University "Stefan cel Mare" Suceava
Faculty of Electrical Engineering
Department of Computers and Automations

SafeWay - Intelligent Street Monitoring System

Team members	Radu VATAVU	raduvro@yahoo.com
	Octavian MANU	octavianm@stud.usv.ro
	Ionel ZAGAN	ionel_z@stud.usv.ro
	Andrei SLAVOIU	ansla@apollo.eed.usv.ro
Team mentor	Eugen COCA, Ph.D.	ecoca@ieee.org
Project website	http://ecoca.eed.usv.ro/safeway	

Abstract

Road traffic safety is a problem that has been given a lot of attention in the last decades and many attempts, social oriented or automatic systems as well, have been proposed in order to reduce the number of injuries by increasing the road security level. We address the same problem in this project, the possibility of reducing traffic accidents by the means of a machine vision-based assistance system, designed to automatically detect dangerous traffic situations involving pedestrians and auto vehicles as well. By intelligently monitoring the road activity using pedestrian and vehicle detection algorithms, the system allows the possibility of warning both drivers and pedestrians of traffic dangerous situations.

Several attempts of automated systems designed for increasing the road safety level have been previously proposed. The very first statement that could derive from analyzing these systems is that they all attempt to solve the problem by focusing on particular aspects, e.g. devices, intelligent or not, to be installed in vehicles and providing mostly audio warning functions, traffic congestion analysis or automated traffic lights switchers. The disadvantages that come with these approaches consist in the fact that there is no warning whatsoever for pedestrians the focus being mainly oriented on the driver, or a bad management of the traffic lights that are not adapting dynamically to the real situations.

From the above, it becomes very clear the necessity of an external and autonomous system with the main task to intelligently evaluate the current road situation and launch the appropriate actions (warnings for both drivers and pedestrians, traffic light switches, signals to a control command center such as the local police department). Our version of such a system is a machine vision based approach which takes into consideration both car and pedestrian information in a given scenario and takes actions in order to protect the two traffic participants in an equally manner.

We believe that a solution designed to increase road traffic safety should act as an external protector, without requesting any input from the traffic participants, without relying on any particular devices and without making any assumption about the driver's or pedestrian's attention to the traffic. The most important issue that our project addresses are that both drivers and pedestrians should have the right to be equally protected whilst in traffic, being behind the wheel or crossing the street should not make a difference. The paper deals with all these aspects in detail and describes such an external automated traffic protector, our vision of how a safe traffic system should be. Several scenarios are also discussed; system interaction to mobile devices such as laptops or handheld computers is also taken into consideration; designs for special low cost devices that address to the traffic participants are presented; the interaction of the system with a decision center such as the police department is also taken into account.

System overview

SafeWay is a machine vision based system designed to reduce possible dangerous situations including both drivers and pedestrians, treating both the traffic participants in an equally manner. The main functionality of the system is to act as an external supervisor, monitoring all the traffic activity in a given area by using pedestrian and vehicle detection algorithms and take proper warning actions when possible accident situations interfere.

The minimum equipment required includes one processing unit (any IBM-PC compatible computer), two surveillance video cameras and one device for other equipment interfacing to the system (Figure 1). The real power and innovation stays in the pedestrians and vehicles detection algorithms using real-time video processing and support vector machine classification.

A serial/Ethernet communication device for controlling output hardware devices has been designed in order to demonstrate the system's ability to connect to other existing traffic equipments (such as LED displays, traffic lights, relays, etc). Interaction and communication to other software equipment is also demonstrated by describing the usefulness of a control center (such as one in the Police Department) where the real field situation can be broadcasted. Another contribution to further improving the system consists in the SafeWay pagers that are permanently connected with the system whenever the person/vehicle that carries them enters the protected area (optional devices, designed just to demonstrate how the system can be further extended to mobile, handheld devices or small pagers).

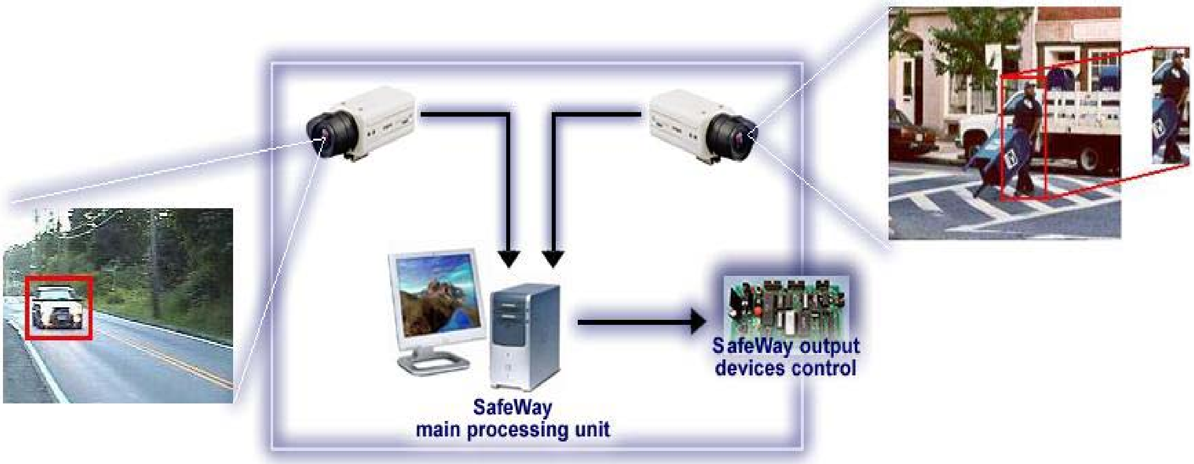


Figure 1. Basic SafeWay configuration: one IBM-PC compatible computer, two video surveillance cameras and one device for other equipment interfacing to the system

A Microsoft Windows operating system together with a Microsoft DirectX distribution is required. Video streaming has been implemented using Microsoft Windows Media SDK hence the Windows Media Codecs are required. Video capture (DirectShow) and video processing (pedestrian and vehicle detection algorithms) require at least 128MB RAM and a 1.2GHz processor. A detailed description of performance results is given in the next section, together with more precise system requirements.

As a simple example, Figure 2 describes one of the possible usages for the SafeWay system: two video cameras monitoring a dangerous curve traffic zone where the visibility is low and both drivers and pedestrians could use some information about the incoming traffic. This can be very easily accomplished by having appropriate information furnished by the SafeWay processing unit to LED display signs as per the image.

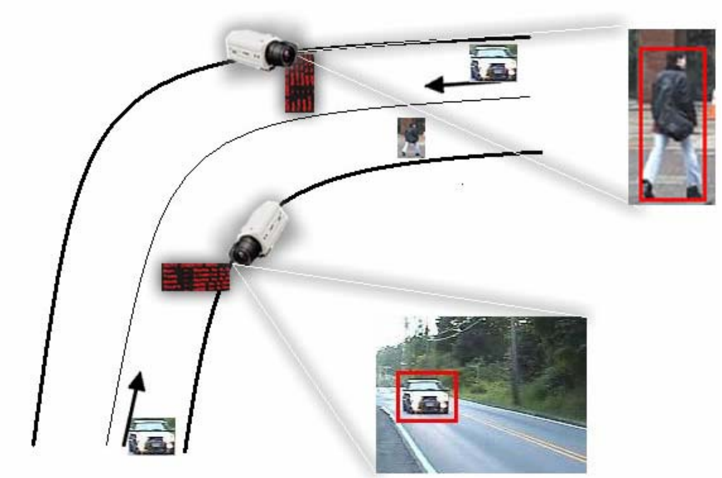


Figure 2. A dangerous road situation

We believe that the SafeWay system introduces a series of innovative approaches and new directions in road traffic security:

- The first and the most important is the perspective that gets introduced: **a system that acts as an external traffic protector, without requesting any input from the traffic participants, without relying on any particular devices and without making any assumption about the driver's or pedestrian's attention to the traffic**.

- Second, but with the same importance, is the fact that the system does not address to any particular traffic participants. **All the participants, both drivers and pedestrians, have the right to be equally protected whilst in traffic**, being behind the wheel or crossing the street should not make a difference.

- Independence of traffic zone. The system does not address to particular road situations and hence can be successfully installed in any dangerous traffic area. The architecture (one processing unit and two surveillance video cameras only) and the ability to connect to other hardware or software equipments give the system an enormous adaptability and flexibility.

- Intelligent traffic monitoring approaches by using **pedestrians and vehicle detection algorithms**. The techniques are based on training support vector machines having as samples Haar wavelet features from pedestrian and vehicle pre-processed inputs.

Implementation and engineering consideration

SafeWay is an automated system designed to intelligently monitor the current road traffic, watching for both cars and pedestrians and taking the appropriate actions when dangerous situation conditions are met. Considering the importance of this task, a series of performance requirements and organization principles have been considered with regards to system's modularity, flexibility and simple interfacing to other hardware and software equipment. An overview of the system's organization and working principles are presented in Figure 3.

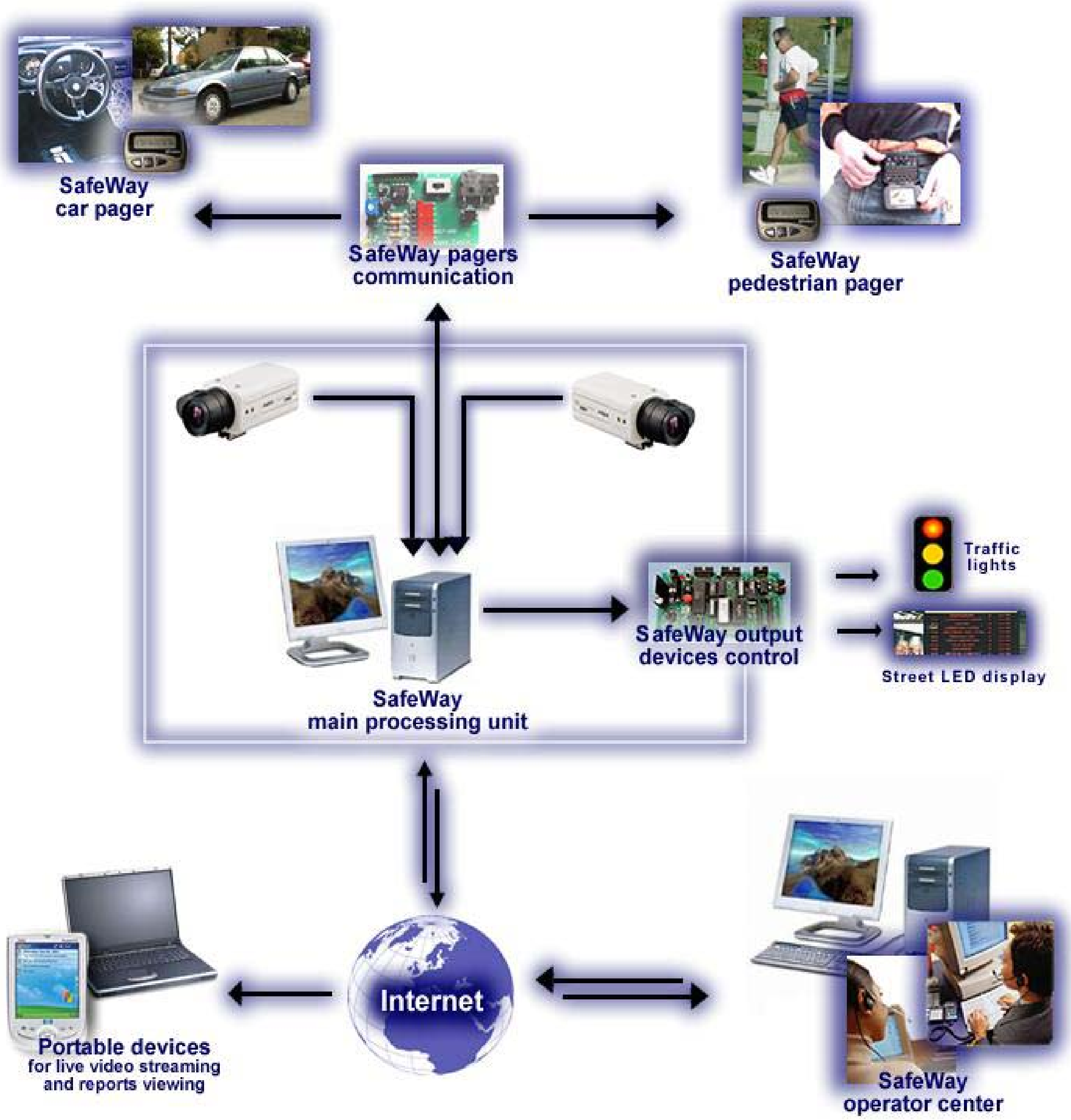


Figure 3. SafeWay system organization and working principles

The SafeWay system has been designed in a structural easy to extend manner, containing 4 major components:

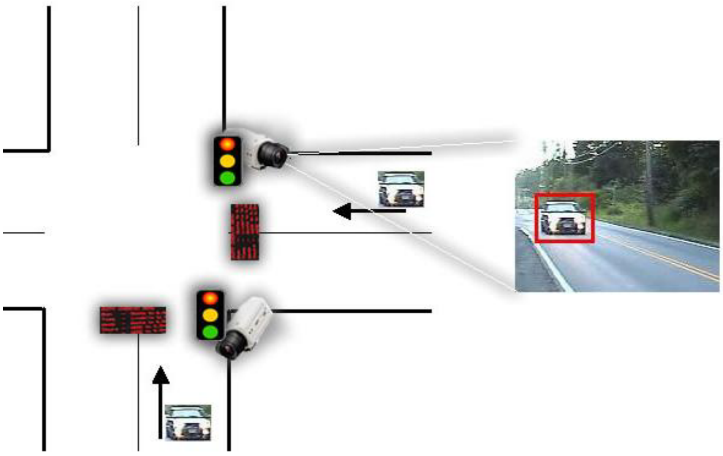
- **SafeWay Main Processing Unit** - consisting of the IBM-PC compatible computer and two video surveillance cameras. All the processing gets done at this level, namely: video capture and analysis, pedestrian detection, vehicle detection, scenario analysis and actions planning

- **SafeWay Output Device Control** - a serial/Ethernet communication device for controlling output hardware devices that has been designed in order to demonstrate the system's ability to connect to other existing traffic equipments (such as traffic lights, LED displays, relays, etc).

- **SafeWay Software Interaction Component** - a module for interacting with external software, including a live video streaming component and a report generator. The purpose is to demonstrate the interaction with an operator center (such as one inside the Traffic Police Department) and the interaction with mobile or handheld devices.

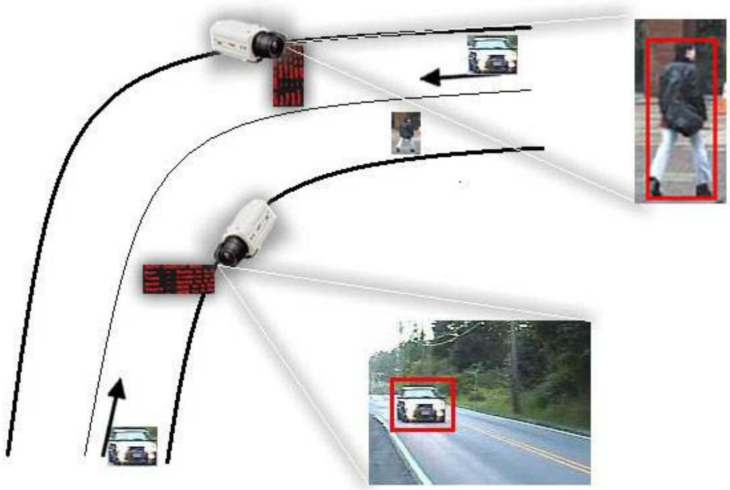
- **SafeWay Pager Communication Module** – that demonstrates how SafeWay can be further extended to mobile, handheld devices or small pager-like devices, permanently connected with the system whenever the person/vehicle that carries them enters the protected area

SafeWay's general functionality and organization make possible adapting the system in any real world situations, a few examples might include: crossroads with or without traffic lights control, highways, roadways or streets with low visibility points, suburban districts roads, etc. A few possible implementations are presented below (see Figure 4), with focus on both drivers and pedestrians.



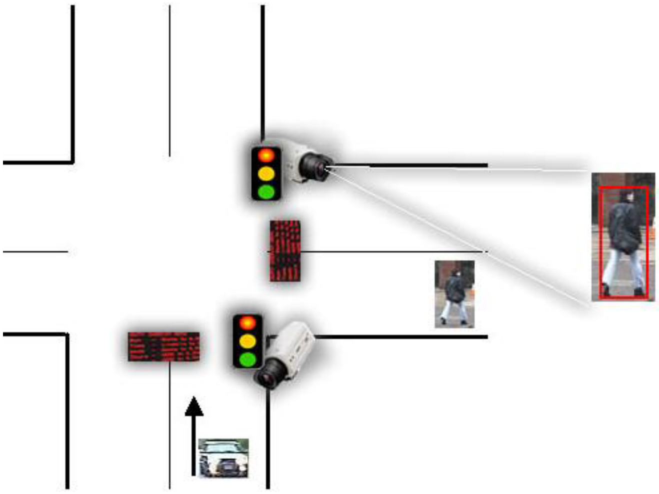
a) Crossroad scenarios

The first example attempts to solve the problem of vehicle accidents in crossroads areas. The only component used is vehicle detection combined with the road warning display. If a car is detected on one street, the system follows the pre-built scenario and automatically takes the proper action for the other.



b) Low road visibility points

Moving a bit further, we can address the problem of low road visibility points. The SafeWay system is placed on a low visibility turn. One video camera is oriented into each entry of the curve. When a car enters the area covered by either one of the cameras, the car driving from the opposite direction is warned by the road display. Also, if a pedestrian is detected on the road, the appropriate warning signal is activated.



In a similar manner to the first scenario, the system proves utility in suburban districts, in this case the focus being on pedestrians. Both the vehicles and pedestrians detection algorithms are being used and the warning component addresses to all traffic participants.

c) Pedestrian safety in suburban districts

Figure 4. Examples of SafeWay working systems

Main Processing Unit

All the processing (pedestrian and vehicle detection, road situation analysis and the warning action planner) is been carried out in the main processing unit with the accent being on the software component.

In a similar manner with the whole system organization and architecture, the software component has been designed and implemented in a sequential manner including 5 layers:

- **Data acquisition layer** (relates to video capture only);

- **Object detection layer**, the most important layer, includes car and pedestrian detectors;
- **Decision layer** aggregates the object detection results and, given a known scenario, decides the appropriate actions;
- **Action layer** takes care of handling the adequate actions, respectively: warns drivers and/or pedestrians of possible dangerous situations, signals to the SafeWay centers together with a live broadcast image and scene analysis, controls the traffic lights and warnings road displays;
- **A video stream-encoding** layer that, accordingly to the analysis carried out in the decision layer, furnishes live video broadcast of the road situation.

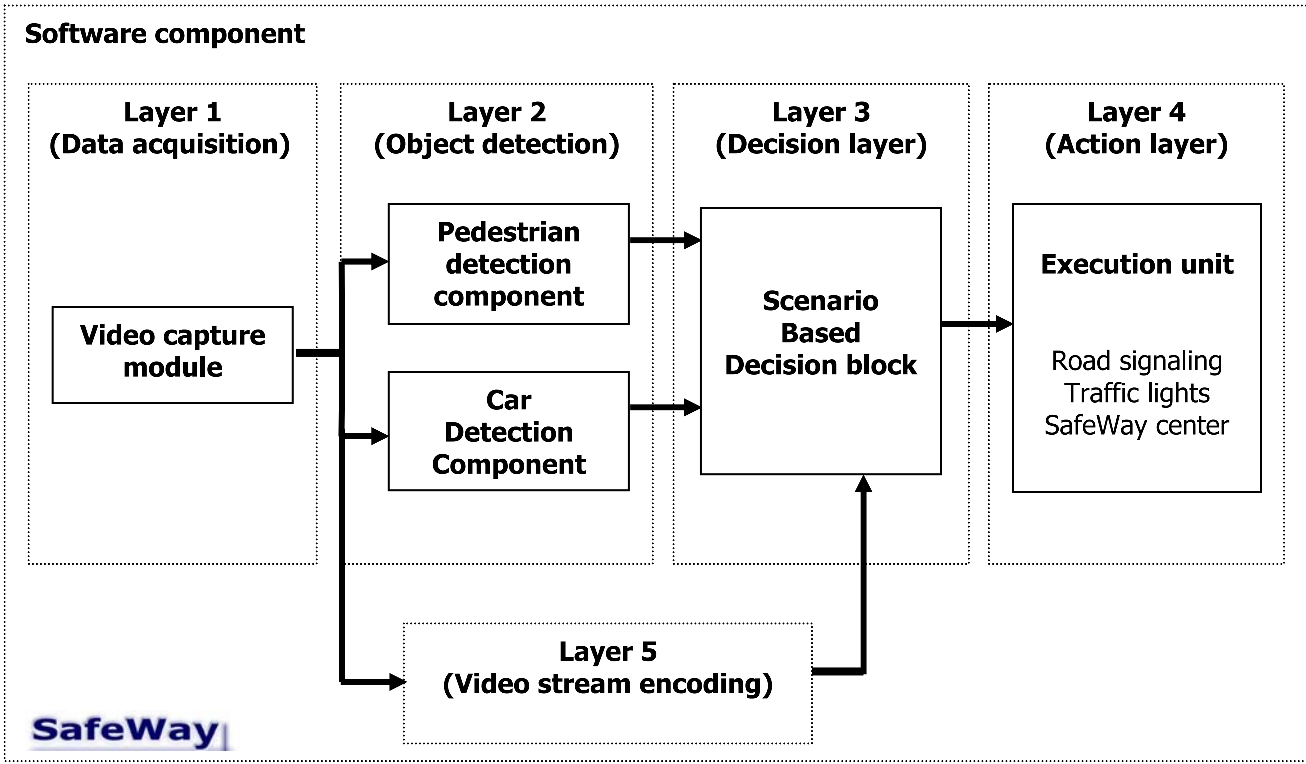


Figure 5. Software component organization

Being by far the most complex of all, the object detection layer can be considered the main component of the project, as it is its classifying results that determine the entire system's behavior and actions. A more detailed description of this processing layer is given in Figure 6 and again, in a sequential layered manner, a series of pre-processing and processing steps are followed:

- **Basic image processing** techniques [6] are been carried out on the raw image data captured from the two video sources. These basic techniques are meant to enhance the image quality before being fed to the object detection components: noise cleaning [2], histogram modification and color image enhancement [5].

- The enhanced image is fed to the **two object detectors**, the pedestrian and vehicle detection modules, both previously trained on a sample set. The detectors are actually support vector machines, trainable machine learning modules, hence there is a clear separation in the layer 2 development process: the **training** phase and the **testing** phase (Figure 7 and Figure 8).

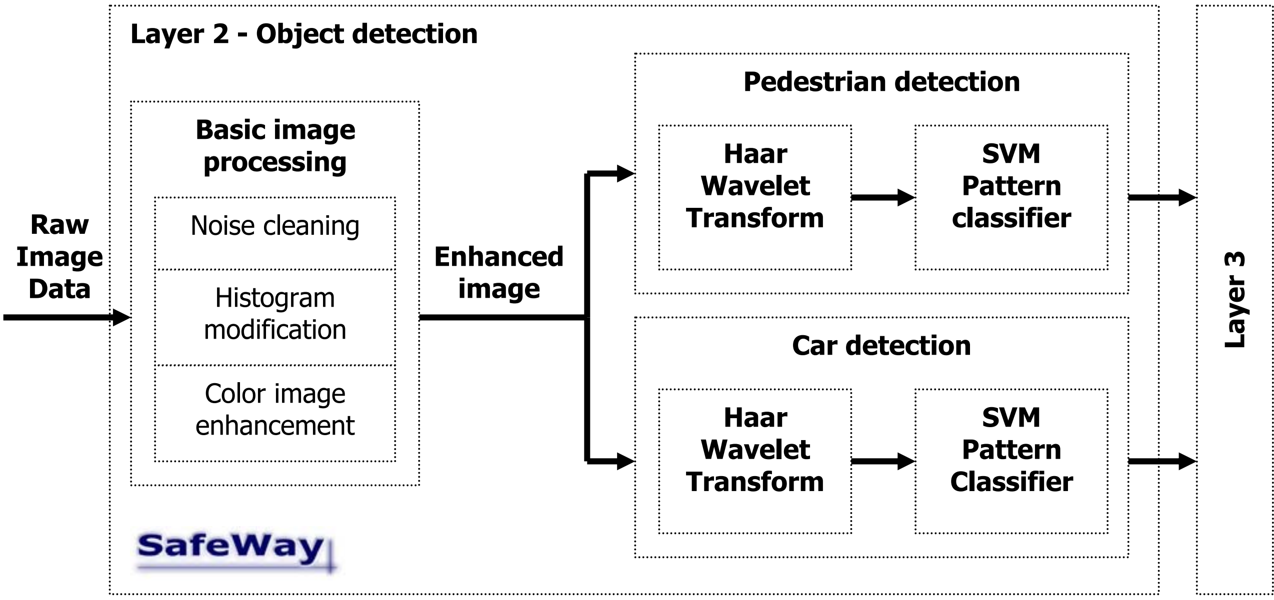


Figure 6. Object detection layer details

Either case, building a pedestrian or a car classifier, the process is basically the same: in the training phase, the system takes as input:

1) A set of images of the object class (pedestrian/car) scaled to the same dimensions and properly aligned so that the focus object is approximately in the center of the image

2) A set of non-object images, i.e. patterns that are not in the desired object class. Based on these sample images that constitute the training set (in the supervised learning approach), an implicit model of the data is constructed and hence, a classifier is built.

In the testing phase, the focus changes on applying the new built classifier to out of training set image samples (the testing set). The principle of detecting the target object consists in sliding a fixed size window over the entire image and decide for each location whether the corresponding pattern represents an object or not.

At each position, the same set of features as the ones in the training phase are being extracted and the resulted pattern classified by the previously data built model. In order to achieve multi scale detection, the image is iteratively resized and processed again with the same fixed size window (the algorithm just presented can be quite expensive, however these shortfalls can be eliminated by using wavelet transformation on the initial image and hence, the extension to real-time systems is straightforward).

The key issues are obviously, the model chosen for object representation and the classifier type. Although the object types can be perceived as simplistic (pedestrian or cars), there is a great variability that accompanies the data, variability that gets translated into color, texture, pose and lack of a consistent background (a simple look at the image in Figure 8 will reveal all these aspects). Hence, the object representation must handle all these shortfalls of the search space and that will achieve a high interclass variability with a low intra-class one.

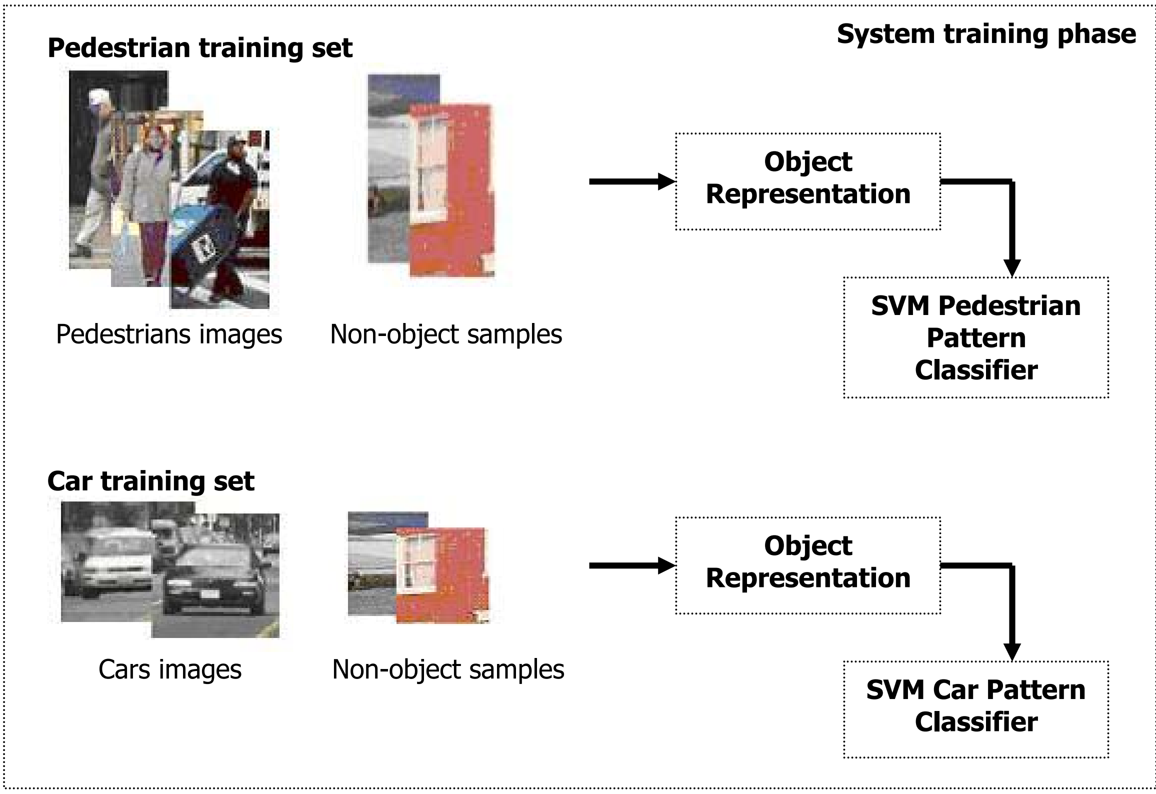


Figure 7. Training phase principles

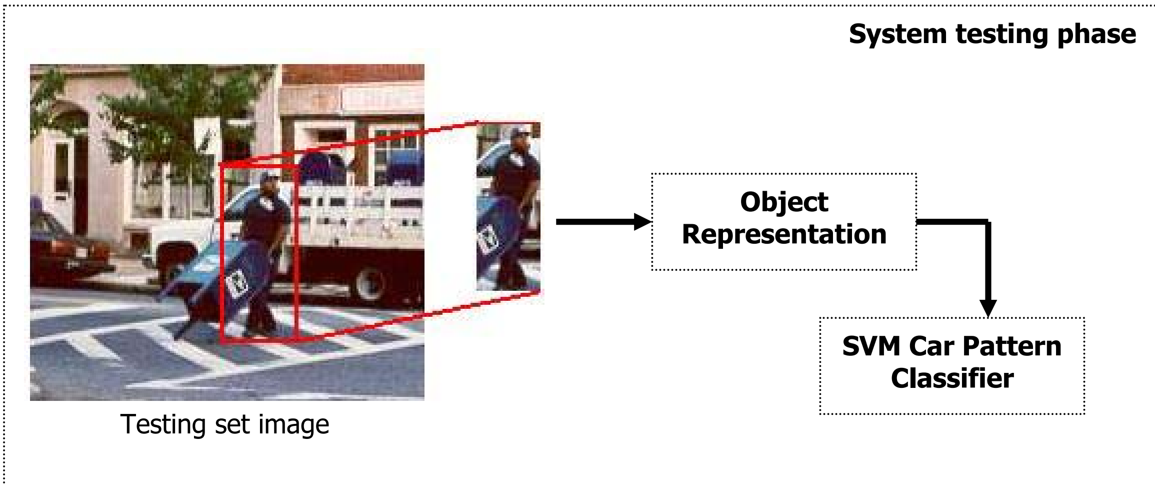


Figure 8. Testing phase principles

Traditional approaches, as pixel based representations or region based ones are very likely to fail if one would consider the permanent changing in color and texture. Edge detection approaches [8] don't prove to work any better cause of the great variability already mentioned that could be found in people clothes, car colors and the always-changing background (see Figure 9).

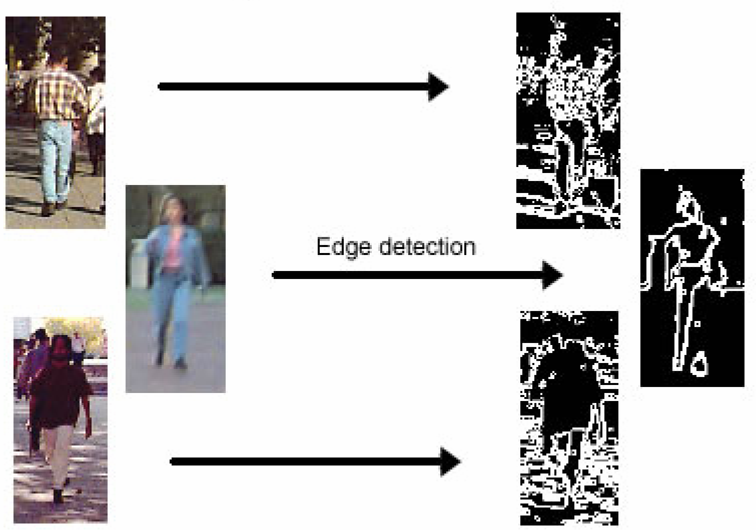


Figure 9. Edge detection does not prove to be suited for detecting pedestrians because of the great variability the images contain. A different technique for representing features is required

A possible relying approach to the problem is to use the set of Haar wavelets (Figure 10), which provide very good responses to local intensity differences in the image and to orientation variations as well. The wavelet transform will compute the image response to the wavelet filters (oriented wavelets – horizontal, vertical and diagonal) leading to fine coarse scale features embedded in the representation.

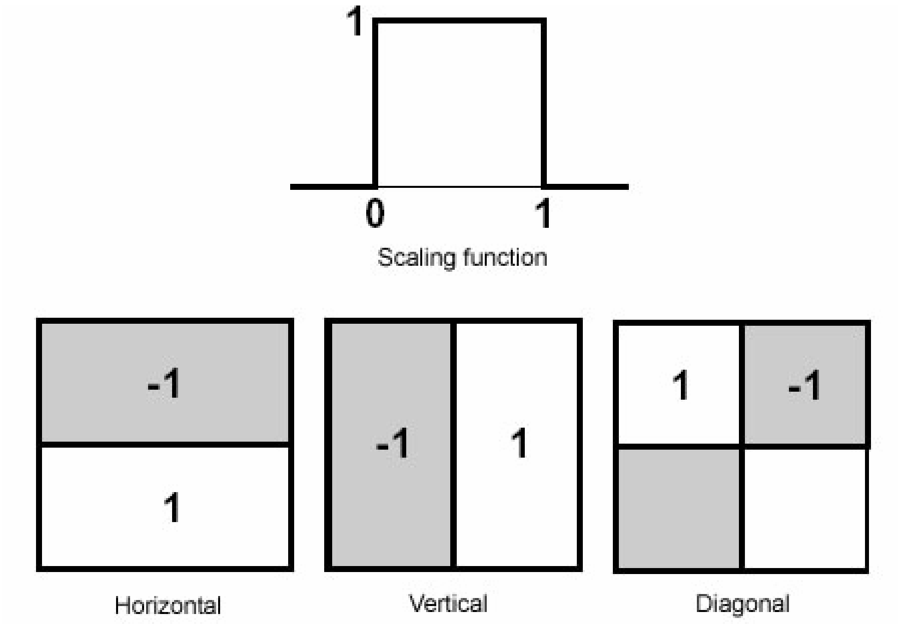


Figure 10. Haar wavelet framework: the Haar scaling function and wavelet and 3 types of 2-dimensional Haar wavelets: horizontal, vertical and diagonal

Average values for the pattern's features obtained with the Haar wavelets are presented in Figure 11 on a pedestrian image and a vehicle one as well. The wavelet representation of the input image has been proven to be very adequate to the purpose of the task, introducing the least level of variability and being far better than the 2 previously mentioned straightforward approaches.

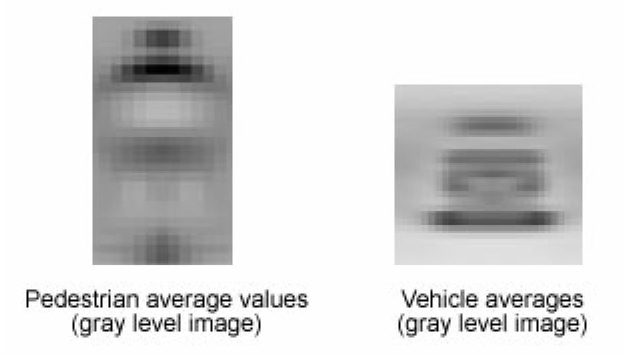


Figure 11. Average values for the object pattern features obtained using the Haar wavelets on pedestrians and vehicles images

The approach taken to pattern classification is based on Support Vector Machines [11] that proved to best achieve two important aspects needed in the system: precisely identify the factors that need to be taken into account in order to learn successfully (coming from statistical learning theory) and handling the complexity of real-world applications that often require advanced models and algorithms (much harder to analyze theoretically).

Support Vector Machines is a technique to train classifiers that is well founded in statistical learning theory, the greatest attraction of using SVMs is that they are capable of learning in high dimensional spaces with very few training samples. SVM accomplish this by minimizing a bound on the empirical error and the complexity of the classifier at the same time. The original data set may not be linearly separable. The support vector machine use a nonlinear kernel to map the data points into a high dimensional feature space in which the classes have a much greater chance of being linearly separable.

The separating boundary is in general of the form [8]:

$$f(x) = \theta\left(\sum_{i=1}^{n} \alpha_i y_i K(x, x_i) + b\right)$$

where n is the number of training samples *(x, y)* y being the label -1/+1 of the training sample x, α_i are nonnegative parameters learned from the data and $K(\cdot,\cdot)$ is a kernel that defines a dot product between projections of the two arguments in some feature space. The kernel $K(x, y) = \exp\left(-\|x - y\|^2\right)$ defines a Gaussian radial basis function. θ is a threshold function [7].

The main feature of SVM is that it finds, among all possible separating surfaces of the form above the one that minimizes the distance between the two classes of points (as measured in the feature space defined by K).

In our case, x will be represented of the feature values obtained by applying the Haar wavelets on the input image. The classification will be decided as follows: if the output (given by the threshold function) is +1, we are dealing with an object of interest (pedestrian or vehicle in our two cases and two classifiers) or -1 indicating a non-object image.

The classification algorithm, once we have two trained classifiers is given below. The main principle is to shift a fixed size window over the image and classify each pattern at row and column (r, c) of window size (m, n) with the two object classifiers.

```
Let I be the input image, captured from one of the video cameras
Let rfactor be the rescaling factor for the image, rfactor=1.25
Let ri, rf and rc be the initial, final and current scales, ri=0.4, rf=1.4
Let Irc be the input image rescaled by rc
Let H(I) be the Haar transform of image I
Let q be a m x n pattern in wavelet space and f the associate feature
vector

Set rc = ri
While rc < rf

      Irc ← Resize I by rc

      H(Irc) ← Compute the Haar wavelet transform for the resized image
      Irc

      For every row, column pair (r, c)
            q ← pattern starting at (r, c) with dimensions (m, n)
            compute the average response of each type of wavelet
            (horizontal, vertical, diagonal) for q
            f ← compute the feature vector by normalizing each wavelet in
            q by its class average
            class ← classify f by using the pedestrian classifier
            if (class == 1) then
                  pattern is a pedestrian at position (r, c)
            else
                  class ← classify f by using the vehicle classifier
                  if (class == 1) then
                        pattern is a vehicle at position (r, c)
                  end if
            end if
      end for

      Update scaling factor rc ← rc*rfactor
end
```

Experimental results

Lab experimental results

To accurately measure our detection system's performance we use a Receiver Operating Characteristic (ROC) [12] curve that quantifies the tradeoffs between detection accuracy and the rate of false positives. The testing procedure measures positive and negative pattern performance separately over different sets of data.

The positive test data consists of aligned but not scaled examples of the object class with a sufficiently large boundary around them such that the detection system can run at several different locations and scales. While these images can be of various sizes, the proportion at which the object occurs in the image is constant from test example to test example. To generate the false positive rate, we have a set of 27 images of different natural and man made scenes that do not contain any examples of the objects we are detecting. The false positive rate is simply the number of detections in this set of data divided by the total number of patterns that are examined.

Table 1 gives the number of positive examples and negative patterns and Figure 12 shows the ROC detection curves for pedestrians and vehicles detection.

	Pedestrian	**Vehicles**
Positive examples	312	212
Negative examples	50,000	50,000

Table 1. Test sizes for pedestrian and vehicle detection classifiers

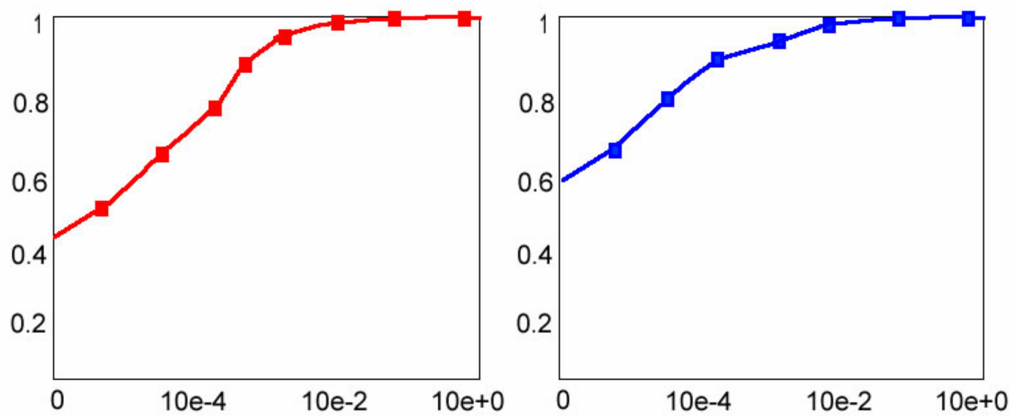


Figure 12. ROC curves for pedestrians (red) and vehicle (blue) detection classifiers

Taking the system outside

The second testing experiment was simulating a real situation, hence porting the system outside and test field performances. We have performed tests at 2 high traffic crossroads and the results demonstrated a bit lower detection rate than in the lab testing, caused by a series of variables: camera settings, camera placement, camera zoom factor, etc. However, the average detection rate has remained in a 10% limit of the lab tests results.

Figure 13. Taking the SafeWay system out side

SafeWay video streaming module

Although the SafeWay system was designed to function without human supervision, it might be desirable to monitor the road activity from the Head Quarters.

In order to achieve this goal, we used the Microsoft Windows Media Encoder 9™ technology [18] to make the image available to the supervisor. The module is implemented in the form of a DLL that is integrated in the main application and can also be used by a standalone server.

Figure 14. SafeWay live streaming demo server

The Windows Media Encoder 9 Series SDK is designed for anyone who wants to develop a Windows Media Encoder application by using a powerful Automation-based application-programming interface (API).

We have chosen to use Windows Media Encoder because of the features it offers:

- **Live content broadcasting**. Live pictures of road conditions can be streamed at multiple trouble spots, alerting drivers to traffic congestion and advising them of alternate routes.

- **User interface customization**. An interface that uses the functionality of the Automation API can be built to capture, encode, and broadcast media streams. Alternatively, the predefined user interfaces within the Automation API can be used for the same purpose.

Due to the various profiles built into WME9 the video can be streamed over any type of connection. This way the supervisor could watch a high quality stream over the local network or a low bandwidth one if he/she is in a low speed remote location.

SafeWay output device control

In order to prove the system's flexibility we designed a serial/Ethernet communication device (Figure 15) to control the hardware output equipments (such as LED displays, traffic lights, relays or devices that already exist and used by the traffic system).

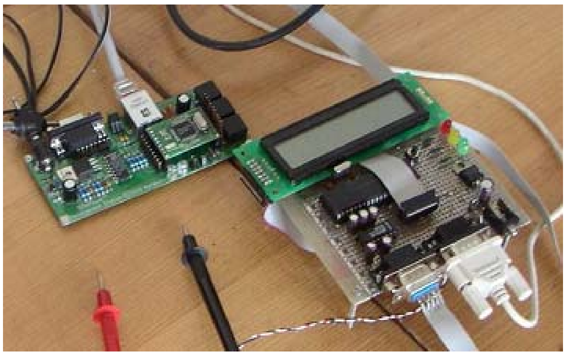


Figure 15. SafeWay output control device with serial/Ethernet communication

We have chosen the PIC 16F877 microcontroller [15] as the core of the SafeWay Hardware Module (Figure 16) because it provides the necessary functionality: high-speed serial port, large number of Input-Output pins, its program and data memory can be extended to meet our project's needs. The microcontroller receives information from the SafeWay Main Processing Unit via a RS-232 serial connection or from Ethernet through Site Player.

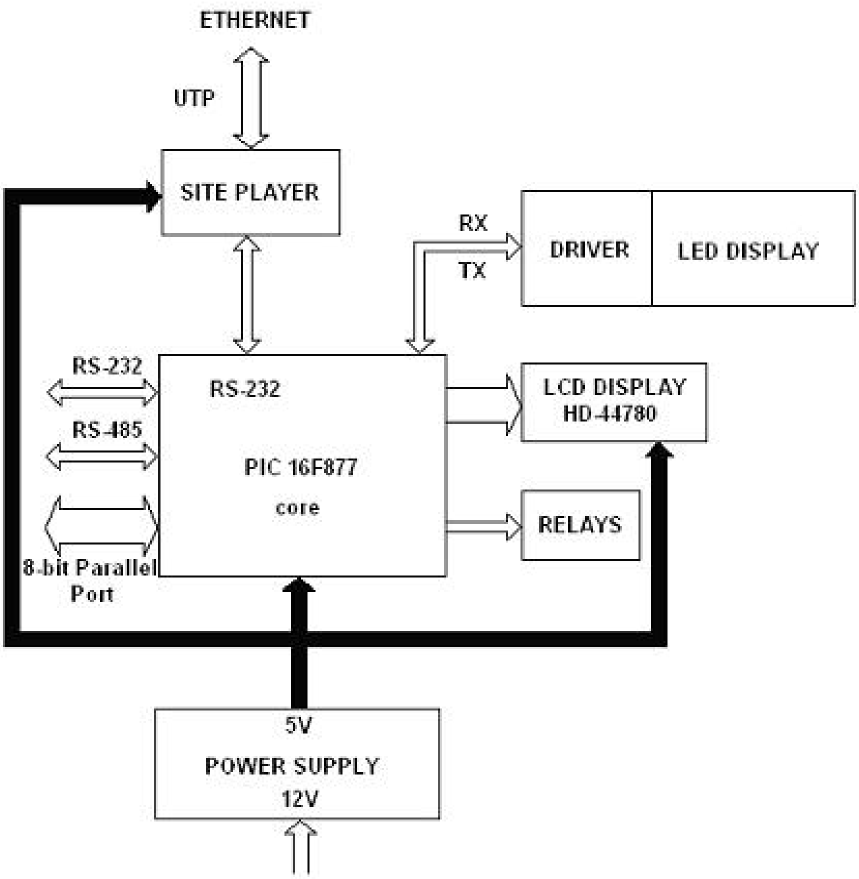


Figure 16. SafeWay Output Device Control - block diagram

We have also used an inexpensive alphanumeric 20X2 Hitachi compatible LCD display for SafeWay Output Device Control indication and testing. We provide an interface for a connection to a more expensive serial dot matrix Hyper-LED Display.

Interfacing the SafeWay Output Device Control to a variety of simple road signaling devices such as traffic lights or warning light is done via a simple scheme using transistors and relays. More complex road signaling, with build-in logic devices, could be connected through serial interface such as RS-232 or industrial RS-485, or by an 8-bit parallel interface.

Since PIC 16F877 doesn't have a dedicated RS-485 driver we used a LTC485, Low Power RS485 Interface Transceiver.

The device processor requests the object values from SitePlayer through the serial port and acts accordingly, in this case, by changing the output values. At defined time intervals new readings are initiated via a serial communication line to a web sever dedicated module. The writing and reading processes, related to the web server memory, are based on keeping a single format for the serial sent/received data. This format is *objects* concept based, which are defined inside the web server, in a specific file. This way many parameters may be defined, such as the beginning address, the type, the length in bytes and the access mode (which may be unidirectional - '$OutputOnly' or bi-directional - '$bidirectional').

The warning texts are displayed using an *Intelligent Liquid Crystal Display Module* (LCD). The next method chose to improve the benefits of the projects represent the command of three relays used to signalize the crossroads. The communication with the LCD module is bi-directional. The microcontroller transmits the commands, addresses and data to the LCD module and receives the response containing the internal state of operating instruction.

The hardware diagram showing the microcontroller and all the interfaces between the software detection component and the road signaling devices are presented in Figure 17.

Project costs

Component	Quantity	Price ($)	Cost ($)
Video camera	2	75	150.00
Video capture card	1	100	100.00
Microcontroller PIC1F876	1	10	10.00
SitePlayer web-server microcontroller	1	30	30.00
Connectors, resistors, capacitors, printed circuit board	-	48	48.00
		Total (USD):	338.00

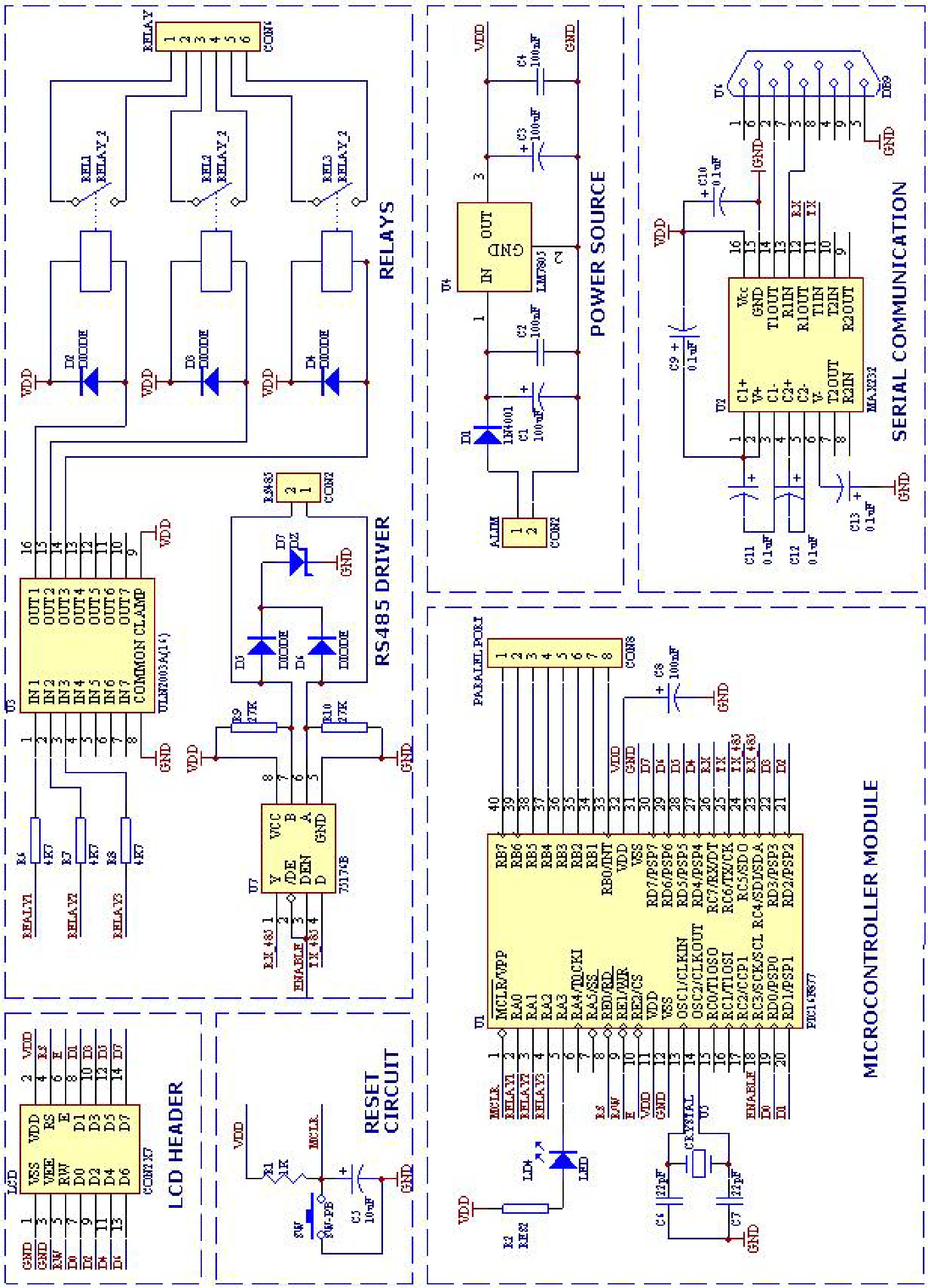


Figure 17. SafeWay Output Device Control - block diagram

Summary

We have addressed in this project the problem of road traffic safety by increasing the security level. SafeWay has been designed as an external traffic protector, street monitoring activity and protecting both drivers and pedestrians. A few new ideas are introduced regarding the road traffic systems' perspectives and also with regards to their possible implementations by using machine vision algorithms.

A working prototype of the SafeWay system was developed (Figure 18) consisting in the main processing unit (pedestrian and vehicles classifiers) and the output interface and control unit.

Figure 18. SafeWay working prototype

Recommendations for further work

Further work is in progress at the moment with regards to the mobile pager-like devices for pedestrians and drivers. We are looking to demonstrate how SafeWay can further be improved by adding mobile communication support and hence, a plus of security. We are also completing the video web streaming component and the automatic report generator.

As a general note, we have succeeded in building a working prototype of a road traffic monitor, including high performance pedestrian and vehicle detection algorithms and two interfaces for external hardware and software interaction to the SafeWay system.

References

1. Mark D. Pesce, *Programming Microsoft DirectShow for Digital Video and Television*, Microsoft Press, 2003

2. William K.Pratt, *Digital Image Processing*, 3rd Edition, John Wiley & Sons, New York, 2001

3. Christopher M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 2003

4. Duda R., Hart P., *Classification and Scene Analysis*, John Wiley & Sons, New York, 1973

5. Sonka M., Hlavac V., Boyle R., *Image Processing, Analysis and Machine Vision*, 2nd Edition, PWS Publishing, 1998

6. Russ J.C., *The Image Processing Handbook*, 2nd Edition, CRC Press.

7. Klaus-Robert Muller, Sebastian Mika, Gunnar Ratsch, Koji Tsuda, Bernhard Scholkopf, *An Introduction to Kernel-Based Learning Algorithms*, IEEE Transactions on Neural Networks, vol.12, no.2, march 2001

8. Constantine P. Papageorgiou, *A Trainable System for Object Detection in Images and Video Sequences*, MIT Artificial Intelligence Laboratory, AI Technical Report No. 1685, may 2000

9. Vapnik V, *Statistical Learning Theory*, John Wiley & Sons, New York, 1998

10. Vaillant R, Monrocq C, Y. Le Cun, *Original approach for the localization of objects in image*s, IEEE Proceedings Vision Image Signal Processing, August 1994

11. C.J. Burges, *Simplified Support Vector Decision Rules*, Proc. 13th International Conference of Machine Learning, San Francisco, 1996

12. K. Sung, *Learning and Example Selection for Object and Pattern Detection*, PhD thesis, MIT AI Lab and Center for Biological and Computational Learning, 1995

13. Baldi P., Hornik K., *Neural networks and principal component analysis: learning from examples without local minima*, Neural Networks (2), 53-58

14. P. Bartlett., *The sample complexity of pattern classification with neural networks: the size of the weights is more important that the size of the network*, IEEE Transactions on Information Theory, 1998

15. Microchip, *PICmicro Mid-Range MCU Family Reference*, 1997

16. Mark Palmer, *Low-Power Real Time Clock*, 1999

17. Stan D' Souza, *Automatic Calibration of the WDT Time-out Period*, 1997

18. Windows Media Encoder 9 Series SDK
http://msdn.microsoft.com/library/en-us/wmencode/htm/windowsmediaencoderautomation.asp

19. DirectShow SDK Documentation
http://msdn.microsoft.com/library/en-us/dndxsdk/html/ds_sdk_redirect.asp

20. Visual Studio .NET Documentation
http://msdn.microsoft.com/library/en-us/dnanchor/html/vsnetanchor.asp