Running License Plate Recognition (LPR) algorithms on smart survillance cameras. A feasibility analysis

Leonardo Dominguez, Juan Pablo D'Amato
CONICET - Univ. Nac. Del Centro de la Prov. Bs. As.
PLADEMA,
Tandil, Argentina
{juan.damato@gmail.com}

Alejandro Perez, Aldo Rubiales, Rosana Barbuzza
CICPBA - Univ. Nac. Del Centro de la Prov. Bs. As.
PLADEMA,
Tandil, Argentina

Abstract - **In many of cities, video surveillance is employed to prevent crime, having hundreds of cameras and sensors monitoring activities during the whole day. Due to the huge amount of video information generated in real time, these surveillance centers are requiring more technology and intelligence to support human operators in many complex situations. There are important analyses that could be realized with this video-data: from criminalistics event detection to particular object recognition. One important tool is License Plate Recognition(LPR) that helps detecting vehicles, that could have been robbed. Although corporative solutions exist, these techniques require a lot of processing power and special located cameras, that not always could be afford by the local government.**

In this context, the proposed project is based on applying open-source LPR algorithms that runs on already existent surveillance cameras. These cameras are observing a complete scene (not just a line as it is commonly used), so LPR algorithms are rather slow, processing only 1 image per second. For this reason, the objective is to improve the performance combining parallel LPR processing and object tracking algorithms. This work describes the ongoing implementation, the techniques currently used for object tracking and LPR implementation, and exposes results regarding the efficiency of the solution.

Keywords: Image processing, license-plate recognition, Security, surveillance, open platform, digital government.

Keywords - component; formatting; style; styling;headings.

I. Introducción

Security has become one of the major shaping elements of government policy. The fear and uncertainty that high levels of violence and crime induce in society [1] are motivating factors for the development of new technologies to aid citizen security. This situation worsens in large cities, where violent events are more frequent. According to [2], between 2000 and 2008, the rate of imprisoned population in Latin America grew around 42%; at the same time, web searches in Spanish terms for "Burglary/Theft" and "Insecurity", for example, have risen continuously in the last 10 years.

There are known systems that help to solve the problem. In the United Kingdom, for example, it is estimated that every person in London is recorded on a common day by around 300 cameras [4]. The project [5] was conceived with a similar purpose, with the main goal of creating a large network of sensors and wireless devices, capable of supervising entire metropolitan areas in an intelligent manner.

In Latin America, and particularly in Argentina, local governments play a primary and active role in taking care of citizen security [6]. To face this problem, large monitoring centers are being built, based on CCTV technology. This model involves the installation of complex video camera networks distributed throughout the cities, requiring exclusive connectivity, and management and video recording software. Videos are stored for prolonged periods of time for further visualization, or to be used as evidence in a court of law. Monitoring tasks are performed by groups of people watching and reporting any event related to crime and accidents, uninterruptedly and with rotating schedules. In case of an emergency, security forces from local and state levels are deployed, working together with surveillance centers.

In this context, Automated License Plate Recognition systems (LPR or APLR) helps in the way that videos could be used. These systems convert the image (picture of the license) to computer data that can be checked against a database. The system includes cameras designed for this application, powerful software that provides consistent and reliable results and a computer (Digital Processing Unit) designed to handle the fast processing required.

Even though there are commercial solutions which solve a considerable amount of the issues involved, they are not always correctly adapted to legislation and country-specific interests. In addition to this, the "closed package" model they present works against the digital government model, which pretends to be open and transparent. At the same time, in general LPRs only work on cameras located at a specific height and observing only a part of a place, in order to obtain a good detection rate.

Furthermore, these systems generally follow a centralized scheme, where all the analysis is performed in a monitoring center, requiring very specific equipment. Given that the cost of embedded processors has decreased considerably, it is natural to consider a distributed design for image processing in order to scale the system rapidly. Under this scheme, solutions must coexist with multiple connected devices which perform analysis on site, reporting only when an irregular situation is detected

In these cases, to achieve an architecture with the ability to adapt to multiple situations in a simple way, it is vital to count with an accessible, transparent and inter-connected platform.

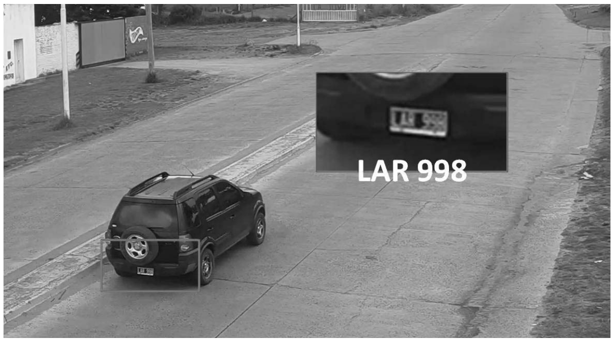

Fig. 1. A LPR detection case

A. *Work proposal*

Security is not only a public concern but also an academic one. Our project emerges as an answer to social demands regarding security and in a close relationship with the continuous rise of camera installations in public spaces. This work proposes a distributed platform for managing and sharing IP cameras with embedded processing via different kinds of networks, taking advantage of a collaborative philosophy aimed to achieve a general goal: public safety. This platform, that was already presented in [13], is extended to support plate recognition.

The platform takes into consideration the automation and integration of tasks related to the detection and tracking of objects and individuals with multiple video cameras. The innovative design of the platform allows for the usage of various analysis techniques depending on the situation, which can even execute on different de-vices (servers, micro-PCs, embedded systems). That is why its architecture is open and distributed, making it able of processing "in situ" to enhance reactivity. Only relevant events are transmitted to a central server, improving scalability without de-creasing global performance.

This paper is organized into five sections. Section 2 summarizes the state of the art and considers existing tools with similar purposes. Section 3 describes the developed architecture and detectors. Section 4 shows preliminary results regarding the LPR analysis feasibility with the issues found. Finally, Section 5 offer conclusions and future work.

II. State of the Art

As a result of enhanced computation power from current market processors, there are abounding video surveillance systems available and recent bibliography addressing real time people tracking, making it a deeply studied subject.

Regarding algorithms, works such as [7] describe some common methods for object detection and tracking. In [8], an interesting comparison between multiple detectors is made, where it is concluded that a combination of them can be useful to re-duce the rate of false positives without compromising real-time analysis. More recently, in [9], general strategies for vehicles tracking and classification from video were reexamined. When the scenes are noisy, and with plenty of objects, new approaches are required. For instance, [10] and [11], that are using deep learning in order to improve character recognition. These algorithms should be combined with object characterization in order to have a good detection rate. Considering integral existing solutions, [12] makes a review of 18 commercial surveillance systems, and a summary of the most used algorithms for object detection, tracking, classification, event detection and trajectory reconstruction.

III. Proposed Solution

The main purpose is to automatically detect license plates and store in a database or notify an operator to react swiftly and take an action. This process can be done by auto-mated object and behavior recognition, in order to select situations where human intervention is required.

Against conventional LPR systems, the main challenge was to used already installed IP surveillance cameras. As it is known in **Figure 2**, the main difference between each system is how cameras are located. In common LPR systems, the camera is focused on a certain part of the image, aiming at the car's front. On the other side, surveillance ones, has a broader focus, trying to take the whole scene.

This configuration comparison is presented in Figure 3.

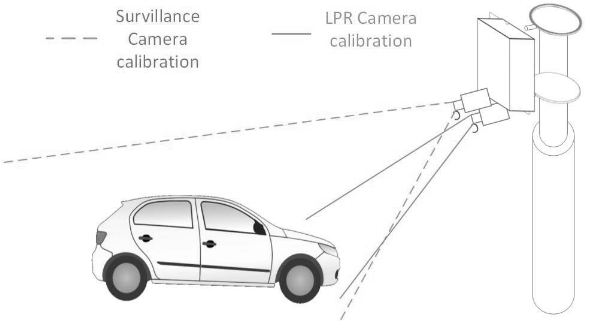

Fig. 2. Different camera configurations

Even that this seems to be a complex task, it is known that new cameras can record at full-HD resolution (1920 x 1080 pixels). With this quality, in many cases the operators could manually recognize the plates, so the idea was to evaluate if could be done by computational means. At the same time, LPR algorithms are high CPU consuming and in general only a part of the image is of interest. In order to reduce this complexity, a tracking based method was used. Only moving objects, that came close to the camera are evaluated. Other object properties, such as color, type or brand are of interest, so they whole object is analyzed

To accomplish this task, we propose to analyze each new frame from a camera stream in multiple ways. First, a fast object detector is applied to the regions of interest, to discern moving parts of the image from the static scene. This constitutes the *real-time* pipeline, and gives as an output an enriched stream with metadata that can outline detected objects. After the detection of a potential object, more complex characteristics are detailed. This new analysis stage involves computationally expensive algorithms (for example, object recognition algorithms); for that reason, it is executed asynchronously. This constitutes the *deferred pipeline.*

Finally, detected objects and associated analysis results are stored in a distributed database that can be accessed from any interested device in the network.

A. Processing Distribution for Real-time Motion Detection

The first step in fast event detection is to localize motion in series of consecutive frames. This problem is well studied in the field of background subtraction algorithms. Many solutions can calculate image pixels variation [13]14]. Methods vary in effectiveness and computational complexity. Currently, the platform makes use of a version of ViBe [4] to improve results.

When cameras are placed on outdoors, factors associated with variations of time of the day, cloud conditions and time of the year are exacerbated. To counteract these effects, motion windows are used as mentioned in [??]. Their goal is to apply a logical operation to process only movements detected inside the window dimensions and filter the ones outside.

Given the computational cost, a dedicated processor is required to execute this task. To improve global efficiency, processing was extracted to the video-camera location. Watertight cabinets were designed to that end, with the capacity to allocate a mini-PC connected directly with the camera. The PC performs the analysis of the objects and communicates the results to the central server, along with the video frame. The output of this step is a set of pixels (grouped or not) that have been classified as relevant to further describe objects in motion.

Object characteristics are then described using tracking techniques, such as its size, velocity direction, and last positions. If an object size is considerably small (too few pixels), it is not considered for further analysis. Otherwise, it is uniquely identified inside the network, registered in the system, and assigned for pending characterization. Object tracking is then based on motion detection events from the real-time pipeline. After an object detection, a new message under the topic new object is created, which includes the object ID, a cropped portion of the image where the object was found and, if necessary, any information obtained as a result of the analysis (for example, direction and speed).

Data and images are serialized as a JSON message and published on the platform. Modules listening to the topic will read, deserialize and process the message. Another message is then published to store the results. An example of a message is shown below. Messages are configured according to the type of analysis being applied. Also, it is possible to specify whether the modules run on a single device/computer or in a distributed environment.

B. LPR algorithms

For each new detected object in real time, a LPR algorithm is invoked. This algorithm has three main steps:

- License plate detection
- Character segmentation
- Character recognition

The most computational expensive task is the first one, where a white rectangle is search in the image. For this task, cascade classifiers are commonly used, and this could run both in a CPU or in a GPU based hardware platform.

The input image is the already detected object, coming from the real- time pipeline. If the object size is too small, the image is discarded. Then, the plate detection is applied. If it is found a match, the algorithm continues, until trying to detect the whole plate. This algorithm could be running locally in the camera or in a centralized server that is receiving the objects detected.

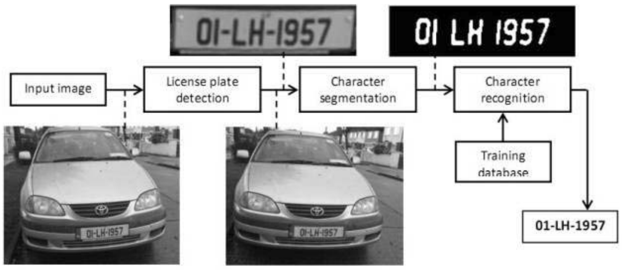

Fig. 3. A LPR detection case

As the images are coming from video, there are several pictures from the same object, and even several objects could be observed. By now, all the object cropped-images are evaluated, but in the future a priorization rule must be applied.

IV. Preliminary Results

The platform is operational, currently on a prototype stage, and under constant development by the authors of this paper. It is instantiated on a college campus located in the city of Tandil, Argentina. Cameras are connected to a local network with an average bandwidth of 100 Mbps. All video streams are registered and managed through a desktop application inside the platform. The database is an SQL Server with an educational license, although a free license would be sufficient for the features required. Cameras can also be accessed via web, with an application implemented with AngularJS, that counts with an adaptable UI accessible from mobile devices and in low bandwidth conditions.

To evaluate the feasibility of applying LPR algorithms, we carried out two kind of tests. On the first side, considering processing speed, in order to quantify if CPUs can handle several analyses at high speed. On the other side, validating if full-HD video quality is enough for this task.

A. LPR performance evaluation

The goal of a first series of test was to assess the impact of processing in real-time video images, trying to detect license plates. For the analysis, we used a variation of the openALPR [15], with different inputs and hardware configurations. The input video comes in 1920x1080 resolution, at about 20 frames-per-second (20fps). The algorithms run on an industrial PC, with an Intel i7 4500U processor, 4GB of RAM and a 6000 Intel graphics card series, supporting GPU programming. This hardware is already in used in our platform.

We studied three configurations:

- LPR applyed to the whole image
- LPR on moving area
- LPR on moving area with GPUs

We applied the process to a 600 frames video. Time results are presented in **Figure 4**, measured in milliseconds:

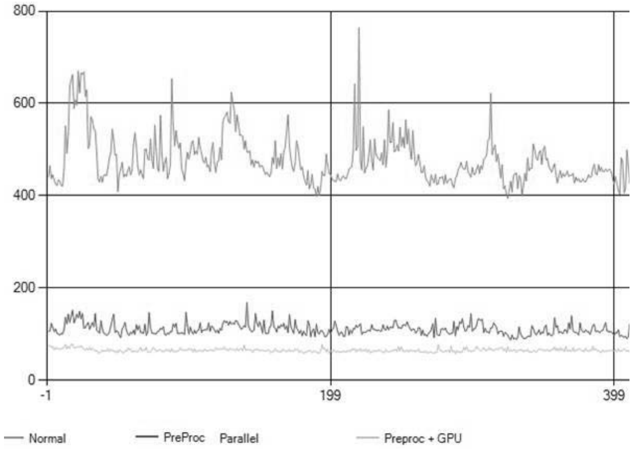

Fig. 4. Comparison of the variants of LPR algorithm

As it was well-known, evaluating the whole image is not feasible, as only a 1 frame per second could be processed. The second case, working on the moving area was better, reaching about 9 fps processing rate, but taking the CPU load to 100%. Finally, running the plate-detection algorithm on the GPU increased speed, getting about 15 fps. In this case, it was even better, because CPU load was only of 30%.

LPR detection performance

A second series of tests were concerned with different camera locations, in order to detect and recognize license plates. We used two cameras, with different points of view. We selected 13 images in each camera that has a near legible plates. Test results are shown in Table 1.

Table 1. : Results on two different camera configurations

	Recognitions	**False positive**
	5/13	8/13
	3/13	10/13

These results are quite reasonable. The second camera had lower resolution quality. Objects on both were relative small, with about 500x500 pixels representing each, which are very bad quality compared to those required in commercial LPRs but enough to show a potential improvement.

V. CONCLUSIONS

In this work, a first approach of using existent surveillance cameras for detecting license plates is presented. An open source platform is used and improved to evaluate only objects that are moving. The algorithms run on a distributed platform that operate multiple cameras and sensors, to support comprehensive security management. The first results are reasonable in performance, reaching near a real-time LPR detection, even for high resolution images. On the other side, the true positives are less than expected. One of the reasons is that the algorithms required to have the plates more focused. On the other side, text is readable, so we think that it could be solved using smarter algorithms that could be trained with this kind of images.

ACKNOWLEDGEMENTS

The current project has received grants from the National Scientific and Technical Research Council (CONICET), project PICT-2016-0236 and from the Scientific Research Council (CICPBA), project ARCT 17.

REFERENCES

[1] Kessler G.: “El sentimiento de inseguridad: sociología del temor al delito.” Siglo Veintiuno Editores. Buenos Aires, Argentina (2009).

[2] Dammert L., Salazar F., Montt C., González P.: “Crimen e inseguridad: indicadores para las américas.” FLACSO-Chile/Banco Int. de Desarrollo. Washington D.C., USA. BID (2010).

[3] Bouwmans D., Porikli F., Hoferlin B., Vacavant A.: Background Modeling and Foreground Detection for Video Surveillance. Chapman and Hall/CRC, London (2014).

[4] Piccardi M.: “Background subtraction techniques: a review.” In: Systems, man and cybernetics, 2004, IEEE international conference on, vol. 4, pp 3099–3104. IEEE (2004).

[5] Calavia, L. Baladón, C. Aguiar, J.M., Carro, B., Sánches-Esguevillas, A.: “A semantic autonomous video surveillance system for dense camera networks in smart cities,” Sensors, 12(8), 10407-10429 (2012).

[6] Carrión F., Pontón C., Armijos B.: “120 Estrategias y 36 experiencias de seguridad ciudadana,” Revista Latinoamericana de Estudios de Seguridad, (2009).

[7] Serby, D., Koller-Meier E., Gool, L.: “Probabilistic object tracking using multiple features.” In: Pattern Recognition, 2004, Proceedings of the 17th International Conference on, vol 2., pp. 184-187. ICPR (2004).

[8] Hall D., Nascimento J., Ribeiro P.: “Comparison of target detection algorithms using adaptive background models.” In: Visual Surveillance and Performance Evaluation of Tracking, 2005, 2nd Joint IEEE Int. Work., pp 113–120. IEEE (2005).

[9] Ojha, S., Sakhare, S.: “Image processing techniques for object tracking in video surveillance - A survey.” In: Pervasive Computing, 2015, Inernational Conference on, pp. 1-6 (2015).

[10] H. Li and C. Shen. Reading car license plates using deep convolutional neural networks and lstms. arXiv preprint arXiv:1601.05610, (2016).

[11] A. Vedaldi M. Jaderberg, K. Simonyan and A. Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. arXiv preprint arXiv:1406.2227, (2014).

[12] Gdanks: “Review of existing smart video surveillance systems capable of being integrated with ADDPRIV project” (2011)

[13] D’Amato J.P. , Dominguez L., Perez A., Rubiales A.: “Plataforma abierta de gestión de cámaras IP y aplicaciones móviles para la seguridad civil ciudadana,” Revista Ibérica de Sistemas e Tecnologias de Inormaçao (20), 48-61 (2016).

[14] Shaikh S. H., Saeed K., Chaki N.: Moving Object Detection Using Background Subtraction. Springer International Publishing, Berlin, Germany (2014).

[15] The Open LPR initiative : web-site = https://github.com/openalpr/openalpr.