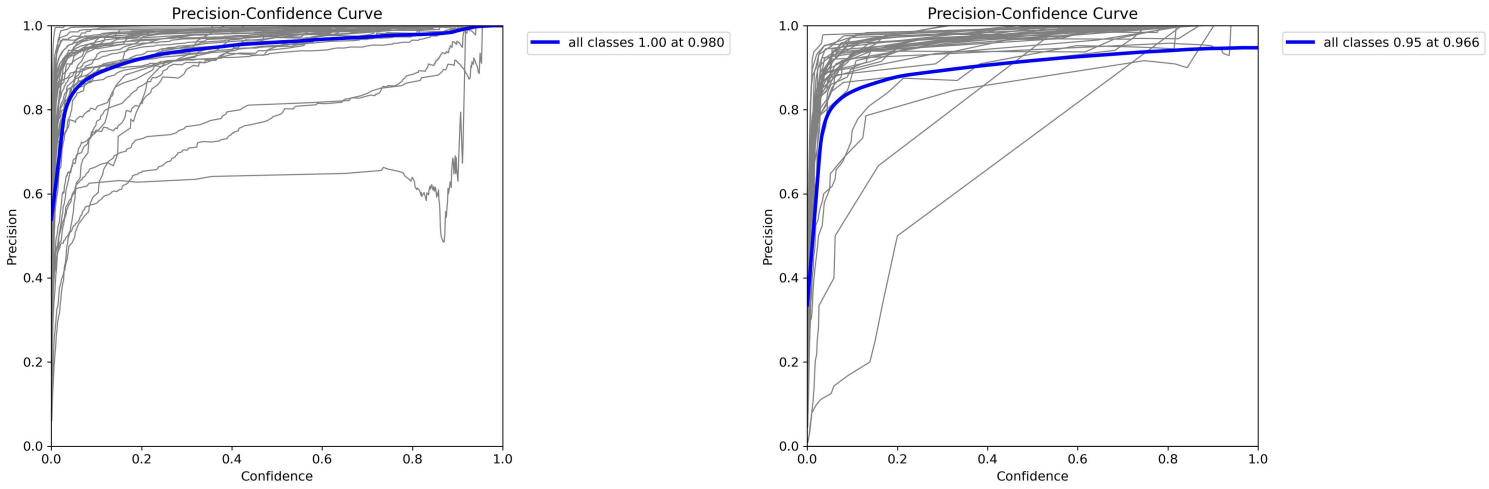

Figure 22. Precision-Confidence Curve for the YOLOv8 Model Trained on the CENPARMI Dataset on the Right and UFPR-ALPR Dataset on the Left

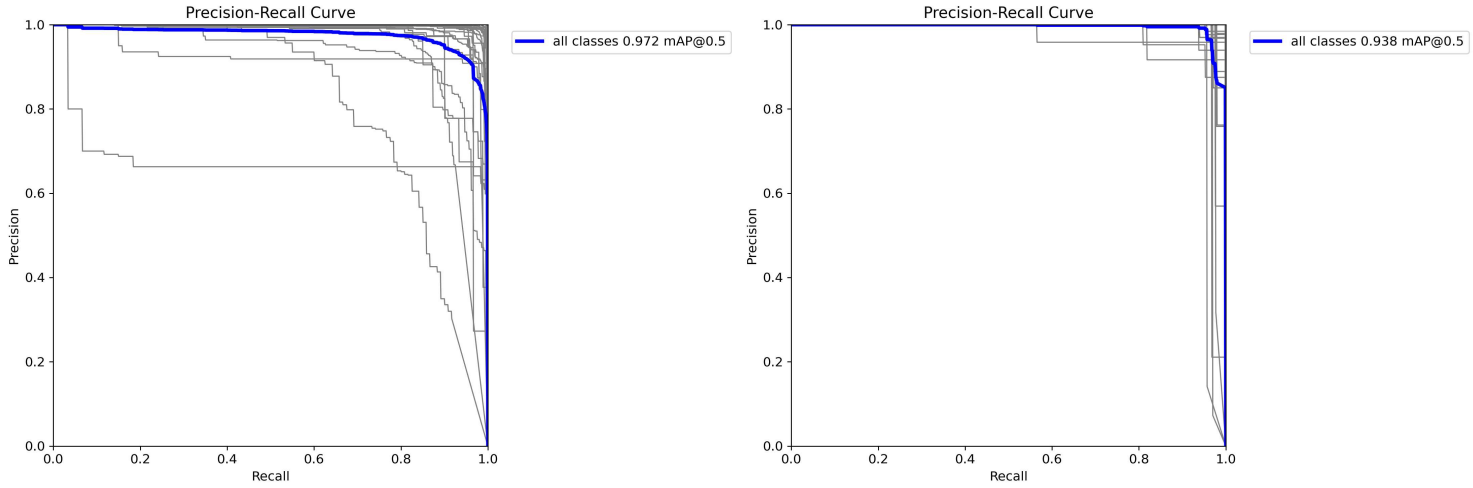

Figure 23. Precision-Recall Curve for the YOLOv8 Model Trained on the CENPARMI Dataset on the Right and UFPR-ALPR Dataset on the Left

Figure 23 compares the precision-recall curves for the character recognition task in license plates using the YOLOv8 model trained on the UFPR-ALPR[29] and CENPARMI datasets. For the UFPR-ALPR[29] dataset, the precision-recall curve shows a high mean average precision (mAP) of 0.972 at an IoU threshold of 0.5. The curve maintains high precision near 1.0 across a broad range of recall values, only dropping slightly as recall approaches 1.0. This indicates that the model performs exceptionally well, balancing high precision and recall.

In contrast, the precision-recall curve for the CENPARMI dataset exhibits a slightly lower mAP of 0.938 at an IoU threshold of 0.5. The curve maintains a flat precision of 1.0 until recall reaches approximately 0.9, followed by a sharp drop in precision. This suggests that while the model performs well with high precision, it maintains a different level of performance across the entire recall range than the UFPR-ALPR[29] model.

The critical difference between the two datasets lies in the mAP and the behavior of the precision-recall curve. The UFPR-ALPR[29] model achieves a higher mAP, indicating better overall performance in balancing precision and recall. The CENPARMI model, while still performing well, shows a more significant drop in precision as recall increases.

In summary, the precision-recall curves clearly show the YOLOv8 model's performance. The model trained on the UFPR-ALPR[29] dataset achieves a higher mAP of 0.972, maintaining high precision across a broad range of recall values. In contrast, the CENPARMI model, with an mAP of 0.938, performs well but shows a sharper decline in precision as recall increases, indicating a slightly less robust performance than the UFPR-ALPR[29] model.

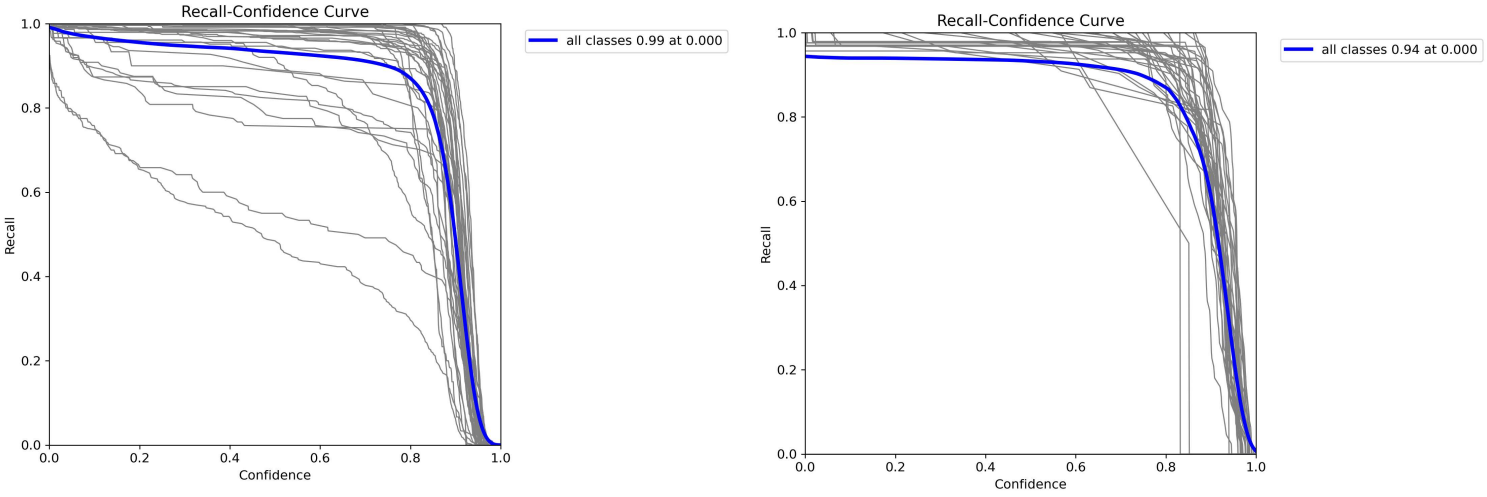

Figure 24. Recall-Confidence Curve for the YOLOv8 Model Trained on the CENPARMI Dataset on the Right and UFPR-ALPR Dataset on the Left

Figure 24 shows the recall-confidence curves for the character recognition task in license plates using the YOLOv8 model trained on the UFPR-ALPR[29] and CENPARMI datasets indicate high performance. In the UFPR-ALPR[29] dataset, the model achieves a peak recall of 0.99 at a confidence threshold of 0.000. The curve maintains high recall near 1.0 across a wide range of confidence levels, suggesting that the model effectively captures most true positives with varying confidence thresholds. Recall remains high until it declines sharply around a threshold of 0.8, indicating the model's conservativeness increases with higher confidence levels.

In contrast, the CENPARMI dataset's recall-confidence curve shows a peak recall of 0.94 at a confidence threshold of 0.000. The curve similarly maintains high recall near 1.0 across most confidence levels but starts to decline at a slightly higher confidence threshold around 0.8,

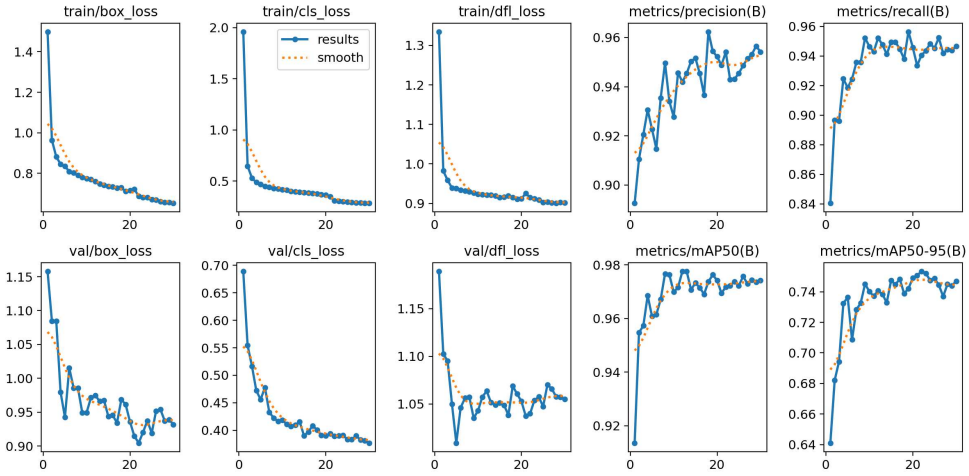

Figure 25. Various Performance Metrics and Loss Values for the YOLOv8 Model Trained on the UFPR-ALPR Dataset

reflecting a similar trend of increasing conservativeness with higher confidence levels, thus demonstrating the models' equal reliability.

Comparing the two datasets, the UFPR-ALPR[29] model achieves a higher peak recall of 0.99 at a low confidence threshold, initially capturing more true positives than the CENPARMI model, which peaks at a recall of 0.94. Both models show a sharp decline in recall around a confidence threshold of 0.8.

In summary, the UFPR-ALPR[29] model demonstrates slightly better recall performance than the CENPARMI model, with higher initial recall and robust performance across a broad range of confidence levels. Both models, however, exhibit a similar trend of increasing conservativeness at higher confidence thresholds, further emphasizing their robustness.

Figure 25 presents a detailed view of the training and validation metrics for the YOLOv8 model trained on the UFPR-ALPR[29] dataset. The train/box loss decreases from 1.4 to 0.7, indicating a significant improvement in bounding box predictions. The train/classification loss drops sharply from 2.0 to 0.5, reflecting a substantial enhancement in class prediction accuracy. The train/DFL loss falls from 1.3 to 0.9, showing a marked improvement in distribution predictions. Training precision stabilizes around 0.96, demonstrating a high level of prediction accuracy with a minimal number of false positives. Recall increases from 0.84 to 0.94, indicating a notable improvement in true positive detection.

In validation, the box loss decreases from 1.15 to 0.95, classification loss drops from 0.7 to 0.4, and DFL loss falls from 1.15 to 1.05, all indicating consistent improvement. The mAP at IoU 0.5

rises and stabilizes near 0.98, reflecting excellent precision and recall balance. The mAP across IoU 0.5 to 0.95 improves to around 0.74, showing robust performance across varying thresholds. The model shows vital performance improvements, decreasing losses, and high precision and recall values. The increasing mAP scores demonstrate the model's effectiveness in detecting and classifying license plates, maintaining robust performance across different IoU thresholds.

Figure 26 displays the training and validation metrics for the YOLOv8 model trained on the CENPARMI dataset. The train/box loss decreases from 1.4 to 0.6, indicating improved bounding box predictions. The train/classification loss drops sharply from 5.0 to 0.5, reflecting significant enhancement in class prediction accuracy. The train/DFL loss falls from 1.3 to 0.9, demonstrating better distribution predictions. Training precision stabilizes around 0.9, indicating high accuracy in predictions with few false positives, while recall increases rapidly to around 0.9, showing improved true positive detection.

In validation, the box loss decreases from 1.1 to 0.8, classification loss drops from 4.5 to 0.5, and DFL loss falls from 1.0 to 0.9, all indicating consistent improvement. The mAP at IoU 0.5 rises and stabilizes near 0.98, reflecting an excellent balance between precision and recall. The mAP across IoU 0.5 to 0.95 improves to around 0.78, showing robust performance across varying thresholds.

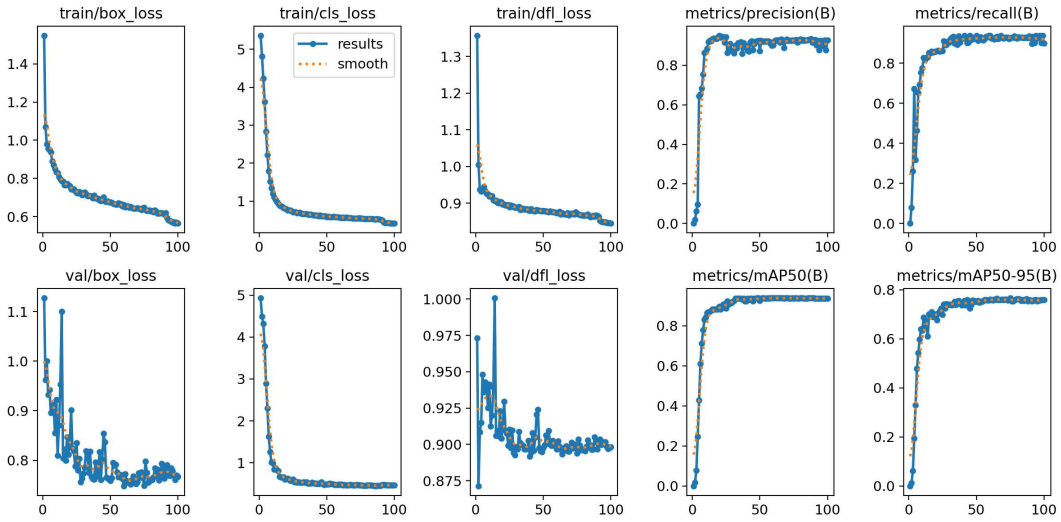

Figure 26. Various Performance Metrics and Loss Values for the YOLOv8 Model Trained on the CENPARMI Dataset

The YOLOv8 model, when trained on the CENPARMI dataset, shows vital performance improvements, decreasing losses, and high precision and recall values. The increasing mAP scores demonstrate the model's effectiveness in detecting and classifying license plates. Importantly, the model maintains robust performance across different IoU thresholds, reinforcing its versatility. The overall results indicate that the model is well-optimized and performs reliably on the CENPARMI dataset, consistently improving training and validation metrics.

The YOLOv8 model shows vital performance improvements on the UFPR-ALPR[29] and CENPARMI datasets, with some differences in specific metrics. For train/box loss, both datasets exhibit a steady decrease, with UFPR-ALPR[29] reducing from 1.4 to 0.7 and CENPARMI from 1.4 to 0.6, indicating slightly better improvements in CENPARMI. The train/classification loss drops significantly in both datasets, from 2.0 to 0.5 for UFPR-ALPR[29] and from 5.0 to 0.5 for CENPARMI, showing a more substantial improvement in the latter despite starting higher.

Both datasets show consistent train/DFL loss improvement, decreasing from 1.3 to 0.9. Training precision stabilizes around 0.96 for UFPR-ALPR[29] and 0.9 for CENPARMI, indicating slightly lower precision in CENPARMI. Recall metrics also improve, with UFPR-ALPR[29] increasing from 0.84 to 0.94 and CENPARMI stabilizing around 0.9.

Both datasets show decreasing val/box loss in validation, with CENPARMI ending lower. The val/classification loss drops more significantly in CENPARMI. Both datasets achieve similar high mAP at IoU 0.5, near 0.98. However, mAP across IoU 0.5 to 0.95 improves to around 0.74 for UFPR-ALPR[29] and 0.78 for CENPARMI, indicating slightly better performance in the latter.

In summary, both datasets show robust performance, with CENPARMI exhibiting slightly better overall results in classification loss, box loss, and mAP across multiple IoU thresholds. The model’s effectiveness in detecting and classifying license plates is evident in both datasets.

Figures 27 and 28 show YOLOv8’s recognition sample results for the CENPARMI and UFPR-ALPR[29] datasets, respectively.

The analysis of the performance metrics for license plate detection and recognition tasks across the UFPR-ALPR [29] and CENPARMI datasets, as shown in Tables 8 and 9, reveals that the model demonstrates superior efficiency with the CENPARMI dataset. For the detection task, detailed in Table 7, the model achieved an AP50:95 score of 79.67% on the CENPARMI dataset, surpassing

the 66.96% score on the UFPR-ALPR [29] dataset, indicating better precision and recall in detecting license plates. In the recognition task, as illustrated in Table 8, the Character Error Rate (CER) for CENPARMI is less than UFPR-ALPR [29] , the recall rate is higher for CENPARMI at 94% compared to 91% for UFPR-ALPR [29]. This suggests that the model detects license plates more accurately in the CENPARMI dataset and recognizes the characters within the plates more effectively. Therefore, the model is more suitable and reliable for automatic license plate recognition applications involving the CENPARMI dataset.

Figure 27. Recognition Sample Results of YOLOv8 Model on the UFPR-ALPR Dataset

Figure 28. Recognition Sample Results of YOLOv8 Model on the CENPARMI Dataset

Dataset	AP50:95
UFPR-ALPR	66.96%
CENPARMI	79.67%

Table 7. AP50:95 Scores for Various Datasets

Dataset	Character Error Rate(CER)	Recall
UFPR-ALPR	7.5%	91%
CENPARMI	3.5%	94%

Table 8. Performance Metrics for Different Models and Datasets

The recognition model's performance was further analyzed using four different datasets separately from Canada and the United States. The results for each dataset are detailed in the Table 9.

The model demonstrated consistent performance across the different datasets, with slight variations attributed to differences in plate designs, font styles, and environmental conditions. State of New York achieved the highest accuracy and the lowest Character Error Rate (CER), likely due to more precise and consistent plate designs.

Province/State	Character Error Rate(CER)	Recall
Quebec	4.5%	90%
Ontario	3.5%	93%
New York	2.9%	98%
California	3.2%	95%

Table 9. Performance Metrics for Different Provinces/States

Chapter 5. Font Evaluation Results and Analysis

In this chapter, we present several sets of outcomes derived from evaluating various fonts based on the recognition results of individual letters. We can draw several significant conclusions through our confusion matrices and a detailed analysis of the task of recognizing license plates. These insights highlight the strengths and weaknesses of different fonts in accurately identifying individual characters, which is crucial for improving the overall effectiveness of license plate recognition systems.

I. CNN+RNN Model with CTC Loss Results

CENPARMI Dataset: Our CENPARMI dataset includes 1600 license plates from diverse California, New York, Ontario, and Quebec environments. Because each dataset has a unique font, we conducted license plate detection and recognition in each province separately.

- **Quebec Dataset:** Based on the confusion matrix in Figure 29, we can observe the following character confusions:

  1. 'Q' and ‘0’:

  - Often need clarification due to their similar shapes.

  - The tail of the 'Q' is sometimes indistinct.

  2. '6' and 'G':

  - The resemblance stems from their similar curves and inner spaces.

  3. 'Q' and 'D':

  - Confusion can occur if the tail of the 'Q' blends with its bowl, resembling a 'D.'

  - This is especially true if both characters have a similar spur.

  4. 'W' and 'M':

  - These characters are confusing due to their similar diagonal strokes and internal structure.

  5. '6' and '4':

  - These can be mistaken for each other due to similarities in their upper structures and angles.

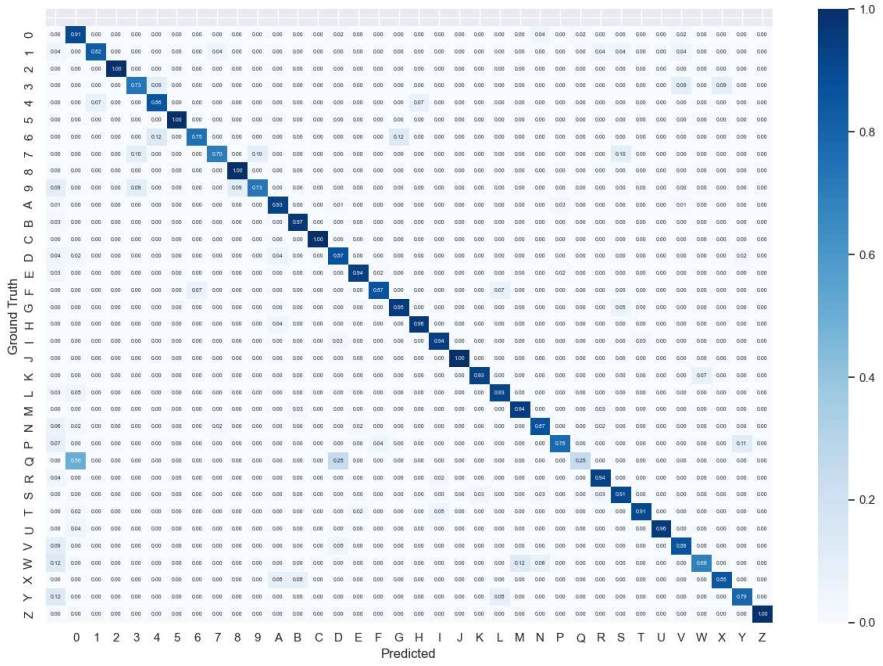

Figure 29. Confusion Matrix for Quebec Province

- **Ontario Dataset:** Based on the confusion matrix in Figure 30, we can observe the following character confusions:

  1. **'Q' and '0':**

     -These characters might be mistaken for each other due to their rounded bowls, especially if the 'Q' has a subtle tail.

  2. **'5' and '2':**

     -These can be confused because of their similar upper curves and strokes.

  3. **'6' and 'G':**

     -These characters might look alike due to their similar shape and internal space.

  4. **'1' and 'I':**

     -These can be indistinguishable when depicted as simple vertical lines.

  5. **'4' and 'A':**

     -Similar to '1' and 'I,' these characters can be indistinguishable when depicted as simple vertical lines.

6. **'V' and ‘Y':**

-These characters can appear similar if the 'Y' has a short or blended tail.

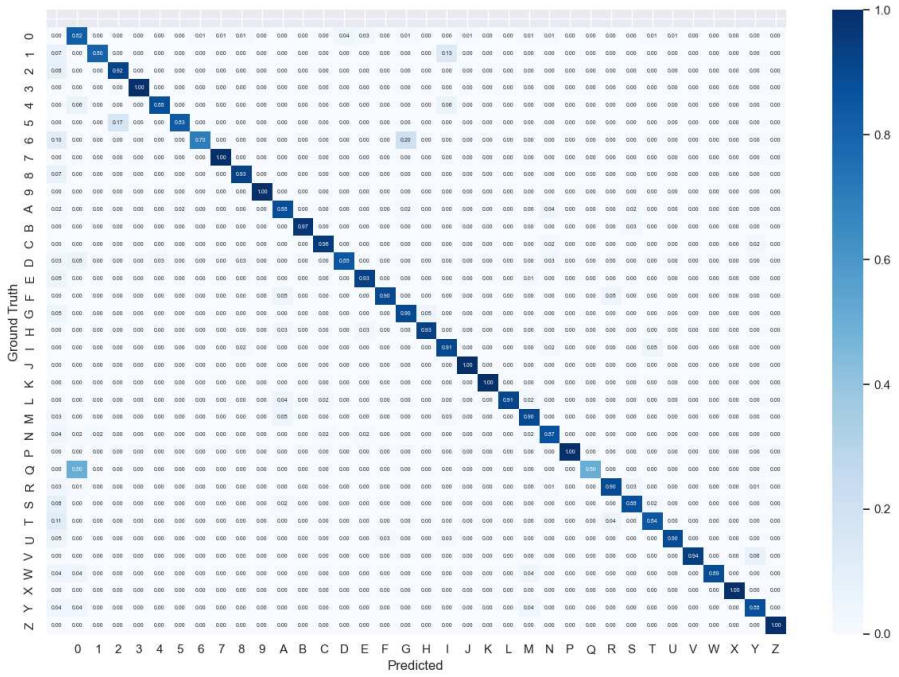

Figure 30. Confusion Matrix for Ontario Province

- **California State Dataset:** Based on the confusion matrix in Figure 31, we can observe the following character confusions:

1. **'O' and ‘0':**

-These characters can be confused due to their similar rounded shapes.

2. **'F' and ‘E':**

-These might be mistaken for each other because of their horizontal strokes, making it hard to notice the missing middle stroke in 'F'.

3. **'L' and 'T':**

-These can appear similar due to their vertical and horizontal line components, especially in bold weights.

4. **'7' and 'Z':**

-These might be confused when '7' has a pronounced horizontal stroke and 'Z' is styled with straight, minimalistic angles.

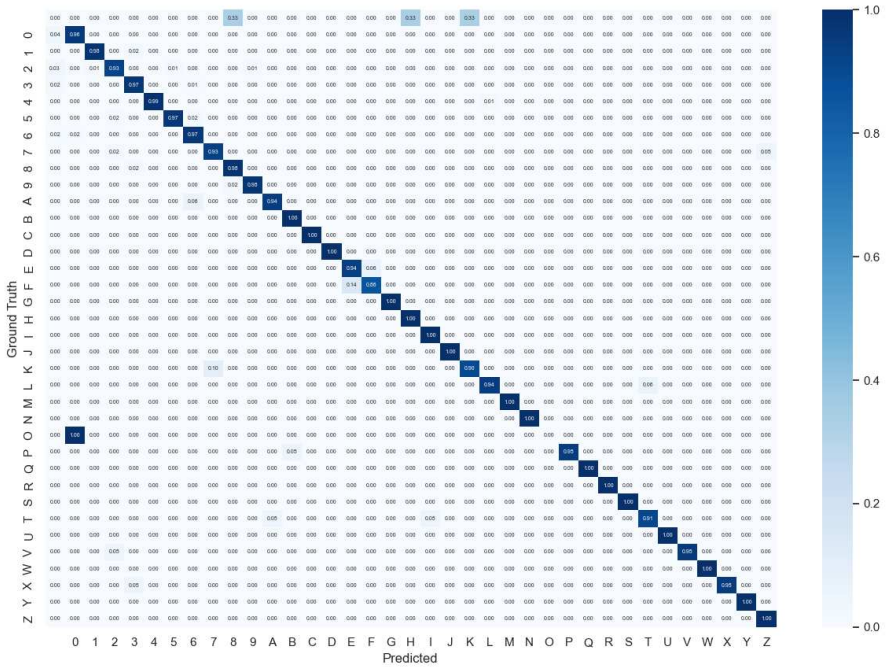

Figure 31. Confusion Matrix for California State

- **New York State Dataset:** Based on the confusion matrix in Figure 32, we can observe the following character confusions:

1. **'O' and '0':**

   -These characters can look very similar due to their tightly rounded shapes.

2. **'J,' 'I,' and 'T':**

   -These letters might be mistaken for one another due to their similar vertical strokes, especially when 'J' has a subtle curve and 'T' has a short horizontal stroke.

3. **'V' and 'U':**

   -These can be confused if 'V' has a sharp vertex resembling a 'U' without the middle crossbar.

4. **'X' and 'L':**

   -These may appear similar if 'X's diagonal strokes blend, looking like intersecting 'L's.

5. **'M' and ‘W':**

-These look like mirror images with unclear middle peaks and valleys in a condensed font.

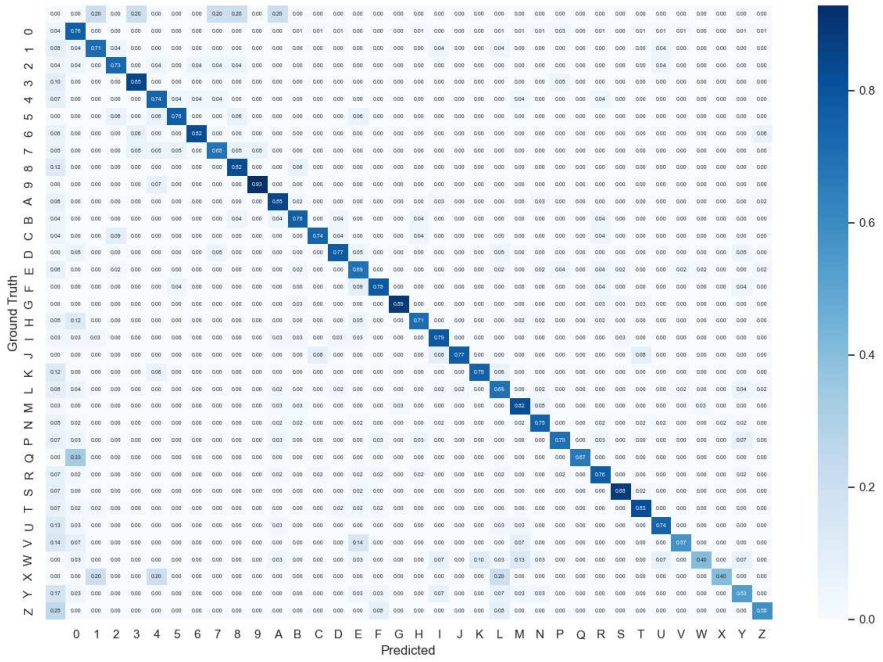

Figure 32. Confusion Matrix for New York State

UFPR-ALPR Dataset: The confusion matrix in Figure 33 highlights common character misidentifications due to their similar appearances.

1. **'O' and ‘0':**

-These characters can look very similar due to their tightly rounded shapes.

2. **'I', '1', and 'T':**

-These can be mistaken for each other when depicted as straight lines, especially if 'T' has a short horizontal stroke.

3. **'M' and 'W':**

-These characters look alike due to their mirrored structures and bold lines.

4. **'B' and '8', 'E' and ‘8’:**

-'B' and '8' are often mistaken for each other due to their similar bowls and counters. Both characters feature rounded parts resembling each other, and the double loops of

'8' can be confused with the loops of 'B' when the spacing is similar.

5. **'C' and 'G':**

-These might be confused if 'G's tail is subtle, making it look like 'C.'

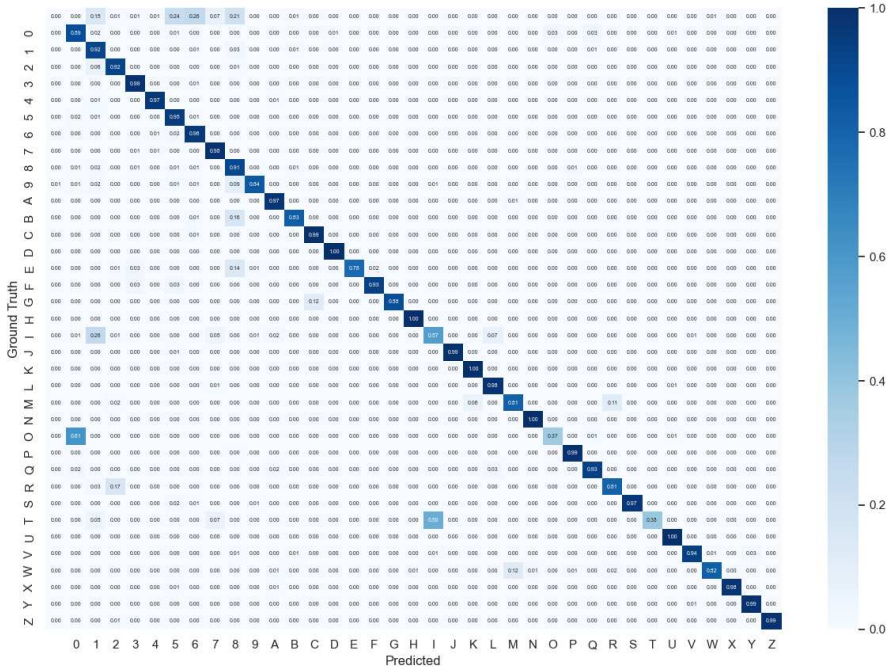

Figure 33. Confusion Matrix for UFPR-ALPR Dataset

II. YOLOv8 Model

CENPARMI Dataset: We present two sets of analyses on this dataset. The CENPARMI dataset was used for the training and testing phases in the first analysis and the confusion matrix in shown in Figure 34. The second analysis tested the model exclusively on the CENPARMI dataset, which it had not seen during the training phase and the confusion matrix is shown in Figure 35. The second analysis aims to demonstrate that the YOLOv8 model is a highly efficient OCR model capable of generalizing well. Specifically, we strive to show that it can accurately detect letters from fonts with features it did not encounter during training. Based on the confusion matrix in Figure 34, we can observe the following character confusions:

1. **'O' and '0':**

-These characters can look very similar due to their tightly rounded shapes.

2. **'I' and 'T':**

-These characters are often mistaken for each other when depicted as straight lines, especially if 'T' has a short horizontal stroke.

3. **' B' and '8':**

-'B' and '8' are often mistaken for each other due to their similar bowls and counters. Both characters feature rounded parts resembling each other, and the double loops of '8' can be confused with the loops of 'B' when the spacing is similar.

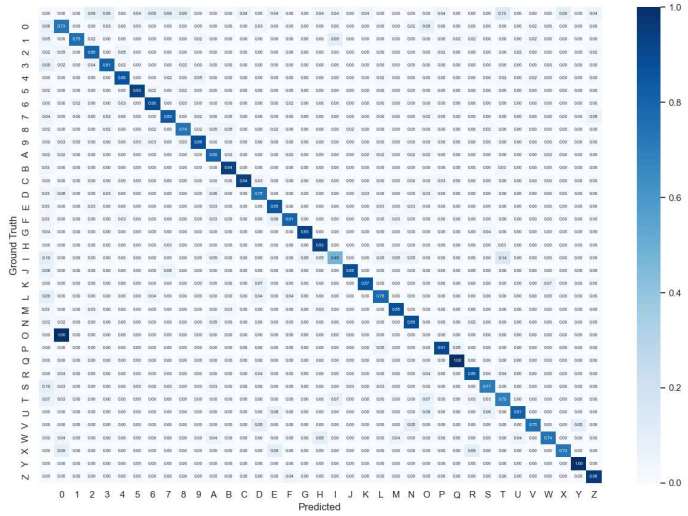

Figure 34. Confusion Matrix for CENPARMI Dataset

Based on the confusion matrix in Figure 34, we can observe the following character confusions:

1. **'O' and ‘0':**

-The confusion between 'O' and '0' arises from their similar bowls, as both characters have rounded shapes that make them difficult to distinguish.

2. **'I' and 'T':**

-The characters 'I' and 'T' are often mistaken for one another because of their similar apexes, where the vertical stems of 'I' can resemble the short horizontal stroke of 'T.' Additionally, identical horizontal strokes at the top or bottom further contribute to the confusion between these characters.

3. **'V' and '4':**

-'V' and '4' can be confused due to their similar diagonal strokes. Both characters

have similar angular shapes that can be confused due to identical placement and design. The similar apexes and spurs, with the junctions and small projections of 'V' looking like parts of '4', further add to this confusion.

4. **'F' and 'W':**

-'F' and 'W' can be confused due to their similar horizontal strokes. 'F' has a horizontal stroke at the top and middle, which can sometimes resemble the diagonal and horizontal structure of 'W,' especially in certain font styles where the middle stroke of 'F' is prominent.

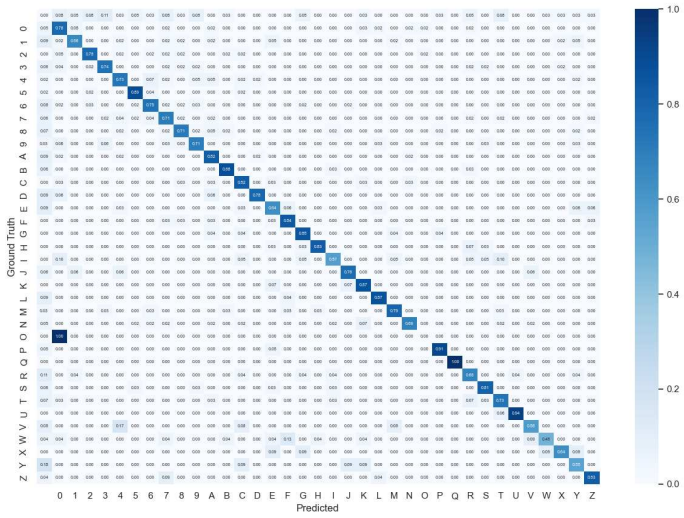

Figure 35. Confusion Matrix for CENPARMI Dataset

Our CENPARMI dataset includes 1600 license plates from diverse places including, California, New York, Ontario, and Quebec environments. Because each dataset has a unique font, we conducted license plate detection and recognition in each province separately.

- **Quebec Dataset:** Based on the confusion matrix in Figure 36, we can observe the following character confusions:

1. **'0' and '6':**

-The characters '0' and '6' are often mistaken for one another because of their similar bowls, where the rounded shape of '0' can resemble the partially closed loop of '6.' Additionally, if the tail of '6' is not clear, it can be further confused with '0.'

2. **'2' and '4':**

-The characters '2' and '4' are often mistaken for one another because of their similar apexes, where the top part of '2' can resemble the horizontal stroke of '4.' Additionally, the similar diagonal stroke in '2' and the vertical line in '4' contribute to the confusion between these characters.

3. **'2' and '9':**

-The characters '2' and '9' are often mistaken for one another because of their similar bowls and tails, where the curved shape of '2' can resemble the loop of '9.' Additionally, if the tail of '9' is not prominent, it can be further confused with '2.'

4. **'M' and 'N' with 'I':**

-The characters 'M' and 'N' with 'I' are often mistaken for one another because of their identical vertical strokes, where the stems of 'M' and 'N' can resemble the simple vertical line of 'I.' This is particularly true in sans-serif fonts where distinguishing elements are minimal.

5. **'8' and 'B':**

-The characters '8' and 'B' are often mistaken for one another because of their similar bowls, where the two loops of '8' can resemble the top and bottom rounded forms of 'B.' In fonts where these forms are symmetrical, the confusion between these characters is further increased.

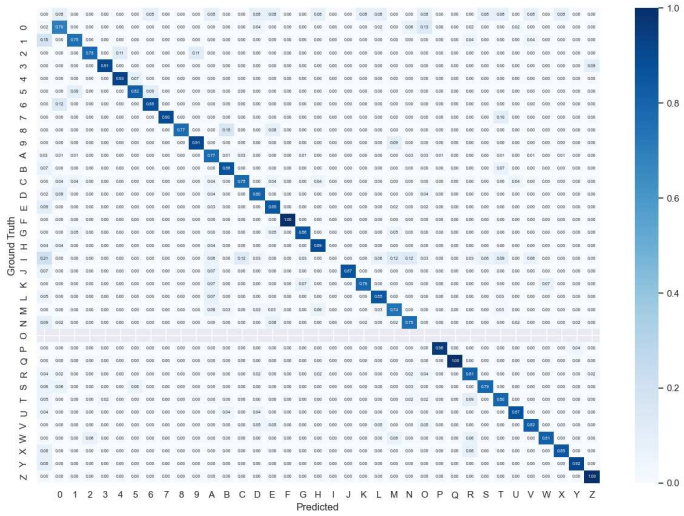

Figure 36. Confusion Matrix for Quebec Province

- **Ontario Dataset:** Based on the confusion matrix in Figure 37, we can observe the following character confusions:

  1. **'0' and ‘Q’:**

     -The characters 'Q' and '0' are often mistaken for one another because of their similar bowls, where the rounded shape of '0' can resemble the circular form of 'Q.' Additionally, if the tail of 'Q' is not clear or prominent, it can be further confused with '0.'

  2. **'2' and ‘5’:**

     -The characters '5' and '2' are often mistaken for one another because of their similar diagonal strokes, where the curved and diagonal parts of '5' can resemble the shape of '2.' In certain fonts, the top part of '5' can also look like the horizontal stroke of '2,' increasing the confusion.

  3. **'J' and ‘4’ with ‘D’:**

     -The characters 'J,' '4,' and 'D' are often mistaken for one another because of their similar curved shapes and tails. The top part of 'J' can resemble the vertical and horizontal strokes of '4,' while the rounded part of 'J' and the curve of '4' can look like the bowl of 'D,' contributing to the confusion between these characters.

  4. **'D' and ‘O’:**

     -The characters 'O' and 'D' are often mistaken for one another because of their similar bowls, where the circular shape of 'O' can resemble the rounded part of 'D.' If the vertical stroke of 'D' is subtle or not prominent, it can be further confused with 'O.'

  5. **'1' and ‘I’:**

     -The characters '1' and 'I' are often mistaken for one another because of their identical vertical strokes, where the simple line of '1' can resemble the line of 'I.' This confusion is particularly prevalent in sans-serif fonts where distinguishing features like serifs are absent.

- **California:** Based on the confusion matrix in Figure 38, we can observe the following character confusions:

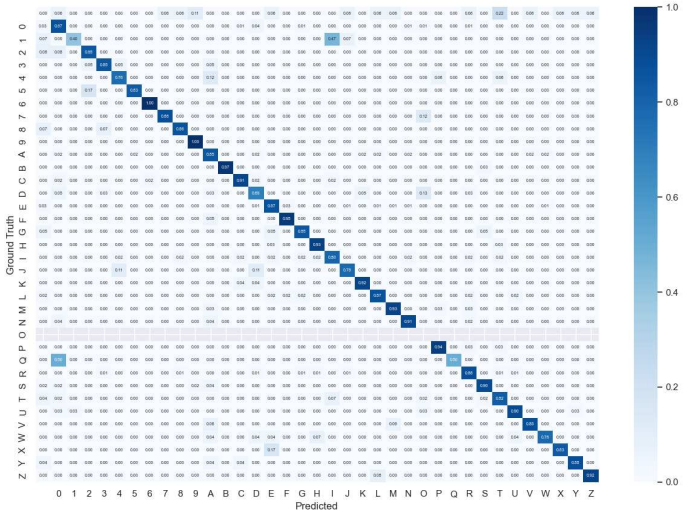

Figure 37. Confusion Matrix for Ontario Province

1. **'0' and ‘Q’ and ‘O’:**

   -The characters '0,' 'Q,' and 'O' are often mistaken for one another because of their similar bowls, where the rounded shapes of '0' and 'O' can resemble the circular form of 'Q.' Additionally, if the tail of 'Q' is not clear or prominent, it can be further confused with '0' and 'O.'

2. **'F' and ‘4’:**

   -The characters 'F' and '4' are often mistaken for one another because of their similar horizontal strokes, where the top and middle horizontal lines of 'F' can resemble the top part and crossbar of '4.' Additionally, the vertical stroke in 'F' can be confused with the vertical part of '4,' increasing the likelihood of confusion.

3. **'F' and ‘E’:**

   -The characters 'F' and 'E' are often mistaken for one another because of their similar horizontal strokes, where the top and middle lines of 'F' can resemble the top and middle lines of 'E.' Additionally, the bottom horizontal stroke of 'E' can be mistaken for an incomplete or worn-out stroke in 'F,' especially in certain fonts or handwriting styles.

4. **'I' and ’T’:**

   -The characters 'I' and 'T' are often mistaken for one another because of their similar vertical strokes, where the vertical stem of 'I' can resemble the vertical part of 'T.' Additionally, identical horizontal strokes at the top or bottom further contribute to

the confusion between these characters.

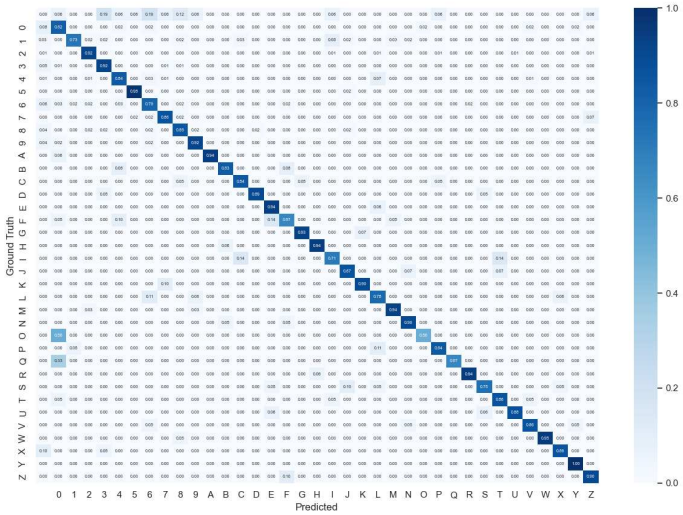

Figure 38. Confusion Matrix for California State

- **New York State Dataset:** Based on the confusion matrix in Figure 39, we can observe the following character confusions:

  1. **'X' and ‘0’:**

     -The characters '0' and 'X' are often mistaken for one another because of their similar circular and intersecting shapes. In certain fonts, the rounded form of '0' can resemble the intersecting diagonal lines of 'X,' especially if the font style makes the 'X' appear more rounded or the '0' more angular.

  2. **'9' and ‘8’:**

     -The characters '9' and '8' are often mistaken for one another because of their similar loops and rounded shapes. The top loop of '8' can resemble the rounded part of '9,' and the bottom loop of '8' can create confusion, especially in fonts where the loops of '8' are not distinctly different in size.

  3. **'5' and ‘6’:**

     -The characters '5' and '6' are often mistaken for one another because of their similar

curved shapes and tails. The top part of '5' can resemble the upper loop of '6,' and the curved bottom of '5' can look like the lower part of '6,' especially in fonts where these characters are designed with similar curvature.

4. **'8' and ‘A’:**

-The characters '8' and 'A' are often mistaken for one another because of their similar top and bottom sections. The two loops of '8' can resemble the triangular form of 'A,' particularly in fonts where the loops of '8' are more angular or the 'A' is designed with a rounded apex.

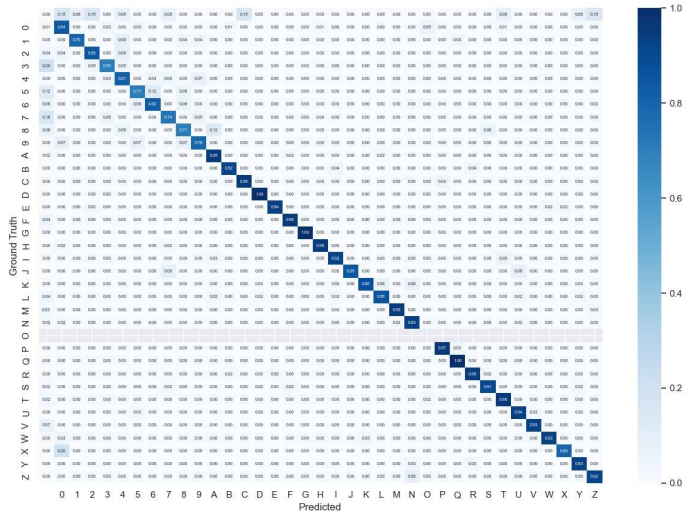

Figure 39. Confusion Matrix for New York State

UFPR-ALPR Dataset: The confusion matrix in Figure 40 highlights common character misidentifications due to their similar appearances.

1. **'O' and ‘0':**

-The confusion between 'O' and '0' arises from their similar bowls, as both characters have rounded shapes that make them difficult to distinguish.

2. **'E' and ‘4':**

- Their similar horizontal and vertical strokes can confuse 'E' and '4'. The structure of 'E' with its three horizontal lines can resemble the shape of '4', especially if the top and middle lines are similar in design.

3. 'C' and '9':

- 'C' and '9' can be confused due to their similar bowls. The open, curved shape of 'C' can resemble the top part of '9', especially if the tail of '9' is not pronounced or evident.

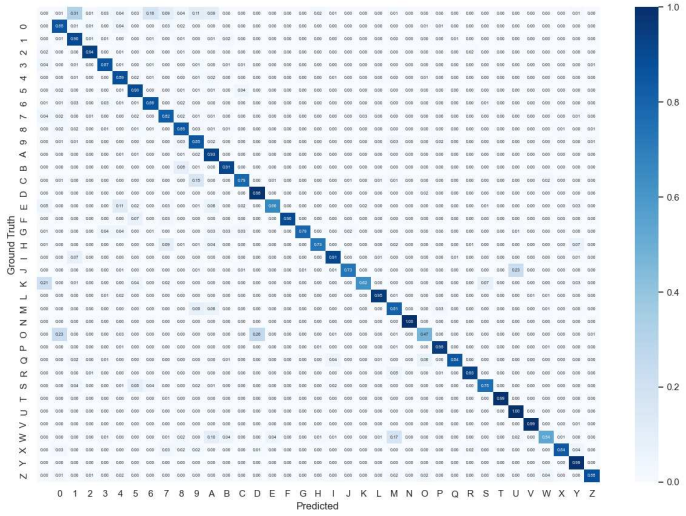

Figure 40. Confusion Matrix for UFPR-ALPR Dataset

4. 'M' and 'W':

-'M' and 'W' are commonly confused because of their similar diagonal strokes; both characters have mirrored structures and bold diagonal lines that look alike. The identical spurs, or small projections at the ends of these strokes, also contribute to the difficulty in distinguishing them.

5. ' B' and '8':

-'B' and '8' are often mistaken for each other due to their similar bowls and counters. Both characters feature rounded parts resembling each other, and the double loops of '8' can be confused with the loops of 'B' when the spacing is similar.

6. ' A' and ‘W':

- 'A' and 'W' can be confused due to their similar diagonal strokes. The structure of 'A,' with its pointed apex and diagonal lines, can resemble the overlapping diagonal strokes of 'W,' especially in specific fonts where the middle strokes of 'W' are less

pronounced.

7. ' I' and ‘1’:

- 'I' and '1' are often mistaken for one another because of their similar vertical stems. Both characters are depicted as straight vertical lines; without additional distinguishing features (such as serifs or hooks), they can appear nearly identical. This similarity in shape makes it difficult to distinguish between the two, especially in specific fonts or styles where the characters are very simplified.

8. ' J' and ‘U’:

- 'J' and 'U' can be confused due to their similar curved shapes. Both characters have rounded bottoms, and if the tail of 'J' is subtle or the curve of 'U' is more pronounced, they can look alike, especially in specific fonts where these characteristics are emphasized

9. ' D' and ‘O’:

- 'D' and 'O' can be confused due to their similar rounded shapes. The curved stroke of 'D' can resemble the rounded shape of 'O', especially in fonts where the vertical stem of 'D' is less pronounced, making it appear more circular.

Chapter 6. Conclusion and Future Work

This chapter concludes the thesis and provides some recommendations for future work.

I. Conclusion

Automatic License plate recognition system (ALPR) with high accuracy is challenging due to environmental factors such as light conditions, rain, and dust. License Plate Detection (LPD) is a significant area of study extensively explored by researchers because of its implications in various domains such as traffic monitoring, safety, and security. Many techniques were employed by researchers to improve the accuracy of automatic license plate detection systems including, image processing, machine learning, and deep learning techniques. This work utilizes more efficient deep-learning algorithms to detect license plates and recognize letters.

Moreover, font style on license plates also significantly affects the recognition tasks. Although font evaluation is crucial and greatly impacts recognition accuracy, it has yet to be thoroughly investigated in the existing literature. In particular, there needs to be research on the font evaluation of Ontario provinces of Canada, California, and New York. Therefore, this work has three main aspects. Firstly, we employed two deep-learning techniques for license plate detection, including Faster-RCNN and YOLOv8. Secondly, we utilized two deep neural networks for character recognition of license plates, including the CNN-RNN model with CTC loss networks and YOLOv8. Thirdly, we assessed font characteristics within the context of license plates. This work examines the two approaches for license plate detection and character recognition. It also analyzes the character error rate for each dataset and each deep learning model. Moreover, it explores and evaluates five different fonts for license plates and shows how the choice of font can affect the readability of license plates.

What makes our approach truly stand out is the utilization of the DINO (DIstillation with NO labels) method, a self-supervised learning technique that employs Vision Transformers (ViT) to train models without labeled data. This cutting-edge method not only enables the detection of multiple objects by promoting consistency in feature representation despite varied input conditions, but also facilitates the development of rich, meaningful feature representations that

can be applied to various downstream tasks. By reducing the reliance on extensive labeled datasets and making better use of available unlabeled data, DINO opens up new possibilities in the field of machine learning.

We trained two deep-learning networks for the detection problem: Faster R-CNN and YOLOv8. These networks were selected due to their robustness and efficiency in object detection tasks. After extensive training and evaluation, we achieved an average precision of 95.72% on the detection task, demonstrating the effectiveness of these models in accurately identifying and localizing license plates in various conditions.

The license plate recognition task, a complex and challenging problem, was tackled using two deep-learning networks. We employed the CNN-RNN model with CTC loss networks and YOLOv8 to recognize the characters. Our analysis of the character error rates led to a significant achievement-a 94% recall ratio for the recognition task using YOLOv8.

We used confusion matrices for the font evaluation problem to explore confusion cases. Moreover, we compared our results with those of one of the commercial products for the optical character recognition(OCR) task.

II. Future Work

Our future work presents novel and impactful directions that extend the current research:

We aim to tackle license plate detection in more challenging contexts by training deep learning networks to detect plates of various sizes, styles, and font types across all vehicles, ensuring robustness and versatility.

Our proposal includes the development of a practical benchmark/testing framework. This framework will evaluate the performance of commercial license plate recognition products and support multiple datasets worldwide, each with different font types. It will be easily integrated with new datasets and products and standardize performance metrics, facilitating comprehensive and comparative evaluation. We intend to create a standard benchmark for typographers to assess new font types for license plate detection systems. This system will allow typographers to input font type glyphs, which will then be automatically evaluated for their suitability in detection and recognition models, identifying potential confusion cases.

Given Quebec's permission for personalized license plates since 2018, we propose extending the Quebec license plate dataset to include more personalized items.

We will collect new datasets from other Canadian provinces that allow personalized plates, such as Ontario, Alberta, Manitoba, and Nova Scotia. We will retrain the detection network with these new datasets and monitor performance and potential confusion cases associated with personalized plates.

References

[1] J. Redmon, and A. Farhadi, "YOLO9000: Better, faster, stronger," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, USA, 2017, pp. 6517-6525. DOI:10.1109/CVPR.2017.690

[2] Ren, S., He, K., Girshick, R. B., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. CoRR, abs/1506.01497.Retrieved from http://arxiv.org/abs/1506.01497

[3] Shi, B., Bai, X., & Yao, C. (2015). An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Application to Scene Text Recognition. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/1507.05717

[4] OpenALPR, https://platerecognizer.com/ , last accessed October, 4, 2018.

[5] Laroca, E. Severo, L. Zanlorensi, L. Oliveira, G. Gon.alves, W. Schwartz, and D. Menotti, "A robust realtime automatic license plate recognition based on the YOLO detector", International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 2018, DOI: 10.1109/IJCNN.2018.8489629.

[6] Y. Li, and C. Y. Suen, "Typeface personality traits and their design characteristics", In Proceedings of the 9th IAPR International Workshop on Document Analysis Systems (DAS '10). ACM, New York, NY, USA, pp. 231-238, 2010, DOI=http://dx.doi.org/10.1145/1815330.1815360

[7] P. Baines, and A. Haslam, "Type & typography", ISBN-13: 978-0823055289, Watson-Guptill, New York, 2005.

[8] D. Hoiem, Y. Chodpathumwan, and Q. Dai, "Diagnosing error in object detectors", In Computer Vision, ECCV 2012 - 12th European Conference on Computer Vision, Proceedings.

PART 3 ed. 2012. pp. 340- 353. (Lecture Notes in Computer Science; PART 3). DOI: 10.1007/978-3-642-33712-3_25

[9] Su, Z., Su, Z., Yu, J., Tan, H., Wan, X., & Qi, K. (2023). MSA-YOLO: A Remote Sensing Object Detection Model Based on Multi-Scale Strip Attention. Sensors, 23(15), 6811.

[10] Zheng L., He X. "Character Segmentation for License Plate Recognition by K-Means Algorithm." In proc. of Image Analysis and Processing, ICIAP 2011. Lecture Notes in Computer Science, vol 6979, pp. 444-453, Springer. Berlin, Heidelberg 2011.

[11] Soungsill Park, Hyoseok Yoon and Seho Park, "Multi-Style License Plate Recognition System using K-Nearest Neighbors," KSII Transactions on Internet and Information Systems, vol. 13, no. 5, pp. 2509-2528, 2019.

[12] K. S. Satwashil and V. R. Pawar, "Integrated natural scene text localization and recognition," In Proc. of International conference of Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2017, pp. 371-374.

[13] V. Joy, A. Binu, and A. Kuttyamma, "License plate detection and recognition using a HOG feature based SVM classifier", International Journal of Innovative Research in Computer and Communication Engineering, Volume 5, Issue 5, pp. 10115-10119, May 2017, DOI: 10.15680/ IJIRCCE.2017.0505235.

[14] A. Vaishnav and M. Mandot, "An integrated automatic number plate recognition for recognizing multi language fonts," in *2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*. IEEE, 2018, pp. 551–556.

[15] Carvalho, A. M., & Prati, R. C. (2020). DTO-SMOTE: Delaunay Tessellation Oversampling for Imbalanced Data Sets. Information. https://doi.org/10.3390/info11120557

[16] P. Wu and Y. Lin, “Research on license plate detection algorithm based on ssd,” in *Proceedings of the 2nd International Conference on Advances in Image Processing*, 2018, pp. 19–23.

[17] S. M. Silva and C. R. Jung, “Real-time brazilian license plate detection and recognition using deep convolutional neural networks,” in *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. IEEE, 2017, pp. 55–62.

[18] T. K. Cheang, Y. S. Chong, and Y. H. Tay, “Segmentation-free vehicle license plate recognition using convnet-rnn,” *arXiv preprint arXiv:1701.06439*, 2017.

[19] X. Wang, Z. Man, M. You, and C. Shen, “Adversarial generation of training examples: applications to moving vehicle license plate recognition,” *arXiv preprint arXiv:1707.03124*, 2017.

[20] B. Suvarnam and V. S. Ch, "Combination of CNN-GRU Model to Recognize Characters of a License Plate number without Segmentation," In Proc. of 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 2019, pp. 317-322.

[21] Q. Fu, Y. Shen, and Z. Guo, “License plate detection using deep cascaded convolutional neural networks in complex scenes”, In: Liu D., Xie S., Li Y., Zhao D., El-Alfy ES. (eds) Neural Information Processing. ICONIP 2017. Lecture Notes in Computer Science, vol 10635. Springer, Cham, DOI: https://doi.org/10.1007/978-3-319-70096-0_71

[22] F. Kurpiel, R. Minetto, and B. T. Nassu, "Convolutional neural networks for license plate detection in images", 2017 IEEE International Conference on Image Processing (ICIP), Beijing, 2017, pp. 3395-3399. DOI: 10.1109/ICIP.2017.8296912.

[23] G. Hsu, A. Ambikapathi, S. Chung, and C. Su, "Robust license plate detection in the wild", 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Lecce, 2017, pp. 1-6. DOI: 10.1109/AVSS.2017.8078493

[24] N. Astawa, G. Caturbawa, M. Sajayasa, and M. Atmaja, “Detection of license plate using sliding window, histogram of oriented gradient, and support vector machines method”, Journal of Physics: Conference Series. Volume 95, Number 1, 2018. Series 953012062. DOI: 10.1088/1742-6596/953/1/012062.

[25] C. Y. Suen, N. Dumont, M. Dyson, Y. Tai, and X. Lu, "Evaluation of fonts for digital publishing and display", 2011 International Conference on Document Analysis and Recognition, Beijing, 2011, pp. 1424- 1436. DOI: 10.1109/ICDAR.2011.307.

[26] K. Larson, and M. Carter, “Chapter 3: Sitka: a collaboration between type design and science”, Series on Language Processing, Pattern Recognition, and Intelligent Systems: Digital Fonts and Reading, World Scientific Publishing Co Pte Ltd, edited by: Mary Dyson and Ching Suen, pp. 37-53 (2016), DOI: 10.1142/9789814759540_0003.

[27] S. Beier, “Chapter 5: Designing legible fonts for distance reading”, Series on Language Processing, Pattern Recognition, and Intelligent Systems: Digital Fonts and Reading, World Scientific Publishing Co Pte Ltd, edited by: Mary Dyson and Ching Suen, pp. 79-93 (2016). DOI: 10.1142/9789814759540_0005

[28] M. Almuhajri, and C. Y. Suen, “Chapter 14: Legibility and readability of Arabic fonts on Personal Digital Assistants PDA”, Series on Language Processing, Pattern Recognition, and Intelligent Systems: Digital Fonts and Reading, World Scientific Publishing Co Pte Ltd, edited by: Mary Dyson and Ching Suen, pp. 248-265 (2016), DOI: 10.1142/9789814759540_0014.

[29] R. Laroca, E. Severo, L. Zanlorensi, L. Oliveira, G. Gonçalves, W. Schwartz, and D. Menotti, “A robust real-time automatic license plate recognition based on the YOLO detector”, International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 2018, DOI: 10.1109/IJCNN.2018.8489629.

[30] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: unified, real-time object detection”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779- 788, 2016.

[31] Ahmed, M., Pagani, A., Liwicki, M., Stricker, D., & Afzal, M. (2021). Survey and Performance Analysis of Deep Learning Based Object Detection in Challenging Environments. Sensors, 21(15), 5116.

[32] Reda, I., Ghazal, M., Shalaby, A., Elmogy, M., Aboulfotouh, A., El-Ghar, M. A., Elmaghraby, A., Keynton, R., & El-Baz, A. (2019). Detecting Prostate Cancer Using A CNN-Based System Without Segmentation. https://doi.org/10.1109/isbi.2019.8759102

[33] Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., … Adam, H. (2019). Searching for MobileNetV3. *arXiv [Cs.CV]*. Retrieved from http://arxiv.org/abs/1905.02244

[34] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., … Zhang, L. (2023). Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. *arXiv [Cs.CV]*. Retrieved from http://arxiv.org/abs/2303.05499

[35] Rabiah.Alqudah. License plate detection using deep learning and font analysis.(2019)