OD Matrix Estimation of Traffic Flow on a Highway Network

Kristian Kovačić
Sergio D. Varela
Supervisor: Asst. Prof. Edouard Ivanjko

int12sva@student.lu.se
kkovacic00@gmail.com

- **Sergio D. Varela**
  - Live in Madrid, was born in Colombia
- **Technical University of Madrid**
  - Established in 1971 but most of the centres were founded during the 18th, 19th or beginning of 20th century
  - 4 campuses with 20 Schools of engineering
  - 35,000 students
- **Telecommunications Engineering School**
  - Established 1957
  - 501 professors/2,600 students.
  - Telecommunications, Biomedicine Degree

- **Double Degree at Lunds Universitet**
  - Founded in 1666
  - 3,994 academic staff, 28,000 students
- **Lund Tekniska Högskola, Faculty of Electrical Engineering**
  - Esp. Design of Processors and Digital Sytems.
    - Digital IC Design
- **Hobbies: running, cooking, TV shows, cars**

Outline

- **Introduction**
- **Vehicle detection**
  - Applied Approach
  - Enhancements
- **License plate recognition**
- **Experimental Results**
  - Problems detected
- **Conclusion & Future work**

Introduction

- **Project Computer Vision Innovations for Safe Traffic (VISTA)**
- **Implementation of modern computer technology for traffic improvement***
  - Surrounding view parking assistance
  - Detection of road side vegetation
  - Automatic headlight detection
  - Lane departure and collision warning
  - Traffic sign detection and recognition
  - Lane detection and recognition

*** P. Cerri, G. Soprani, P. Zani, et. al. – „*Computer Vision at the Hyundai Autonomous Challenge*" – 2011, 14th International IEEE Conference on Intelligent Transportation Systems, Washington, US**

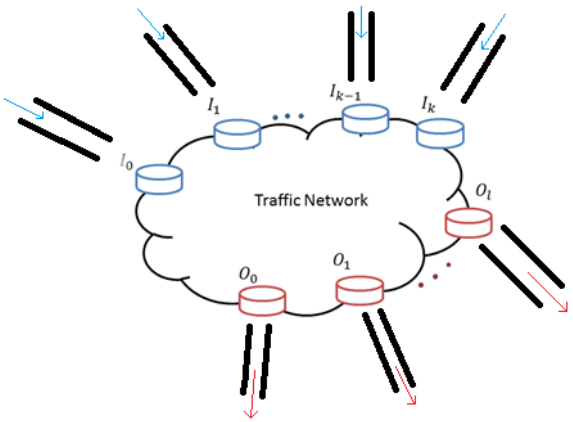
Introduction

- **Participants:**
  - Faculty of Electrical Engineering and Computing (UNIZG-FER) as leading institution
  - Faculty of Transport and Traffic Sciences (UNIZG-FTTS) as partner institution
- **Visiting student contribution to the project VISTA**
  - Testing and evaluation work package
    - Vehicle detection
    - Vehicle identification using license plates

- **Traffic management**
  - Dynamic behavior of a traffic system derived from known parameters
  - Driver information system
  - Rerouting links between nodes in a traffic network
- **Origin-Destination analysis of highways traffic**
  - From traffic network only Inputs and Outputs known, not what's inside
  - Create a matrix that estimates the traffic flow between Inputs and Outputs
  - Possibility to estimate the route of a traced vehicle

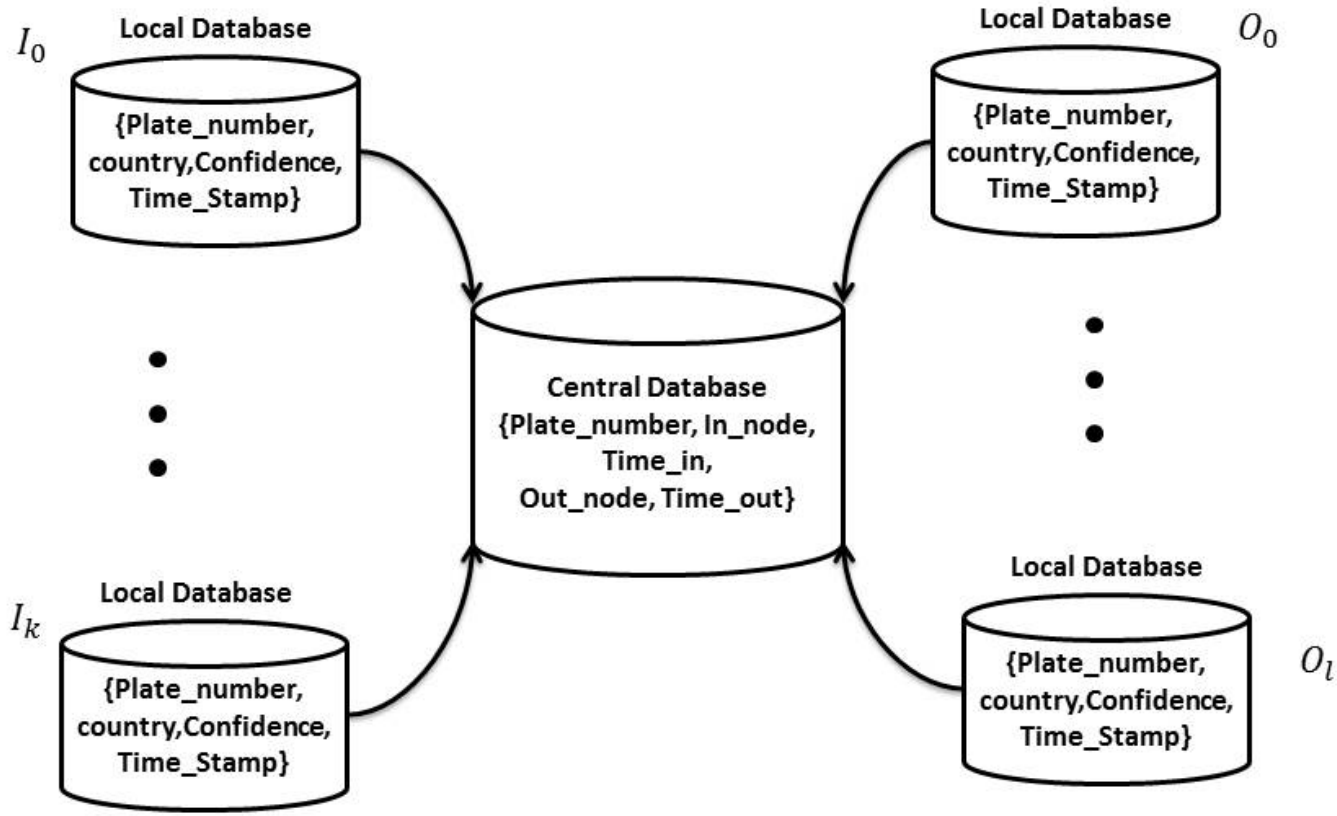
$$\mathbf{T} = \begin{pmatrix} T_{11} & \cdots & T_{1l} \\ \vdots & \ddots & \vdots \\ T_{k1} & \cdots & T_{kl} \end{pmatrix}$$

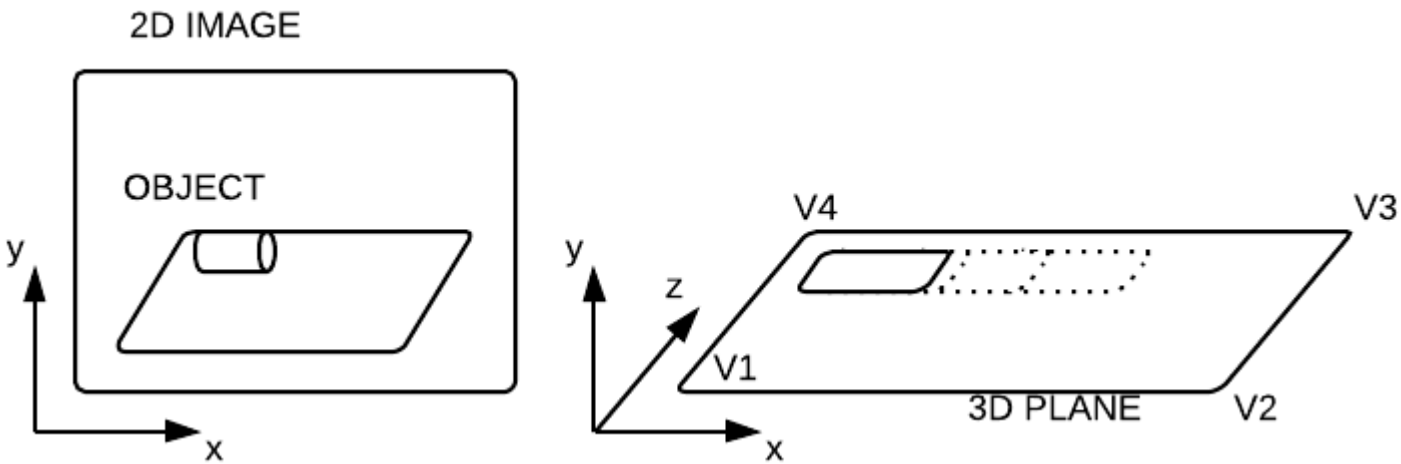
$$T_{ij} = \sum Car_{InNode_i \rightarrow OutNode_j}$$



- **For an effective monitoring of traffic network, key parameters are needed**
  - Traffic flow [v/h]
  - Average speed of a flow [km/h]
  - Queue length on a road [m]
  - Clasification of a vehicle [car/truck/bus/motorcycle]
  - Connectivity between traffic network nodes

Structure of local and central databases



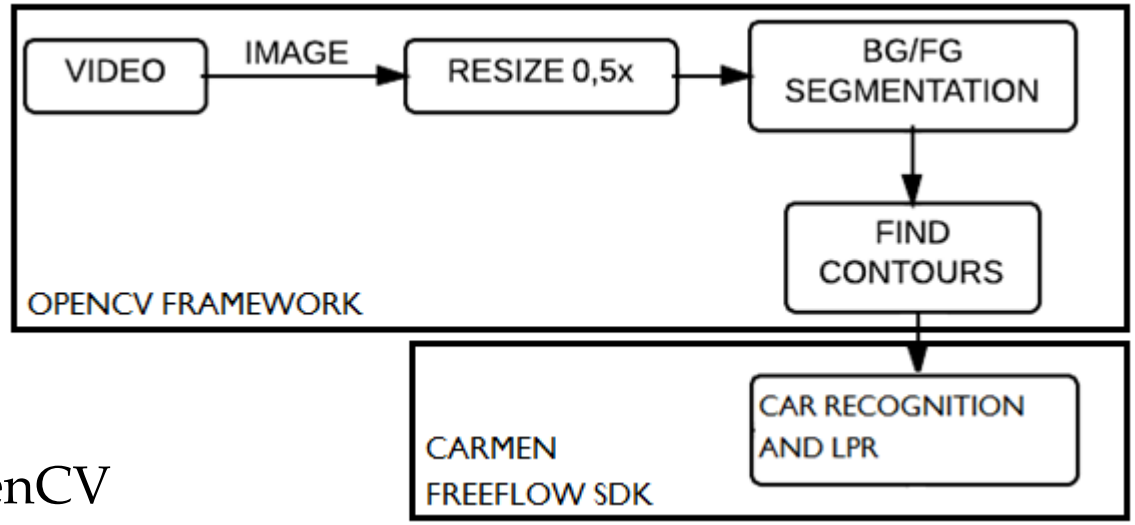


- **Motion detection***
- **Edge detection (parking space)****
  - V1 = const, V2 = const, V3 = const, V4 = const
- **Camera calibration**
- **Physics engine (rules)**

* **V. Braut, M. Culjak, V. Vukotic, S. Segvic, M. Sevrovic, H. Gold – *„Estimating OD matrices at intersections in airborne video – a pilot study"* – MIPRO, 2012 Proceedings of the 35th International Convention, pp. 977-982**

** **G. Wenshuo, Z. Xiaoguang, Y. Lei, L. Huizhong – *„Edge Detection of Images Based on Improved Sobel Operator and Genetic Algorithms"* – 2010. 3rd IEEE International Conference on vol. 5., pp. 67-71**

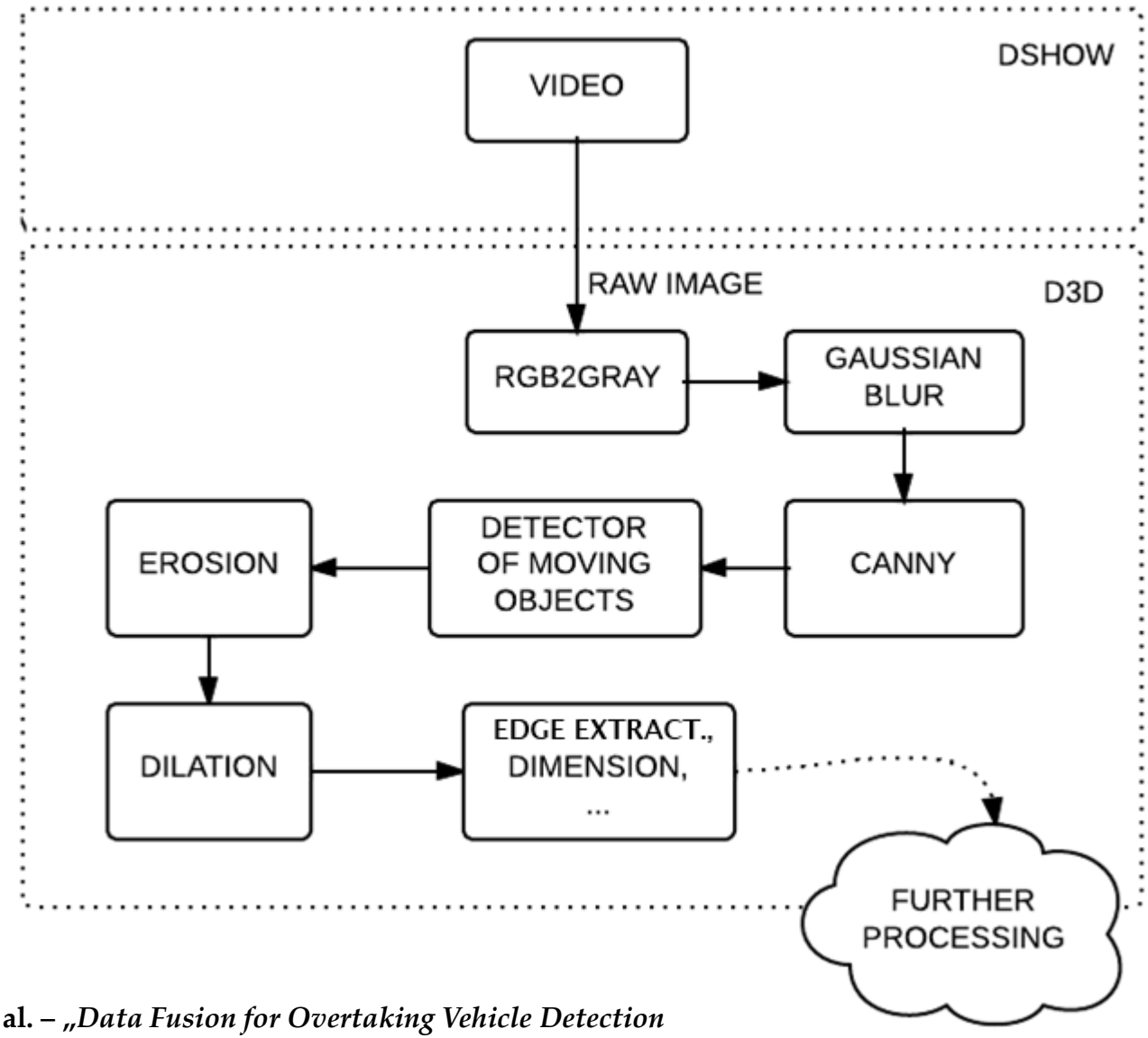
- **Based on OpenCV and CARMEN frameworks**



  - OpenCV
    - Decode an image from a file
    - Downsampling the image
    - Bg/fg segmentation of the image*
    - Object contour finder (based on edge detection)
  - CARMEN
    - Perform license plate recognition (LPR) on the specific image

***Z. Zivkovic – „Improved adaptive Gausian mixture model for background subtraction" – International Conference on Pattern Recognition, UK, 2004.**

- **Program work flow**

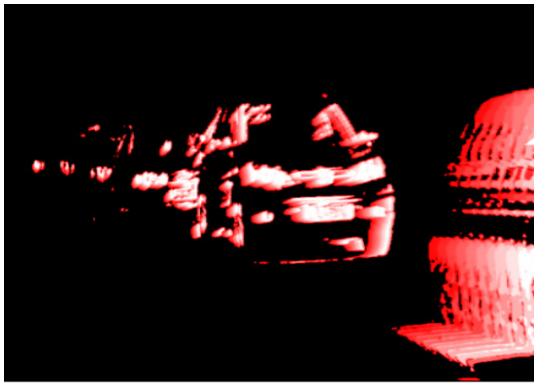


* F. Garcia, P. Cerri, A. Broggi. et- al. – *„Data Fusion for Overtaking Vehicle Detection based on Radar and Optical Flow"* – 2012, Intelligent Vehicles Symposium, Alcalá de Henares, Spain

- **Filters used in program**

1. GAUSSIAN BLUR
2. CANNY

1. GAUSSIAN BLUR
2. FG/BG SEGMENTATION



1. CONVERSION TO BINARY IMAGE
2. OBJECT GROUPING

+

Detected vehicle

License Plate Recognition

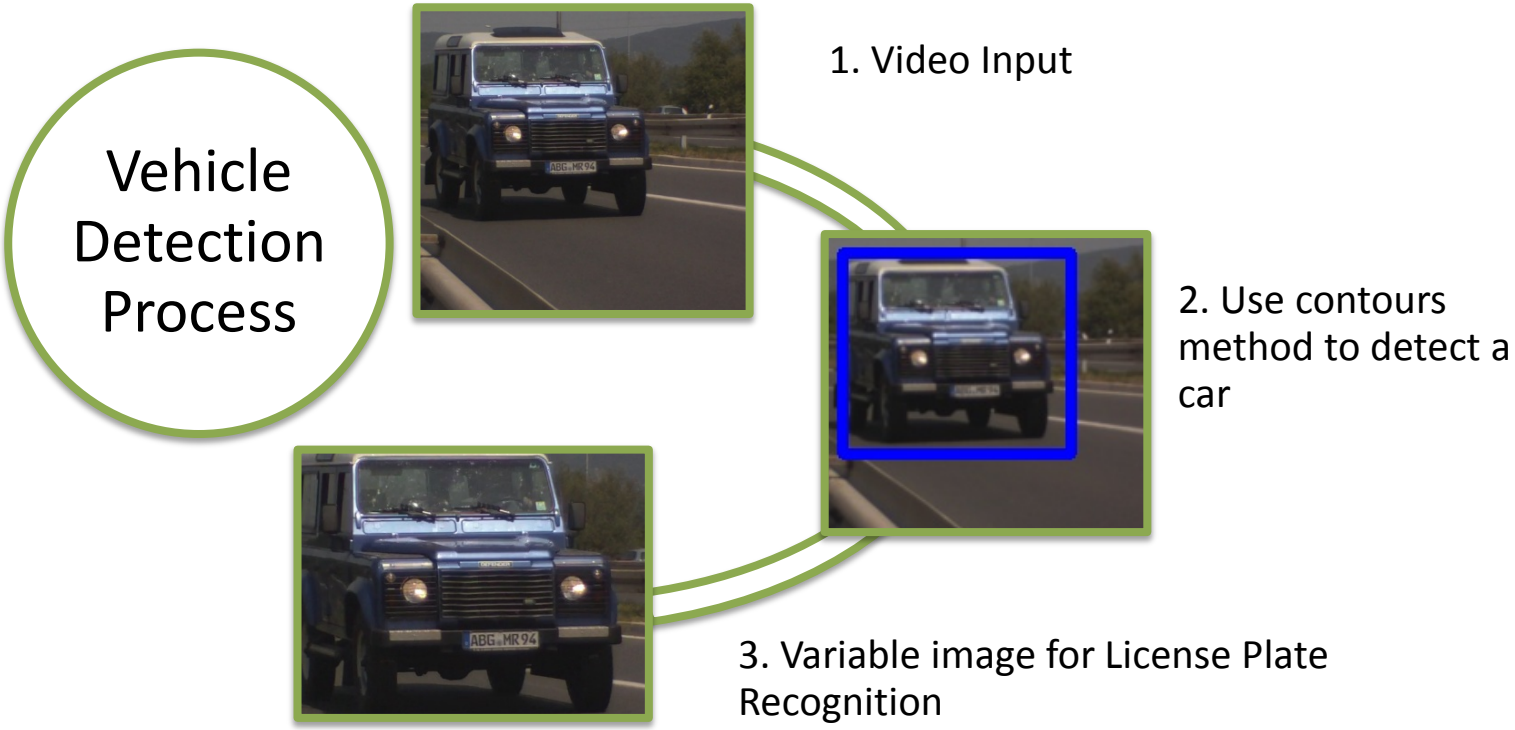
- **Plate recognition is carried out by CARMEN® ANPR Engine**
  - Developed since 1997 by ARH Inc. based in Hungary
  - Software provides number plate and other attributes
    - Plate type → Country
    - Coordinates where the plate was recognized
    - Confidence of the recognized plate

ARH
Adaptive Recognition Hungary

Experimental Results

- **When a car is recognized, a smaller image is send to Recognize() method, which is the one that does the License Plate Recognition**
- **ANPR Engine used to recognize the number plate**
- **Constrains to analyse the traffic flow were set after video analysis**
  - Confidence Trigger to accept a plate as correct 40 (%)
  - Time between two different cars 0.5-1 second

- **Real Time System video Detection**
- **First Estimation: Processing of the recorded video**
  - OpenCV: Contours method to recognize a car



- **For performing LPR on vehicle, an image needs to be cut**
- **Cutting the image can be done using**
  - Variable image size (defined in run-time)
    - Not efficient method to recognize a car plate due to slow execution
  - Constant image size (hardcoded)
    - Faster due to fixed image coordinates for LPR
    - No contours computation
- **Calculation time for each frame in a video**
  - Variable image size ≈ 380 [ticks]
  - Fixed image size ≈ 177 [ticks]
  - Difference ≈ 46 [%]

Experimental Results

Developed application

CRV (Running) - Microsoft Visual Studio Express 2012 for Windows Desktop

FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS TEST WINDOW HELP

Continue Debug Win32

Process: [6448] CRV.exe Suspend Thread: [6108] msvcr80.dll thread Stack Frame: 77b7fd71

No Source Available lpr.cpp main.cpp

(Global Scope) main(int argc, char * argv[], char * envp[])

```
        pyrDown(matImgSm, matImgHD);
        imshow(wndName1, matImgSm);

        bgSub(matImgHD, fgMaskOpened);
        bgSub.getBackgroundImage(bgMask);


        cont_init = clock();
        findContours(fgMaskOpened, contours, hierarchy, CV_RETR_EXTERNAL, CV_CHAIN_APPROX_TC89_KCOS);
        for (int idx = 0; idx < contours.size(); idx++)
        {
            Rect    box     = boundingRect(contours[idx]);
            INT     area    = box.width * box.height;

            if ((area < areaROImax)&&(area >= areaROImin))
            {

                Point   p1  (box.x, box.y);
                Point   p2  (box.x + box.width, box.y + box.height);
                rectangle(fgMaskOpened, p1, p2, Scalar(255, 0, 0), 5);
                rectangle(matImgHD, p1, p2, Scalar(255, 0, 0), 5);

                if (counter >= FRAMES)
                {
                    img_rec_init = clock();
                    timeStamp = vCapHD.get(CV_CAP_PROP_POS_MSEC);
                    /*pHD1.x = p1.x*4;
                    pHD1.y = p1.y*4;
                    pHD2.x = p2.x*4;
```

100 %

Locals: Name | Value | Type

Call Stack: Name

Error List | Locals | Watch 1 | Threads | Call Stack | Immediate Window

Ready

C:\Users\sramirez\Documents\Visual Studio 2012\Projects\CR

```
4000
```

- **To optimize the algorithm and avoid repetitions a video with variable traffic flow on highway was studied and the next cases were found**
  - Two different plates according to CARMEN software in consecutive frames
  - New plate (correct/wrong) recognized with a time lapse difference sufficient to be valid as a new car
  - If a plate was recognized as wrong by CARMEN confidence level, plate is evaluated in next frames for further checking and correction

- **Problems found when the video example was analysed**
  - Environment conditions (sunlight, shadows, camera movements)
  - Camera position

- **Possible solutions**
  - Add filters (sharpener and contrast/brightness) to mitigate the environmental effects
  - Analyse just one lane when camera is placed in red point, if the camera is placed in the green point, all lanes are included

- **Two cases were analysed**
  - Without filters
  - With sharpener filter which is used on an image before it is sent to CARMEN LPR

	Total evaluation time	Total cars count	Correct cars	Mean confidence	Wrong cars	Corrected cars
Analysis with sharpener filter	1760 s	534	507	70.9633	27	94
Analysis without sharpener filter	1760 s	532	515	74.7599	17	80

	Contours for loop		Processing time of an image with car		Recognize Method - LPR	
	Max time [ticks]	Min time [ticks]	Max time [ticks]	Min time [ticks]	Max time [ticks]	Min time [ticks]
Analysis with sharpener filter	1718	100	1709	100	1650	50
Analysis without sharpener filter	1218	106	1206	90	1138	34

Conclusion & Future work

- With the current resources, the program has been optimized as much as the resources allow it
- Alternative to current program
  - Execution of system demanding algorithms on GPU rather than CPU
  - CPU execution with vector and multi-thread (multi-core) support
  - Implementation of feedback information for shader (image processing)
- The project is currently being expanded to several nodes to enable whole traffic network OD estimation
- Tracking multiple vehicles in same and additional road lanes

OD Matrix Estimation of Traffic Flow on a Highway Network

Kristian Kovačić
Sergio D. Varela
Supervisor: Asst. Prof. Edouard Ivanjko

int12sva@student.lu.se
kkovacic00@gmail.com