

Improving License Plate Identification in Morocco: Intelligent Region Segmentation Approach, Multi-Font and Multi-Condition Training

El Mehdi Ben Laoula^{1,*}, Marouane Midaoui², Mohamed Youssfi¹, Omar Bouattane¹

¹2IACS Laboratory, ENSET Mohammedia, University Hassan II of Casablanca, 28830, Morocco

²M2S2I Laboratory, ENSET Mohammedia, University Hassan II of Casablanca, 28830, Morocco

ARTICLE INFO

Article history:

Received: 09 March, 2023

Accepted: 04 June, 2023

Online: 25 June, 2023

Keywords:

License plate recognition

YOLOv5

Detectron2

Intelligent region segmentation

Customized dataset

Moroccan license plate issues

Fonts-based data

ABSTRACT

The exponential growth in the number of automobiles over the past few decades has created a pressing need for a robust license plate identification system that can perform effectively under various conditions. In Morocco, as in other regions, local authorities, public organizations, and private companies require a reliable License Plate Recognition (LPR) system that takes into account all plates specifications (HWP, VWP, DP, YP, and WWP) and multiple fonts used. This research paper introduces an intelligent LPR system implemented using the YOLOv5 and Detectron2 frameworks, which have been trained on a customized dataset comprising multiple fonts (such as CRE, HSRP, FE-S, etc.) and accounting for different circumstances such as illumination, climate, and lighting conditions. The proposed model incorporates an intelligent region segmentation approach that adapts to the plate's type, thereby enhancing recognition accuracy and overcoming conventional issues related to plate separators. With the use of image preprocessing and temporal redundancy optimization, the model achieves a precision of 97,181% when handling problematic plates, including those with specific illumination patterns, separators, degradations, and other challenges, with little advantage to YOLOv5 over Detectron2.

1. Introduction

The global vehicle population has experienced substantial growth in recent decades fueled by a combination of factors, including demographic changes, lifestyle shifts, and advancements in the automotive industry. To accommodate this growth, numerous countries have developed their own vehicle registration systems, wherein each vehicle, including cars, trucks, and motorcycles, is assigned a unique license plate. These license plates typically consist of a combination of letters and numbers, serving as an alphanumeric identifier that may also be associated with the vehicle owner for easy identification.

To enable effective vehicle tracking and activity monitoring, the development of automatic number plate recognition (ANPR) systems took place. These systems employ optical character recognition (OCR) technology to analyze pre-captured images of license plates, which are captured by dedicated cameras. Through

this analysis, ANPR extracts the plate numbers and thereby identifies vehicles and their owners [1]. This advancement eliminates the need for manual identification of license plates, previously carried out by human agents and prone to errors. ANPR finds widespread use in law enforcement agencies for enforcement purposes and is also utilized by highway agencies to implement road pricing [2]. Furthermore, ANPR is employed in automated parking systems to facilitate charging processes [3].

Due to the progress in computer science and technology, alongside advancements in databases, Automatic Number Plate Recognition (ANPR) has emerged as a crucial component of traffic management systems within smart cities [4-6]. ANPR is recognized as a valuable tool for gathering traffic data and enhancing road efficiency and safety, aligning with the primary objectives of Intelligent Transportation Systems (ITS) [7]. The ANPR procedure encompasses a series of techniques and automated algorithms, typically comprising four steps: capturing an image of the vehicle, detecting the license plate, segmenting

*Corresponding Author: El Mehdi Ben Laoula, mehdi.benlaoula@gmail.com

www.astesj.com

<https://dx.doi.org/10.25046/aj080329>

the characters on the plate, and ultimately recognizing the characters.

2. Literature review

2.1. License Plate Recognition System

With the advent of artificial intelligence and especially deep learning, several license plate recognition system was built with interesting achievements. In fact, authors in [8] utilized a compact yet powerful network to classify characters on plates extracted by the Single Shot MultiBox Detector (SSD) [9]. Subsequently, various extensions of these neural networks have been proposed. In [10], a system for Automatic License Plate Recognition (ALPR) specific to Chinese license plates was introduced, utilizing two Convolutional Neural Networks (CNNs) based on the YOLO2 framework. The system was compared to other version of the framework and achieved a detection precision of 99.35% and a recognition precision exceeding 97.89% at a speed of 12.19ms. Another robust and efficient ALPR system based on Deep Neural Network was experienced in India [11]. It presents a License Plate Detection (LPD), followed by pre-processing of detected license plates and License Plate Recognition (LPR) using the LSTM Tesseract OCR Engine. Experimental results demonstrate high accuracy, with a 99% LPD accuracy and 95% LPR accuracy comparable to commercial ANPR systems such as Open-ALPR and Plate Recognizer.

In [12], a sliding window technique was proposed for identifying license plates in Taiwan. The system achieved a license plate detection accuracy of approximately 98.22% and a license plate recognition accuracy of 78%, with each image requiring 800ms for processing. Additionally, researchers introduced a novel ALPR system based on YOLOv2 in [13]. Their focus was on capturing license plates in uncontrolled scenarios with potential view distortions. The system employed a unique CNN capable of identifying and correcting multiple distorted license plates within a single image, using an Optical Character Recognition (OCR) approach. The final results were promising.

For recognizing Jordanian license plates, an ALPR system based on YOLOv3 was proposed [14]. This system underwent testing on genuine videos from YouTube, achieving an accuracy of 87% in recognition. Similarly, contributors of [15] implemented a YOLO framework to detect and recognize Iranian license plates. Their system demonstrated an accuracy of 95.05% after testing over 5000 images. Another Iranian study [16] has compiled a complete dataset comprising 19,937 car images and 27,745 license plate characters, annotated with the entire license plate information. This dataset was experienced in license plate detection using several optimization YOLOv5 and detectron2 frameworks.

2.2. Temporal redundancy

The incorporation of a temporal redundancy stage within the proposed license plate recognition architecture has gained significant attention in the research community. Several studies have explored the importance of this stage in improving the accuracy and efficiency of license plate recognition systems. The benefits of temporal redundancy are widely recognized, the implementation and optimization of this stage can vary depending on the specific system requirements. Factors such as dataset size,

computational resources, and real-time constraints need to be carefully considered to ensure efficient and accurate license plate recognition.

One notable study [17] demonstrated the effectiveness of temporal redundancy in enhancing recognition accuracy. They presented a high-accuracy pole number recognition framework for high-speed rail catenary systems, overcoming challenges such as illumination changes, image blurs, and occlusions. Our approach combines a cascaded CNN-based Detection and Recognition model (DR-YOLO) with a temporal redundancy approach, achieving accurate results through global and local features and context-based combination of adjacent frames. Extensive experimental testing validates the effectiveness and efficiency of their approach in real-world working environments. In another related work [18], authors experienced the use of temporal redundancy in license plate detection. The results of their method reached an overall recognition rate of 86% and achieved an outstanding accuracy of 99% for four-letter plates. Furthermore, the incorporation of temporal redundancy significantly enhanced the recognition rate to 96%. Compared to Sighthound and OpenALPR, this method outperforms them by 9% and 4.9% respectively, showcasing its superiority. Also, authors of [19] explored temporal redundancy to stabilize OCR output in videos. They introduced an end-to-end Automatic License Plate Recognition (ALPR) method based on a hierarchical Convolutional Neural Network (CNN), leveraging synthetic and augmented data to enhance recognition rates yielding superior accuracy compared to academic methods and a commercial system on Brazilian and European license plate datasets.

2.3. License plate recognition in Morocco

In [20], the authors introduced a Moroccan license plate recognition system consisting of two steps: hypothesis generation and verification. They utilized the Connected Component Analysis technique (CCAT) to detect rectangles considered as license plate candidates. Then, edge detection was applied within these candidates, followed by the close curves method to confirm their status as license plates and segment the characters. The experiment yielded promising results, with an accuracy of 96.37% when tested on three Moroccan road videos. In addition, authors of [21] proposed a three-phase method. Firstly, license plate localization under various environmental conditions was accomplished through a combination of edge extraction and morphological operations. Secondly, the segmentation process exploited the specific features of Moroccan license plates. Finally, optical character recognition relied on the Tesseract framework, known for its accuracy as an open-source OCR solution. The authors demonstrated the method's ability to recognize multiple license plates in real-time under different acquisition constraints, although no accuracy rate was provided.

Furthermore, [22] presented a robust method for detecting and localizing Moroccan license plates from images. This approach relied on edge features and characteristics of license plate characters. The model's robustness was verified using various images capturing Moroccan license plates from different distances and angles. The experimental results indicated a precision rate of nearly 95% with a recall rate of 81% and a standard quality measure of 87.44%. Moreover, authors of [23] contributed to the

field of Moroccan license plate recognition with their work. They developed a dataset specifically for Moroccan license plate OCR applications, consisting of 705 unique and labeled images collected manually. This dataset is freely available for use and compatible with CNN models like Yolov3.

Finally, a one-stage modified tiny-Yolov3 model was proposed for real-time Moroccan license plate recognition [24], enhanced with transfer learning techniques. This method achieved an excellent balance between speed and accuracy, executing the detection and recognition process in a single phase. It achieved an accuracy of 98.45% and a speed of 59.5 Frames Per Second (FPS).

2.4. Plate detection issues

License plate detection systems face significant challenges in accurately identifying and recognizing license plates in Morocco. The country's license plate specifications, including multiple plate types, various fonts, separators, variable character spacing, partially written Arabic characters, and plate modifications, create complexities that hinder the recognition process. These unique specifications require advanced algorithms and techniques to overcome the challenges and ensure accurate license plate detection and recognition.

➤ Plate formatting (PF)

- Multiple plate types in use :** In Morocco, there are multiple types of license plates used for different purposes. The most common type is the horizontal plate (HWP), which consists of three sections. The first section indicates the prefecture or province to which the vehicle is attached. The second section represents the registration series, which consists of Arabic letters in a specific order. The third section indicates the order of registration with digits. Some vehicles have a two-line plate (VWP), where the first line contains the first and second sections separated by a vertical line, and the second line displays the digits of the third section. Also, specific plates for local authorities (DP), which include an Arabic character indicating the concerned authority and a generic number specific to the vehicle. Diplomatic, consular agents, representatives, experts, and officials of international or regional organizations in Morocco have their own specific plates (YP) divided into two parts. Additionally, there are plates for vehicles related to international cooperation and temporary importation. In addition, new automobiles have special plates for provisional entry into service (WW) and vehicle purchased or sold by automobile dealers (W18). These plates are exclusively delivered by importers, manufacturers, or traders of new motor vehicles to buyers in Morocco. Figure 1 shows the multiple plates in use in the country.

- Multiple fonts in use :** Local lawmakers in Morocco have granted car owners the autonomy to choose the desired font for their license plates. The fonts listed in Table 3 are commonly utilized options. These fonts include Clarendon Regular Extra (CRE), High Security Registration Plate (HSRP), FE-Schrift (FE-S), Ingeborg Heavy Italic font (IHIF), Metalform Gothic JNL font (MGJF), Morton otf (400) (MOTF), and Moroccan Rekika Font (MRF).

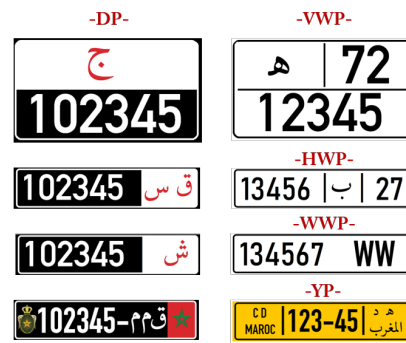


Figure 1: multiple plates in use in Morocco

- Separators:** Different separators are used by Moroccan plate makers. Even though the vertical line is the most typical separator, some constructors use hyphens, while others prefer slashes or don't use separators at all, as shown in figure 2. The configuration of plates is altered by the choice of separators or lack thereof, and this problem persists for plate identification systems.



Figure 2: multiple separators in use in Moroccan plates

- Distance between characters:** Morocco has a large number of plate constructors, thus their plates are not uniform, and the spacing between characters is not constant in all altered plates. The fact that the distance between characters differs from the plates poses a significant problem for LPR systems.
- Arabic letters specifications:** Moroccan license plates use Arabic characters that are partially written. For instance, "ب" has a dot protruding from its body. Additionally, small markings like a "ء" can be used to denote certain characters or they can stand alone (as in the case of the "ا" character). Because dots or short marks are left out, these types of characters cannot be accurately identified during the character segmentation phase [21].
- Plate modification and damage (PMD)**
 - Additions:** Even though local government agencies have waged a significant campaign against extra features on vehicle license plates, some owners still choose to add drawings, logos, stickers, or cameras to plates as presented in figure 3.



Figure 3: Plates with irregular additions

- **Degradation:** In the context of Morocco's developing status, vehicle owners often encounter difficulties in maintaining their license plates in pristine condition. This deterioration can manifest in different forms, such as scratches on metallic plates that resemble lines or additional features that may be mistaken for characters if they possess similar dimensions. Another common issue involves the degradation of character paint, resulting in misleading interpretations or even the omission of altered characters. These damages present a significant challenge for automated license plate recognition systems, as they can adversely affect the accuracy and reliability of the recognition process.

➤ **Visibility and image quality (VIQ)**

- **Illumination:** As the sample presented in figure 4, due to the light shift in the shadow zone, the presence of objects' shadows might present a number of difficulties that must be resolved in order to prevent any false positive detections. We use the median filtering technique to remove specific forms of noise in order to complete this objective.



Figure 4: multiple plates captured with shadow

- **Camera noise:** Issue caused by the sensor used can alter the taken image. Here, we discuss camera vibration, which can amplify blur, as well as weather-related noise. Vehicle position and speed can cause technical degradation by making it challenging to see images or video sequences.

- **Plate inclination and distortion:** Due to various factors such as improper mounting or external forces, license plates can become tilted or distorted, making it difficult for automated systems to accurately capture and interpret the characters. Plate inclination refers to the angle at which the plate is positioned, which can vary significantly from one vehicle to another. This variation in plate inclination introduces complexity in the recognition process, as the characters may appear skewed or slanted, leading to potential recognition errors. Plate distortion can occur due to factors such as physical damage, temperature changes, or poor plate material quality. Distortions may cause the characters to appear stretched, compressed, or warped [25, 26].

3. Proposed solution

3.1. YOLOv5

YOLO is a real-time object detection algorithm that directly predicts bounding box coordinates and class probabilities without the region of interest extraction, resulting in improved detection speed compared to faster R-CNN [27]. The latest version, YOLOv5, introduced by Ultralytics in 2020, outperforms previous versions in both speed and accuracy. Written in Python, YOLOv5 is easier to install and integrate with IoT devices, and it utilizes a new PyTorch framework for training and deployment, enhancing object detection results. During training, YOLOv5 incorporates online data augmentation techniques such as scaling, color space modifications, and mosaic augmentation. YOLOv5 offers four

models, including YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x, by adjusting the width and depth of the backbone network [28]. The backbone, a convolutional neural network, collects and compresses visual features, leveraging the focus structure and stacking image edges to reduce calculations and speed up the process. The CSP1_x and CSP2_x modules split and combine feature mappings, increasing accuracy and reducing calculation time. The SPP network separates contextual features and expands the receptive field. YOLOv5 also includes a neck network that utilizes PANet and FPN structures to fuse and combine features from different layers, improving feature extraction and prediction [29-31]. Figure 5 displays the YOLOv5 network.

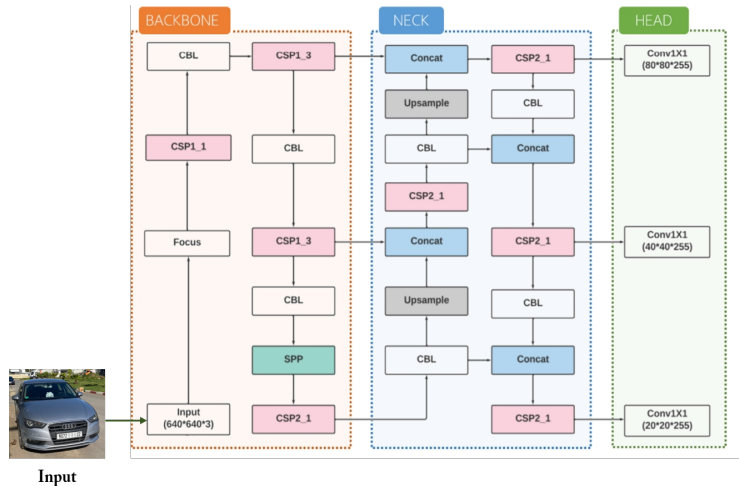


Figure 5: Network topology of YOLOv5s [27]

3.2. Detectron2

Detectron2 is a powerful open-source framework for object detection and instance segmentation [32]. It is built on top of PyTorch [33] and provides a modular and flexible platform for training and deploying state-of-the-art deep learning models. With a focus on research and production deployment, Detectron2 offers a wide range of functionalities and features that enable efficient and accurate object detection tasks. It includes a comprehensive set of pre-trained models and allows users to easily customize and extend the framework to suit their specific needs [32]. The architecture of Detectron2 is based on a modular design, with different components such as backbone networks, feature extractors, and prediction heads [34], allowing for easy experimentation and fine-tuning of models. It also supports distributed training and inference, enabling efficient utilization of resources and scalability [35]. Detectron2 has gained popularity in the computer vision community due to its performance, versatility, and user-friendly interface, making it a valuable tool for researchers and developers working on object detection and instance segmentation tasks. Figure 6 displays the network of Detectron2.

3.3. Model

In light of the aforementioned, the proposed Moroccan Automatic License Plate Recognition System (MALPR), shown in Figure 7, is intended to meet the requirements of the nation while minimizing as many difficulties as possible. It addresses the

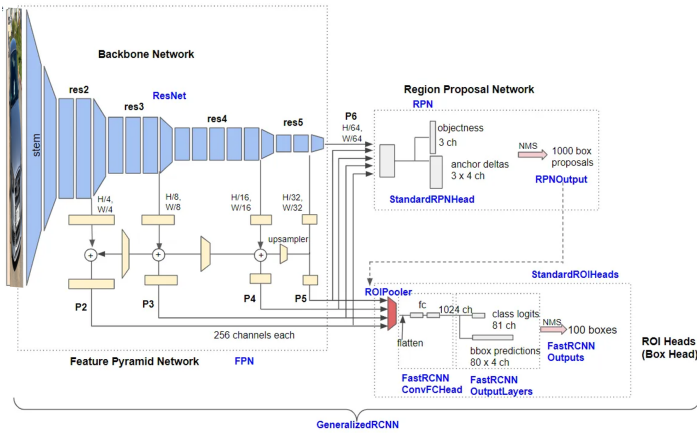


Figure 6: Network topology of Detectron2 [36]

recognition of all types of license plates used in Morocco, including both local and foreign vehicles. There are two main parts to the MALPR system [37]. An SDK-based system with embedded IoT devices like a camera, GPS, and GSM as well as a neural network framework for image analysis constitute the first part of the system. The second one is an API server-side system that carries out further processing, like character segmentation and digits recognition.

The proposed architecture involves capturing a real-time video from a camera and converting it into a specific number of frames per second, based on the deployment location of the device. For instance, if the system is used for detecting parking activity, a low frame rate of one or two frames per second would suffice. However, in areas with higher traffic density, such as highways, a higher frame rate would be necessary to enhance the accuracy of the detection.

Initially, as presented in figure 7, the device performs an analysis of the video and processes the frames to improve their quality and increase the accuracy of predictions through techniques such as compression, greyscale conversion, etc. [37, 38]. Subsequently, the captured vehicle is classified using the neural network framework [10-16] in object detection, vehicle classification, and plate localization within the frame. The better the vehicle classification, the more accurately the plate can be located within the image. Neural network can easily locate the plate with a simple configuration. Once the plate is detected, it is cropped and sent as a binary large object (BLOB). On the server side, the software development kit (SDK) completes the process by performing region segmentation, character detection, and gathering recognized characters to construct the final output. The current architecture extends beyond the steps outlined above.

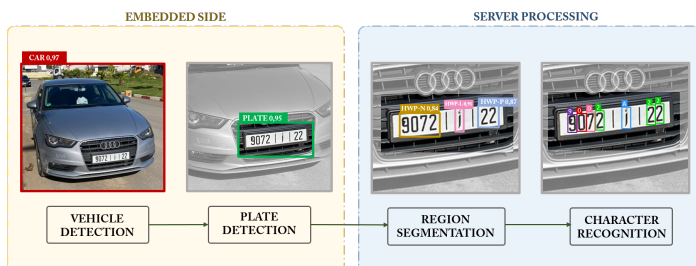


Figure 7: Moroccan License plate overview.

The Moroccan license plate detection system involves several key steps to accurately identify and process license plates in real-time. The system begins by capturing a video feed from a camera and converting it into frames per second based on the location of the device. The frames undergo analysis and processing to enhance their quality and accuracy using techniques like compression and grayscale conversion. The neural network (Yolo, detectron2, CRAFT, etc.) is then employed to classify the captured vehicles and locate the license plates within the frames. The output of this stage consists of the coordinates of the predicted plate as presented in equation (1). Once the plate is detected, it is cropped and sent for further processing.

$$y = (pc, bx, by, bh, bw, c) \tag{1}$$

With bw and bh are the width and height of the rectangle, c stands for the class found and bx and by are the coordinate of the center of the box. pc corresponds to the confidence of the prediction:

$$pc = Pr(\text{Object}) * IoU \tag{2}$$

With IoU corresponds to the area of overlap between the predicted BB and the ground-truth BB [39] which corresponds to the labeled BB from the testing set that specify where is the object.

The system incorporates region segmentation to categorize the license plate into major digit regions, allowing for individual analysis of each part. The plate undergoes color inversion if necessary, and character recognition techniques are applied to extract characters and numbers from separated regions. The recognized characters are assembled to form the final output.

Additionally, the system employs temporal redundancy, where the largest plate within the frame is selected as a basis for prediction. Comparative analysis is conducted with previously stored data to assess the similarity of predicted characters. Plates with higher dimensions and better prediction precision are prioritized, while redundant results are rejected to avoid duplication and optimize efficiency.

Overall, the Moroccan license plate detection system integrates video analysis, plate detection, cropping, region segmentation, character recognition, and temporal redundancy stages to achieve accurate and efficient recognition of license plates, enabling further processing and utilization of the obtained information.

3.4. Improvements

Temporal redundancy. The incorporation of a temporal redundancy stage within the proposed license plate recognition architecture plays a pivotal role in elevating the accuracy rates and optimizing the utilization of the entire frame. This stage serves as a crucial component that effectively verifies the output and facilitates a comparative analysis with previously obtained results, ultimately leading to the refinement and advancement of the model's overall performance.

To ensure a comprehensive evaluation, the model begins with meticulously selecting the output derived from the largest plate within the frame. The rationale behind this selection lies in the inherent understanding that plates with larger dimensions often provide a more accurate basis for prediction. Remarkably,

empirical evidence has demonstrated that this approach can lead to an impressive increase in the accuracy of the proposed model by up to 20% [19]. Such an improvement is substantial, indicating the crucial role played by this temporal redundancy stage in the enhancement of recognition capabilities.

In recognizing the vast number of vehicles traversing Moroccan roads, estimated to exceed an impressive count of 4 million, it becomes evident that the likelihood of detecting multiple plates simultaneously with similar digit patterns, occupying identical positions, and preserving the same order is exceedingly low. Therefore, it becomes viable to employ the system to compare the predicted characters from the cropped license plate to an existing collection of previously conserved data, in order to ascertain their level of similarity and make informed decisions based on such assessments.

The initial step involves verifying the presence of an existing output within the redundancy collection for comparative purposes. If an older output is identified, the model proceeds to scrutinize whether the predicted license plate corresponds to the same plate as the last output stored within the redundancy collection. This verification process is accomplished through the establishment of a similarity rate, serving as a threshold to determine the degree of resemblance between the predicted characters and the existing collection.

Subsequently, in instances where the similarity rate surpasses the pre-established threshold, confirming that the detected characters unequivocally belong to the same license plate, a thorough comparison ensues to ascertain the most relevant output. Notably, the determination of relevance primarily revolves around the consideration of plate dimensions and the level of prediction precision. As shown in figure 8, plates with larger dimensions are inherently attributed with a higher potential for providing more accurate predictions, thereby warranting their prioritization within the selection process.

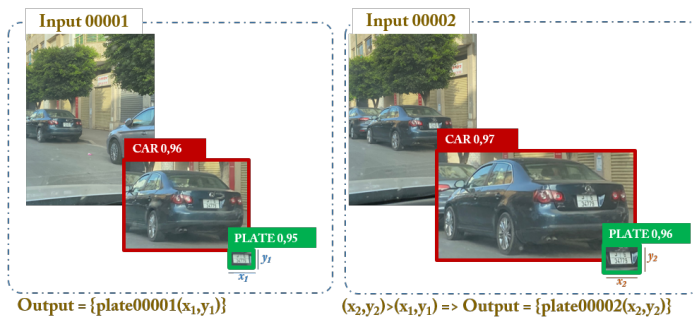


Figure 8: Temporal Redundancy optimization sample.

However, in scenarios where the output unequivocally replicates a redundant result with a similarity rate of 100%, it indicates that the information has already been transmitted to the embedded side. Consequently, to avoid redundancy and unwarranted repetition, such results are promptly rejected, effectively curtailing the transmission of duplicate outputs to the API side.

The temporal redundancy stage constitutes a critical juncture within the license plate recognition system, serving as a formidable mechanism for bolstering accuracy rates. By incorporating comparative analyses with previously obtained results, the system

can intelligently discern the most relevant output, leveraging the dimensions of the license plate and the level of prediction precision as key criteria. In doing so, the system avoids the needless transmission of duplicate information to the API side, thus optimizing the overall efficiency of the recognition process.

Algorithm 1: Temporal redundancy

```

Algorithm temporalRedundancyCheck(newOutput,
redundancyCollection)
largestPlateOutput <- selectLargestPlateOutput(newOutput)
if redundancyCollection.isEmpty() then
    redundancyCollection.add(largestPlateOutput)
    return largestPlateOutput
end if
existingOutput <- redundancyCollection.getLastOutput()
similRate <- calculateSimilRate(largestPlateOutput, existingOutput)
if similRate > threshold then
    if similRate = 100 then
        return "Redundant output"
    else
        if largestPlateOutput.dimensions > existingOutput.dimensions
then
            redundancyCollection.remove(existingOutput)
            redundancyCollection.add(largestPlateOutput)
            return largestPlateOutput
        else
            return existingOutput
        end if
    end if
else
    redundancyCollection.add(largestPlateOutput)
    return largestPlateOutput
end if
End Algorithm
    
```

Region segmentation. The proposed model makes a significant addition to the field of region segmentation. Before extracting each character, it divides the license plate into its major digit sections. This step is essential to ensuring that every component of the plate is examined separately and no area is missed. The regions are created to be as large as feasible during the training phase in order to support all potential typefaces during the test phase. In the second stage of isolating the digits for analysis, extreme care is used to prevent overlapping regions. This model does not take separators into account, such as the lengthy Arabic letter "أ" and does not divide the alphabet into discrete pieces, in contrast to other Moroccan license plate recognition systems. Figure 9 presented the region segmentation.

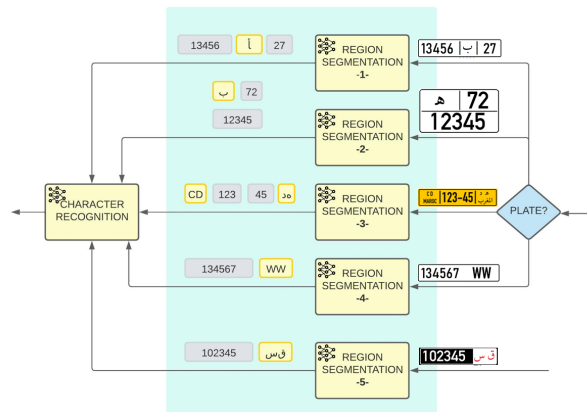


Figure 9: Region segmentation overview

Image processing : Stage at the beginning of the workflow where the video input is processed in order to remove small components and noises, to increase the quality necessary for further operations (binarization, contrast maximization, Gaussian blur filter and Adaptive threshold) [40] as well as to reduce the computation cost.

- **Image binarization** consist on converting the frame to black and white [40, 41]. And essential processing for the digits detection where the input is considered as a collection of subcomponents (text, background and picture). To each pixel, a local threshold is set in order to reduce noise, illumination issues and source type-related degradations.
- **Thresholding** is used in the aim of classifying dark pixels as black and others as white. The most important is to set the optimal threshold value for particular image, hence the use of adaptive thresholding. A threshold can be selected by user manually or it can be selected by an algorithm automatically which is known as automatic thresholding. Also called automatic tresholding, this mode is used to set automatically the threshold value by an algorithm when it becomes difficult or almost impossible to select optimal one [42].
- **Histogram Equalization (HE)** is the method that standardize the intensities of the input image by spreading out the most recurrent ones, which results in an improved contrast. Figure 10 shows the impact of applying the histogram equalization technique on a vehicle image. Accordingly, HE equalize the light intensity and improve dark and indistinguishable license [15].



Figure 10: Vehicle image before and after HE optimization.

4. Experiment and results

4.1. Dataset

Machine learning systems require a ivolume of high-quality data. Although efforts have been made to solve the issue of license plate detection, it is still difficult to identify license plates in uncontrolled and unconstrained contexts. In fact, whether attempting to detect license plates that are rotated, in uneven illumination, in snowy conditions, or in a dimly lit environment, the majority of offered algorithms have low accuracy. The vast majority of researchers have trained and tested their detectors on really short datasets, which only contain a small number of unique

photos or modest variations in angles, limiting their usefulness to particular scenarios.

In fact, a unique dataset is created based on the plate types (HWP, VWP, YP, WWP, and DP) and fonts used (CRE, HSRP, FE-S, IHIF, MGJF, MOTF, and MRF) to test the proposed method. This dataset, shown in figure 11, consists of 11617 photos of unique vehicles taken from Moroccan roads in a variety of settings (including location, time, rotation, background, lighting, and weather). These photos are organized based on the plate style and font used. After being cropped, each annotated plate is segmented (HWP-N, HWP-P, HWP-L, etc.). This serves as a segmentation data that may be trained on and annotated.

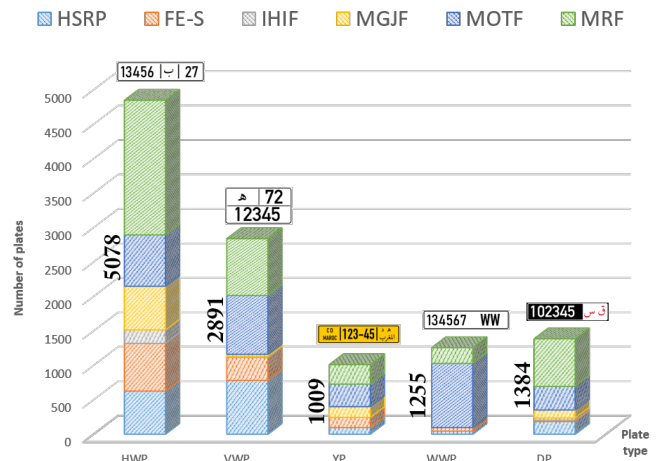


Figure 11: Dataset composition (fonts and plate types).

Also, specific folders are built to test the present model on problematic plates: Plates Formating (PF) having plates with difference distance between characters with almost all fonts in use and with several types of separators, Visibility and Image Quality (VIQ), having plates with illumination and shadowing issues, pictures with camera noise and plates under different inclinasion. Also a folder named (PMD) Plate Modification and Damage is built with plates with degradation and those having additions to test the robustness of the model to deal with these issues.

4.2. Training

The dataset is labeled using Labelling [43]. This desktop software analyzes the image annotation process for training Deep learning models in modern image recognition systems. This tool creates a classes.txt file and saved annotation with the following structure, with the first character corresponding to the order of the class in class.txt. The next four values are the coordinates of the BB annotated.

To perform the training we used NVIDIA GeForce RTX 3070 (total memory 16196 MB) build on AMD Ryzen 9 3900XT 12-Core Processor computer with 16384 MB RAM and Windows 10 Pro N 64-bit (10.0, Build 19045). The model was built with Python-3.9.13, torch-1.9.1+cu111 CUDA:0.

The usual approach of the presented model involves categorizing outcomes into four cases denoted by T and F, representing true or false predictions. The letters P and N indicate whether the instance is expected to be part of a positive or negative class. Evaluating the model's performance involves examining the

distribution of these prediction outcomes, which can be formed by various combinations of these categories. To gauge the model's precision, the following metrics are commonly employed.

$$Accuracy (A) = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (3)$$

$$Precision (P) = \frac{TP}{TP + FP} \times 100\% \quad (4)$$

$$Recall (R) = \frac{TP}{TP + FN} \times 100\% \quad (5)$$

$$F_1 = 2 \times \frac{P \times R}{P + R} \quad (6)$$

$$mAP = \frac{1}{N(T)} \sum_{r \in T} AP_r \quad (7)$$

True positive (TP) refers to a test result that correctly detects the presence of a certain characteristic, while true negative (TN) indicates accurate identification of the absence of the region or character. False positive (FP) occurs when a result wrongly suggests the presence of a specific region or character, and false negative (FN) represents a test result that falsely indicates the absence of a particular condition or attribute. Figure 12 illustrates three types of losses: classification loss, objectness loss, and box loss. The box loss evaluates the algorithm's ability to accurately determine an object's center and how well the estimated bounding box encompasses the object. Objectness measures the probability of an object's existence within a suggested region of interest. A higher objectness indicates a higher likelihood of an object being present in the image window. Classification loss demonstrates how effectively the algorithm can assign the correct class to a given object. The model's precision, recall, and mean average precision show significant improvement initially, reaching stability around 50 epochs. The validation box, objectness, and classification losses also showed a noticeable decrease until approximately epoch 50. We employed early stopping to select the best weights for the model.

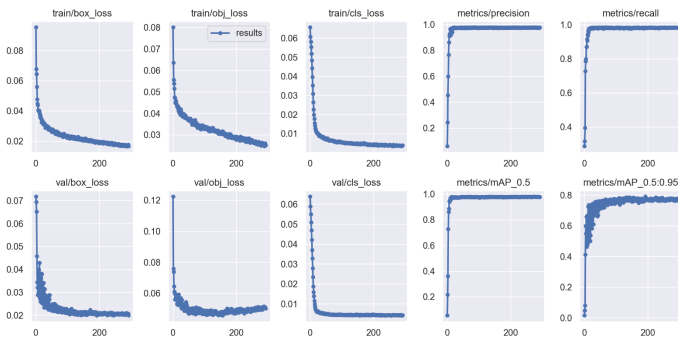


Figure 12: Plots of box loss, objectness loss, classification loss, precision, recall and mean average precision (mAP) over training and validation epochs.

4.3. Result and discussion

After performing precision of 95.492%, a recall of 98.259% and mAP 50% up to 97.768% in training, the model performed excellent rates on problematic dataset. In fact, as shown in figure

www.astesj.com

13, almost all PF images were detected and correctly predicted and the model showed very good results when tested on VIQ and PMD datasets with little advantage to Yolov5 over Deteron2. The average speed of all detection stages (vehicle detection, plate type, plate segmentation, and plate characters) is up to 135.3ms when run under experimentation configuration.

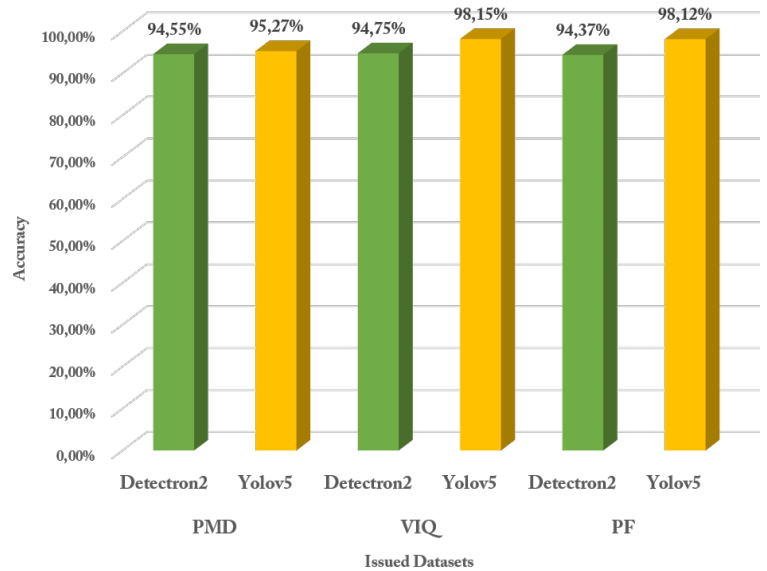


Figure 13: Model precision on issued datasets.

By implementing the described architecture, the model has effectively addressed the aforementioned issues. The current model guarantees the following outcomes:

- Accurate recognition of characters confined within regions, addressing the problem of separators.
- Elimination of false recognition of separators as characters, specifically the Arabic letter "ا".
- Proper identification of the desired result from region segmentation, ensuring that letters are not misidentified as numbers (such as "1" and Arabic letter "أ", "و" and the number 9, etc.).
- Enhanced recognition of Arabic characters "ب", "ت", "س", and "ش" that often have separated parts (dots), through optimized contour finding solutions.
- Improved handling of repeated inputs by considering the most precise result obtained from the most relevant frame.
- Optimization of the training dataset to include a wide range of fonts used on license plates, degraded digits, variations in shadows, and different lighting conditions.
- Capability to add new license plates (foreign plates) by incorporating new categories in plate detection (HWP, VWP, YP, WWP, DP, FRP, GERP, etc.).

5. Conclusion

Based on the aforementioned considerations, the increasing number of vehicles on the roads has led to a growing demand for a reliable and versatile license plate recognition system. This requirement is shared not only in Morocco but also in other regions, where local governments, government agencies, and private businesses seek a robust License Plate Recognition (LPR) system that can effectively handle the specific plate specifications (HWP,

VWP, DP, YP, and WWP) and typefaces used by plate manufacturers. To address this need, a CNN-based intelligent LPR system is introduced in this study. The system was trained on diverse datasets focusing on multiple fonts (CRE, HSRP, FE-S, etc.) and environmental factors such as illumination, climate, and lighting conditions and tested using both Yolov5 and Detectron2 frameworks. Notably, the model incorporates an intelligent region segmentation stage that adapts to the specific type of license plate. This segmentation process greatly enhances recognition precision and successfully resolves previous separator-related challenges. The results obtained from testing the trained model indicate its exceptional capability in accurately identifying automobiles, license plates of various types and fonts, as well as individual digits and plate components. Specifically, the model achieved an average precision rate of 97,18% when tested on issued plates.

Data availability statement

The data used in this study was collected manually by the authors and sorted by type of plate: WWP, VWP, YP, and DP. After further analyzing the collected pictures, the data was sorted into four folders: PDEG, PDI, PSS, and PADD. A part of this datasets used in the analysis will be available upon request by contacting the corresponding author.

Conflict of interest

The authors of this manuscript declare that they have no financial or personal relationships with other people or organizations that could inappropriately influence their work. The authors confirm that this article is original, has not already been published in any other journal, and is not currently under consideration by any other journal. The authors also confirm that all the data presented in this manuscript are original and authentic.

References

- [1] C. Patel, D. Shah, A. Patel, "Automatic Number Plate Recognition System (ANPR): A Survey," *International Journal of Computer Applications*, **69**(9), 21–33, 2013, doi:10.5120/11871-7665.
- [2] S. Bouchelaghem, M. Omar, "Reliable and secure distributed smart road pricing system for smart cities," *IEEE Transactions on Intelligent Transportation Systems*, **20**(5), 1592–1603, 2019, doi:10.1109/TITS.2018.2842754.
- [3] S.S. Omran, J.A. Jarallah, "Iraqi car license plate recognition using OCR," in: Baghdad, I., ed., in 2017 Annual Conference on New Trends in Information and Communications Technology Applications, NTICT 2017, 298–303, 2017, doi:10.1109/NTICT.2017.7976127.
- [4] K. Yogheedha, A.S.A. Nasir, H. Jaafar, S.M. Mamduh, "Automatic Vehicle License Plate Recognition System Based on Image Processing and Template Matching Approach," 2018 International Conference on Computational Approach in Smart Systems Design and Applications, ICASSDA 2018, 1–8, 2018, doi:10.1109/ICASSDA.2018.8477639.
- [5] C. Bila, F. Sivrikaya, M.A. Khan, S. Albayrak, "Vehicles of the Future: A Survey of Research on Safety Issues," *IEEE Transactions on Intelligent Transportation Systems*, **18**(5), 1046–1065, 2017, doi:10.1109/TITS.2016.2600300.
- [6] S.K. Lakshmanaprabu, K. Shankar, S. Sheeba Rani, E. Abdulhay, N. Arunkumar, G. Ramirez, J. Uthayakumar, "An effect of big data technology with ant colony optimization based routing in vehicular ad hoc networks: Towards smart cities," *Journal of Cleaner Production*, **217**, 584–593, 2019, doi:10.1016/j.jclepro.2019.01.115.
- [7] M. Soyuturk, K.N. Muhammad, M.N. Avcil, B. Kantarci, J. Matthews, "From vehicular networks to vehicular clouds in smart cities," *Morgan Kaufmann*: 149–171, 2016, doi:10.1016/B978-0-12-803454-5.00008-0.
- [8] Q. Wang, "License plate recognition via convolutional neural networks," *Conf. Softw. Eng. Serv. Sci. ICSESS*, 2017-November, 2018, doi:10.1109/ICSESS.2017.8343061.
- [9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, A.C. Berg, "SSD: Single shot multibox detector," *CVPR* 2. tesseract-ocr, 2016, doi:10.1007/978-3-319-46448-0_2.
- [10] X. Hou, M. Fu, X. Wu, Z. Huang, S. Sun, "Vehicle license plate recognition system based on deep learning deployed to PYNQ," *ISCIT 2018 - 18th International Symposium on Communication and Information Technology*, 2018, 422–427, 2018, doi:10.1109/ISCIT.2018.8587934.
- [11] J. Singh, B. Bhushan, "Real Time Indian License Plate Detection using Deep Neural Networks and Optical Character Recognition using LSTM Tesseract," in *Proceedings - 2019 International Conference on Computing, Communication, and Intelligent Systems, ICCIS 2019, Communication, and Intelligent Systems (ICCIS)*, Greater Noida, India: 347–352, 2019, doi:10.1109/ICCIS48478.2019.8974469.
- [12] Hendry, R.C. Chen, "Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning," *Image and Vision Computing*, **87**, 47–56, 2019, doi:10.1016/J.IMAVIS.2019.04.007.
- [13] S.M. Silva, C.R. Jung, "License plate detection and recognition in unconstrained scenarios," *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11216 LNCS, 593–609, 2018, doi:10.1007/978-3-030-01258-8_36.
- [14] S. Alghalyne, "Real-time Jordanian license plate recognition using deep learning," *Journal of King Saud University - Computer and Information Sciences*, 2020, doi:10.1016/j.jksuci.2020.09.018.
- [15] A. Tourani, A. Shahbahrami, S. Soroori, S. Khazaei, C.Y. Suen, "A robust deep learning approach for automatic Iranian vehicle license plate detection and recognition for surveillance systems," *IEEE Access*, **8**, 201317–201330, 2020, doi:10.1109/ACCESS.2020.3035992.
- [16] M. Rahmani, M. Sabaghian, S.M. Moghadami, M.M. Talaie, M. Naghibi, M.A. Keyvanrad, "IR-LPR: A Large Scale Iranian License Plate Recognition Dataset," in 2022 12th International Conference on Computer and Knowledge Engineering, ICCKE 2022, Mashhad, Iran, 0: 53–58, 2022, doi:10.1109/ICCKE57176.2022.9960129.
- [17] Y. Yang, W. Zhang, Z. He, D. Li, "High-speed rail pole number recognition through deep representation and temporal redundancy," *Neurocomputing*, **415**(16), 201–214, 2020, doi:10.1016/j.neucom.2020.07.086.
- [18] W. Riaz, A. Azeem, G. Chenqiang, Z. Yuxi, Saifullah, W. Khalid, "YOLO Based Recognition Method for Automatic License Plate Recognition," in *Proceedings of 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2020, 1109/AEECA49918.2020.9213506: 87–90, 2020*, doi:10.1109/AEECA49918.2020.9213506.
- [19] S.M. Silva, C.R. Jung, "Real-time license plate detection and recognition using deep convolutional neural networks," *Journal of Visual Communication and Image Representation*, **71**, 10277, 2020, doi:10.1016/j.jvcir.2020.102773.
- [20] I. Slimani, A. Zaarane, A. Hamdoun, I. Atouf, "Vehicle license plate localization and recognition system for intelligent transportation applications," 2019 6th International Conference on Control, Decision and Information Technologies, CoDIT 2019, **2019**(6), 1592–1597, 2019, doi:10.1109/CoDIT.2019.8820446.
- [21] F. Zahra Taki, A. El Belrhiti El Alaoui, "Moroccan License Plate recognition using a hybrid method and license plate features," *ResearchGate*, (March), 2018.
- [22] H. Anoual, S. El Fkhi, A. Jilbab, D. Aboutajdine, "Vehicle license plate detection in images," *International Conference on Multimedia Computing and Systems -Proceedings*, 2011, doi:10.1109/ICMCS.2011.5945680.
- [23] A. Alahyane, M. El Fakir, S. Benjelloun, I. Chairi, "Open data for Moroccan license plates for OCR applications : data collection, labeling, and model construction," *Dec*, **19**(7), 146–158, 2021, doi:10.48550/arXiv.2104.08244.
- [24] A. Fadili, M. E. Aroussi, and Y. Fakhri, "A one-stage modified Tiny-YOLOv3 method for real-time Moroccan license plate recognition," *International Journal of Computer Science and Information Security (IJCSIS)*, **19**(7), 146–158, Jul. 2021. doi: 10.5281/zenodo.5164655.
- [25] C.H. Lin, C.H. Wu, "A Lightweight, High-Performance Multi-Angle License Plate Recognition Model," in *International Conference on Advanced Mechatronic Systems, ICAMechS, 09/ICAMechS.2019.8861688: 235–240, 2019*, doi:10.1109/ICAMechS.2019.8861688.
- [26] D. Xiao, L. Zhang, J. Li, J. Li, "Robust license plate detection and recognition with automatic rectification," *Journal of Electronic Imaging*, **30**(01), 1, 2021, doi:10.1117/1.jei.30.1.013002.
- [27] G. Jocher, A. Stoken, J. Borovec, NanoCode012, et al., "ultralytics/yolov5: v3.0," 2020, doi:10.5281/zenodo.3983579.

- [28] K. He, X. Zhang, S. Ren, J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," arXiv, 2014, doi:10.1007/978-3-319-10578-9_23.
- [29] W. Jia, S. Xu, Z. Liang, Y. Zhao, H. Min, S. Li, Y. Yu, "Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector," IET Image Processing, **15**(14), 3623–3637, 2021, doi:10.1049/ipr2.12295.
- [30] X. Xu, Y. Jiang, W. Chen, Y. Huang, Y. Zhang, X. Sun, "DAMO-YOLO : A Report on Real-Time Object Detection Design," 2022.
- [31] K. Wang, J.H. Liew, Y. Zou, D. Zhou, J. Feng, "PANet: Few-shot image semantic segmentation with prototype alignment," Proceedings of the IEEE International Conference on Computer Vision, 2019-October, 9196–9205, 2019, doi:10.1109/ICCV.2019.00929.
- [32] A. Venkata, S. Abhishek, S. Kotni, "Detectron2 Object Detection & Manipulating Images using Cartoonization," Article in International Journal of Engineering and Technical Research, 2022.
- [33] S. Imambi, K.B. Prakash, G.R. Kanagachidambaresan, "PyTorch," Springer International Publishing, Cham: 87–104, 2021, doi:10.1007/978-3-030-57077-4_10.
- [34] V. Pham, C. Pham, T. Dang, "Road Damage Detection and Classification with Detectron2 and Faster R-CNN," Proceedings - 2020 IEEE International Conference on Big Data, Big Data 2020, 5592–5601, 2020, doi:10.1109/BIGDATA50022.2020.9378027.
- [35] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, R. Girshick, "Detectron2: A PyTorch-based modular object detection library," in Meta AI, 1–10, 2019, doi:10.1007/978-3-030-58553-6_49.
- [36] M. Ackermann, D. Iren, S. Wesselmecking, D. Shetty, U. Krupp, "Automated segmentation of martensite-austenite islands in bainitic steel," Materials Characterization, **191**, 2022, doi:10.1016/j.matchar.2022.112091.
- [37] S. Du, M. Ibrahim, M. Shehata, W. Badawy, "Automatic license plate recognition (ALPR): A state-of-the-art review," IEEE Transactions on Circuits and Systems for Video Technology, **23**(2), 311–325, 2013, doi:10.1109/TCSVT.2012.2203741.
- [38] H. Rezaatoughi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2019-June, 658–666, 2019, doi:10.1109/CVPR.2019.00075.
- [39] G.R. Gonçalves, D. Menotti, W.R. Schwartz, "License plate recognition based on temporal redundancy," IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, 2577–2582, 2016, doi:10.1109/ITSC.2016.7795970.
- [40] H. Michalak, K. Okarma, "Improvement of image binarization methods using image preprocessing with local entropy filtering for alphanumerical character recognition purposes," Entropy, **21**(6), 1–18, 2019, doi:10.3390/e21060562.
- [41] M.R. Gupta, N.P. Jacobson, E.K. Garcia, "OCR binarization and image preprocessing for searching historical documents," Pattern Recognition, **40**(2), 389–397, 2007, doi:10.1016/j.patcog.2006.04.043.
- [42] P. Roy, S. Dutta, N. Dey, G. Dey, S. Chakraborty, R. Ray, "Adaptive thresholding: A comparative study," 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies, ICCICCT 2014, 1182–1186, 2014, doi:10.1109/ICCICCT.2014.6993140.
- [43] L.Y. Chan, A. Zimmer, J.L. Da Silva, T. Brandmeier, "European Union Dataset and Annotation Tool for Real Time Automatic License Plate Detection and Blurring," 2020 IEEE 23rd International Conference on Intelligent Transportation Systems, ITSC 2020, 2020, doi:10.1109/ITSC45102.2020.9294240.