DeMS White Paper

Stanley Salvatierra[1]*, Alvaro Hurtado[2]

Abstract
Privacy concerns and the high demand of GPU capabilities are some of the main problems for real-time video processing when dealing with security and safety in the industry and urban areas, thus, generating high demand for suitable emerging technologies. The rising power of small computers and the increasing capabilities of machine learning techniques are allowing us to design devices capable of detecting events in real time without the need of Internet connectivity or cloud computing access; therefore saving money and solving the increasing privacy problems for business and government institutions. In Deep Micro Systems we succeeded to combine several techniques to detect events in real-time video and exploit the combined information of several cameras connected to a single device, achieving a reasonable frame rate and sufficient resolution to capture the required details.

Keywords
Computer Vision — Smart Cities — Off-line computing — Artificial Intelligence

[1] *Chief Technology Officer, DeMS, Smart Cities*
[2] *Chief Executive Officer, DeMS, Smart Cities*
***Stanley Salvatierra**: s.salvatierra@deepmicrosystems.com

Contents

Introduction

Privacy concerns, reduced Internet connectivity, and low computational power are some of the main reasons to improve real-time video processing in local devices. The requirements for Internet connectivity and GPU services are some of the enormous problems that are particularly limiting in developing countries.

Computer vision has proved to be of great help for industries and governments aiming to increase security and safety at competitive prices. Embedded systems have limited power; thus, the need for novel computer vision techniques and new hardware are essential to handle more complex tasks.

Some specific tasks can offer some facilities to real-time video monitoring. That is the case of fixed cameras, surveillance cameras and security cameras that can handle only the changing pixels of video to reduce complexity or can make use of computer vision techniques to stabilize the frames in case of a sudden shake of the camera.

Simple computer vision techniques require low computational power and can be tailored to focus machine learning techniques on particular tasks.

New and more powerful small computers are also appearing such as Orange Pi, Raspberry Pi 3 B + and even neural sticks like Intel Movidius, which are capable of handling modern machine learning

techniques.

1. Deep Micro Systems

Deep Micro Systems is a Bolivian Startup in the fields of Computer Vision and Smart Cities, with the mission to inform and to provide solutions to critical problems in the industry and government.

Road Security   The high number of roads accidents is currently of great concern for the Bolivian government with about 1300 deaths every year. New technologies are needed to reduce these large numbers. As Internet connectivity in Bolivia is expensive and unreliable compared to our neighboring countries, the need for off-line processing is a major concern. To address this problem, Deep Micro Systems designed an All-In-One device capable of detecting traffic Infringements, producing a visual proof of the act and at the same time generating valuable real-time data for city planning.

Our camera is capable of detecting events in real-time video identifying the vehicle violating the law.

Applications of computer vision   Besides road infringement detection, Deep Micro Systems works with computer vision systems with a broad perspective with applications in industry and government. Some of the other projects that Deep Micro Systems is planning to get involved in the short term are:

- Optimal city design, parking, roads and flows
- Traffic accidents forecast
- Smart traffic light managing
- Industrial applications of computer vision.

Industry   Local Bolivian business such as Enalbo and IBCO SRL expressed their interest in detecting real-time events such as security infringements of their personal, abnormal functioning of equipment and other failures.

All-In-One devices

The lowering costs of pocket computers and the increasing power and efficiency of machine learning techniques allow us to build small devices capable of solving important problems in urban areas and

Figure 1. All-In-One computer vision device

the industry without the need of Internet connectivity, guaranteeing privacy and cheaper uninterrupted work.

Computer processors such as Cortex-A53 (ARMv8) 64-bit SoC with 1.4GHz, and ARM H3 Quad-core Cortex-A7 with 1.6GHz are capable of handling complex tasks and can make use of the new neural chips to get even faster results in real time.

All these components allow us to build custom devices for solving specific problems, providing Ethernet connectivity, GPS, GSM/GPRS, Bluetooth, WiFi and any sensors to be required.

Currently, Lu-Cam, our smart traffic enforcement camera is an 8x9x16 cm device capable, as seen in figure 1,and it is able of detecting events in real-time videos. Rather than getting frame features with convolutional neural networks, that would take several seconds to complete, we make use of the advantage of knowing the primary purpose, extracting only relevant features and making an improvement to less of a tenth of the time it would require otherwise.

Real time monitoring and data analysis

Such All-In-One Devices hold a huge advantage, allowing to pre-process data. For example, real-time road monitoring could generate real-time histograms of infringements with proper labeling as shown in figure 2. Then a central computer can handle data from several devices to make decisions,
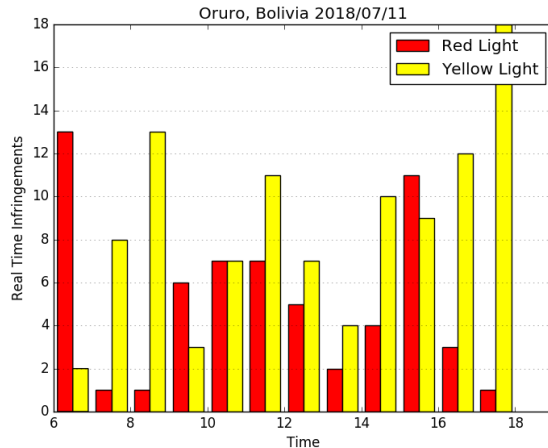
Figure 2. Real time histogram

improving efficiency or automating complex processes.

2. On the Edge Computing

Conventional visual monitoring systems, CCTV, IP cameras, send video transmission, analog or digital, to a data center to be stored and eventually processed or reviewed. Embedded smart cameras process image data directly on-site allowing several advantages over the last systems, in this paper we present:

- *Our approach* to an on-board smart camera computing device.
- *Hardware and Software description*.
- *Actual real Use Case implementation* of our product.

Nevertheless, there is a strong demand for mobile vision solutions ranging from object recognition to advanced human-machine interfaces.

Our Approach

The problem of on-board computing and further analysis of real time video is subject of current research. We follow the next procedure which is focused on 3 main cycles that are plot in figure 4 and are stated as follows:

1 *Algorithm Design*. If the client requirements does not have precedents, we partially hand

craft a Computer Vision Algorithm specific for our client use case for data mining, cleaning and analysis.

2 Data Analysis for client use cases and results presentation according to initial requirements.

3 *Improvement*. Improve original algorithm using the proprietary collected data with the use of Machine Learning techniques, together with a hardware upgrade if required.

We dedicate a specialized team to solve the problems in each cycle. The case study that we present in the whole proposal, completed 2 of the 3 states of our cycle of work, and we are currently working on an improvement of our main hardware and algorithm to provide the best results to our clients. The collected data is a huge advantage above other competitors and allows us to improve existing products and create new ones.

Hardware and Software description

Below there is a description of the hardware and software we use.

Software Description

If the requirements are very specific and no historic data is available, we design an algorithm with traditional computer vision techniques according to the use case with focus on the embedded hardware; we maintain a constant monitoring of the program for the event that client is looking for and at the same time we collect data for three main purposes: Improve the current algorithm with Machine Learning techniques; Bring data for further applications; Create new useful labeled data in the field of work.

Our main tools for the creation of custom algorithms are Python and C++ as main languages. Open-CV is our compendium of multiple purpose Computer Vision techniques which are used to hand craft the desired behavior of the program and data recollection.

Hardware Description

The figure 5 shows some main aspects of our smart-camera and contains:

Figure 3. Time synchronization between cameras
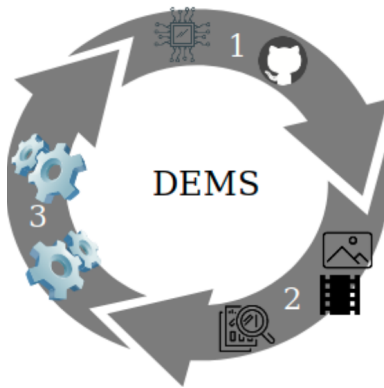
Figure 4. Cycle of work

- A low resolution camera, used to detect flow in real time.

- A 8 MP camera sensor is used for capture the main aspects of interest when the low resolution camera triggers the signal of event of interest, in real time.

- We also develop custom printed circuit boards to handle different sensors or actuators like the infra-red filter switch for night vision improvements. Custom boards can include several sensors or actuators like GPS, GSM and others

The compute capabilities of our embedded device are described below:

- SoC: Broadcom BCM2837B0 quad-core A53 (ARMv8) 64-bit @ 1.4GHz
- GPU: Broadcom Videocore-IV 256 MB VRAM
- RAM: 1GB LPDDR2 SDRAM
- Networking: Gigabit Ethernet.
- 802.11b/g/n/ac Wi-Fi

- Bluetooth: Bluetooth 4.2, Bluetooth Low Energy (BLE)
- Storage: Micro-SD
- Multiple GPIO header
- Ports: HDMI, analogue audio-video jack, USB 2.0, Ethernet
- Camera Serial Interface (CSI), Display Serial Interface (DSI)
- Optional - Neural Network Compute Capabilities according to use case.

Actual real Use Case implementation

In figure 6 we present an implementation schematics of our camera.

1 - On Board Computation according to client requirements, communication to a private cloud can be achieved if required.
2 - Mounting pole, in some applications the height of the device plays an important role in the algorithm behavior.
3-8 - Specific working environment, in this case for traffic infringements detection.
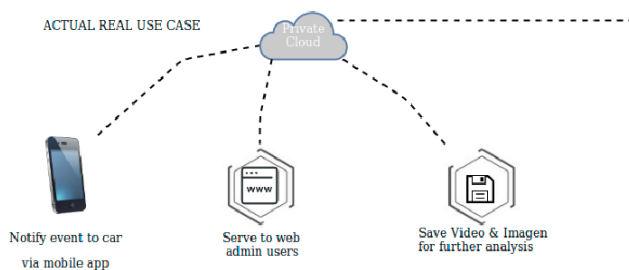4 - Optimal vision angle for plate detection.
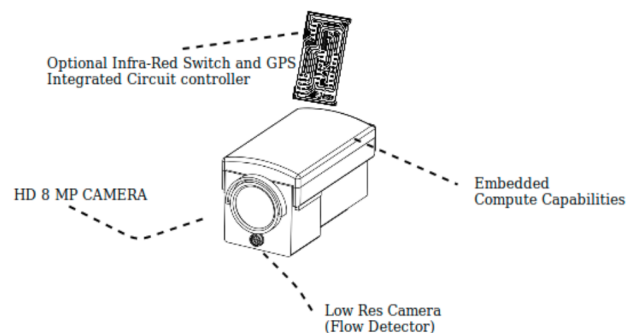
Figure 5. Hardware schematics
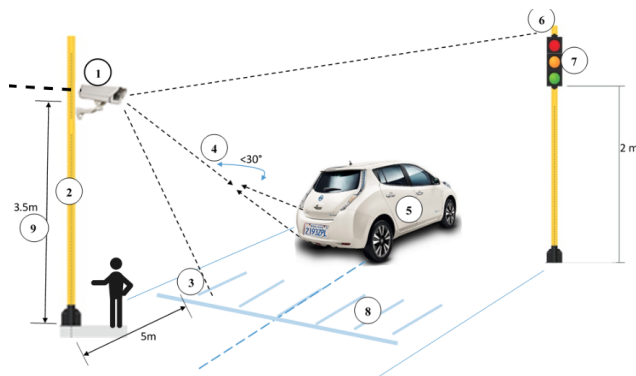
Figure 6. Actual use case of our embedded camera

Figure 7. On-line process for the information detected in the road.

    5 - Target object, in this case, cars.

    6 - Traffic light mounting pole,

    7 - Traffic Light signal. The camera is capable of detecting color change if the traffic light is within it's range of optimal working, it is implemented by software but can be implemented as hardware if the client requires.

Figure 3 shows the extract of 5 seconds video from low resolution camera and a picture of High Resolution for target object with region of interest, this two assets, video and picture is the output of our custom algorithm in real time.

As stated before, if an Internet connection is available, we can make a wingspan of the information of video/picture in real time and serve it to different users as required, or save it in a server for further analysis, figure 7 shows the camera and system online behavior.

Currently we are training a Neural Network Algorithm to support the Hand Crafted algorithm with the information of video and image generated from the beginning of this year 2018. Following our cycle of work of Figure 6, we have generated

enough labeled data for our use case and we are ready to start a new cycle of production.

You can see the partial results of this project in our page www.demsbo.com and our Github project https://github.com/alvarohurtadobo/prototipo.

3. Further applications
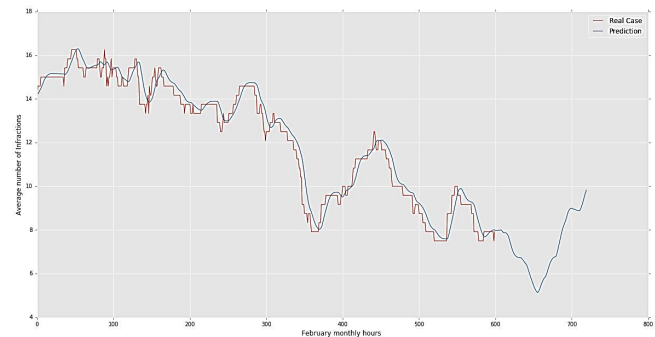
Figure 8. Real time Traffic Forecast

The applications for this kind of devices are endless. Real time monitoring and data analysis can lead us to a better understanding of cause problems and possible solutions to them thanks to data science and big data techniques.

Forecasting data is another example of an application of monitoring systems, in the case of real time traffic monitoring it can lead to traffic forecasting, saving thousands in time, fuel and possible accidents happening in streets and roads across Bolivia and the world, forecast is possible with a little amount of devices like shown in figure 8

Real time simulation is also possible thank to the real time and flow data in the devices, allowing us to extrapolate data to points where it is not possible to place a monitoring device. Like aggressive environments in industries to complex roads in cities.

Acknowledgments