International Journal of Advances in Image Processing Techniques– IJIPT
Volume 2: Issue 1 [ISSN : 2372-3998]
Publication Date : 30 April, 2015

Collaborative Monitoring of Traffic in Intersections using Video and Image Processing

[Konstantinos Kokkinos, Dimitrios Ventzas]

***Abstract*—Effective monitoring and traffic management at modern urban area road systems contribute significantly to resolving traffic problems and enhance the quality of life. Congestion caused by lack of capacity and strength of the traffic network due to traffic bursts essentially directs the flow control management in urban area intersections which also imposes restrictions on the development of a city. To address this problem, we develop algorithms for the identification of vehicles, their license plate recognition and the tracking of pedestrians when crossing intersections. The proposed framework is developed in a C# standalone application which collaborates with matlab libraries for the image and video processing and it is an open-ended environment for future extensions .**

***Keywords*—collaborative simulations, video/image processing vehicle identification, plate recognition, pedestrian detection.**

I. Introduction

Urban Traffic Control Systems play a major role in vehicle and driver safety as well as in pedestrian safety when crossing intersections. Most intelligent transportation systems research though concentrate on highway monitoring and tracking. This is due to the fact that, highways have attracted considerable attention as they carry the most part of the traffic, at the expense of other parts of the road network [1]. However, intersection traffic data and behavior analysis is considered equally important to reduce accidents, improve the accessibility of the intersection and finally solve the traffic jam caused by drivers and pedestrians. Consequently, several researchers have developed traffic monitoring systems which aim to analyze/characterize and manage traffic flow in urban intersections [2]. Most of these methodologies are based ground video detection techniques and radar technology which is further processed to produce and clarify analytics and to make the collected data more actionable. In general, these systems categorize the data traffic data in order to: *(a)* discover traditional traffic parameters and *(b)* develop traffic incident detection techniques. For both of the aforementioned cases there exists a plethora of commercial products (see [3, 4] and the references within).

The general methodology of collaborative traffic monitoring systems includes basically the development of a set of interdependent components that together form the basis for constructing an assessment tool both for the researchers and the stakeholders within the decision making context.

Konstantinos Kokkinos, Dimitrios Ventzas

Computer Engineering Dept. / Technological Education Institute of Larissa Greece

One method in designing a collaborative monitoring system is to couple existing models or components together so that, they become a single, larger software. This approach follows the *tight integration paradigm* which cannot overcome the following major disadvantage: when the components expand to monitor or model additional processes, they extend beyond the expertise of the original developers thus, they become *component creeping initiators* i.e., they gradually creep into neighboring research disciplines. An alternate approach is to couple the components using a *loose integration paradigm*. Loosely integrated m odels/components are written as software libraries which can then be integrated into a modeling framework [5]. By designing these components to follow a standard interface specification, it is possible to create an open-ended generic framework that facilitates component-to-component communication during processing/monitoring. Along these lines and based on the demands of the stakeholders, we present in this paper a collaborative road-intersection monitoring prototype which identifies five major milestones for its development: *(a)* system conceptualization, *(b)* component development, *(c)* component coupling, *(d)* monitoring and *(e)* assessment analysis as Figure 1 illustrates:

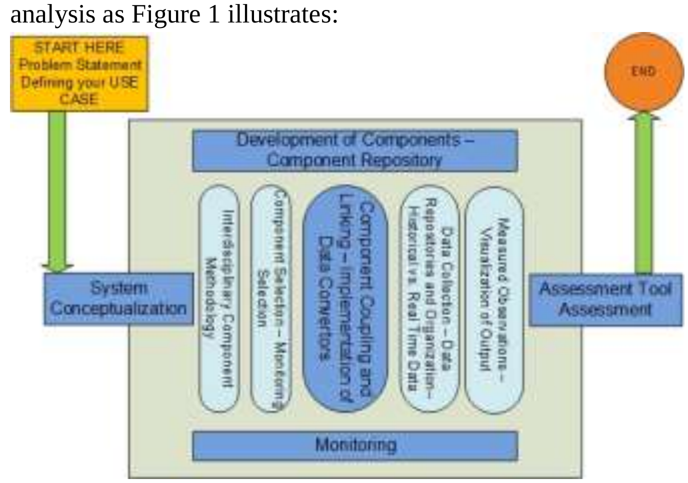


Figure 1. Implementation Milestones for a Collaborative Road-Intersection Monitoring System.

Following the above lines of inquiry, we use the video cameras as the primary source of input to our system since it is preferable to other alternatives due to the fact that it is an almost cost-free solution using the pre-installed infrastructure by the competent authorities. Furthermore, this solution requires limited human control or operation [6]. Most of the research that concentrates on vehicle identification at road intersections has as primary objective to monitor vehicle and pedestrian activities for event predictions. Recently, a thorough analysis on traffic accidents occurred at intersections was presented in [7]. This work attempts to identify and classify the involved vehicles in the use cases studied but the research in general was based on coefficient of variances for the statistical results and it was compared with the multiphase regression technique introduced in [8]. On the other hand research in Veeraraghavan et al. [1] targets to minimize occlusion effects in order to detect robust objects and track them in the intersection scene (including, vehicles, pedestrians etc.). In all cases, the technique is to extract the background and then to segment, classify and track the objects under motion. The tracking of the objects is usually done by performing a filter on every frame of the video (Kalman filters are the most popular). For the case of vehicle identification through license plate recognition, additional methodologies need to apply either in retroactive or in real time processing. The incident detection modules in the monitoring systems is attached as a plugin and it is based on detecting overlapped contour regions of individual vehicles or other objects in time.

In this paper we present our monitoring framework for traffic in intersections. The rest of the paper is structured as follows: In section II we illustrate our model for identification and tracking of vehicles, in section III we present our methodology for license plate recognition whereas in section IV the procedure for the pedestrian detection crossing urban roads. We additionally present a thorough discussion of how our models can be coupled and intercommunicate in an integrated monitoring system by temporally synchronizing our models using the OPENMI standard in section V. Conclusions and futures challenges are then followed.

II. Vehicle Identification Process

Vehicle monitoring systems use tracking models which calculate the location of a vehicle in a tracking zone. By repeating this process in a time series of video frames, we can discover the contour, calculate the speed and count passing vehicles in an intersection. Since this process is extremely computationally intensive, we use frame templates which are generated by the extraction of the background in the scene. For our analysis we use a set of steps as they have established in [9]:

- Video frame acquisition.
- Background extraction.
- Scene classification (according to the illumination variability due to day/night/weather conditions).
- Calculation of the mean background intensity.
- Application of morphological transforms.
- Calculation of vehicle contours, speed and other statistics.

We now show the details of each step which constitute the vehicle tracking and monitoring mechanism:

A. Video Frame Acquisition

Several videos have been gathered by the local authorities for research purposes and also manually at different locations of the city of Larissa, Greece. The manual data acquisition pre-assumes setting video cameras at fixed locations in intersections of various congestion characteristics (80-145 vehicles per hour) in each road zone under perspective. Furthermore videos have been distributed somewhat equally relatively to the day time and night time sampling and under different weather conditions. For every video we extract only one frame per second otherwise the computational burden cannot be overcome.

B. Background Extraction, Background Intensity and Classification

The most important functionality is the scene background discovery and its extraction. To achieve so, we compare sequential frames by computing the pixel-mean value after we transform them in grayscale mode so that the color variation would be normalized. We assign the first frame in the sequence to be our reference point and for each of the following frames we find the pixel-to-pixel intensity differences. For value differences more than 15 we assign the value of 255 (black) to the pixel. This results a new image of white background (pixel value =0) containing black objects (blobs).

Due to illumination levels differences during the day, we need to provide our system with a scene classification functionality as mentioned before. We note that, the intensity of the background on the carriageway part of the frame is changed by changing the brightness and therefore, it can be a criterion to be used for the classifier. Finding the mean intensity of the background is now explained:

Let for each frame F, f_B denotes the background and f_D the rest of the frame where f_{DM} symbolizes the mean intensity of f_D. After the background extraction obviously $f_D = F - f_B$. Let also N be the pixel count in the observation area, N_C the pixel count that corresponds to a vehicle, f_{CM} the average intensity of the frame-part that corresponds to the vehicle and f_R the average intensity of the frame part that corresponds to the carriageway then,

$$\begin{aligned}\overline{f_D} &= \frac{1}{N}[(N-N_C).\overline{f_R}+N_C\,\overline{f_C}] \\ &= \frac{1}{N}[N\,\overline{f_R}+N_C(\overline{f_C}-\overline{f_R})] \\ &= \overline{f_R}+\frac{N_C}{N}(\overline{f_C}-\overline{f_R})\end{aligned} \tag{1}$$

Relatively to the percentage of the frame which is occupied by each vehicle (that corresponds to the contour or the volume), this can be calculated as the ratio of the number of pixels that corresponds to a vehicle over the total number of frame pixels and for that reason $f_{DM} = (N_C / N) . f_{CM}$.

C. Morphological Transformations and Contours

After the background extraction and the identification of the vehicles the result is a frame with black and white pixels (binary image). In the back blobs we apply a dilation process to achieve accentuation of their contents. The dilation in our system is analog: it essentially emphasizes the black blobs because we want the objects since it is imperative to improve the final visual effect. The frame that emerges from the above analysis goes through further processing in order to place a red border around the object detected. To do this, all white pixels which are neighboring the black blobs are colored red. It is noted also that we applied a small squaring frame around the object of 8 pixel distance for better appearance of the recognized object. Figure 2 shows the functionality of the vehicle recognition in an intersection in our framework.

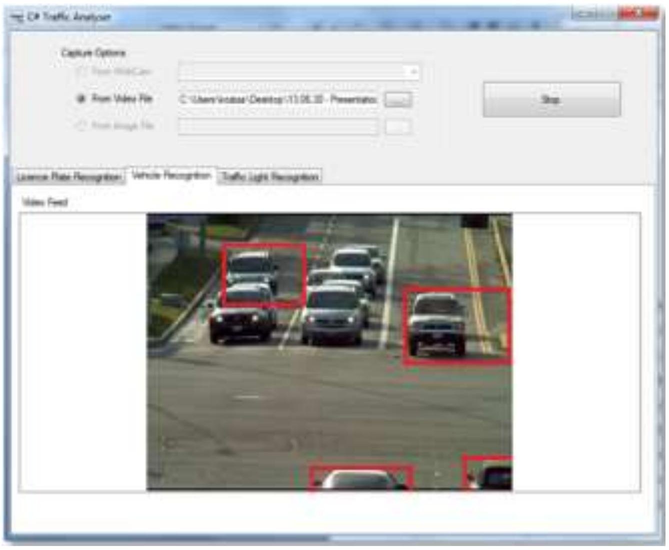


Figure 2. The Functionality of the Intersection Monitoring System for the Vehicle Identification.

III. License Plate Recognition

The proposed system accepts as input video frames of vehicles in most popular codecs (.jpeg, .bmp, etc.). The algorithm for the license plate recognition basically uses the following three steps: *(a)* the first step deals with the detection of a parallelogram among all candidate rectangles (frame segments) which contain the license plate. This is done using the morphological characteristics of the luminance of the edges of the rectangle thereof, *(b)* in the second part of the algorithm we focus on the selected rectangle from the first process and calculate the position and the orientation of the plate. The rectangle is then rotated, aligned and normalized and *(c)* the last part of the algorithm detect all the characters present in the plate and rejects all extraneous characters (character which are not part of the actual number).

The identification of the characters is done by cross referencing with predefined templates of characters and numbers incorporated in the system database. The matching process is relatively simple and not computationally intensive given the small set of characters. However, when the system is extended to several countries' character set, this deteriorates the overall performance. Further deterioration also occurs when frames under processing refer to bright environments (snow, bright sunny days, etc.). For not extreme cases, the process response is within high levels of success in recognizing the characters of the plates. In Figure 3, we illustrate the functionality of plates' recognition in our monitoring system. It is worth to refer also to four important features of the system in this process namely (a) white color intensity identification (b) color edge detection (c) frame segmentation and merging and (d) transformations. The first is important to identify since all license plates are a small rectangle with white background. The second is equally important in order to distinguish this rectangle among other candidate similar rectangles since all plates are contained within a dark colored boundary. The first two sub-processes are used for segmenting and combining the two sets of candidate frames such they satisfy the two first assumptions. Finally, the transformations in the end help for the alignment and optimize the final visual representation of the result.

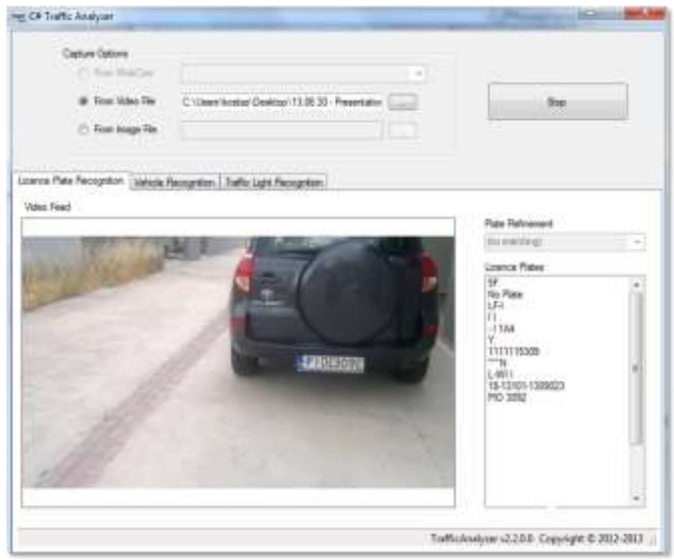


Figure 3. The Functionality of the Intersection Monitoring System for the License Plate Recognition.

A. *White Color Intensity Identification*

The detection of the intensity of the white color is used to find regions in the frame with bright white color. The idea behind the use of this sub-process is that all plates contain a white background which is highly reflective. This reduces the risk of false positive identifications of other areas such as taillights. In our case we establish an intensity threshold for eliminating the processing of frame-rectangles of lesser interest. However threshold values may vary for the case of variable weather conditions.

B. *Color Edge Detection*

The brightness profile is not sufficient to uniquely identify the area of interest for further processing as mentioned before. However a unique feature of all frames rectangles that contain license plates is that they also contain many edges from white to dark color and vice versa. An edge detection algorithm finds the sudden changes in the intensity due to the edges appear in the plate contour. The edge detection is done by convolution of the image with appropriate masks. We incorporated this feature using a filter from matlab that checks the sudden variations in intensity processing the pixels of the frame row by row.

C. Segmentation and Merging

Once we have calculated the two frames (one that contains the brightness clusters and the second that contains the edge clusters) the next step is to combine the results in a structural way. That is, to number all the candidate frame sections in binary mode and also number the candidate edges discovered. According to the area of the sections and the dimensions of the edges we make the final match for the best-chosen rectangle.

D. Transformations

For the case when the chosen rectangle appears on the frame with a slope, we must rotate it to correct the horizontal alignment. To determine the slope angle we use a Hugh transformation [10] operating on the frame in such a way that, each black pixel in the edge map generates a sine curve in Hough space as a direct mapping from the coordinates of the pixels.

IV. Pedestrian Detection

To track objects which are of smaller dimensions in road under traffic conditions is a much more challenging task that the vehicle tracking. Research in [11 – 12] and the references therein suggest three basic methodologies to obtain and track pedestrians in a traffic oriented scene: *feature tracking*, *pattern matching*, and *background subtraction*.

With the method of feature tracking we try to track points of high gradient difference such as corners, joints, or other points on sharp curvature [13]. Alternatively, with pattern matching we build a library of templates and we try to match new object with similar objects on the corpus however, the accuracy of the system is not guaranteed. Finally, with background subtraction (the most common), we monitor the scene for some time and we compute for every pixel the mean intensity value. When we select a frame for tracking, we subtract the background part to reveal the objects. Several problems can arise also with this method mostly due to camera vibrations especially when we record the passive scene (i.e. the scene without objects) and the variations in intensity due to weather conditions.

In our work, we use the method of background extraction followed by the well-known *Histogram* of *Oriented Gradients* (HOG) tracking mechanism [14 -15] to discover pedestrians on crossroads.

A. Background Extraction

We first convert the frame into grayscale mode using the standard luminance formula in [13]:

$$GRAYSCALE=0.299*RED + 0.587*GREEN +0.114*BLUE \qquad (2)$$

Since now the frame is comprised grayscale values, we consider as background the be the part of each frame where

$$F_B(x,y) = Mean(M(x,y)) \qquad (3)$$

F_B denoting the background frame part and $M(x,y)$ is the running average matrix containing previous intensity values present at location (x,y).

The extraction of the background relatively to the original frame is done in binary mode in such a way that pixels are set to 1 if their absolute difference between their intensity and the background intensity is greater than a certain threshold, otherwise are set to 0.

B. Tracking Pedestrians

The basic idea behind the use of the HOG methodology is that the objects under tracking in the frame can be easily determined using the distribution of their local intensity gradients or the edge directions even without knowing their exact location. For that reason, we count the number of occurrences of gradient orientation in cells of the image (i.e. portions of the image), usually called *bins*.

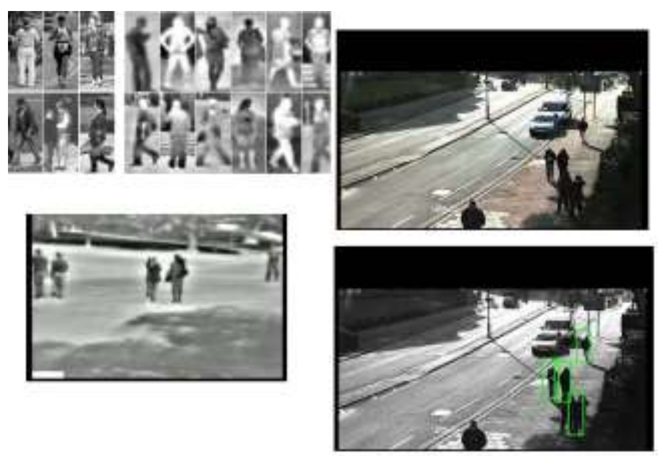


Figure 4. The Functionality of the Intersection Monitoring System for the License Plate Recognition.

This method is similar to that of edge orientation histograms but differs in that it is computed on a set of uniformly spaced bins and uses overlapping local contrast normalization for improved accuracy. We used the code in matlab from [16] to achieve the HOG transform of the grayscale image after the background extraction. Examples of pedestrian tracking are shown on Figure 4.

V. Model Collaboration Using OpenMI

Apart from modelling the three aforementioned functionalities in our system, we also attempt to provide the user with an integrated monitoring scheme for a better traffic management. This can be achieved by mutual data interchanging between the three components. By component intercommunication and collaboration we mean the initiation of a trigger mechanism used by one component to alarm another for the tracked position of an object at time *t*. This can be very important especially when the system can be used for accident identification and management due to involvement of both vehicles and pedestrians. We attempt to innovate towards this goal with the use of the OpenMI standard [17].

The OpenMI (Open Modeling Interface) Standard is a software component interface definition which was originally conceived to facilitate the simulation of interacting processes, particularly environmental processes. The main functionality of OpenMI is that, it enables independent processes to collaborate by exchanging data as they ran, time step by time step. However, it was soon realized that OpenMI could be a generic solution to any software component data interchanging and be applied to any case as long as the components are spatially or temporally synchronized.

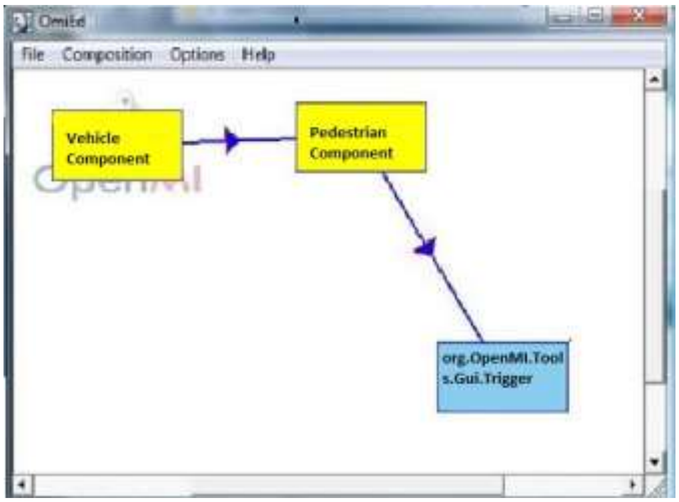


Figure 5. The Collaboration of the traffic software components through OpenMI (a snapshot of the OpenMI Configuration Editor).

To build an OpenMI-compliant system, every component has to become a *LinkableComponent* of the OpenMI. To achieve this we had to implement a code wrapper for each component so that, it communicates with the OpenMI *IEngine* core engine of the system. Furthermore, temporal synchronization of the components is achieved naturally by using the same video as input, and processing the same frame in every time step when the models run simultaneously. Unfortunately, the computational burden for this functionality cannot be overcome in a stand-alone application and for that reason we used timers. The use of timers apparently slows the overall system performance since all components have to finish at the same time. However, this functionality can be applied in a retroactive mode, where the post-processing can identify the exact time of where a pedestrian appears in the scene and the exact location of the vehicles at that time.

We also added to the functional requirements of the framework the ability of the user to select which tracking variables should be available through the OpenMI data exchange. Since all components are migrated to OpenMI it is practical to always show all intermediate output data variables. This is solved by saving the variable names in the same place where the manifest file (.omi file) of the OpenMI configuration editor is saved. It is still an open issue to decide the set of data variables stored in the system and the structure of the relational database schema applied in the framework. In Figure 5 we illustrate the configuration editor and present the connections between the components.

VI. Conclusions and Future Challenges

A monitoring framework to track vehicles, pedestrians and to recognize the license plates of vehicles at intersections has been implemented. The system is solely base in image and video processing. Relatively to the vehicle tracking we presented a method to extract the background from the scene and according to luminance levels to categorize it based in three different categories that meet the weather criteria we imposed. Contours of tracked vehicles are calculated using morphological transformations on the grayscale images after the background extraction. The license plate recognition technique implemented in our system initially discovers the best rectangle among all candidates for the license plate using the morphological characteristics of the luminance of the boundary edges. Afterwards the method fixes the orientation of this candidate rectangle by aligning its edges. Then the extraction of unimportant characters is implemented. The final discovery of characters is based on matching with stored character templates for different countries. For the pedestrian tracking component a simple background extraction was implemented which is followed by the application of the HOG tracking method to achieve better performance. Finally the three components are migrated into the OpenMI standard gaining the ability to interact with one another and interchange mutual variables of interest. It is still an open issue the choice of variables needed to be stored in the database as well its overall schema.

This project was the first attempt to create an integrated and collaborative framework that captures the most common functionalities when monitor and manage an intersection's traffic based on video and image processing. We plan to extend our system in the following ways:

- Add new software components such as traffic light detector, small vehicles detection (bicycles, motorcycles) and detection of parked vehicles.
- Integrate a detailed database that not only manages the huge data store of videos and images but also establishes the set of parameters needed to be focused on for better monitoring and management.
- Provide the necessary analysis and attempt to implement a traffic congestion and an accident detection component.

Acknowledgment

This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: **ARCHIMEDES III**. Investing in knowledge society through the European Social Fund.

References

[1] H. Veeraraghavan, O. Masoud, and N.P. Papanikolopoulos, "Computer vision algorithms for intersection monitoring", IEEE Transactions on Intelligent Transportation Systems, 4(2):78–89, June 2003.

[2] Autoscope. [Online]. Avalilable: http://www.autoscope.com. (Last accessed on 8/21/2014).

[3] N. Buch, S. A. Velastin and J. Orwell, "A Review of Computer Vision Techniques for the Analysis of Urban Traffic," IEEE Trans. Intell. Transp. Syst., no. 99, pp. 1-20, 2011.

[4] T. Rodriguez and N. Garcia, "An adaptive real-time traffic monitoring system," Machine Vision and Applications, January 2009.

[5] J. Löwy, "Programming .NET components", (2nd ed. ed.). O'Reilly Media Inc., 2005.

[6] Pedro F. Q. Loureiro and Rosaldo J. F. Rossetti, "Video Processing Techniques for Traffic Information Acquisition Using Uncontrolled Video Streams", Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems, St. Louis, MO, USA, October 3-7, 2009.

[7] O. Akoz, and M.E. Karsligil, "Video-based traffic accident analysis at intersections using partial vehicle trajectories", IEEE International Conference on Image Processing, September 2010.

[8] O. Akoz, and M.E. Karsligil, "Severity detection of traffic accidents at intersections based on vehicle motion analysis and multiphase linear regression", IEEE International Conference on Intelligent Transportation Systems, September 2010.

[9] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using optical sensors: a review", IEEE International Conference on Intelligent Transportation Systems, pp. 585–590, 2004.

[10] Hough Transformation, [Online source code]. Available at : http://www.mathworks.com/help/images/ref/hough.html, last accessed on 8/21/2014.

[11] J. Zheng, Y. Wang, N.L. Nihan, & M. Hallenbeck, "Detecting cycle failures at signalized intersections using video image processing", Computer-Aided Civil and Infrastructure Engineering, 21(6), 425–35, 2006.

[12] R.T. Collins, "Mean-shift Blob Tracking through Scale Space", in Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03), Vol. **2**, pp. II 234–40, 2003.

[13] Y. Malinovskiy, J. Zheng and Y. Wang, "Model-Free Video Detection and Tracking of Pedestrians and Bicyclists", Computer-Aided Civil and Infrastructure Engineering, Vol. 24, Issue 3, pp. 157–168, 2009.

[14] Histogram of Oriented Gradients method, [Online], http://en.wikipedia.org/wiki/Histogram_of_oriented_gradients , last accessed on 8/21/2014.

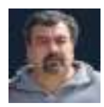
[15] L. Zhang, B. Wu, and R. Nevatia, "Pedestrian Detection inInfrared Images Based on Local Shape Features," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1-8, 2007.

[16] Histogram of Oriented Gradients (HOG) matlab code, [found Online] http://www.mathworks.com/matlabcentral/fileexchange/46408-histogram-of-oriented-gradients--hog--code-using-matlab , last accessed on 8/21/2014.

[17] OpenMI-Standard, [Online], http://www.openmi.org, last accessed on 8/21/2014.

About Author (s):

Dr. Konstantinos Kokkinos was awarded a B.Sc. in Physics, from the Aristotle's University of Thessaloniki, Greece (1989), a M.Sc. in Computers Science and a PhD in Computer Science from Computer Science Dept. Western Michigan University, U.S.A. (1995 and 2002 respectively). Dr. Kokkinos is a member of IEEE Society, an OpenMI-Association member, a full time instructor in the Computer Informatics Dept. of TEI-Larissa, Greece, a lecturer at Civil Engineering Dept., University of Thessaly, Greece and a reviewer of several IEEE journals. His major research interests include Integrated Modelling and Simulation, Process Control, Network Congestion, Traffic Load Balancing and Web Services. Dr. Kokkinos has participated in 7 NSF and European (FP) funded projects. He has published over 25 journal and conference papers.



Dr. Dimitrios Ventzas (chartered Electrical & Mechanical Eng, SMISA) is professor at Informatics & Telecommunications Dept. at TEI of Larissa, Greece with a PhD in Control Eng. and expertise in DSP, Image and Video Processing, Sensors & Measurements and Biomedical Eng. He has been a Research assistant at University of Bradford, UK in multiphase flow, Systems Engineer at HARSA SA, Greek Railways and AMBER SA. He has taught at TEI Athens, Lamia and Larissa, Greece and at the University of Thessaly graduate and postgraduate courses at the Mechanical Eng. and Agricultural Eng. departments. He is the author of various lecture notes and 20 / 50 papers at international Journals / Conferences with over 30 citations.