information

Article

Intelligent Video Surveillance Systems for Vehicle Identification Based on Multinet Architecture

Jacobo González-Cepeda *, Álvaro Ramajo and José María Armingol

Department of Electric, Electronic and Automatic Engineering, Carlos III University, 28912 Madrid, Spain; aramajo@pa.uc3m.es (Á.R.); armingol@ing.uc3m.es (J.M.A.)
* Correspondence: 100307736@alumnos.uc3m.es

Abstract: Security cameras have been proven to be particularly useful in preventing and combating crime through identification tasks. Here, two areas can be mainly distinguished: person and vehicle identification. Automatic license plate readers are the most widely used tool for vehicle identification. Although these systems are very effective, they are not reliable enough in certain circumstances. For example, due to traffic jams, vehicle position or weather conditions, the sensors cannot capture an image of the entire license plate. However, there is still a lot of additional information in the image which may also be of interest, and that needs to be analysed quickly and accurately. The correct use of the processing mechanisms can significantly reduce analysis time, increasing the efficiency of video cameras significantly. To solve this problem, we have designed a solution based on two technologies: license plate recognition and vehicle re-identification. For its development and testing, we have also created several datasets recreating a real environment. In addition, during this article, it is also possible to read about some of the main artificial intelligence techniques for these technologies, as they have served as the starting point for this research.

Keywords: surveillance systems; vehicle re-identification; ALPR; real-time applications; deep learning algorithms

check for updates

Citation: González-Cepeda, J.; Ramajo, Á.; Armingol, J.M. Intelligent Video Surveillance Systems for Vehicle Identification Based on Multinet Architecture. *Information* **2022**, *13*, 325. https://doi.org/10.3390/info13070325

Academic Editors: Danilo Avola, Daniele Pannone and Alessio Fagioli

Received: 5 May 2022
Accepted: 2 July 2022
Published: 6 July 2022

1. Introduction

Surveillance systems are only one part of a complete security solution, which also includes several elements such as physical barriers or deterrents. Surveillance systems consist of different components such as video cameras, detectors or different IoT devices, such as trackers or smart sensors. This topic is extremely wide, and there are many areas that can be investigated in detail. In fact, it is possible to find several avenues of research from different points of view, such as [1] or [2], which provide an overview of the concept of surveillance systems.

Intelligent video surveillance systems would be a specific type of system, whose functionality is based on the combined use of images with signal processing methods. In particular, intelligent video surveillance is an important topic within video processing and computer vision, because of its potential in very different fields such as military, health prevention or public security [3,4]. In this case, this article will focus on vehicle identification that is applied to security by the means of intelligent video surveillance systems. In particular, a specific solution that tries to combine different capabilities will be presented, with the ultimate purpose of being robust against variable conditions.

1.1. Structure

This article has two purposes: the first one, as mentioned above, is to design and develop a complete security solution for vehicle identification (combining re-identification methods with license plate reading) under variable operating conditions. For this, we have first attempted to identify these variable conditions and improvement areas. Then,

work has been performed directly on the signal processing system, focusing on artificial intelligence techniques such as those described in the following paragraphs (mainly deep learning) and combining them in a multinet solution. The second purpose is to show different methods and solutions that we consider (through experience) to be the most practical, optimal and updated, and compile them in a small survey. We want to collect what we consider to be the most applicable solutions for real environments in relation to these specific tasks (vehicle re-identification and license plate reading). It is important to highlight that these methods and solutions have served as a starting point for this research, and have helped with thinking about possible solutions.

This will be the paper's structure. Right after this paragraph, we will define some concepts that are important for setting the framework. Section 2 will identify the state-of-the-art in license plate readers and different methods. Section 3 will be similar to Section 2, but concerning the collection of vehicle re-identification methods. In Section 4, certain considerations for designing a solution will be detailed. In Section 5, our own solution will be detailed, based on a combination of the methods described above. Finally, certain conclusions will be outlined.

1.2. Video Surveillance System: Definition

An artificial vision system can be defined as a set of elements that are designed to capture, process, store and transmit information collected in frames [5]. There are many different types, depending on their configuration, purposes or capabilities.

A video surveillance system is an instrument that is designed to have visual control over a certain place, continuously (that is, through a succession of frames), reinforcing the security of the site under control [6]. It would be a specific type of vision system.

To delve deeper into some of the defining characteristics of a video surveillance system, it is first necessary to know its general components, as this will help to better understand the different areas of improvement. They consist of a lighting mechanism, optics, capture sensor, signal processing system, storage and transmission [7]. The presence of all these elements is not compulsory. The purpose of the system will condition its design. For example, a simple camera can itself constitute a vision system, consisting of lighting, optics, capture sensor and storage.

1.3. Video Surveillance System: Uses

A video surveillance system could simply seem to be just a camera for monitoring a scenario. However, there are multiple factors that need to be considered, in order to design the most suitable system. The challenge is to identify all these factors and to manage all of them at the same time.

The first and the most important one is to define the purpose of the system. For example, it would not be the same if the camera is intended for traffic control, compared to robbery prevention. In the first case, the focus would be on cars; in the second one, the aim would be to collect as much detail as possible (such as license plates or faces).

The second factor is the study of the operational circumstances. This term refers to identifying whether our system, for example, will work only during the day, at night or over 24 h; if it will be installed inside or outside, or if we can have an electric grid connection or only a battery supply.

Last but not least is to determine whether or not our system will operate in real time. This is very important, because it will not only affect image transmission, but also the image processing method.

From the point of view of security, cameras have a dual purpose. On the one hand, they act as a deterrent. On the other hand, they are a fundamental element for investigations and crime solving. They are one of the main tools that are used to collect evidence that can be brought to court proceedings.

Precisely because of its wide variety of uses and configurations, the first step is to be clear about the targets. An excellent example can be a public car park. It is common to find

several types of cameras within these scenarios. Some of them are focused on entry and exit access, and on capturing the license plates of vehicles (as shown in Figure 1). Thus, they record the number of parked vehicles and the time spent inside the parking lot. The rest of the cameras are usually used for prevention, to avoid theft or damage to vehicles.

Figure 1. Example of an access control camera for license plate reading [8].

Focusing on the first types of cameras, they will need the capability for identifying and registering the license plates of the vehicles entering and leaving. To this end, all elements of the system shall be installed to obtain images of license plates with the highest possible quality and contrast.

In this specific scenario, the signal processing system will recognise the license plates, extract the characters, store them and register the different vehicles. The rest of the elements of the system will enable this task. Hence, the camera and lighting will be large, visible, and they will focus directly on the license plate. In addition, the work of acquiring the frames of the license plate is easier, thanks to the fact that the car must remain stopped until the process has been completed.

Most of security systems operate similar to this previous one. They are designed for controlled environments with static cameras installed, accompanied by large optics (which contribute to a better resolution in the image) and lighting systems that are designed to eliminate any type of aberration. However, these ideal circumstances are not always possible, such as, for example, on a highway.

1.4. Motivation: Identifying the Scenario and Looking for Solutions

The specific scenario in the use of video surveillance systems will be vehicle identification and tracking. The vast majority of commercial systems focus mainly on license plates. This is why most of them employ specific cameras to facilitate signal processing. However, there are many situations where these cameras do not work properly.

These situations can be mainly due to two factors: vehicles position in images and lightning conditions. The first one directly affects the information available in the frames obtained. For example, when a vehicle exceeds the speed limit while it is overtaking another car, a situation can happen where the speed trap captures both vehicles and it is impossible to obtain all of the characters of the license plate corresponding to the offender. Or, when a camera is monitoring a certain place such as a roundabout or a corner, it may happen that the sensors capture the image of the target car, but that the frame does not include the license plate.

The second scenario occurs when weather or lightning conditions affect the capacities of the sensors. By law, European cars have license plate backlighting. Although this was thought to facilitate license plate reading, it is very common in the evening or at night that this indirect light can burn the images captured, causing the opposite effect.

As previously mentioned, cameras are fundamental for investigation and crime solving. That being said, all of the information provided can be critical. A perfect example may be a robbery that is recorded by a security camera. Due to the above-mentioned scenarios, this camera may have captured images of the vehicle used by the offenders, but not its license plate. In this case, we have very important but incomplete information. Although this camera has not been able to read the license plate, maybe there is another one that

could obtain the information in time. Without an automatic image processing tool, this may take hours or days, and time is a vital resource. The later that data is analysed, the less chance there is to capture the authors.

Some systems try to solve these problems by offering additional capacities such as detecting the brand or the colour of the cars (such as, for example, OpenALPR). This could be very useful, but it is not enough in some scenarios where, for example, we want to detect a certain vehicle while it drives along a motorway, and we need to determine where it leaves the motorway. On the other hand, many traffic control cameras do not have enough resolution to detect a car with just its license plate.

Vehicle tracking for security requires a high standard of precision. In fact, in critical situations where there is no margin of error, there will be always an officer verifying the information. A problem appears when there is a large amount of information to analyse. Linking with the robbery example, traditional license plate readers would not be useful because we do not have the license plate characters, and pure vehicle re-identification systems may provide too much information. However, a combination of both elements can guide the investigation faster because it would be possible to both synthesise the images of cars that can be matched to our target, and to know the license plate data. This is the main advance compared to other different solutions shown here that act in isolation.

Through this research, we want to look for a robust solution that not only improves license plate reading capacities, but also one that can be applied in a wide range of scenarios and that satisfies real needs in security systems that are used for vehicle identification.

1.4.1. Two-Factor Authentication

Two-factor authentication consists of the use of two different elements that are combined to identify something unambiguously. It is widely used for control access [9–12]. For example, some banking operations require both username/password and biometrics.

This concept can also be applied in vehicle identification. What happens if we are tracking certain cars through cameras, and we do not have a correct image of the license plate? Maybe, we can obtain just a partial image, but we can perfectly distinguish the shape of the vehicle. Thus, the idea is to employ a two-factor authentication system in vehicle recognition, considering both the license plate and the image of the target vehicle (not only the brand, model or colour). This means that our system will not discard a vehicle if it is not able to read the entire license plate correctly.

1.4.2. Identification, Recognition and Re-Identification

In security, depending on the aim pursued, there are four levels of "precision", called the "DORI concept" [13] (acronym for detection, observation, recognition and identification). This standard, included in the IEC EN62676-4: 2015 International Standard, defines the resolution in pixels that an image must have, so that the detection, observation, recognition and identification of objects/people in images can be conducted in it. It is usually included by surveillance cameras manufacturers in their datasheets.

These terms are directly related to their size within a frame, and in turn, to their distance from the sensor. As we can see in Figure 2, the closer the target is to the sensors, the more information we have regarding it, which implies a shift between the different terms mentioned.

Detection (Figure 2a) is the ability of the system to capture some movement or event. This would be the first step to "activate" the perimetral security. It can be associated with a motion detection system. In fact, it is usually combined with physical sensors or with movement detection mechanisms (such as event alerts caused by pixel variations, or by infrared detectors). Using the first image as an example, detection would be the capability to detect a new active element; in this case, a person.

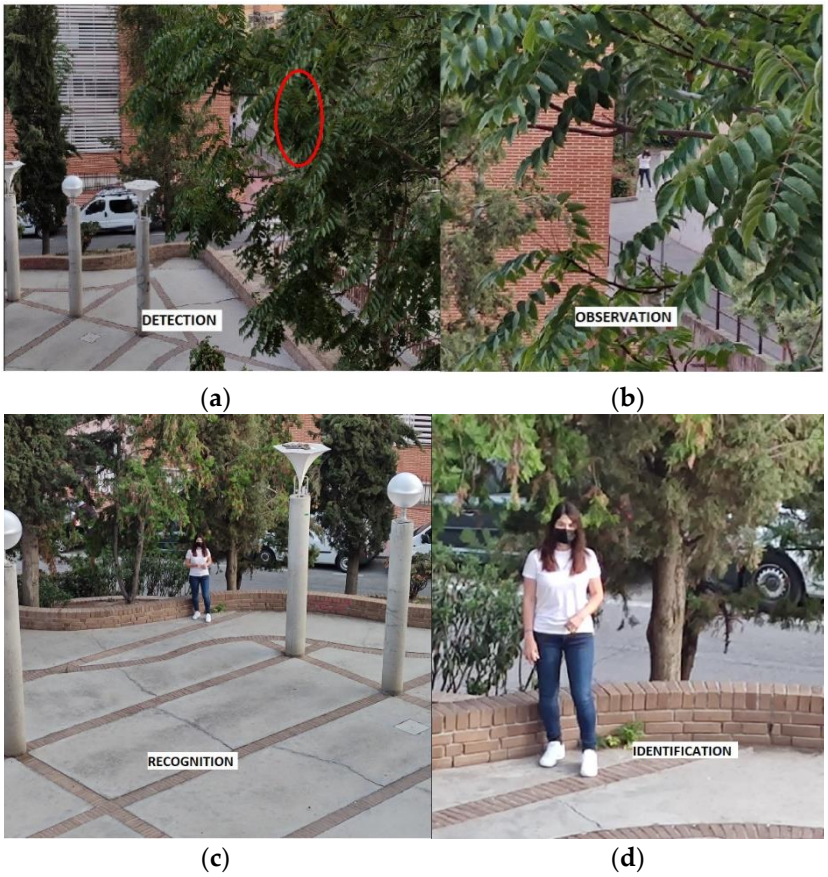


Figure 2. Example of DORI concept: (**a**) Detection, (**b**) Observation, (**c**) Recognition and (**d**) Identification.

Observation (Figure 2b) would correspond to the capacity to appreciate the possible movements of this new asset. It will allow for an analysis or a study of its "intentions". As is shown in the second image, it is usually possible when the new target comes closer to the camera, which allows it to be seen with better resolution, but not enough to recognise it.

Recognition (Figure 2c) is the capability to observe whether the asset is previously known or not. In this case, this term is usually linked with the CCTV operator, as this is the entity that is able to make this recognition. In Figure 2, it is the distance or the number of pixels when the user is able to know who is in the images.

Finally, identification (Figure 2d) is the ability to see enough characteristic elements in the image to make the asset recognisable on subsequent occasions. In the previous example, it is possible to observe enough of the person's features to give an unambiguous picture of him or her (as in the images). These last two capabilities, recognition and identification, are the main objectives pursued by a video surveillance system. In fact, the vast majority of research lines have focused on developing these capabilities, exploiting the increment in the resolution of the sensors. The higher the resolution, the sharper the image, and the greater the distance at which that recognition and identification can be conducted.

The recognition and identification tasks developed by video surveillance systems tend to be mainly performed upon people and vehicles. To identify a person, there are different biometric characteristics that are individual and impossible to replicate, such as the face, voice, eyes or even the arrangement of blood vessels [14,15] (in vehicles, it is normal to resort to license plates).

All these systems need images with a minimum resolution to be able to fulfil their objectives. Hence, lighting mechanisms, and optics and sensors try to obtain the images in the best possible conditions, so that the treatment of these signals offers the best results. Even as they are different procedures, the operation behind them is exactly the same; that is, detect a face/license plate, extract it, and match it with a database to know who the

vehicle's owner is, or who that person is. However, is this an identification? Or is it really a recognition? In order to develop this idea, we must delve deeper into the differences between identification and recognition.

Due to the large computational cost required by the artificial intelligence algorithms applied, the vast majority of identification systems actually perform large-scale recognition tasks [16]. An identification can be understood as a recognition in which a sample "n" collides with a base "N", where "N" contains "n", as well as a large number of elements.

For example, when referring to a facial identification system, the system does not really identify any faces. It will compare a sample against a very large database (such as an ID Card database). Faces are encoded as an array of unique and unrepeatable features, and compared with the rest of the features arrays stored.

The first case would refer to a classification network (such as the one explained in the previous section), in which there would be as many categories as the number of people among whom the identification is made. This generates a problem. Under a traditional approach, to perform a correct classification, it would be necessary to have a very specific dataset with as many categories as elements (faces) to classify. In addition, being very similar elements (faces), it is also necessary to have a very large volume of images. It is not the same that the classifier distinguishes between people and vehicles; for example, compared to only between different types of vehicles.

To simplify this problem, a different approach must be applied. For example, let us imagine an access control system with facial recognition. The system, when it has the frame corresponding to the person who wants to enter, will compare that face with those that stored in its database. That is, it will indicate whether or not that face is in the database. Broadly speaking, this problem can be understood as a binary classification, in where there are really two categories: yes (the image corresponds to one of the stored ones) or no (it does not correspond). Under this perspective, it is possible to simplify a classification problem to a mere comparison. This is called re-identification [17,18], and it is the approach chosen to develop our investigation upon.

2. Related Works: ALPR

Our solution is a combination of two capacities that have been previously mentioned: license plate recognition and vehicle re-identification. As mentioned in the previous section, the idea is to apply the two-factor authentication concept to vehicle recognition, which offers the final user a wider range of capabilities for identifying a target. This idea is very relevant, because in critical situations (real-time tracking), it is not possible to discard a target due to image occlusion (which could affect license plate recognition). In fact, in real conditions, an officer would check for possible positives, even if they were false.

Therefore, this section will compile state-of-art in license plate recognition, which is one of the starting points of this research. Automatic License Plate Recognition (ALPR) is the common name given to the systems designed to read the characters inside a license plate. They can be mainly divided into two different categories, as multi-stage and single-stage methods [19]. Both methods are based on the DORI concept mentioned before. They detect, recognise and identify the license plate and the characters inside.

2.1. Multistage Methods

In these methods, we can mainly identify three steps. The first step is license plate detection, where the license plate is located inside an image, and it is usually bounded by the region of interest (ROI). The second one (which is not compulsory), acts on that ROI to facilitate the identification of the characters by applying traditional image processing techniques such as segmentation or thresholding. The last one is the proper recognition of the characters inside the license plate, applying OCR (Optical Character Recognition). Figure 3 shows a scheme of the whole process.

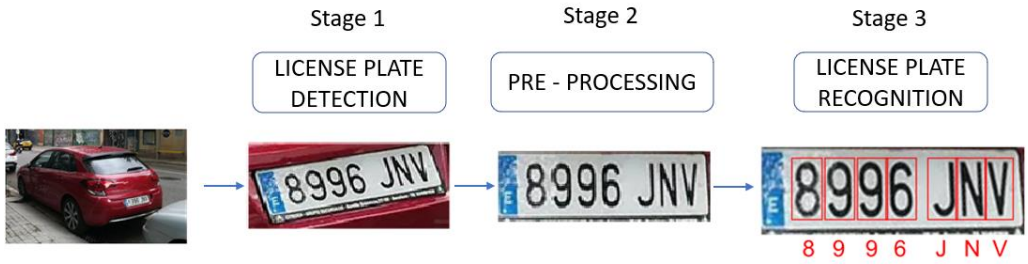


Figure 3. Representation of main stages in a multi-stage license plate recognition system.

2.1.1. License Plate Detection

In [19], license plate detection methods are divided into traditional computer vision techniques and classifiers. Figure 4 (directly extracted from [19]) shows this division:

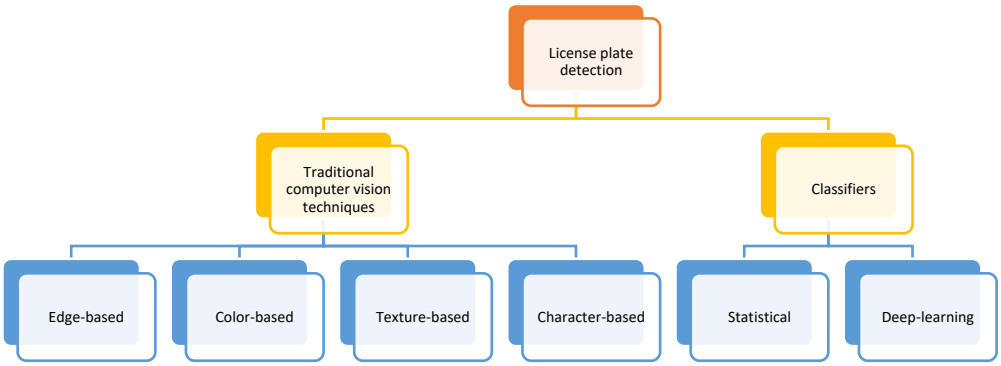


Figure 4. License plate detection methods.

From a practical point of view, this classification can be reduced to traditional computer vision techniques and deep learning classifiers. The first group, especially edge-based methods, can be implemented through OpenCV libraries. An example is [20], which uses a combination between geometric transformations and Haar cascades [21–23] to detect license plates in real time. The idea through traditional techniques is to take advantage of the characteristics and the dimensions of the license plates (which are usually standardised [24]).

On the other hand, deep-learning classifiers are trained to detect license plates and to extract an ROI. Here, the development of YOLO [25] has been crucial. Their different versions and evolutions (especially "Tiny" ones) have reduced inference time and computational costs, being able to operate in real time. In fact, thanks to these classifiers, it is becoming common to use them to pre-detect cars in images, reducing the amount of information processed.

In summary, we can conclude that deep learning methods tend to be more robust, but they require more resources as they are computationally more complex [19] than the "traditional methods". The influence of these computational costs will be deeply explained in Section 4. However, we have to balance this dichotomy between results and costs. In fact, as will be showed in the end of this section, this issue lies in guiding the recent research.

2.1.2. Pre-Processing

Although this step could be omitted, it is very much recommended. It will depend on the quality of the information previously detected.

Here, the aim is to prepare the characters on the inside of the license plate to be easily legible. In this case, we differ from [19], because we consider the traditional preprocessing and character segmentation as only one step altogether, as they are thought to help character recognition.

In this step, the most common techniques (which can also be implemented through the OpenCV libraries) are binarization, thresholding and morphological transformations such

as erosion or dilation, which remove noise from the image and highlight the characters against the background.

Sometimes, it will be necessary to make hard geometrical transformations to turn our image to be as horizontal as possible. This will depend on the character recognition method. This is where deep learning is offering new possibilities, with "spatial transformer networks" [26]. For example, WPOD-net [27] employs spatial transformation and turn images to a horizontal plane.

2.1.3. Characters Recognition

Now that the images are ready to be read, the next step is to extract the characters from the license plate. This process is called Optical Characters Recognition (OCR). The study by [19] shows three different possibilities: compare all the pixel values of the raw image data directly with the predefined templates (template and pattern matching), and use different image processing and machine learning techniques to extract the features, before classifying the segments (character recognition using feature extractors) and deep learning techniques. In [19], we can find more information regarding the first two methods. In this paper, we are going to go deeper into deep learning techniques, which are widely used nowadays.

Tesseract [28] has become the flagship OCR. Many authors [29–32] have developed several methods where Tesseract performs the OCR function. This net was designed for written character recognition. It is very easy to implement, does not have much computational cost, and works well in controlled environments. Thus, it works well on horizontal text with good segmentation. However, it presents several problems when the text is not correctly aligned, or when the characters are a bit blurred. This problem worsens in real-time applications with changing circumstances. For example, in [20], the results throw an accuracy of 91%. In real time, this result descends to 75% (these values just refer to character recognition).

This solution may work for simple models with low resources. However, right now, it is possible to obtain better results (if we have access to GPU units) with specific CNNs that are trained to perform character recognition, such as LPRnet [33] (which obtains a 95% precision rate on Chinese license plates) or OCR-net [34], which raises almost 94% on European license plates (OCR-net is an interpretation of YOLOv2, trained to identify characters in images).

2.2. Single Stage Methods

License plate recognition can be considered as a specific object detection situation. The idea through these systems is to combine detection and recognition in just one single stage, considering the similarities between the two processes [35], becoming faster and more efficient than the two-stage methods [35,36]. Behind this, the idea is the use of Deep Learning algorithms (usually Convolutional Neural Networks or CNNs), such as VGG16 or EfficientNet, which have been trained to make character identification directly from the image.

In this case, it is possible to include [33] or [34] in this group. In fact, NVIDIA offers LPRnet as a dependency included in their toolkit (just trained for Chinese and Indian license plates). Nowadays, one of the current trends is to create or adapt CNNs to directly extract the characters of license plates without making any previous step (or at least very fast ones). In fact, the right training in these systems can enable the creation of robust solutions that are less affected by changing environmental conditions.

2.3. Up-to-Date Solutions

This section will give a brief overview of the different updated solutions launched this year, which will help us to understand the current trends in license plate recognition. We can distinguish two different ones. On the one hand, we can find multi-stage CNN-based developments, readapting well-known nets such as YOLO (in different versions) that have

been trained with specific datasets, and training methods such as transfer learning [37–40]. For example [37], shows an end-to-end double stage method for Persian characters, where license plate recognition is the final step right after car detection and license plate detection, with an accuracy of 99.37% in character recognition. Another example is [38], which is a single-stage method based as well in CNNs for multiple-font characters, with an accuracy of 98.13% in several types of vehicles (such as cars or motorbikes).

On the other hand, there is another very interesting approach that consists of deploying these systems for low-resource devices in real time, emulating real environments. For example, in [41], the authors run their tool in a CPU-based system with 8 GB of RAM, and obtain a precision of 66.1% with MobileNet SSDv2 [42] with a 27.2 FPS rate; or in [43], with metrics of detection of 90% and a recognition rate of 98.73% with just Raspberry Pi3B+ as hardware support. Going further, we can find Android-based systems such as [44] or [45], which are designed to be deployed in mobile phones and operate in real time. These procedures and their results are shown in Table 1.

Table 1. Example of up-to-date solutions for license plate recognition.

Model	Characters	Accuracy	Hardware Requirements
Pirgazi et al. [37]	Persian	99.37%	High
Kaur et al. [38]	Multiple	98.13%	High
Hossain et al. [39]	Bangladeshi	96.31%	High
Zanid et al. [40]	Iranian	98.86%	High
Ashrafi et al. [41]	Bangladeshi	66.1%	Low
Padmasiri et al. [42]	Chinese (CCPD Dataset)	98.73%	Low

2.4. Datasets: Problems and Solutions

One of the main problems for improving ALPRs is having access to datasets. Legislation plays an important role, making it very difficult to share license plate datasets, especially in Europe. In fact, it is easier to access Asian license plate datasets (as Chinese, Pakistani or Indian) (see Table 2). For example, we can find GAP-LP [46] (Turkish), with 9175 license plates, CCPD [36] (Chinese, 250,000 license plates), or UFP-ALPR [47] (Chinese, 4500 images). On the other hand, the biggest open-access European dataset is OpenALPR–EU [48], with only 108 images. The most comprehensive European dataset found is TLPD (THI license plate dataset), [49] with 18,000 labelled European license plates. The problem is that this dataset is not public access.

Table 2. Main license plate datasets.

Name	Location	N° of Images	Vehicles
GAP-LP [46]	Turkey	9175	Cars
CCPD [36]	China	250,000	Cars
UFP-ALPR [47]	China	4500	Various (cars, trucks, etc.)
OpenALPR-EU [48]	Europe	108	Various
TLPD [49]	Europe	18,000	Various

License plates can be very different, depending on the country (even more, when considering Asian characters), making it very difficult to obtain good generic results. To solve this, it is very important to train our system with a specific dataset that should meet the following requirements:

- Variable lighting conditions: day, night, rain, fog . . . ;
- Different angles and viewpoints;
- Several license plates per frame;
- Different backgrounds: street, road, paths . . . ;
- Different qualities and aspect ratios;
- Different sensors.

The dataset has to cover as many real conditions as possible, and not only ideal ones. That being said, the license plates must be recognisable in the different images, and they have to be big enough (at least 5000 different samples). We have developed this idea, creating our own datasets, which will be detailed in Section 5.

3. Related Works: Vehicle Re-Identification

As mentioned in the previous section, the second part of this surveillance system will also focus on the visual characteristics of the whole car. In the introduction, we define object re-identification (Object ReID) as the ability to recognise a specific object that has previously been identified, or "the efforts to associate a particular object through different observations" [50]. This term has been widely used for people, as the possibility "to associate people across camera views at different locations and times" [5]. In fact, re-identification has traditionally been used in faces rather than cars. Facenet [51], launched in 2015, is a good example. Right after it was developed, many companies started to offer facial recognition security solutions, such as passport identification. Although vehicles have many more characteristic patterns than faces, the re-identification of vehicles is more difficult than that of people [52]. That is why even though we can find traditional methods as well as deep learning ones, deep learning development has led to a massive use in Object ReID, displacing traditional methods. It can be considered to be more efficient, because these methods do not focus only on the specific details of the cars, but also on the whole image, selecting several features in the same process [50,52]. That being said, in this section, we are going to describe the most important deep learning procedures for vehicle re-identification.

3.1. Main Deep Learning Procedures for Vehicle Re-Identification

There are different research articles that explore Vehicle Re-identification [53–56]. For example, "A survey of advances in vision-based vehicle re-identification" [57], compiles different methods of vehicle re-identification, ranging from procedures combining the use of external sensors (presence detectors), to deep learning procedures. In contrast, "A Survey of Vehicle Re-Identification Based on Deep Learning" [53] focuses itself on the use of deep learning algorithms. In this case, Convolutional Neural Networks (CNNs) also play a very important role, due to their capacity to extract features from images.

3.1.1. Methods Based on Local Features

The general idea behind these procedures is to obtain a feature vector at the output of the network, based on a number of specific image details. These methods can capture unique visual features in local areas and improve perception, which greatly helps to distinguish between different vehicles, and increases the accuracy of vehicle re-identification. In addition, many methods try to combine local features with global features to improve precision. However, the main disadvantage is the need for high computational resources, to allow for a correct training of a very dense network; this is precisely due to the need to be able to correctly highlight these local features.

3.1.2. Methods Based on Representation Learning

One of the main problems for the on-local features methods is that they require virtually no disruption between the original image and the rest of the correlative images. In case of similar images that are different from the original (which is quite common, either because of a change in the camera angle, or because they are obtained by a different camera), the accuracy of the system decreases significantly, which translates into a loss of effectiveness in real scenarios.

Representation learning works in the readjustment of the weights of the network. It aims to make the network itself capable of focusing on the characteristic points of interest, instead of us pointing the network through the different convolutional layers. The idea is that the network, at the end of the training, will be able to automatically detect those

characteristic patterns that specifically identify the image of each vehicle, but without having a predefined focus.

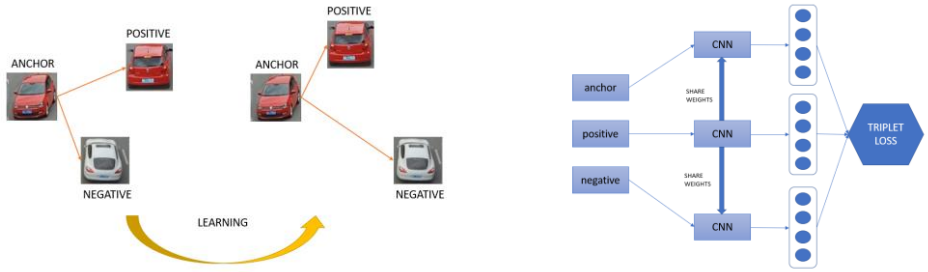
3.1.3. Methods Based on Metric Learning

This method is based on the metrics (i.e., feature vectors) obtained by two or more sets of images at the output of a convolutional neural network. Since a comparison between the metrics is necessary, it is closely linked to the use of Siamese networks. This type of procedure has been widely used, especially in people re-identification.

Within this group of methods, we can highlight mainly two subtypes:

The first is based on the use of a function called "contrastive–loss". At the output of a Siamese network, two feature arrays are obtained. This function calculates the loss function of the network with the absolute value of the Euclidean distance between the two arrays of each of the input images. So, if both images are similar, the value of the loss function will be similar to 0, and if they are different, it will be close to 1. This is further explained in Section 5.2.

The other large subtype is the one that uses the "triplet–loss" as "Facenet" does [51]. In this case, instead of operating with two networks, it uses three convolutional networks simultaneously, in which there would be three input images. An image would be the identified one, which is called "anchor"; a second one would be practically equal to the anchor, which would be the positive image, and a third one would be different, called the negative (Figure 5a). Therefore, there would be two different Euclidean distances, one between the anchor and the positive image, and one between the anchor and the negative image. In this way, the loss function would make a comparison between both Euclidean distances, and the learning of the network would adjust the weights (see Figure 5b).



(**a**) (**b**)

Figure 5. Representation of the "anchor" (**a**) and a network that applies the triplet loss function (**b**).

As the main advantage, this type of mechanism usually presents quite accurate metrics. However, they have problems when it comes to performing correct training, with it being necessary to apply advanced techniques, since this can easily fall into overtraining.

3.1.4. Methods Based on Unsupervised Learning

In this case, the idea behind the system is to use a type of unsupervised training CNN, called GANs (Generative Adversarial Networks), to perform re-identification tasks. These networks have two main modules: a generator and a discriminator. The generator creates new virtual images through real images, introducing pseudorandom patterns, which in some cases, are invaluable to the human eye. The discriminator is in charge of discerning whether, after this subsequent modification, the original image and the one obtained at the discriminator output are similar or not. Thus, it is not necessary to have a dataset that contains labelled images.

This type of network was initially created to protect against steganographic attacks (in which digital noise is included in an image, so that if it is analysed visually, it may be the same as the original, but if its digital encoding is analysed, it is completely different). It is a type of unsupervised learning (unlike the rest of the systems, which are considered supervised), since the network does not know whether the pairs of images analysed by the

discriminator are similar or not to each other; it is the network that does this autonomously. In fact, this is the purpose of the learning process.

The idea, therefore, in vehicle re-identification, is that the generator induces geometric modifications in the images so that the network is able to discriminate between similar and dissimilar images (Figure 6).

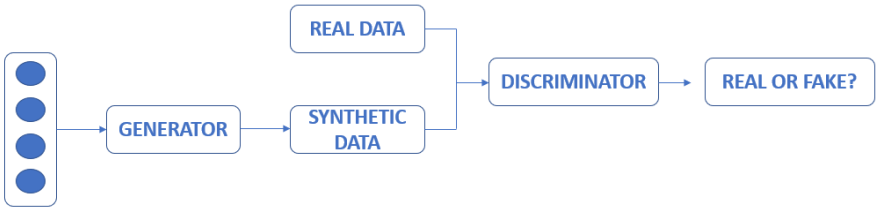


Figure 6. Representation of a GAN.

These types of nets are particularly useful in situations where the vehicle has to be identified under different planes or with different cameras. On the contrary, they are very difficult to train, as they are quite unstable models.

3.1.5. Methods Based on Attention Mechanisms

Finally, the most innovative technique is the application of attention mechanisms [58] at the exit of a network. These mechanisms, considered as an evolution of Residual Neural Networks (or RNNs), were initially conceived for use in natural language processing. The general idea behind RNNs in natural language processing is that each word is encoded with regard to previous words. This presents problems with long-term memory, due to the carry-over of the value of all the gradients. Attention mechanisms solve this problem because they can use all the resources that are available to operate (the only limit), through an encoder–decoder sequence (transformers).

The application of attention mechanisms in images emulates the concept of RNN in natural language processing. Thus, different regions of the image are encoded regarding the previous regions of that image. The system then gives the proper relevance to different parts of the images.

The input in a transformer is a vector corresponding to a text fragment (encoder). If we use them in images, this encoder is a feature map that is extracted from the output of another net (for example, a CNN). As a result of the application of an attention mechanism, the image is first divided into n parts, and it calculates the representations of each specific part of the image. When the network generates a detail of that image, the attention mechanism is focused on the relevant parts of the image, so that the decoder only uses specific parts of the image to encode it. This is called "attention maps". A graphical representation can be seen on Figure 7.

Attention mechanisms facilitate re-identification, in a way that is analogous to what the human being would do, by focusing attention on certain characteristic regions of the image.

The image above shows attention maps that have been taken from some vehicle images. For example, they can be detected in areas such as windshield glows, stickers or specific paints. However, by focusing on eye-catching details, re-identification ability is lost when the details are more subtle, the background of the image resembles the vehicle or when there is no large multitude of images correctly labelled for training.

These procedures are considered as the state of the art in vehicle re-identification. In fact, the chosen solution that will be described later is based on them [56,58–60].

Figure 7. Example representation of attention maps [53].

3.2. Up-to-Date Solutions

As in the previous section, this section will compile some of the current solutions for vehicle re-identification that have been published this year (see Table 3). The research seems mainly focused in two methods: attention mechanisms and metrics learning. The study by [61] is an example of the first ones. The authors propose a "three-branch adaptive attention network for vehicle Re-ID".

Table 3. Example of up-to-date solutions for vehicle re-identification.

Model	Characteristics	Metrics		
		mAP	r@ank1	r@ank5
GRMF [61]	Multi-granularity feature learning	0.882	0.957	0.991
VARID [62]	Inter- and intra-view triplet loss	0.793	0.962	0.992
SN++ [63]	Support neighbours loss	0.757	0.951	0.981
Meng et al. [64]	3D viewpoint alignment	0.832	0.987	0.992

The studies in [62,63] are examples of metrics learning methods. The first one presents a "novel viewpoint-aware triplet loss" to solve re-identification considering intra-view triplet loss (between different classes) and inter-view triplet loss (similarities in the same class) as the final output of the net. The second one proposes the "support neighbours (SN) loss" derived from the KNN (k nearest neighbours) algorithm, and it is also valid for person and vehicle re-identification. The study in [64] is another very interesting and novel approach, due to its combination of "3D viewpoint alignment".

3.3. Datasets

In a classification problem, datasets have a deep impact on network metrics. The more categories to classify, the more images are needed. We have already mentioned that re-identification can be considered a specific classification model. Therefore, datasets are fundamental (as is the case for license plates). The particularity in re-identification is the similarity between the classified objects. This requires a large sample of car images, with very different viewpoints.

One of the first datasets was published in 2013 by Stanford University [65] (Figure 8). The original purpose was for making three-dimensional object classification, which were very similar. However, it started to serve as an initial point for several studies focused on the development of CNNs. In fact, this dataset can be used as a starting point to evaluate certain CNNs as classifiers.

Figure 8. Example of Images from Stanford Cars Dataset [65].

Lately, in 2016, three datasets appeared, with the idea of making vehicle re-identification possible, as with faces: VehicleID [66], VeRi [67], VeRi-776 [68] and VeRi-Wild [55] (Figure 9). For example, VeRi arises from the need to be able to perform vehicle surveillance and tracking in urban environments, offering a new and different solution for re-identification tasks. According to its authors, the purpose is to establish a working environment that is as close as possible to a real scenario, where it does not only depend on a catalogue of a single image per vehicle brand and model, but on several images of the same vehicle. To do this, VeRi contains images of 619 different vehicles, captured by 20 different cameras from various points of view. Therefore, this dataset constitutes a very important starting point for the development of vehicle re-identification systems.

Figure 9. Examples of images from the VeRi dataset [67] and VeRi-Wild [55].

To increase the re-identification possibilities, the same authors of the previous dataset created "VeRi—776". It is an evolution of VeRi, which increases the total volume of images (up to 20% more), with up to 776 different vehicles contained in 50,000 images. It also incorporates new bounding boxes of the license plates in these images, reinforcing the re-identification by complementing the images of vehicles with the characters of the license plates (something fundamental in the work of police investigation). Finally, it includes an annotation of the spatial–temporal relationship, providing the location of the camera capturing the image, the direction of the captured vehicle and the trajectory it is taking.

VeRi-Wild is the ultimate expression of a dataset obtained through real-time video captures. For his purpose, continuous video sequences were obtained 24 h a day for a full month, captured by 174 different cameras. This achieved 416,314 images of vehicles with 40,671 different identities, with different optic variants such as occlusions, variant trajectories, perspectives, etc.

We consider these three datasets, due to their resemblance to the reality of work within security, as well as their wide range of data, to be very important tools for developing a valid and useful re-identification tool for real work.

That being said, there are other datasets (see Table 4) with a big influence in re-identification studies such as PKU Vehicle [69], which contains tens of millions of vehicle

images captured by real-world surveillance cameras in several Chinese cities, "including several locations (e.g., highways, streets, intersections), weather conditions (e.g., sunny, rainy, foggy), illuminations (e.g., daytime and evening), shooting angles (e.g., front, side, rear), different resolutions (e.g., 480 P, 640 P, 720 P, 1080 P, 2 K) and hundreds of vehicle brands". Other examples are Vehicle-1M [70], also Chinese, with 55,527 vehicles of 400 different vehicle models and 936,051 images, or CityFlow [71] which is the world's first large dataset [72] containing cross-camera car tracking and re-identification. It has 3.25 h of surveillance from 40 different cameras at 10 intersections in an American city, in both residential areas and highways. The dataset includes 229,680 labelled vehicles of 666 different vehicles. Each car has passed through at least two cameras, and the dataset provides raw video, camera distribution, and multi-view analysis.

Table 4. Main vehicle re-identification datasets.

Name	Nº of Images	Nº of Vehicles	Characteristics
Stanford Cars Dataset [65]	16,185	196	Car images
VehicleID [66]	221,763	250	Surveillance Cameras
VeRi [67]	40,000	619	Surveillance Cameras
VeRi-776 [68]	49,360	776	Surveillance Cameras
VeRi-Wild [55]	416,314	40,671	Surveillance Cameras
PKU Vehicle [69]	10,000,000		Surveillance Cameras
Vehicle 1M [70]	936,051	26,267	Surveillance Cameras
CityFlow [71]	229,680	666	Surveillance Cameras
VeRi-UAV [73]	17,515	454	Aerial images

Finally, the massive proliferation of UAV, mainly in security, has opened up a new field in vehicle re-identification research. The authors of [73] propose "the view-decision based compound matching learning model (VD-CML)". To verify the effectiveness of their proposal, they have created the first vehicle re-identification dataset (VeRi-UAV) captured by an UAV. This dataset has impelled new research, such as [74,75].

4. Practical Considerations for Real Scenario Use

4.1. Speed Processing and Computational Costs

Depending on the location of image processing, there are mainly two types of configurations in surveillance systems: local or remote processing. The first one means that the sensors and processing hardware are physically connected; the second one employs wireless connections (it could be WiFi, LTE, etc.).

Local processing systems are much faster, because the images are instantaneously processed, while remote processing systems are limited by the transfer rate between the different components. This transfer rate can vary greatly. It will depend on many factors such as mobile phone coverage, frame rate, frame size or compression format. For example, a FullHD sensor may last between 1 and 1.5 s to send each frame using LTE transmission in ideal conditions (tests runs under real conditions). If our target travels at 120 km/h, in just one second, it will cover approximately 33 m. Probably, with that distance, it would not be possible to keep a target under control. Thus, our system may not be effective. This means that real-time operational conditions require fast, almost instantaneous, image processing.

Although some of the networks previously mentioned may take just a few milliseconds to process an image, they require huge amounts of hardware, not only for training the net, but also to run it. So, if we want to operate in real time, we may need to have the hardware locally connected with the sensors. This is not always possible, for three main reasons. First, because this hardware is usually very expensive. Second, because they require direct connection with the electric grid (which is not available in certain scenarios) and third, because hardware can be too big to be installed near the sensors.

Under these circumstances, it may be better to opt for less precise but less resource-consuming systems. For example, some ALPR systems are ready to operate through CPU

instead of GPU, although they sacrifice precision; such as [41] or [43]. It is important to mention that in critical scenarios, there will always be a human controller behind the system, so these tools may deliver more false positives rather than false negatives (and the human controller will discriminate if it is right or not).

4.2. Sensors and Support Elements

As we have mentioned in previous paragraphs, images taken by the sensors must be as neat as possible. The example explained in the introduction (the car park), shows how the sensor responsible for the ALPR functions is settled in a static position (taking a static image of the license plate) with infrared illumination and a fixed lens. With this configuration, the OCR can work much easier. First, this is because images will always be similar and will fit with the license plate (because it can be tested before installation). Second, because infrared illumination makes an automatic thresholding (taking advantage of the reflective background of the license plates), simplifying character segmentation. Third, because the optics of the sensor will reduce aberrations in the images. That it is why most car parks force cars to stop in a certain position, to identify the license plate number. In fact, with these aids, we can perform character reading with just a simple tool such as Tesseract.

Another important consideration that affects both ALPR and re-identification is the area covered that is by the sensors. With the proper scene, resource requirements can be reduced, because the signal processing mechanisms will have less and better information to analyse. As has previously been mentioned, this is how several ALPR systems increase their accuracy.

With this simple explanation, we want to highlight the importance of the rest of the components of a surveillance system, such as the lighting mechanism, the optics, and the capture sensor, to improve the accuracy. Even though we do not cover these aspects in this paper, they must be deeply analysed.

5. Proposed Method: Double-Factor Authentication for Vehicle Identification

In this section, we are going to describe our proposed solution (including future evolutions). As stated in the introduction, most of the above-mentioned proposals for vehicle identification act in isolation, or they combine just one AI technology with different physical sensors (as in [64]). Ours tries to show a new approach by offering both license plate recognition and vehicle re-identification; at the same time, combining everything in a multinet solution. This is intended to take advantage of those images that, as mentioned above, provide partial information on the target vehicles. This could be seen as a practical application of two-factor authentication for vehicle tracking.

To achieve this, it has been necessary to create a new dataset for number plate recognition training (to increase the capacities) as well as two practical datasets for assessing vehicle re-identification.

From a hardware point of view, this is a GPU-based system, but since it has to run in real time, we have tried to select models that could offer the minimum possible inference time.

5.1. Model Architecture

This bimodal system has three different modules: object detection, license plate recognition and vehicle re-identification, as can be seen in Figure 10:

- YOLOv5 [76]: This is the first object detector, which gives the bounding boxes for all the vehicles present in the image.
- YOLOv5 multi-stage: This is the first parallel branch, responsible for performing license plate detection, as well as character recognition, in a multi-stage method.
- FastReid [60]: This network runs vehicle re-identification.

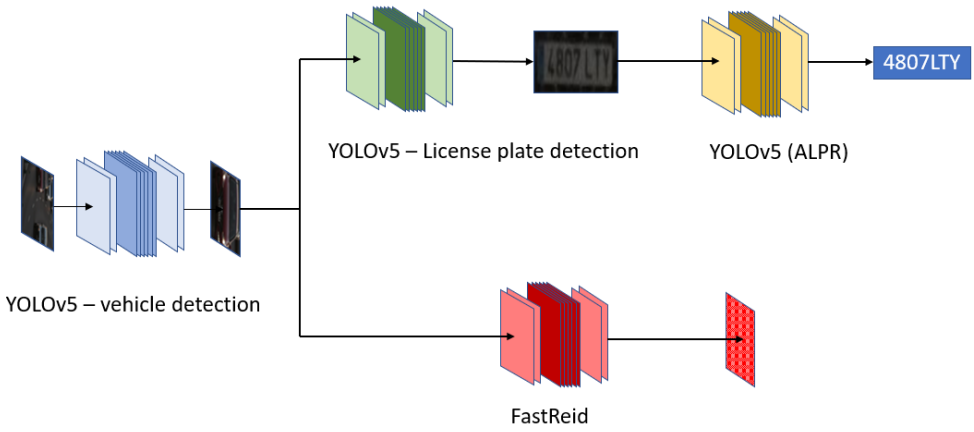


Figure 10. Proposed multi-neural network architecture.

5.1.1. Object Detection

Object detection is twofold in this work. Firstly, the aim is to indicate the ROIs within the image that correspond to a vehicle. One of the general requirements of the system is the ability to be implemented in real time. This is the main reason for choosing YOLOv5 over other models such as YOLOv4 [77], EfficientDet [78], ATSS [79], ASFF [80] or CenterMask [81]. This first YOLOv5 has been trained with the COCO dataset [82].

5.1.2. License Plate Recognition

Two different CNNs are responsible for this step. So, this ALPR can be catalogued as being a multi-stage method. On the one hand, we use a secondYOLOv5 that is specifically trained for license plate detection with our custom dataset (which will be later detailed). It acts right after the first YOLOv5 that extracts car images. This first step helps to obtain the license plate ROI, where it will act as the second module that is responsible for license plate character recognition. We have chosen YOLOv5 mainly for these two reasons. Firstly, because it is very fast in performing object detection. The "s version" (YOLOv5s) can performs iterations in just 0.6 ms under ideal conditions. This tiny inference time is very important for creating a real-time solution, and so we have prioritised speed over accuracy. Secondly, YOLOv5 has proven to be very robust to spatial variations, which is very important in license plate recognition.

The second module is also a YOLOv5 that has been trained with a specific dataset created for direct character recognition on the characters within the previously detected number plates (Figure 11). This particular research has been inspired by OCR-net [27], which is a modified YOLO network, which is similar to the one discussed in the plate contour detection, but with characters. However, the training dataset was considerably extended using synthetic and augmented data, to cope with the diverse characteristics of license plates from different parts of the world.

Figure 11. Character recognition.

The main problem of this net is its compatibility problems with the latest GPU versions. This leads to a large increase in inference time, which has motivated us to do something similar, but with YOLOv5 (the latest YOLO version).

5.1.3. Vehicle Re-Identification

This is based on the extraction of the visual characteristics of vehicles, including shape, perspective and colour. For this reason, it is more robust against variable conditions, since the details and regions to be analysed are larger and more easily scalable. Right after the first YOLOv5 (the one at the entrance of the entire system) extracts the ROI of the car detected, the image is cropped and becomes the input of the re-identification net. This cropped image goes through the backbone and the aggregation layers [61] and generates a 4096 features vector, and this re-identification is concluded by calculating the Euclidean distance between the 4096-feature vectors from the visual recognition model. Each of the processed images will have its corresponding feature vector, which is compared with the features vector of the objective vehicle, to calculate the distance.

5.2. Datasets

5.2.1. Datasets for Vehicle Re-Identification Testing

We have created two different datasets that recreate real working scenarios to test the system, using self-made images. The first one has been called the Highway Gantry Dataset (HGD) (Figure 12). It contains images taken from an elevated position in a highway (gantry). Cars can be seen with identical perspective and lightning without occlusions, imitating a speed trap. It includes a total of 458 images, corresponding to 200 vehicle models.

Figure 12. Highway Gantry Dataset sample.

The second dataset has been called the Operational Urban Dataset (OUD) (Figure 13). It is divided into two different recording locations (v1 and v2), with a multitude of different perspectives and occlusions between the vehicles and with the vegetation. Each of the scenes has been captured simultaneously by two different cameras (c1 and c2), emulating a real operational environment. In addition, by having two input sources, it allows for searching for vehicles annotated from one camera in the other, with different perspectives, which is the main objective pursued in this work. It groups a total of 1255 images of 69 classes, with slightly different annotation criteria. V1 contains all types of images, including very distant and partial views, while v2 only collects vehicles with a minimum recognizable size.

Figure 13. Operational Urban Dataset sample.

5.2.2. Dataset for License Plate Recognition

As mentioned, one of the main problems in ALPR systems is having access to specific license plate datasets. Thus, we have created a specific dataset with different images of European license plates (mainly Spanish) from very different views, angles and lightning conditions (including day and night). It contains more than 2035 images, 2531 license plates and over 15,000 characters, both manually labelled and rescaled to 640 × 640 pixels per labelled image. We have called it the Spanish ALPR Dataset (Figure 14).

Figure 14. Spanish ALPR Dataset examples.

This dataset is twofold. The first part (license plates) has been used for license plate detection and ROI extraction against YOLOv5s. For character recognition, we have used the second part (the labelled characters) to train YOLOv5s again, obtaining a 37 categories classifier (27 letters and 10 numbers).

5.3. Training

5.3.1. Vehicle Re-Identification Model Training

Vehicle re-identification is based on FastReid Toolbox; a comparison between some state-of-the-art models and several trained backbone models has been tested. For vehicle re-identification purposes, the FastReid Toolbox Repository [83] offers already pretrained and optimised architectures. As a backbone, different training strategies with the EfficientNet [84] family have been performed, with max pooling and convolutional layers appended to their output for fine tuning.

Firstly, it has been trained with Standford Cars dataset [65] (Figure 15). To adjust the dropout and learning rate, the first test training was performed with a reduced version of the dataset. This first value refers to the ratio of neural networks in certain layers that are randomly "turned off" during training. In this way, feature extraction is performed via several paths (the "firing" neurons) and thus, the model is better generalised. The learning rate refers to the speed at which the weights are updated. A reduced value allows many more weights to be added, but at the cost of a longer training time, so it is advisable to adjust it optimally.

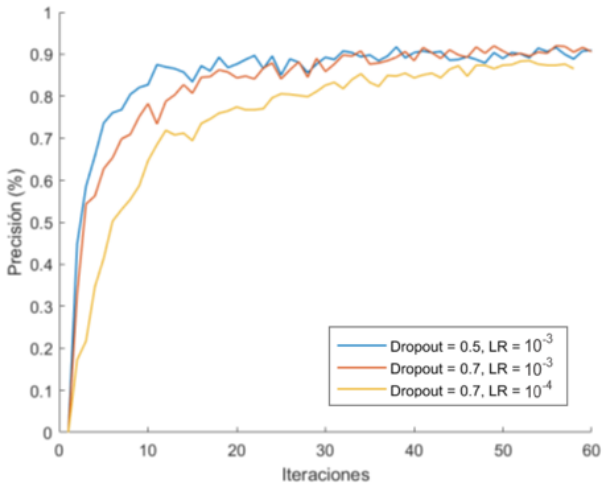


Figure 15. Dropout and learning rate effect.

Two remarkable results in this graph can be observed. The first one is that a dropout of 0.7 generalizes better than with 0.5. However, it takes slightly longer in the early epochs, and the trend is in favour of the former, as it reaches a significantly lower maximum at 0.5. Additionally, the most suitable learning rate seems to be 10^{-3}, because it maximizes the precision faster. With this consideration, as well as other tests, we have tested the performance of three versions of the EfficientNet network: B0, B3 and B7 (Figure 16). The output has been configured using a maximum global pooling for each of the output filters, and a dense classification layer with the previously adjusted dropout.

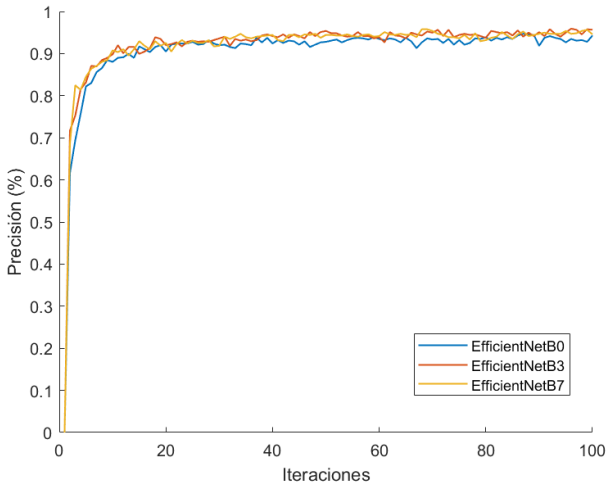


Figure 16. EfficientNet B0-B3-B7 comparison in Stanford-Cars training.

The graphics show a very similar performance between the three models, and so we have adopted B0 model, as it is the lightest and the fastest. Additionally, we have conducted another training run (Figure 17) with the VeRi-776 dataset [67], changing output classes. The precision maintains similarity between the three models, confirming the election of EfficientNetB0.

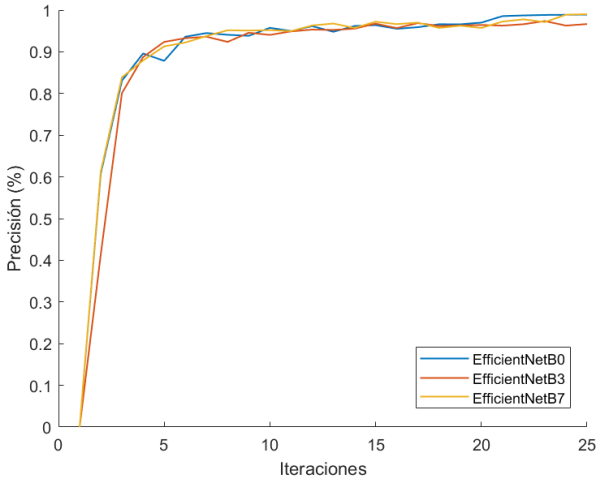


Figure 17. EfficientNet B0-B3-B7 comparison in VeRi-776 training.

5.3.2. License Plate Detection/Character Recognition Training

The Spanish ALPR Dataset has been used to train both license plate detection and character recognition. In the first case, we have made a fast-training run (Figure 18) with 50 epochs in YOLOv5s and the Adam optimizer, obtaining this result:

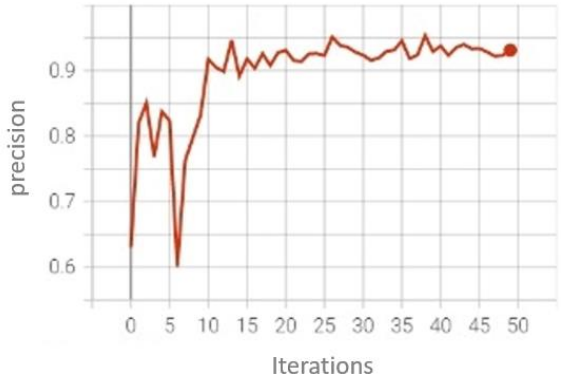


Figure 18. License plate detection training graph.

In character recognition, we have also trained with 50 epochs in the YOLOv5s version (Figure 19) with the stochastic gradient descent optimizer, obtaining this result:

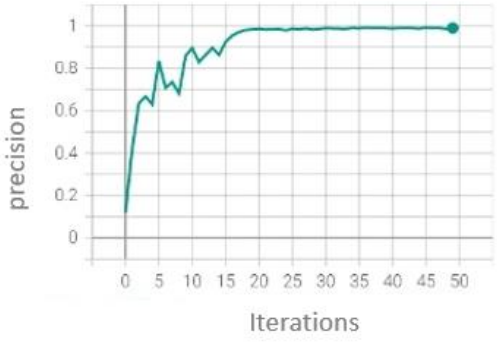


Figure 19. Character recognition training graph.

5.4. Results

The following tables show the final training results. It is important to mention that we are combining two different methods for vehicle identification within real scenarios, so these metrics are relative to our own datasets, which have the aim of recreating these real

conditions. On the one hand, we will show results for vehicle re-identification, tested on the HGD and OUD datasets (Tables 5 and 6). On the other hand, we will show the license plate detection and recognition results, trained and tested on our own dataset. The use of a specific dataset for ALPR training is due to the lack of big enough European license plate datasets.

Table 5. Accuracy in positive–negative pair test in HGD and OUD datasets for vehicle re-identification.

	Precision				
Model	HGD	OUD v1c1	OUD v2c1	OUD v1 [1]	OUD v2 [1]
FastReid (VeRi-776)	97.9%	94.0%	96.6%	87.8%	91.5%

[1] Dataset with 2 different input cameras.

Table 6. Rank@1 and rank@10 metrics for vehicle re-identification in OUD datasets.

	Rank@1		**Rank@10**	
Model	OUD v1	OUD v2	OUD v1	OUD v2
FastReid (VeRi-776)	75.4%	90.6%	94.0%	99.4%

That being said, these are the final results:

Additionally, rank@n metrics have been calculated, with rank@1 and rank@10 in particular.

In this case, we can make a comparison with [49]. This collects one of the largest European license plate datasets for license plate detection (exclusively). TLPD contains over 18,000 different images of license plates, with the bounding box labelled. This is similar to our dataset (in the acquisition conditions); although ours is smaller, with 2351 license plates, it is also ready for license plate character recognition, offering additional capacities.

Table 7 will show the results of our model, trained with the Spanish ALPR Dataset compared with the models proposed in [49], trained with TLPD, and only for license plate detection:

Table 7. Metrics of YOLOv5s trained with Spanish ALPR Dataset.

Model	**Dataset**	**Function**	**Precision**	**Recall**
Fast-YOLO	TLPD [49]	License plate detection	90.08%	90.11%
Tiny-YOLO v3	TLPD [49]	License plate detection	91.44%	92.32%
YOLOv5s	Spanish ALPR Dataset	License plate detection	93.1%	94.3%
YOLOv5s	Spanish ALPR Dataset	Character Recognition	98.4%	95.1%

If we attend to the inference time (Table 8), the results throw that it is possible to perform a whole vehicle identification in less than 10 ms, although only under very ideal conditions. This time can easily be incremented, due to several factors, such as the overall data of the processed image.

Table 8. Inference time in visual re-identification.

Model	**Time (ms)**
YOLOv5s car detection	0.7
YOLOv5s license plate detection	0.7
FastReid (VeRi-776)	4.04
YOLOv5s character recognition	1

5.5. Future Works

There are mainly two lines of research. The first one is training YOLOv5s for both cars and license plate detection at the same time, and using just one YOLOv5s at the entrance of the net, instead of two in a row. The second one would be increasing the Spanish ALPR dataset to improve metrics in training. We want to also to collect more license plates with vowels, as well as consonants, to improve the results. The problem is that the Spanish license plates have not had vowels since 2000, so it is very difficult to find older vehicles.

6. Conclusions

The aim of this article was to develop new solutions for vehicle identification in real time, and under real operational conditions. To this end, we first pointed out some problems that concern traditional surveillance systems that are used for vehicle identification, mainly due to the images taken by the sensors under operational conditions. In addition, we have highlighted the possibility of solving them by the means of two-factor authentication mechanisms: a combination of license plates and vehicle characteristics. To help us in our research, we have compiled the state-of-the-art methods of license plate reading, as well as vehicle re-identification. After identifying the scarcity of suitable European license plate datasets, we created our own one, to develop the system. We have also created two others to test vehicle re-identification under real conditions. Thus, we believe that we offer a fresh point of view in vehicle identification, through the combination of two robust solutions, based on updated CNNs, as well as the creation of specific datasets.

Author Contributions: This work is the result of research conducted by the authors in the field of security. Conceptualization, J.G.-C., Á.R. and J.M.A.; methodology, J.G-C. and J.M.A.; software, J.G.-C. and Á.R.; validation, J.G.-C., Á.R. and J.M.A.; formal analysis, J.G-C. and J.M.A.; investigation, J.G.-C. and Á.R.; resources, J.M.A.; data curation, J.G.-C. and Á.R.; writing—original draft preparation, J.G.-C.; writing—review and editing, J.G.-C., Á.R. and J.M.A.; project administration, J.M.A.; funding acquisition, J.M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been funded by the following research projects: CCAD (AEI), SEGVAUTO-4.0-CM and AMBULATE (CM).

Data Availability Statement: Not applicable.

Acknowledgments: Grants PID2019-104793RB-C31 and PDC2021-121517-C31, funded by MCIN/AEI/10.13039/501100011033, and by the European Union, "NextGenerationEU/PRTR" and the Comunidad de Madrid, through SEGVAUTO-4.0-CM (P2018/EMT-4362). New paradigm for emergency transport services management: ambulance. AMBULATE-CM. This article is part of the agreement between the Comunidad de Madrid (Consejería de Educación, Universidades, Ciencia y Portavocía) and uc3m for the direct award of aid to fund research projects on SARS-CoV-2 and COVID-19 disease financed with the React-UE resources of the European Regional Development Fund, "A way for Europe".

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ghosh, G.; Sood, M.; Verma, S. Internet of things based video surveillance systems for security applications. *J. Comput. Theor. Nanosci.* **2020**, *17*, 2582–2588. [CrossRef]
2. Elharrouss, O.; Almaadeed, N.; Al-Maadeed, S. A review of video surveillance systems. *J. Vis. Commun. Image Represent.* **2021**, *77*, 103116. [CrossRef]
3. Zhang, S.; Chan, S.C.; Qiu, R.D.; Ng, K.T.; Hung, Y.S.; Lu, W. On the design and implementation of a high definition multi-view intelligent video surveillance system. In Proceedings of the 2012 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC 2012), Hong Kong, China, 12–15 August 2012; pp. 353–357. [CrossRef]
4. Sreenu, G.; Durai, S. Intelligent video surveillance: A review through deep learning techniques for crowd analysis. *J. Big Data* **2019**, *6*, 48. [CrossRef]
5. Fernandes, A.O.; Moreira, L.F.E.; Mata, J.M. Machine vision applications and development aspects. In Proceedings of the 2011 9th IEEE International Conference on Control and Automation (ICCA), Santiago, Chile, 19–21 December 2011; pp. 1274–1278. [CrossRef]

6. Wang, V.; Tucker, J.V. Surveillance and identity: Conceptual framework and formal models. *J. Cybersecur.* **2017**, *3*, 145–158. [CrossRef]
7. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing*; Prentice Hall: Upper Saddle River, NJ, USA, 2008; ISBN 13: 9788131712863.
8. Safie, S.; Azmi, N.M.A.N.; Yusof, R.; Yunus, M.R.M.; Sayuti, M.F.Z.C.; Fai, K.K. Object Localization and Detection for Real-Time Automatic License Plate Detection (ALPR) System Using RetinaNet Algorithm. In *Intelligent Systems and Applications*; IntelliSys 2019; Advances in Intelligent Systems and Computing; Bi, Y., Bhatia, R., Kapoor, S., Eds.; Springer: Cham, Switzerland, 2020; Volume 1037.
9. Aloul, F.; Zahidi, S.; El-Hajj, W. Two factor authentication using mobile phones. In Proceedings of the 2009 IEEE/ACS International Conference on Computer Systems and Applications, Rabat, Morocco, 10–13 May 2009; pp. 641–644.
10. De Cristofaro, E.; Du, H.; Freudiger, J.; Norcie, G. A comparative usability study of two-factor authentication. *arXiv* **2013**, arXiv:1309.5344.
11. Gope, P.; Sikdar, B. Lightweight and privacy-preserving two-factor authentication scheme for IoT devices. *IEEE Internet Things J.* **2018**, *6*, 580–589. [CrossRef]
12. Lee, S.; Ong, I.; Lim, H.T.; Lee, H.J. Two factor authentication for cloud computing. *J. Inf. Commun. Converg. Eng.* **2010**, *8*, 427–432. [CrossRef]
13. *IEC EN62676-4*; Video Surveillance Systems for Use in Security Applications—Part 4: Application Guidelines. International Standard: Geneva, Switzerland, 2015. Available online: https://standards.globalspec.com/std/9939964/EN%2062676-4 (accessed on 29 June 2022).
14. Bouchrika, I. A survey of using biometrics for smart visual surveillance: Gait recognition. In *Surveillance in Action*; Springer: Cham, Switzerland, 2018; pp. 3–23.
15. Devasena, C.L.; Revathí, R.; Hemalatha, M. Video Surveillance Systems—A Survey. *Int. J. Comput. Sci. Issues (IJCSI)* **2011**, *8*, 635–642.
16. Renninger, L.W.; Malik, J. When is scene identification just texture recognition? *Vis. Res.* **2004**, *44*, 2301–2311. [CrossRef]
17. Gong, S.; Xiang, T. Person Re-identification. In *Visual Analysis of Behaviour*; Springer: London, UK, 2011. [CrossRef]
18. Layne, R.; Hospedales, T.M.; Gong, S.; Mary, Q. Person re-identification by attributes. *BMVC* **2012**, *2*, 8.
19. Shashirangana, J.; Padmasiri, H.; Meedeniya, D.; Perera, C. Automated license plate recognition: A survey on methods and techniques. *IEEE Access* **2020**, *9*, 11203–11225. [CrossRef]
20. García Serrano, A. *Aplicación de Sistemas de Percepción Para la Seguridad Vial*; Departamento de Ingeniería Eléctrica, Electrónica y Automática, Universidad Carlos III: Madrid, Spain, 2020.
21. Guevara, M.L.; Echeverry, J.D.; Urueña, W.A. Detección de rostros en imágenes digitales usando clasificadores en cascada. *Sci. Tech.* **2008**, *1*, 38.
22. Sharma, P.S.; Roy, P.K.; Ahmad, N.; Ahuja, J.; Kumar, N. Localisation of License Plate and Character Recognition Using Haar Cascade. In Proceedings of the 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 13–15 March 2019; pp. 971–974.
23. Cuimei, L.; Zhiliang, Q.; Nan, J.; Jianhua, W. Human face detection algorithm via Haar cascade classifier combined with three additional classifiers. In Proceedings of the 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Yangzhou, China, 20–22 October 2017; pp. 483–487.
24. Real Decreto 2822/1998, de 23 de Diciembre, por el que se Aprueba el Reglamento General de Vehículos. Spain (1998, mod. 2021). Available online: https://www.boe.es/buscar/act.php?id=BOE-A-1999-1826 (accessed on 29 June 2022).
25. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
26. Jaderberg, M.; Simonyan, K.; Zisserman, A. Spatial transformer networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. Available online: https://proceedings.neurips.cc/paper/2015/hash/33ceb07bf4eeb3da587e268d663aba1a-Abstract.html (accessed on 29 June 2022).
27. Silva, S.M.; Jung, C.R. License plate detection and recognition in unconstrained scenarios. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 580–596.
28. Smith, R. An overview of the Tesseract OCR engine. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Curitiba, Brazil, 23–26 September 2007; Volume 2, pp. 629–633.
29. Patel, C.; Patel, A.; Patel, D. Optical character recognition by open-source OCR tool tesseract: A case study. *Int. J. Comput. Appl.* **2012**, *55*, 50–56. [CrossRef]
30. Singh, J.; Bhushan, B. Real Time Indian License Plate Detection using Deep Neural Networks and Optical Character Recognition using LSTM Tesseract. In Proceedings of the 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 18–19 October 2019; pp. 347–352.
31. Goel, T.; Tripathi, K.C.; Sharma, M.L. Single Line License Plate Detection Using OPENCV and tesseract. *Int. Res. J. Eng. Technol.* **2020**, *07*, 5884–5887.
32. Dias, C.; Jagetiya, A.; Chaurasia, S. Anonymous vehicle detection for secure campuses: A framework for license plate recognition using deep learning. In Proceedings of the 2019 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT), Jaipur, India, 28–29 September 2019; pp. 79–82.
33. Zherzdev, S.; Gruzdev, A. Lprnet: License plate recognition via deep neural networks. *arXiv* **2018**, arXiv:1806.10447.

34. Silva, S.M.; Jung, C.R. Real-time brazilian license plate detection and recognition using deep convolutional neural networks. In Proceedings of the 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Niteroi, Brazil, 17–20 October 2017; pp. 55–62.
35. Li, H.; Wang, P.; Shen, C. Toward end-to-end car license plate detection and recognition with deep neural networks. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 1126–1136. [CrossRef]
36. Xu, Z.; Yang, W.; Meng, A.; Lu, N.; Huang, H.; Ying, C.; Huang, L. Towards end-to-end license plate detection and recognition: A large dataset and baseline. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 255–271.
37. Pirgazi, J.; Kallehbasti, M.M.P.; Ghanbari Sorkhi, A. An End-to-End Deep Learning Approach for Plate Recognition in Intelligent Transportation Systems. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 3364921. [CrossRef]
38. Kaur, P.; Kumar, Y.; Ahmed, S.; Alhumam, A.; Singla, R.; Ijaz, M.F. Automatic License Plate Recognition System for Vehicles Using a CNN. *CMC-Comput. Mater. Contin.* **2022**, *71*, 35–50.
39. Hossain, S.N.; Hassan, M.; Masba, M.; Al, M. Automatic License Plate Recognition System for Bangladeshi Vehicles Using Deep Neural Network. In *Proceedings of the International Conference on Big Data, IoT, and Machine Learning*; Springer: Singapore, 2022; pp. 91–102.
40. Zandi, M.S.; Rajabi, R. Deep Learning Based Framework for Iranian License Plate Detection and Recognition. *Multimedia Tools Appl.* **2022**, *81*, 15841–15858.
41. Ashrafee, A.; Khan, A.M.; Irbaz, M.S.; Nasim, A.; Abdullah, M.D. Real-time Bangla License Plate Recognition System for Low Resource Video-based Applications. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 4–8 January 2022; pp. 479–488.
42. Chiu, Y.C.; Tsai, C.Y.; Ruan, M.D.; Shen, G.Y.; Lee, T.T. Mobilenet-ssdv2: An improved object detection model for embedded systems. In Proceedings of the 2020 International Conference on System Science and Engineering (ICSSE), Kagawa, Japan, 31 August–3 September 2020; pp. 1–5.
43. Padmasiri, H.; Shashirangana, J.; Meedeniya, D.; Rana, O.; Perera, C. Automated License Plate Recognition for Resource-Constrained Environments. *Sensors* **2022**, *22*, 1434. [CrossRef] [PubMed]
44. Ali, F.; Rathor, H.; Akram, W. License Plate Recognition System. In Proceedings of the 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 4–5 March 2021; pp. 1053–1055. [CrossRef]
45. Yang, C.; Zhou, L. Design and Implementation of License Plate Recognition System Based on Android. In *Proceedings of the 11th International Conference on Computer Engineering and Networks*; Springer: Singapore, 2022; pp. 211–219.
46. Kessentini, Y.; Besbes, M.D.; Ammar, S.; Chabbouh, A. A two-stage deep neural network for multi-norm license plate detection and recognition. *Expert Syst. Appl.* **2019**, *136*, 159–170. [CrossRef]
47. Laroca, R.; Severo, E.; Zanlorensi, L.A.; Oliveira, L.S.; Gonçalves, G.R.; Schwartz, W.R.; Menotti, D. A robust real-time automatic license plate recognition based on the YOLO detector. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–10.
48. OpenALPR. Openalpr-Eu Dataset. 2016. Available online: https://github.com/openalpr/benchmarks/tree/master/endtoend/eu (accessed on 29 June 2022).
49. Chan, L.Y.; Zimmer, A.; da Silva, J.L.; Brandmeier, T. European Union Dataset and Annotation Tool for Real Time Automatic License Plate Detection and Blurring. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–6.
50. Yang, H.; Cai, J.; Zhu, M.; Liu, C.; Wang, Y. Traffic-Informed Multi-Camera Sensing (TIMS) System Based on Vehicle Re-Identification. *IEEE Trans. Intell. Transp. Syst.* **2022**. [CrossRef]
51. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 815–823.
52. Wang, Y. Deep learning technology for re-identification of people and vehicles. In Proceedings of the 2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), Changchun, China, 25–27 February 2022; pp. 972–975.
53. Wang, H.; Hou, J.; Chen, N. A survey of vehicle re-identification based on deep learning. *IEEE Access* **2019**, *7*, 172443–172469. [CrossRef]
54. Mai, L.; Chen, X.Z.; Yu, C.W.; Chen, Y.L. Multi-view Vehicle Re-Identification Method Based on Siamese Convolutional Neural Network Structure. In Proceedings of the 2020 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan), Taoyuan, Taiwan, 28–30 September 2020; pp. 1–2.
55. Lou, Y.; Bai, Y.; Liu, J.; Wang, S.; Duan, L. Veri-wild: A large dataset and a new method for vehicle re-identification in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3235–3243.
56. Zheng, Z.; Ruan, T.; Wei, Y.; Yang, Y.; Mei, T. VehicleNet: Learning robust visual representation for vehicle re-identification. *IEEE Trans. Multimed.* **2020**, *23*, 2683–2693. [CrossRef]
57. Khan, S.D.; Ullah, H. A survey of advances in vision-based vehicle re-identification. *Comput. Vis. Image Underst.* **2019**, *182*, 50–63. [CrossRef]

58. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. Available online: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html (accessed on 29 June 2022).
59. Bai, S.; Zheng, Z.; Wang, X.; Lin, J.; Zhang, Z.; Zhou, C.; Yang, H.; Yang, Y. Connecting language and vision for natural language-based vehicle retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 4034–4043.
60. He, L.; Liao, X.; Liu, W.; Liu, X.; Cheng, P.; Mei, T. Fastreid: A pytorch toolbox for general instance re-identification. *arXiv* **2020**, arXiv:2006.02631.
61. Tian, X.; Pang, X.; Jiang, G.; Meng, Q.; Zheng, Y. Vehicle Re-Identification Based on Global Relational Attention and Multi-Granularity Feature Learning. *IEEE Access* **2022**, *10*, 17674–17682. [CrossRef]
62. Li, Y.; Liu, K.; Jin, Y.; Wang, T.; Lin, W. VARID: Viewpoint-Aware Re-IDentification of Vehicle Based on Triplet Loss. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 1381–1390. [CrossRef]
63. Li, K.; Ding, Z.; Li, K.; Zhang, Y.; Fu, Y. Vehicle and Person Re-Identification with Support Neighbor Loss. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 826–838. [CrossRef] [PubMed]
64. Meng, D.; Li, L.; Liu, X.; Gao, L.; Huang, Q. Viewpoint Alignment and Discriminative Parts Enhancement in 3D Space for Vehicle ReID. *IEEE Trans. Multimed.* **2022**. [CrossRef]
65. Krause, J.; Stark, M.; Deng, J.; Li, F.-F. 3D object representations for fine-grained categorization. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 2–8 December 2013.
66. Liu, H.; Tian, Y.; Yang, Y.; Pang, L.; Huang, T. Deep relative distance learning: Tell the difference between similar vehicles. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2167–2175.
67. Liu, X.; Liu, W.; Ma, H.; Fu, H. Large-scale vehicle re-identification in urban surveillance videos. In Proceedings of the 2016 IEEE International Conference on Multimedia and Expo (ICME), Seattle, WA, USA, 11–15 July 2016; pp. 1–6.
68. Liu, X.; Liu, W.; Mei, T.; Ma, H. A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 869–884.
69. Bai, Y.; Lou, Y.; Gao, F.; Wang, S.; Wu, Y.; Duan, L.Y. Group-sensitive triplet embedding for vehicle reidentification. *IEEE Trans. Multimed.* **2018**, *20*, 2385–2399. [CrossRef]
70. Guo, H.; Zhao, C.; Liu, Z.; Wang, J.; Lu, H. Learning Coarse-to-Fine Structured Feature Embedding for Vehicle Re-Identification. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence AAAI18, New Orleans, LA, USA, 2–7 February 2018; pp. 6853–6860.
71. Tang, Z.; Naphade, M.; Liu, M.Y.; Yang, X.; Birchfield, S.; Wang, S.; Kumar, R.; Anastasiu, D.; Hwang, J.N. Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8797–8806.
72. ElRashidy, A.; Ghoneima, M.; Abd El Munim, H.E.; Hammad, S. Recent Advances in Vision-based Vehicle Re-identification Datasets and Methods. In Proceedings of the 2021 16th International Conference on Computer Engineering and Systems (ICCES), Cairo, Egypt, 15–16 December 2021; pp. 1–6.
73. Song, Y.; Liu, C.; Zhang, W.; Nie, Z.; Chen, L. View-Decision Based Compound Match Learning for Vehicle Re-identification in UAV Surveillance. In Proceedings of the 2020 39th Chinese Control Conference (CCC), Shenyang, China, 27–29 July 2020; pp. 6594–6601.
74. Liu, C.; Song, Y.; Chang, F.; Li, S.; Ke, R.; Wang, Y. Posture Calibration Based Cross-View & Hard-Sensitive Metric Learning for UAV-Based Vehicle Re-Identification. *IEEE Trans. Intell. Transp. Syst.* **2022**. [CrossRef]
75. Yao, A.; Qi, J.; Zhong, P. Self-aligned Spatial Feature Extraction Network for UAV Vehicle Re-identification. *arXiv* **2022**, arXiv:2201.02836.
76. Jocher, G. YoloV5 by Ultralytics. Available online: https://github.com/ultralytics/yolov5 (accessed on 29 June 2022).
77. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
78. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
79. Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; Li, S.Z. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9759–9768.
80. Liu, S.; Huang, D.; Wang, Y. Learning spatial fusion for single-shot object detection. *arXiv* **2019**, arXiv:1911.09516.
81. Lee, Y.; Park, J. Centermask: Real-time anchor-free instance segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 13906–13915.
82. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 740–755.
83. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland; pp. 21–37.
84. JDAI Computer Vision. Fast-Reid Repository. 2021. Available online: https://github.com/JDAI-CV/fast-reid (accessed on 29 June 2022).