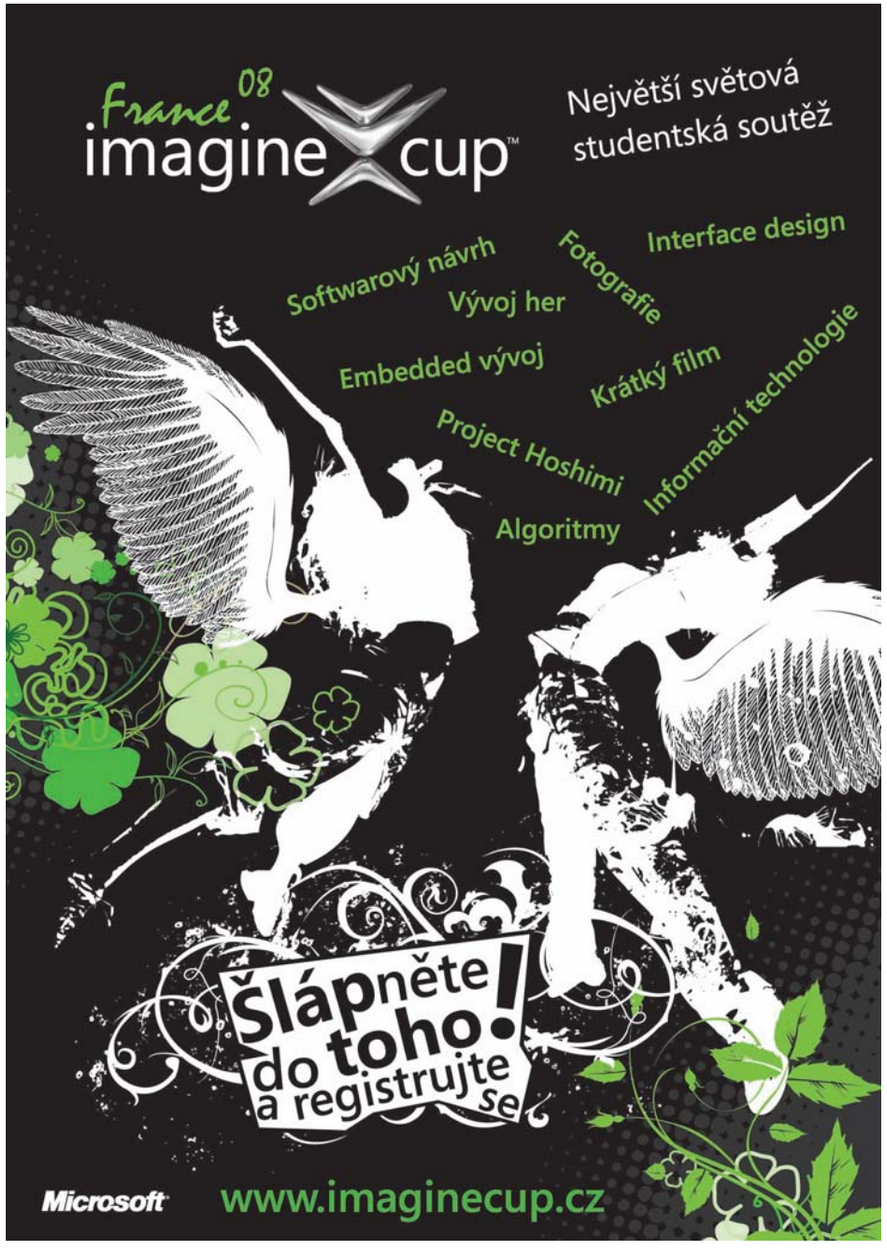


VSB - Technical University of Ostrava
Faculty of Electrical Engineering and Computer Science

Embedded Systems & Microsoft Technologies

Student Research Papers

Jan Martinovič, Jiří Dvorský (Eds.)

ISBN 978-80-248-1650-0

Ostrava, November 2007

Microsoft®

Embedded Systems & Microsoft Technologies
Student Research Papers

Faculty of Electrical Engineering and Computer Science,
VŠB – Technical University of Ostrava

ISBN 978-80-248-1650-0

Embedded Systems & Microsoft Technologies

Student Research Papers

Organized by

VŠB – Technical University of Ostrava
Faculty of Electrical Engineering and Computer Science
Ostrava, 20th November 2007

Embedded Systems & Microsoft Technologies
© Jan Martinovič, Jiří Dvorský, editors

ISBN 978-80-248-1650-0

This work is subject to copyright. All rights reserved. Reproduction or publication of this material, even partial, is allowed only with the editors' permission.

Technical editors:
Pavel Moravec pavel.moravec@vsb.cz
Jiří Dvorský jiri.dvorsky@vsb.cz

Faculty of Electrical Engineering and Computer Science,
VŠB – Technical University of Ostrava

Page count: 68
Impression: 600
Edition: 1st
First published: 2007

This proceedings was typeset by PDFLaTeX.
Printed and bound in Ostrava, Czech Republic by TiskServis Jiří Pustina.

Published by Faculty of Electrical Engineering and Computer Science, VŠB – Technical University of Ostrava

Acknowledgement

These research papers are supported by:

Table of Contents

Server Status Monitoring Using .NET Architecture and XML Web Services 1
Lubomír Walder

Personal Telemetric System – Guardian 5
Dalibor Janckulík, Ondřej Krejcar, Jan Martinovič, Petr Vašíček

Counting People in Videos 15
David Hurych

Working with MS SQL Server 2005 and CLR 25
Michal Gacek

Refactoring robotic 3D data capturing device to Microsoft Embedded platform 32
Jan Stankovič, Ladislav Kárník, David Vala, Ondřej Fialík

Monitoring the Position of a Mobile Device 39
Michal Pešat, Jan Augusta, Elišška Ochodková

Embedded Vehicle Systems and Networks 43
Michal Krumnikl, Jan Martinovič

VŠB – Technical Univesity of Ostrava, Faculty of Electrical Engineering and Computer Science 49

VŠB – Technical Univesity of Ostrava, Faculty of Mining and Geology . . . 56

Author Index 59

Server Status Monitoring Using .NET Architecture and XML Web Services

Lubomír Walder

Department of Computer Science,
Faculty of Electrical Engineering and Computer Science,
VŠB – Technical University of Ostrava,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
lubomir.walder@gmail.com

Abstract. A brief introduction to a software solution which provides remote monitoring of network machines with Microsoft Windows operating systems.

1 The Idea

In large networks with tens of servers and a countless number of end-user computers, every system administrator deals with the same challenge: how to have all those network machines, printers, routers or switches under control, how to know their status, health or performance without constant running from place to place and checking them in person. There are lots of software solutions helping to monitor all those machines or network characteristics: one can use Nagios[1], BigSister[2] or another free or commercial solution; but our idea is a bit different. The basic thought was to create a simple, user friendly, secure and network independent application for monitoring computers with Microsoft Windows operating systems. The main difference is in that those measured values are not displayed via web interface, but they are shown on a small mobile device like PDA or any other with the Windows Mobile operating system.

2 The Architecture

Company networks usually consist of lots of machines connected through switches and routers; and the whole network is secured from the danger and mess of the Internet with a network firewall. The network administrator, with his PDA, could be just somewhere out of his network reach, connected to the Internet via GPRS or otherwise. Either his PDA or our monitored machine could be hidden behind the NAT router and have only a private IP address. At the time of designing our application, we considered the worst case: the company network is hidden behind the NAT router with a very strictly configured

[1] http://www.nagios.org
[2] http://www.bigsister.ch

firewall, with other possible NAT routers in this network. Both machines, either the PDA or the monitored PC, have only local dynamic IP addresses, and thus they are passive without any possibility to accept incoming connections. To ensure communication independence from network firewalls, NAT routers and passive endpoints, we decided to divide our application into three standalone parts. Figure 1 shows the application architecture.

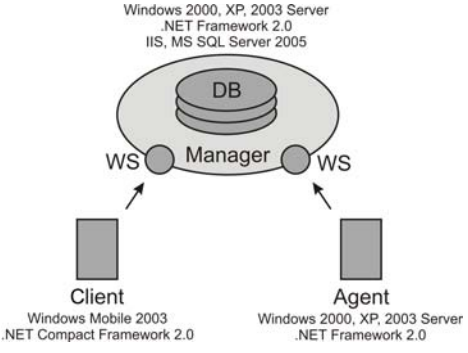


Fig. 1. The Architecture

2.1 Agent

A small application running on each monitored computer is called the Agent. The Agent is an ordinary Windows application and consists of several probes. Each probe monitors one value, e.g. we can have a probe monitoring the CPU load, another probe gathering data about the free disk space, yet another one monitoring the network adapter load and so on. Currently, all probes gather data from the system performance counters. The data is periodically collected, buffered and sent to the Manager via web service interface. In case the probe value reaches or exceeds a predefined value, an alarm is sent to the Manager immediately. Figure 2 shows the main window of the Agent.

2.2 Manager

The Manager is a publicly accessible web application consisting of a database and two web service interfaces. It receives the data from every Agent, stores it into the database and mediates it to Clients. The Manager also provides authorization for both Agents and Clients.

2.3 Client

The Client is an application made for Windows Mobile 2003 and higher operating systems. The function of this application is to periodically download the data

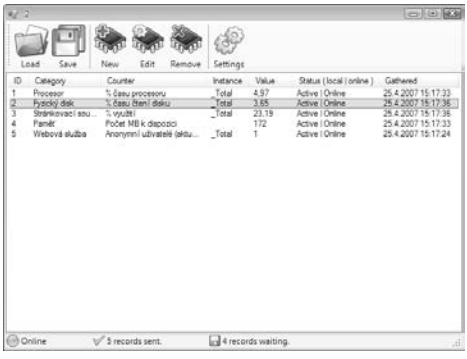


Fig. 2. Main Window of the Agent Application

from the Manager and to visualize it within the user interface. It also provides simple alarm checking and warnings to alert the user to critical situations. Figure 3 shows the main window of the Client application.

Fig. 3. Main Window of the Client Application

3 The Future

Our software system is currently at the stage of a prototype. It offers only one method of checking the computer status (the system performance counters); only basic monitoring functions for the Client; and the role of the Manager is relatively passive. In the future, we plan to extend the variety of monitoring probes to WMI probes, PING probes, and SNMP probes to allow the Agent

to monitor even the surrounding SNMP-capable network devices. Also, a web interface or a client application for Windows operating systems comes forward. As regards the Manager, we propose giving it the possibility to monitor the Agents to see whether they are still alive and to raise an alarm if the Agent is not responding for defined time. Currently this alive-check is under the Client's competence.

Personal Telemetric System – Guardian

Dalibor Janckulík, Ondřej Krejcar, Jan Martinovič, Petr Vašíček

Faculty of Electrical Engineering and Computer Science,
VŠB – Technical University of Ostrava
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
{dalibor.janckulik.st1, ondrej,krejcar, jan.martinovic}@vsb.cz,
petr.vasicek@centrum.cz

Abstract. This project deals with the problems of utilization of mobile equipment working in the biomedicine field, particularly telemedicine. This field is relatively new; it focuses on the observation of life functions from a distance. Practically developing system works with an ECG sensor connected to mobile equipment, such as PDA/Embedded, based on Microsoft Windows operating system. The whole system is based on the architecture of .NET Compact Framework, and other products, such as SQL Server by Microsoft too. This work also deals with the communication of mobile equipment with sensors and with the server via Bluetooth, WiFi, and GPRS/EDGE. The mobile equipment used serves primarily for measuring and processing of data from the sensors and their visualization as a graph. The data is also given to the server for further processing and the analysis of current health of the patients, due to small efficiency of the mobile equipment [1]. The main task we deal with in the server part of application is receiving of the data via web services and further processing, management and analysis of this data. For the analysis of received data and further evaluation of the electrocardiogram, there is a self-organizing neural network [2].

1 Motivation

Many middle-aged people like businessmen, CEOs, managers and other have very hectic lives with much stress and without good ways of living. Sometimes these people have a collapse, breakdown or heart attack and must be in hospitals or health resorts for a long time to regenerate their bodies. The time that they spend in these institutions, is nonutilisable and very long for them. Possibilities of today physics are restricted by many of prescripts so patients cannot use some of the newest techniques (like hyperbaric or arctic chambers), which make it possible to reduce the regeneration time by weeks or months. For example, these chambers are restricted to patients in the first six months after heart attack due to no information about patients' conditions during the procedure.

Here is the main area of utilization of our telemetric system. Of course the use of our system is not limited only to businessmen, but it is targeted to middle-aged people with some knowledge about new technology like mobile phones. The

price of client devices of our system is not low, so we suppose people who can invest to these sorts of assistants.

The basic idea is to create a system that controls important information about the state of a wheelchair-bound person (monitoring of ECG and pulse in early phases, then other optional values like temperature or oxidation of blood ...), his situation in time and place (GPS) and an axis tilt of his body or wheelchair (gyroscope).

Values are measured with the help of existing equipment, which communicates with the module for processing via Bluetooth wireless communication technology.

Most of the data (according to heftiness) is processed directly in PDA or Embedded equipment to a form that is acceptable for simple visualization. Two variants are possible in case of embedded equipment – with visualization and without visualization (entity with/without visualization module). Data is continually sent by means of GPRS or WiFi to a server, where it is being processed and evaluated in detail. Processing and evaluating on the server consists of - receiving data, saving data to data storage, visualization in an advanced form (possibility to recur to the older graph, zoom on a histogram (graph with historical trend), copying from the graphs, printing graphs), automatic evaluation of the critical states with the help of advanced technologies (algorithms) that use Artificial intelligence to notify the operator about the critical state and its archiving.

Application in PDA, Embedded equipment is comfortable, with minimum time - the first configuration, but also configuration after downfall of application. The level of visualization will be lower. Visualization of three to six samples of ECG, after switch-over to application, makes it possible to browse history according to data in data memory. The described system can be used with small modifications for monitoring of patients in hospitals or people working in extremely hard conditions. The biggest limitation is the availability of measuring devices in acceptable and adaptable sizes or comfortable enough to have one around [1].

The measuring device was tested in extreme conditions in a cryogen room[1] in Teplice (−136°C), where the final system will be installed. Implementation of the data transmission security was not solved. The whole system is classified as "work in progress" system and it is in a testing phase where we found mistakes and repaired them.

2 System Architecture

This system consists of several interconnected parts that can communicate among themselves, so they can approach their function. These parts are: measuring sensor, PDA or embedded device, and server for data processing and evaluation. Beside these, the most important parts that are worth mentioning

[1] Cryogen room – room with very low temperature, nowadays using low temperature for medical acts

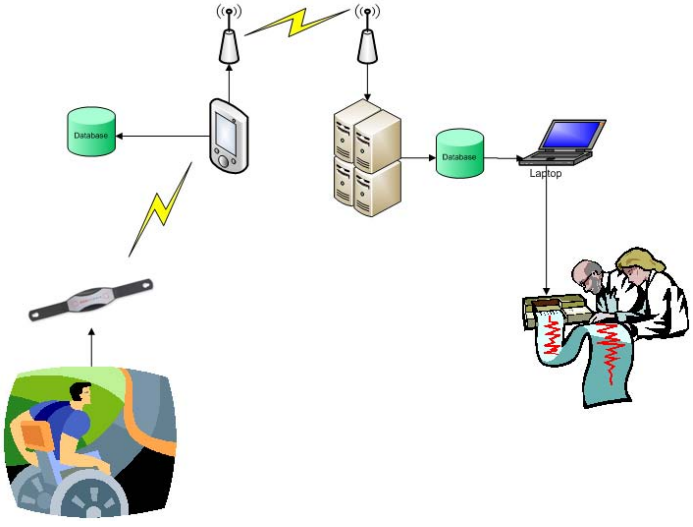


Fig. 1. Measurement Schema

are various kinds of accessories, such as GPRS, WiFi, GSM, Bluetooth or GPS modules. By means of them we can communicate. We mostly use the fastest technology in signal coverage.

Data acquisition and data transmission are the most important parts of this system. The Responsible for correct working are: correct sensor configuration, sensor calibration, data transfer synchronization, and mutual communication between sensors and data receiver.

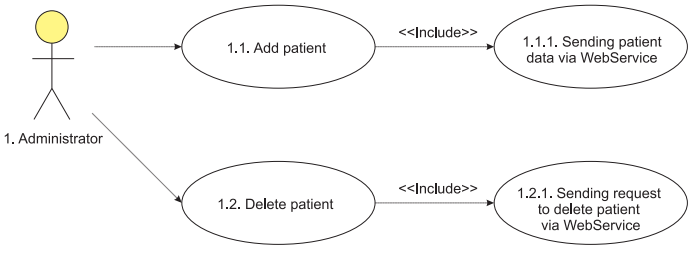


Fig. 2. Administrator's Possibilities

The administrator can add or remove a new patient by means of a PDA program. This step is represented in a server with WEB service. Patient's data is removed or added into a database file.

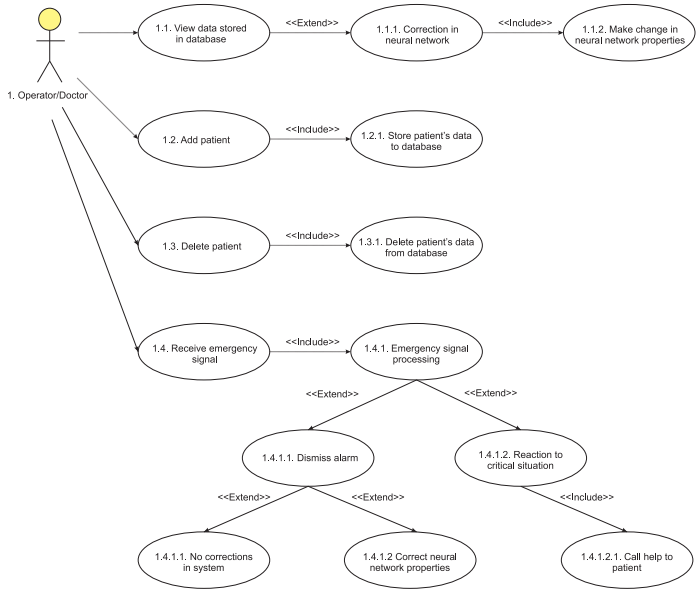


Fig. 3. Operator's/Doctor's Role

The system can display saved data from a database file. The doctor can configure or set a neurone network. A change of that is shown in the XML file enshrining, where the neurone network setting of the patient is kept.

The doctor receives information about worsening of patient's status. In case of doctor's reaction, he sends for an expert assistance, such as a helicopter or an ambulance. In case of false alarm, he can configure a neurone network or leave it unchanged if that was a sporadic incorrect interpretation.

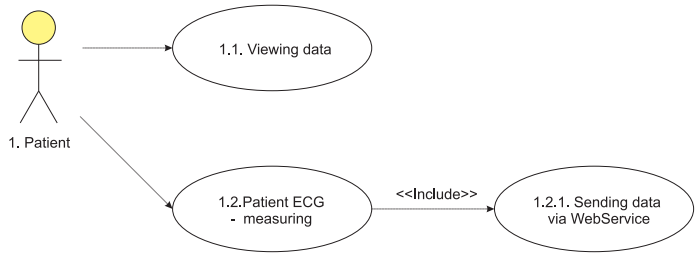


Fig. 4. Patient's Role

The patient can browse data concerning his health status. Measured data is sent to server by WEB service.

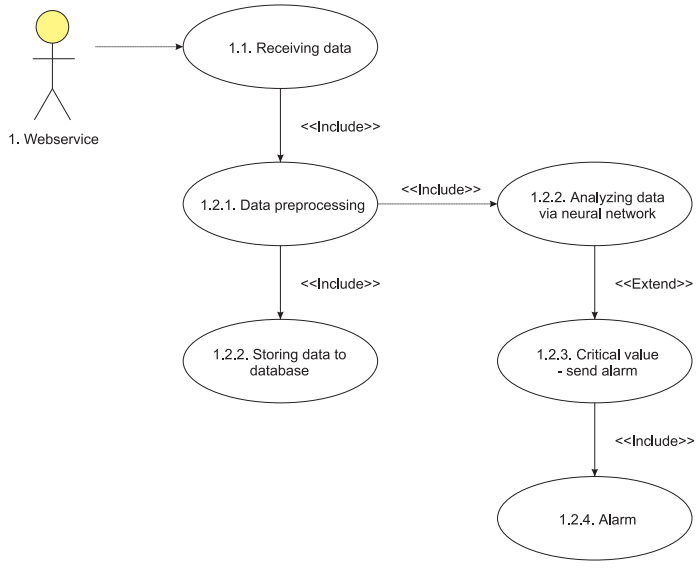


Fig. 5. Web Service

Client's data are not only received but also preprocessed (data checking, risk elimination etc.). Measured data is saved. And now, it is possible to analyze the data using a neurone network. If the analysis shows that the measured data from ECG is critical, a warning is sent. That notifies a doctor of incoming data.

2.1 Mobile Part

The headstone of the whole system is an Embedded or PDA device. The difference in applications for measurement units is the possibility to visualize the measured data in both Real-time Graph and Historical Trend Graph, which can be omitted on an embedded device.

PDA is a much better choice for Personal Healthcare, where the patient is already healthy and needs to review his condition, or for multiple person usage. Embedded devices can be designed for one user, with the option to use an external display used for settings or with the possibility of usage in extreme conditions.

The application is communicating with an ECG Measurement Unit (Corbelt or Blue Keg) through a virtual serial port using wireless Bluetooth technology. Then, after pushing a button, all necessary parameters are set and the communication may begin. Measured data is stored on a Memory Card in a database in MS SQL Server 2005 Mobile Edition [1].

During implementation, several problems were found, resulting from faults and simplicity of the .NET Compact Framework architecture.

– Communication – the Framework SP1 is needed for the support of Virtual Serial Port for HP devices with Windows Mobile 2003 operating system.
– Virtualization – graph visualization without Draving.2D name space.
– Data storage – writing speed, data storage demands, automatic incrementation of the ID of data record.

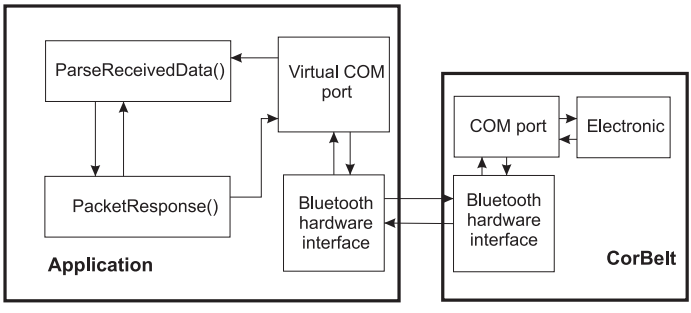


Fig. 6. Bluetooth Communication

The performance of available devices seems insufficient for sequential access; parsing of incoming packets is heavily time-consuming. Pseudo paralleling[2] is required. If Windows Mobile OS versions 2003 to 5.0 are used, the processing of data from a professional EKG is not realizable due to thread count limitations. A newer operating system can be used to solve this [3].

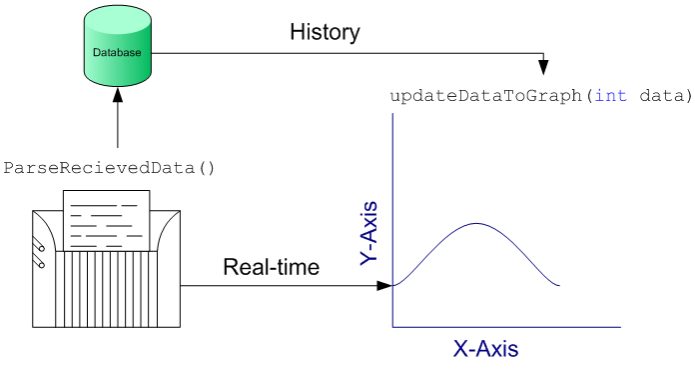


Fig. 7. Packet Parsing

Current application is highly specialized and written to accommodate specific hardware. Usage of any other hardware is not possible. This is due to different

[2] pseudoparallel – process switching

methods of packet folding, which are unique on each device [4]. This is partly caused by the length of the Telemedicine branch.

Operating of the device is simplified as much as possible with the least possible number of steps regarding user registration, measurement device connection and the measurement itself.

2.2 Server Part

In order to run a server, an operating system supporting IIS is needed. IIS is an Internet Information Server application allowing users to connect to the web server by the well-known HTTP protocol [3].

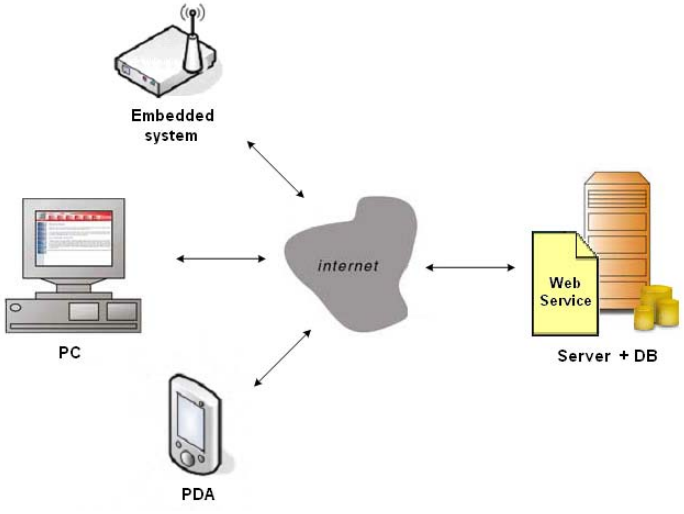


Fig. 8. Web Service Communication

The web service transfers data between the server and PDA/Embedded devices. It reads the data, sends acknowledgments, stores the data in the database and reads it from there. The service is built upon ASP.NET 2.0 technology. The SOAP protocol is used for the transport of data, which is in XML format. That is an advantage since it allows communication of multiple different technologies and platforms.

The Wireless ECG approaches a real professional ECG with data rate as high as 800 records per second [4]. That makes 48,000 records per minute and 2,880,000 per hour. Considering 100 patients, the value gets to 288,000,000 records per hour. Even if the server accepted only 50 records per second, the sum of records for 100 patients per hour would be 18 million. That is an extreme load for both the server and the database system; hence a better way of storing data is needed.

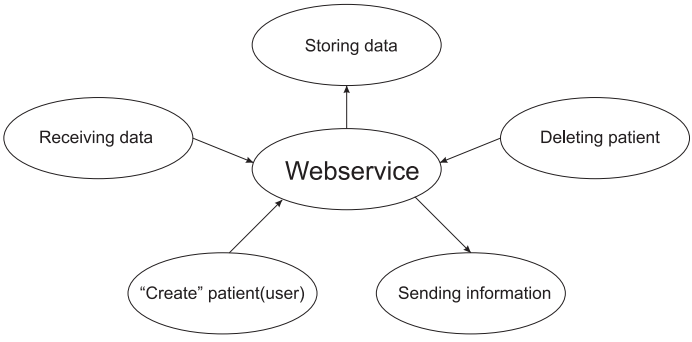


Fig. 9. Web Service Possibilities

Methods that devices communicating with the web service can use include:

- receiving measured data
- receiving patient data
- deleting a patient
- patient data sending

To observe measured data effectively, visualization is needed. A type of graph as used in professional solutions is an ideal solution. To achieve this in a server application, a freeware Zed Graph library can be used.

For data analysis, neural nets are a convenient solution. However, there are problems in the automatic detection of critical states. Every person has a specific ECG pattern. What is completely normal for one person can indicate crisis for another. The Neural net has to learn to distinguish critical states of each patient separately.

Basic characteristics of a neural net:

- 10×10 neurons
- weights with decimal point values for precise results
- learning is based on 3 – 4 minutes of recording, which is approximately 36,000 – 45,000 recorded values
- incorrect values are filtered out
- filtering decreases the amount of values to about 10%, which is still good enough for learning
- the learning cycle with 4,200 values takes approximately 30 seconds (CPU PIII 1.2 GHz) with C# implementation
- after each data reception, the net responds with YES/NO (accepted/declined)
- interruption can be invoked on demand, thus making correction of the net by a new value [2]

To make the specialist's or operator's intervention possible, the system must be provided with a user-friendly interface, possibly imitating those on medical appliances. This area is still in an early phase of development.

3 Data Analysis

Data acquired from the measuring device is incomprehensible for a man, because it is represented by a HEX[3] format packet. The data needs to be stripped of redundant information like packet numbering and transformed to a recognizable state – a graph.

This is done during the packet parsing in PDA/Embedded device, where HEX information 2 bytes in length is transformed to a binary state, where the data is carried on the first 12 bits. It is transformed to a decimal state and sent for further processing – sending the data to the server, storing it in a local database or visualizing it.

The visualization using PDA/Embedded is just a simple visualization of a curve in real-time or historical graph. In the server application, the visualization is far more complex, with the possibility of storing the current curve as an image, printing, or zooming.

Neural net of the SOM[4] type on the server is 10×10 in size. The initial weight of each neuron is random. The weights are assigned progressively by learning. Finally, the whole structure is ready to accept data to analyze. Each patient has his own neural net stored on a server in XML format.

4 Summary

The evolution of Telemedicine is unstoppable and apparent; therefore ways need to be found to improve the quality of hospital services, spa services or hazardous environment workplaces. The area of software products working in embedded devices in hazardous environments or PDAs in personal healthcare is still open and unoccupied.

Personal healthcare products are freely available [5] and with minor hardware modifications they can be used for data acquisition using wireless Bluetooth or ZigBee technologies. By means of data transfers, the acquired data can be gathered in database systems providing access to our personal doctor, who can be alerted in case of trouble.

Why are there systems available to protect our property and not our health? Is it not the most valuable property of ours?

References

1. Janckulík D.: Personální telemetrický systém - Osobní mobilní ordinace, bachelor thesis, VŠB - Technical University of Ostrava, 2007, (in Czech)
2. Vašíček P.:, Osobní biotelemetrický systém, master thesis, VŠB - Technical University of Ostrava, 2007, (in Czech)
3. http://www.microsoft.com

[3] HEX – hexadecimal order
[4] Self-Organizing Maps – artificial neuron network type

4. http://www.corscience.de, Corbelt, BlueEKG datasheet, checked January 30, 2006
5. http://www.taylorgifts.com, checked April 14, 2007
6. Kohonen T.: Self-organizing maps, 3rd edition, Springer, 2001

Counting People in Videos

David Hurych

Department of Computer Science,
Faculty of Electrical Engineering and Computer Science,
VŠB – Technical University of Ostrava,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
David.Hurych@seznam.cz

Abstract. This work briefly describes the development of a system capable of monitoring and counting people walking across an area captured by a digital video camera. The input for the system is recorded video or data provided directly from a stationary video camera. The solution is based on the techniques of corner detection, optical flow computation, clustering of points, and monitoring objects. The work also includes testing of the designed system and the following modifications for increased efficiency. The conclusion describes the results of the experiments and summarizes the pros and cons of the solution.

1 Introduction

Monitoring the motion of people is already important in many areas, for instance in securing buildings with surveillance cameras, where it is necessary to monitor and analyze the motion of people and their activities. It can also be used for automatic monitoring and counting of people passing through an entrance to find out the attendance at the monitored facility. The results of automatic monitoring of people can also be used to find out the number of people who passed through the monitored area at a given part of the day or a specific hour. When we have the direction of the motion of individual people, the system can be used to monitor the filling of the capacity of a given building, which can be useful for various safety reasons. Since the system of monitoring and counting of people shall be based mainly on methods of motion detection and velocity field analysis, it will be relatively easy to change it into a different system that will monitor for example, the motion of cars. Such a system has many possible uses.

2 Solution Method and Application Structure

The programme was created to be tested and used with a camera developed at the Department of Computer Science of the Faculty of Electrical Engineering and Computer Science of the Technical University of Ostrava. The camera has an integrated computer with a Pentium M 1.8 GHz CPU, 512 MB RAM, and Windows XP Embedded OS.

Before starting the work, it was necessary to decide what camera position for monitoring the scene should be chosen and what methods of counting people should be used. One of the possibilities is a vertically positioned camera at the vertex. This position is very useful because the moving people are not overlapping and can be relatively easily discerned in space. This approach was already used and tested several times [1–3], which is why I decided to use a camera position that will capture the moving people from the profile. Here we have to take into account the limitations from overlapping people, and we also have to expect that people will move mostly from right to left or vice versa and not from or towards the camera , because their motion would be very badly detectable. The advantage of this position is the possibility to capture a scene from multiple angles of view, and not only a small area from above.

The assignment was to use the method of velocity field analysis. Velocity field basically describes the motion in the image. My solution uses optical flow computation only for several selected points in the image in order to maintain the highest possible computation speed. The automatic selection of points where motion is to be detected uses the corner detection algorithm. We then detect the motion of these points between two reference frames by means of the Lucas-Kanade method. We have to perform a suitably adjusted clustering method over the points where motion was detected in order to best detect to which objects individual groups of points belong. Not all clusters detected in a scene belong to moving people. In the usable clusters we have to observe their motion in the image scene and deal with possible overlapping and losses. Once the observed cluster leaves the image scene, we decide about its inclusion in passing people. The methods used in the solution are described in the chapters below.

3 Detection of Points of Interest

To detect the points where motion is to be identified, we used the corner detection algorithm by Shi and Tomasi [5]. Detected corners will serve as the points that can be easily identified and monitored in the image. The method is based on the Harris corner detector. This detector is very popular thanks to its resistance to rotation, change in scale, and patch illumination, and also thanks to its resistance to noise. The Harris corner detector is based on calculating the local autocorrelation function which measures the local changes in brightness between two slightly shifted image patches [7]. With the shift $(\Delta x, \Delta y)$ and point (x, y), the autocorrelation function is defined as

$$c\left(x,y\right) = \sum_{W} \left[I\left(x_i, y_i\right) - I\left(x_i + \Delta x, y_i + \Delta y\right)\right]^2, \tag{1}$$

where (x_i, y_i) points of the W (Gaussian) window centered to the point (x, y). Lets approximate this shifted image by a first-order Taylor expansion

$$I\left(x_i + \Delta x, y_i + \Delta y\right) \approx I\left(x_i, y_i\right) + \left[I_x\left(x_i, y_i\right) I_y\left(x_i, y_i\right)\right] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}, \tag{2}$$

where I_x and I_y are partial derivations in x and y directions. By supplying the equation (2) to equation (1) and modifying the resulting equation, we get

$$c(x,y) = \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} \begin{bmatrix} \sum_W \left(I_x\left(x_i,y_i\right)\right)^2 & \sum_W I_x\left(x_i,y_i\right) I_y\left(x_i,y_i\right) \\ \sum_W I_x\left(x_i,y_i\right) I_y\left(x_i,y_i\right) & \sum_W \left(I_y\left(x_i,y_i\right)\right)^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$
$$= \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} C(x,y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}. \tag{3}$$

Let λ_1, λ_2 be eigenvalues of the matrix C. Shi and Tomasi showed that a good measure for identifying a corner is

$$min\left(\lambda_1, \lambda_2\right) > \lambda, \tag{4}$$

where λ is the predefined threshold that must suffice to reduce the influence of noise in the image. With this threshold we also influence the quality of detected corners and therefore the number of points for further processing. If both eigenvalues are very different from zero, just as the partial derivations, we can say that we found a corner and we can consider the given point with its close neighbourhood of several pixels the point of interest which can be well identified in the scene and which is suitable for optical flow computation. The corners in a real scene detected with this method are marked yellow in the left part of Figure 1.

4 Optical Flow Computation

In this work we use the Lucas-Kanade method which is one of the most used methods and it is based on detecting the changes in brightness in the image. It was created by Bruce D. Lucas and Takeo Kanade. The method assumes that optical flow in a small local environment is constant, which solves *the aperture problem*. Applying the *optical flow equation* $I_x u + I_y v + It = 0$ on several surrounding pixels reduces the whole computation problem to solving a system of linear equations. However, this only applies if all points used for one computation have the same velocity. The system of equations with two unknowns for two pixels has only one solution. It is however more effective to combine equations for more than just two pixels. In this case, the approximate solution is found by using the method of the least squares. The equations are usually weighted. The vector of the optical flow is obtained by solving the equation

$$\boldsymbol{v}_{opt} = \sum_{x=p_x-\omega_x}^{p_x+\omega_x} \sum_{y=p_y-\omega_y}^{p_y+\omega_y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}^{-1} \begin{bmatrix} \delta I I_x \\ \delta I I_y \end{bmatrix}. \tag{5}$$

This equation gives an approximate solution to the optical flow in the local neighbourhood of a pixel measuring $(2\omega_x + 1) \times (2\omega_y + 1)$ – what is known as the integration window. We get a more precise solution of optical flow from an iterative solution of this equation [8, 6].

The basic algorithm [4] tries to find and correctly locate the patch from one image in the second image. The aim of the Lucas-Kanade algorithm is to minimize the sum of error squares between two images – between the patch and the part of the image where we are looking for the patch. The method is very computation demanding so it is necessary to reduce the area of the image where we are looking for the patch. That is why the method works reliably only for small motion (in pixel units between two frames). This problem is solved by what we call the pyramidal computation - i.e. the method used in the solution is *the pyramidal iterative Lucas-Kanade method* [6]. The basis of the algorithm is that the resolution of two images is gradually decreased and optical flow is computed in multiple resolutions. In low resolution the original motion of e.g. 20 pixels deceased to 2 pixels and it can be computed using the Lucas-Kanade method. Small motion, on the other hand, is computed in high resolution, thus covering all motion velocities in the image. The disadvantage is however, that repeated computation is required for two reference images. For our particular use, it was verified that we only need three pyramid levels for computing the optical flow with decreased resolution. In Figure 1 on the right we can see the points in a real scene with detected motion marked in yellow.

Fig. 1. Left – detected corners. Right – corners with detected motion

5 Clustering of Points

After filtering the points that moved between frames, we have to find out to which objects these points belong. The basic clustering algorithm consists of the following three steps:

1. The input for the algorithm is a set of points. Create a separate cluster from each point.
2. Check the clusters and make a single cluster from the two clusters that are close enough.
3. Repeat step 2 until there are only clusters that are so far from one another that no two can be joined into one. The output of the algorithm is a set of points arranged in clusters.

Modifications of the basic algorithm consist mainly in its optimization in detecting the proximity of points. This means comparing each two points no more than once. The program checks the points of one cluster, checks the remaining clusters for each of its points and detects if they contain a point through which the clusters could be joined. This means that two clusters can be joined into one if at least two of their points are close enough. If it finds such a point in another cluster, all points of the cluster are assigned to the first cluster. However, after checking the current it is necessary to check the points of former separate clusters because through their points other clusters can be joined too.

Figure 2 shows the procedure of joining clusters. The program gradually checks the points (marked green) of the processed cluster and for each of its points compares the distances from the points of the remaining clusters to the right of the processed cluster. When any of the distances of two points is short enough, it adds the points of the cluster on the right to the processed cluster (left part of the figure). The right part of the figure shows that the points added to the processed cluster must still be compared to the remaining clusters through which they could be joined.

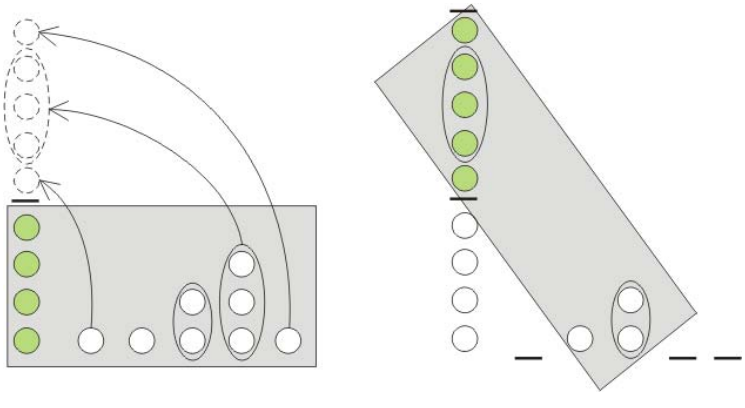


Fig. 2. Clustering procedure

5.1 Filtering Clusters

Not all detected clusters of points can be considered to be a moving person. It is therefore necessary to find out which clusters are suitable and which can be discarded. The main sign that can be considered a feature representing a person is the proportion of height to width. Furthermore, the velocity of a cluster must be within certain limits. When a person walks past a camera, then we expect certain velocity limits in the cluster that represents the person. The clusters that pass this series of tests are then classified for processing.

6 Monitoring Clusters between Frames

Now we can already distinguish individual objects between two frames. In the next stage it was necessary to obtain information about the motion of objects within the scene. For monitoring a cluster of points we first had to determine what we actually need to monitor. I chose monitoring of only a single point for each cluster, the point being the cluster's centre of gravity. The coordinates $T(x, y)$ of the centre of gravity T of the points in a single cluster are computed as follows

$$T_x = \frac{1}{n}\left(\sum_{i=1}^{n} A_{xi}\right)$$
$$T_y = \frac{1}{n}\left(\sum_{i=1}^{n} A_{yi}\right),$$

where n is the number of points in the cluster, A_{xi} is the x-coordinate of the i^{th} point of the cluster and A_{yi} is the y-coordinate of the i^{th} point of the cluster. When we have the point which represents the object, we can monitor it during its motion in the scene. For each observed cluster, the expected position of the centre of gravity in the following frame is pre-calculated between each two frames. Then the currently detected clusters are compared to the observed clusters to discover similarities. Based on the shortest distance criterion, one cluster is selected from the suitable clusters, which is assumed to be the successor of the observed cluster. When none of the detected clusters is suitable, we register the loss of the observed cluster which is still monitored for a certain number of frames, so that it can possibly be found again. We calculate the assumed position P_{n+1} using the vector that is defined by the first and the last point of the trajectory. The length of this vector is then divided by the number n of the preceding motion vectors, and we can now determine the assumed position of the object

$$P_{n+1} = P_n + \frac{P_n - P_1}{n}.$$

To update the observed cluster of points with one of the newly found clusters we first need to compare their characteristics. With respect to minor inaccuracies in identifying the points of interest and computing the optical flow, we cannot expect that two reference clusters would be absolutely identical, also due to the changes in the width and height of the cluster caused by irregular motion of a person. However, we can expect only minimal changes in characteristics, so two clusters can be considered identical if the differences are within certain limits.

Theoretically, there can be multiple suitable candidates found for a single observed cluster, so we need to select the best one. The selection criterion is the shortest distance of the centre of gravity $T(x, y)$ of the suitable clusters from the assumed position for the observed cluster $P(x, y)$ according to the formula

$$MIN\ {}_{i=1}^{v}\left(\sqrt{(P_x - T_{xi})^2 + (P_y - T_{yi})^2}\right),$$

where v is the number of suitable clusters and T_i is the centre of gravity of the i^{th} of the suitable clusters. When the best candidate is selected, the observed cluster (object) must be updated by taking over all the characteristics of the newly found cluster. Furthermore, another position is inserted in the observed trajectory, which is the coordinates of the new centre of gravity.

7 Counting People

As we have already mentioned above, the programme distinguishes people from other moving objects mainly by the velocity of their movement and the proportion of the height to the width of points. For an observed object to be included among passing people, it must meet two conditions. It must travel a certain minimal distance across the image and it must pass through the counting line in the centre of the image.

8 Optimization

The program was designed on a computer with an AMD 2100+ XP CPU and the camera for which it was designed has a CPU of similar performance, namely Pentium M 1.8 GHz. In the original resolution of 640×480 the program processed sample videos for about three times longer then their actual lengths. It was therefore necessary to optimize the program. The actual algorithm for monitoring and counting clusters is not CPU demanding, as it always works only with several clusters at once (not more then ten) and does not make any complicated computations. The computation times of individual parts of the algorithm show that the most computation time is taken by methods of identifying the points of interest and computing the optical flow.

The points of interest are computed using the *cvGoodFeaturesToTrack()* method from the free *OpenCV* library by Intel. By setting several parameters (such as the threshold for eigenvalues λ_1, λ_2 of the covariance matrix) we can determine the quality of corners and consequently the number of detected points. To compute the optical flow we also use a method from the *OpenCV* library called *cvCalcOpticalFlowPyrLK()*. The method computes the optical flow by means of *the pyramidal iterative Lucas-Kanade* [6] method. The method also provides several parameters for setting the computation quality and speed. The first parameter is the integration window dimensions, another is the number of iterations. However, by optimizing the input parameters of these functions we only achieved minor improvement in the computation speed. That is why I used the alternative of reducing the original video resolution to 320×240. Reducing image quality does not have a large effect on motion detection in the image, which is already shown by the pyramidal reduction of resolution in the optical flow computation method used. By reducing the image resolution to half, we achieved computation times sufficient for processing data from the camera in real time without loss of frames at 30 fps.

Another experiment was testing the alternative incremental clustering algorithm called *leader* [9, 10]. The algorithm showed to be unusable, as it depends on the order of input points.

9 Analysis of Processed Data

When monitoring clusters (objects) in real time and counting them at a single moment, it is not really possible to analyze the complex relations between them. That is why important information about detected clusters is stored during the computation in order to make more complex analysis with the relations to other clusters and to test the counting of people only after processing a longer part of the video. The about which clusters appeared at the same time in the same place (close surroundings) in the image and their mutual relations is surely useful.

The original idea was to correctly count a group of two or more people. It is necessary to identify the relations between clusters in time and space. For each two clusters, their time intervals of appearing in the scene are compared and it is determined whether there was at least partial overlapping, and this information is stored. Now we know which clusters were in the scene at the same time. This however does not mean that they also must be in the same place in the image. So the groups formed thanks to identifying the relations in time must be divided according to their relation in space. The analysis procedure is shown in Figure 3.

This way we obtain several groups and we can assume that each group represents one object in the scene. The object can be either one person or one group of people. The data in a group can be further analyzed to determine how many people it may represent. This gives us an alternative method of counting people.

Table 1 shows the results of counting people. The actual number of people who passed in front of the camera is compared with the results obtained from automatic counting of passing people in real time and the results of the subsequent data analysis.

Table 1. Results of automatic counting of people

video name	actual number of people captured by camera	number of people detected by the monitoring algorithm	number of people detected by subsequent data analysis
MVI_6236	1	1	1
MVI_6237	2	2	2
MVI_6239	3	3	3
MVI_6240	4	2	3
MVI_6242	4	2	4
MVI_6243	1	1	1
MVI_6245	6	5	6

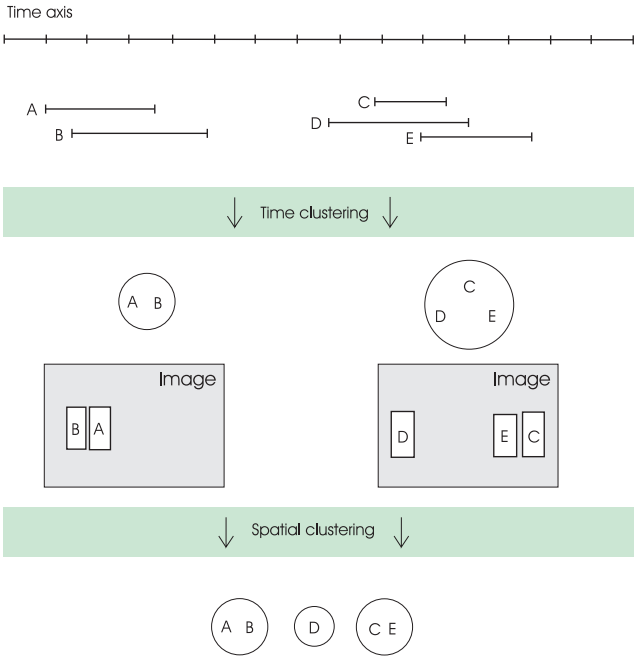


Fig. 3. Illustration of space-time clustering

10 Conclusion

The monitoring algorithm reliably counts individual passing people. It also correctly counts multiple people in the image who don't overlap. It can correctly count two people walking in the opposite direction towards each other who cross in the image. The algorithm shows robustness in frequent loss of information about the monitored objects. When a group of several people appears in the image, they are either counted as a single person if they overlap almost completely, or if the overlap is only partial and the group occasionally splits into individual persons and consists of two people, then there is some chance of correct counting. The algorithm fails to count correctly groups of more than two people, or multiple people in a too crowded scene. The testing showed that the algorithm can correctly count one to three people appearing at the same time in the scene who don't overlap (or rarely overlap).

The success rate of counting people by subsequent data analysis is better. Thanks to considering the time-space relations between the observed clusters and distinguishing the number of people in a group using the average value of the proportion of height to width, it is successful in correctly counting even passing groups of two people. However, the algorithm doesn't count correctly groups of three or more people with complete overlapping of people. We can therefore say that we obtain better results from subsequent analysis of data.

References

1. Rossi, M., Bozzoli, A. *Tracking and Counting Moving People,* IRST - Instituto per la Ricerca Scientifica e Tecnologica, I-38050 Povo, Trento, Italy, 1994.
2. Terada, Kenji, Kurokawa, Natsuki *A Method of Counting the Passing People by Using the Method of the Template Matching,* Faculty of Engineering, The University of Tokushima, Japan, 1998.
3. Terada, K., Yoshida, D., OE, S., Yamaguchi, K. *A Counting Method of the Number of Passing People Using a Stereo Camera,* Faculty of Engineering, The University of Tokushima, Tokushima, Sogo Security Service Co., Ltd., Tokyo, Japan 1999.
4. Lucas, Bruce D., Kanade, Takeo *An Iterative Image Registration Technique with an Application to Stereo Vision,* Computer Science Department, Carnegie-Mellon University, Pittsburg, Pensylvania 1981.
5. Shi, Jianbo, Tomasi, Carlo *Good Features to Track,* Computer Science Department, Cornell University - Ithaca, Stanford University - Stanford, 1994.
6. Bouguet, Jean-Yves *Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the Algorithm,* Microprocessor Research Labs, Intel Corporation.
7. Derpanis, Konstantinos G. *The Harris Corner Detector,* 2004.
8. INTEL, *Open Source Computer Vision Library - Reference Manual,* Intel Corporation.
9. Jain, A. K., Murty, M. N., Flynn, P. J. *Data Clustering: A Review,* Michigan State University, Indian Institute of Science, The Ohio State University.
10. Kummamuru, K., Lotlikar, R., Roy, S., Singal, K., Krishnapuram, R. *A Hierarchical Monothetic Document Clustering Algorithm for Summarization and Browsing Search Results,* New Delhi, North Guwahati, India.

Working with MS SQL Server 2005 and CLR

Michal Gacek

Department of Computer Science,
Faculty of Electrical Engineering and Computer Science,
VŠB – Technical University of Ostrava,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
gacek@post.cz

Abstract. Abstract: This work describes how the new functions offered by SQL Server 2005 together with the development environment of Visual Studio 2003 can be used in real applications. In my own application dealing with searching in text by means of what is called chains I implemented my own SQL Server data type, user defined functions and stored procedures. I implemented all of the above-mentioned three new possibilities directly in my own database using assembly, which not only shortens the time for searching but also simplifies distribution of the resulting application, while at the same time everything is secured just like the data in the database.

1 Introduction

In my work [8] dealing with searching in text based on chains[1]. I encountered the problem of how to effectively solve storing the chains in a database. It would be best to store chains as fields of integers. I therefore decided to try out a new function in MS SQL Server and create my own field data type for storing my chains. I also decided that it would be good to transfer the search functions into user defined functions and implement them into the database together with the type and stored procedures which handle filling the database data.

This way everything will be implemented directly in the database and secured the same way as the data stored in the database. This will not only simplify distribution but further development will also be much easier. The programmer developing an application based on this database only needs to know the interface required for communication with the functions, and he/she can use them directly without knowing anything about their internal arrangement, i.e. he/she can directly start developing his/her database.

Furthermore, thanks to the compatibility of MS SQL Server and MS Visual Studio, it is very easy to create these databases with a user defined code, be it UDT, UDF, or SP. Everything is already prepared in Visual Studio as templates. All you need to do is select an SQL Project, choose a template and start. Visual Studio will also take care of loading into the database, thus considerably speeding

[1] chain - a field of integers representing the positions of the given term in the given document

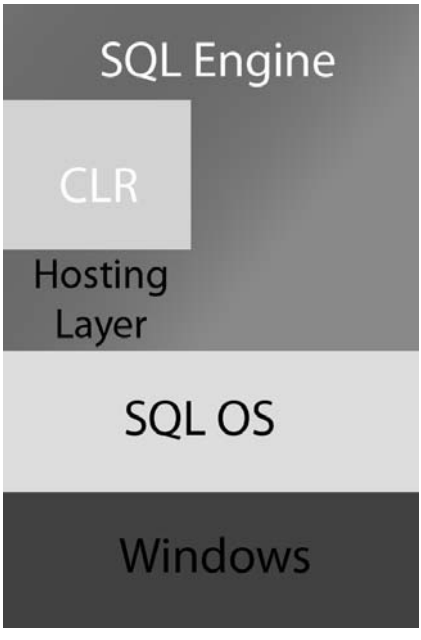


Fig. 1. CLR layer integration principle

up the development, which I personally experienced when I was loading the first attempts manually.

Another clear benefit is using an arbitrary programming language in the Visual Studio environment; it is only up to the programmer which he/she is used to and prefers using. This makes the database something more than just data storage system; we can make sure directly in the database that the data are used as effectively as possible, so that the resulting application is as effective as possible.

2 Selection

Selection of a higher programming language is not always beneficial, and it is therefore necessary to make the right decision and choose whether it is really optimal to work in T-SQL or whether it is better to try and transfer everything into a higher programming language. We don't need to search hard to find an illustrative example – let me mention here the simple example taken from [3] where T-SQL is the clear winner, but as always there are two sides of the same coin and many examples showing the opposite, such as the code in T-SQL for converting the binary numeral format into hexadecimal format of a number which takes about 37 lines while the same code in C# only takes 5 lines.

As an example, here is a sample of the SELECT command in T-SQL and the equivalent in C# is given in [3].

```
SELECT * FROM T_CUSTOMERS WHERE C_CITY = @name
```

```
using(SqlConnection conn = new SqlConnect("context connection=true"))
{
    conn.Open();
    SqlCommand cmd = new SqlCommand("SELECT * FROM T_CUSTOMERS
        WHERE C_CITY = @name", conn);
    SqlDataReader rdr = cmd.ExecuteReader();
    SqlContext.GetPipe().Send(rdr);
}
```

3 User Defined Type CLR

3.1 Why UDT

User defined types allow expanding the system of types with various types such as scalar types the need of which follows from the application logic. Their use is significant in quantities which are given by general rules of a scientific field and will not change. They are not suitable in cases where their frequent changing is expected depending for instance on legislation. That would mean that it would be necessary to redefine the UDT (User Defined Type) after each change.

In user defined types, we can define for instance the rules for classification, aggregation, or our own arithmetic operations. UDTs can be used as columns in database tables, variables and parameters and return values for user defined functions. UDTs implicitly support conversion from varchar, or it is possible to use explicit conversion or write your own conversion function.

3.2 Own UDT Implementation

To create your own UDT you need only a few basic things. Using a template in Visual Studio in an SQL project will make things even easier as mentioned above. I define the required libraries:

```
using System;
using System.Data.SqlTypes;   // INullable
using System.Data.Sql;   // SqlUserDefinedType
using System.IO;   // BinaryReader
```

Then I start defining the parameters defining my UDT. We choose the required features of our UDT such as the name, whether we want to classify them using operands, and whether the format should be Native or UserDefined which gives the programmer more freedom but requires better knowledge because it is necessary to implement the Read and Write methods, which the programmer does not need to do in the Native format [6].

```
[ Serializable ] // attribute which means that all fields must be serializable
[SqlUserDefinedType(Format.UserDefined,Name="Retizek", IsByteOrdered=true,
    MaxByteSize=8000)] // attribute which says that it is a user defined type
    .
```

We thus have the libraries and parameters and we only need to define the required functions based on the application. There are several methods the programmer must definitely implement. If you choose the UserDefined format, you also need to implement the Read and Write methods. After the implementation, you need to set the *Sp_configure'CLR enabled', 1 RECONFIGURE* parameter [6], and then use the Visual Studio to translate, use Deployed, and try your own UDT.

3.3 Required UDT Methods

The Parse method handles transfers from a text value to a data type value. The reverse transfer is handled by the ToString method.

```
[SqlMethod(DataAccess=DataAccessKind.None, SystemDataAccess=
    SystemDataAccessKind.None)]
public static SqlInt32Array Parse(SqlString text)
public override string ToString()
public static SqlInt32Array Null
public bool IsNull
```

Unless it is a Native type, we also have to define

```
public void Read(BinaryReader r)
public void Write(BinaryWriter w)
```

4 Stored Procedure CLR

Stored procedures, as the name suggests, are stored directly in the database together with other objects. This means that you can store in the database not only data, but also a part of or the whole application logic for processing these data. This allows not only easier distribution of applications, but it also contributes to reliability because each application logic is secured and backed up just as the actual data. Stored procedures are usually stored in the database in a precompiled form. We create the **public static** method in the corresponding class. To implement the stored procedure, we have to specify the [SqlProcedure()] attribute [4].

For a method for entering data into the database to become a stored procedure, we have to make minor modifications. First we have to change the method to **static**, then precede it with [SqlProcedure()]. Then it is necessary to modify the connection string as follows: SqlConnection con = **new** SqlConnection("Context␣Connection=true") . Then it is necessary to change the input types of parameters

to CLR data types (SQL Server). The overview of important types conversion can be found in the types table in my other work [8]. After these modifications we can import the method into the database as a stored procedure.

My stored procedures deal with storing data in the database and can be used both for individual entering of data, i.e. adding data to the database, or (as is my expectation) for filling the whole database with data in which the application is supposed to search. It is not expected that the data will change; instead, they will be increasing.

5 User Defined Function CLR

A function is a special case of a stored procedure which contrary to a procedure returns a value in the block in which it was called. This value is calculated or obtained in the function body. We create a **public static** method. After implementing the UDF (User Defined Function) we have to specify the [SqlFunction()]attribute [5].

Basically any .NET function can be used as a T-SQL function. The corresponding class must be **public**, and it is required that the function is **static**. CLR assembly containing the class and function must be catalogued. The rest is the same as in the stored procedure.

I implemented two UDFs, one for searching all documents a returning the field of documents meeting the condition that they contain at least one instance of two searched terms within the specified distance. The second searches the specific document and returns the initial positions of all terms in the document which meet the requirements.

6 Search Speed Compared to Standard Solution

For the test we used a collection of documents called *Medlars Collection*[2]. The collection contains 1,033 English abstracts from medical science, one abstract being one entry in the documents table. Furthermore, there are 8,553 significant terms in the terms table and 57,334 positions of these terms in the position table.

The tests were performed on three computers representing standard computers. In the first test we measured the speed at which the application finds all the documents containing the specified terms in the specified distance. The second test measured the time of finding all the positions of terms in the given document in the specified distance. All times are the average search times, see table 1.

The standard solution does not make use of chains, and it stores each position of the given term in the given document in a new line. The number of lines compared to using chains increased in the used database by approximately 50%. However, the number of lines depends, to a considerable degree, on the documents and the frequency of terms in them.

[2] Available at ftp://ftp.cs.cornell.edu/pub/smart

Table 1. Experimental results

	Finding documents		Searching a document	
	Using CLR [s]	Standard method [s]	Using CLR [s]	Standard method [s]
PC1 Celeron 2.4GHz 256 MB RAM	1.219	9.93	0.016	0.35
PC2 Mobile AMD Athlon XP-M 2400+ 256 MB RAM	1.219	10.1	0.016	0.35
PC3 AMD Sempron 2800+ 512 MB RAM	1.031	9.76	0.015	0.31

7 Conclusion

The main aim of this work was to explore the new functions of MS SQL Server 2005, namely UDF (user defined functions), UDT (user defined types) and stored procedures, and to create a library implementing these functions. All functions created within this work were designed to be usable for searching text based on chains.

This work dealt with using the new functions of MS SQL Server 2005. It showed that it is very appropriate not to use database servers only as data storage systems, but that implementing the required functions directly in the database can be beneficial not only to its speed, security and ease of distribution, but also to further development of the application by other programmers who can base their own applications on this interface.

References

1. Architektura .NET, http://download.microsoft.com/download/8/6/c/86c09926-affc-4e14-bec0-3c45cd989436/Prehled_architektury_NET.pdf, checked March 10, 2007, in Czech
2. Příručka vývojáře pro přechod na platformu Microsoft .NET, http://download.microsoft.com/download/8/6/c/86c09926-affc-4e14-bec0-3c45cd989436/DevelopersGuide2Moving2MicrosoftNET.pdf, checked March 10, 2007, in Czech
3. Praktický úvod do noviniek SQL Serveru 2005 v oblasti relačných databáz, http://download.microsoft.com/download/1/B/0/1B055693-B82B-491F-9FD6-4FE9B17C77DF/Zbornik_Novinky_SQLServer2005.pdf, checked March 10, 2007, in Czech
4. How to: Create and Run a CLR SQL Server Stored Procedure, http://msdn2.microsoft.com/en-us/library/5czye81z(VS.80).aspx, checked March 21, 2007
5. How to: Create and Run a CLR SQL Server User-Defined Function, http://msdn2.microsoft.com/en-us/library/w2kae45k(vs.80).aspx, checked March 21, 2007

6. CLR User-Defined Types, http://msdn2.microsoft.com/en-us/library/ms131120.aspx, checked March 21, 2007
7. Martinovič J.: Metody dotazování pro textové databáze, master thesis, VŠB-TU Ostrava, 2004, in Czech
8. Data Analysis by the Features of MS SQL Server 2005, bachelor thesis, VŠB-TU Ostrava, 2007, in Czech

Refactoring robotic 3D data capturing device to Microsoft Embedded platform

Jan Stankovič[1], Ladislav Kárník[2], David Vala[3], Ondřej Fialík[1]

[1]Institute of Geoinformatics, Faculty of Mining and Geology, VŠB – Technical University of Ostrava, 17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
jan.stankovic@vsb.cz
[2]Department of Robotics, Faculty of Mechanical Engineering, VŠB – Technical University of Ostrava, 17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
ladislav.karnik@vsb.cz
[3]Department of Measurement and Control, Faculty of Electrical Engineering and Computer Science,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
david.vala@vsb.cz

Abstract. Construction parameters of previously developed robotic device for 3D data capturing were strongly limited by heavy computing demands for image capture and processing. In the time of construction we were forced to use only slightly modified ATX platform, that required high amounts of energy and space, as well as weight of suitable power sources. The aim of this project was to rebuild the concept with use of Embedded platform utilizing all it's benefits like small size, low energy but high processing power CPU and optimized operating system compatible with our 3D processing application.

1 Project Motivation

The project can be seen as a preparation for the proposed project of cooperative robots (Grant Agency of Czech Republic 2007/2008) – the aim of the project is to research systems for remote collecting of metric spatial data in dangerous environments based on a group of mobile robotic devices with digital visual, radio-navigation, inertial, and additional sensors, cooperating in real time. The prototype of a device meeting these requirements has already been created by our team, but it has many limitations which can be considerably eliminated by implementing new hardware and Microsoft Embedded technology.

2 Previous robot and Need for Modernization

The work was assigned because of the need to miniaturize the hardware for controlling the robot and processing images. It is also necessary to transfer the software for controlling the robot and processing the 3D image to a MS Embedded PC with MS Windows XP Embedded. This will provide the required increase

in computer performance for working with 3D and a noticeable decrease in power consumption, increased operating time of the device, and decreased total weight of the device.

Fig. 1. Previous robot prototype

The previous robot setup was unsatisfactory; the controlling module was a standard PC which has high power consumption, and when the robot moves through the terrain, errors can occur when writing 3D data from firewire cameras due to the standard hard drive used. It is therefore necessary to implement a new technology, ideally an embedded device which has lower power consumption and which also works more reliably in the terrain. For this purpose, we chose, ordered, and received the Arcom APOLLO Development Kit (http://www.arcom.co.uk/devkit-xp-apollo.htm) for Windows XP Embedded which meets all our required specifications.

Device specifications:

- 1.6 GHz Intel Pentium M CPU
- 512 Mbyte PC2700 DDR memory
- Intel Extreme Graphics 2 dual video controller
- Onboard 10/100baseTx and 1000baseTx Ethernet controllers
- 4 x RS232 serial ports
- 1 x RS485 / RS422 / RS232 port
- Parallel port
- PS/2 mouse / keyboard
- Dual IEEE1394a Firewire port
- USB 2.0 bus driver

Fig. 2. Apollo development kit

- Cardbus controller support for CompactFlash memory and I/O (Wi-Fi, Bluetooth) cards
- IDE drive
- GPIO port driver (for accessing switches and LCD character display on 1U APOLLO ICE)
- TSC1 analogue touchscreen controller (with 10-bit resolution)

3 Procedure

The module was assembled and Windows XP Embedded was created for it as required using Windows XP Embedded Studio.

Hardware configuration of the device was detected using the Target Analyzer programme. The file with these data was entered in Target Designer where the necessary components were chosen automatically. The components required for working with the robot (e.g. remote desktop management) were then added manually. A system image was created from the resulting system configuration. The image was transferred to a flash card and implemented in the APOLLO module.

After launching the system, the MSCN programme was installed (developed by Doc. Dr. Ing. Eduard Sojka – Faculty of Electrical Engineering and Computer Science of the Technical University of Ostrava FEI) which is designed for collecting and processing 3D data. It was also necessary to install the drivers for the DFK 31BF03 firewire cameras (resolution: 1024 x 768 pixel in colour, 30 fps). This constituted the first problem because the operation system did not contain the required dynamic libraries which had to be found and manually transferred into the system.

Another problem occurred after connecting the cameras to the module. The manufacturer states that the module contains two independent firewire ports

Fig. 3. Working on software portation with Apollo

with a capacity of 400Mb/s, but the cameras, each of which has a bitrate of 300 Mb/s, did not provide correct images after connecting. It was therefore necessary to connect a PCI card to the module with another firewire interface. After that everything worked correctly.

In the next stage, we will connect a GPS device, gyroscopes and other sensors to the module, and perform test measuring and monitoring.

Simultaneously with this work, the robot itself is being redesigned, which is currently financed from other very limited sources. Many university departments are participating in this work, such as the Department of Measurement and Control (Faculty of Electrical Engineering and Computer Science), Department of Computer Science (Faculty of Electrical Engineering and Computer Science), the Institute of Geoinformatics (Faculty of Mining and Geology), Department of Robotics (Faculty of mechanical engineering), and the Institute of environmental engineering (Faculty of Mining and Geology).

The original crawler undercarriage has been replaced by wheels. It is necessary to replace the robot's power supply and all the electronic systems. Since the position of the centre of gravity was changed, it was necessary to redesign the construction of the pivot arm carrying the photogrammetric equipment with firewire cameras.

A grant application was submitted to the Czech Science Foundation for complete redesign of the Cooperative Multi-Robot System for obtaining and processing metric spatial data in real time. The proposal was found interesting but it was rejected due to lack of finance and resources.

Fig. 4. Design of the new robot chassis from Pro Engineer

Fig. 5. State of the art – development/testing

Fig. 6. New Robot

4 Equipment Used

- Embedded APOLLO Development Kit from Arcom – purchased from the Microsoft grant
- 2x DFK 31BF03 firewire camera
- Microsoft Virtual PC
- Microsoft Windows XP Embedded + Embedded Studio
- special software MSCN

5 Using the Purchased Equipment and the MS Embedded Platform in Teaching

The Arcon Apollo Development Kit with Windows XP Embedded was used for practical teaching in the Mobile Geoinformation Technologies course. Students were first introduced to the Windows XP Embedded platform through a lecture (by Dr. Stankovic), and they were later familiarized with its particular application in the Apollo module. This demonstration and the following explanation were done by a student (Mr. Fialík). Students had the opportunity to try everything themselves and to objectively evaluate the benefits of the Win XP Embedded platform.

References

1. Rapant P., Stankovič J., Sojka E., Gavlovsky E.: *Visual system for metric 3D data gathering and processing in real time.* Proceedings of the First International Symposium on Geo-information for Disaster Management, March 2005. In: Oosterom,

P. van, Zlatanova, S., Fendel, E. M. (eds.): Geo-information for Disaster Management. Springer-Verlag, Berlin, 2005. 1434 str. ISBN 3-540-24988-5. Str. 609-617

2. Sojka E., Rapant P., Stankovic J., Bohmova D., Kapica R.: *3D data gathering and processing in real time using mobile photogrammetric platform*. In proceedings of GISPLANET 2005, Estoril, Lisboa, Portugal. CDROM ISBN 972-97367-5-8
3. Stankovic J., Brunetti P., Flor R., Furlanello R., Minati R., Neteler M., Lazzari G.: *Going Mobile: open source platforms*, WILMA Workshop, The first line - Wi-Fi / Powerline, 2002, Trento, Itálie. Available at URL http://www.wilmaproject.org/bffmnsl.pdf
4. Kárník, L., Buzek, V. *The specifications and analysis of working environment for designed model of four-wheel chassis*. In: Transactions of the VŠB - Technical University of Ostrava, vol. XLIX, Mechanical Series, No.1, Ostrava: VŠB - TUO, 2003, pp. 85-89; ISBN 80-248-0239-2, ISSN 1210-0471

Monitoring the Position of a Mobile Device

Michal Pešat, Jan Augusta, Elišška Ochodková

Department of Computer Science,
Faculty of Electrical Engineering and Computer Science,
VŠB – Technical University of Ostrava,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
{michal.pesat.st, jan.augusta.st, eliska.ochodkova}@vsb.cz

Abstract. The created applications allow monitoring the position of a mobile device and displaying the position on a map. The applications running on a mobile device process and appropriately record the position of this device, and then send this position via a web service to the server. The position of the device can be displayed on a map both on the mobile device and on the server. We use two technologies to locate the device. Wi-Fi technology is used mainly inside buildings (but also outside buildings) where GPS, the other technology, is often unusable.

1 Monitoring the Position by Means of WiFi

The aim of this work was to design and implement an application for mobile devices with Wi-Fi technology and test one of the available algorithms for determining the position by means of this technology. The created application sends the resulting data to the server where they can be further monitored. The whole project is based on .NET architecture.

The term Wi-Fi describes a wireless network technology based on the IEEE 802.11 standard. This technology was developed for connecting mobile devices inside buildings to the Internet. Later, this standard was also used for connections outside buildings at larger distances.

1.1 Mobile Client Application

The main task of the client for mobile devices (such as Pocket PC, hereinafter referred to as PPC) is to determine the position and send the resulting data to the server for monitoring purposes. Positions are determined by using the method of location marks which are the values of received signals from hotspots. Unlike other methods, this method does not require any additional hardware and it only needs several hotspots and a mobile device with a Wi-Fi card. This method consists of two stages, as shown on Fig. 1:

– The first stage is for describing the relation of the selected location and the signals from surrounding hotspots. The more hotspots there are, the better the quality of the reference point. By measuring several reference points we get the whole network which we will use in the second stage.

– In the locating stage we use the data obtained in the preceding stage and the currently measured values, and these data are used for calculations. In our case, we use the K-nearest neighbours algorithm which gives us an ordered list of differences in distances between the current point and each point in the reference network. This list is passed on for determining the position itself. One of the possibilities for determining the position is to use a centroid. This is a simple method using the arithmetic mean of coordinates, while setting up the number of points in the application from which this mean is calculated, as show on Fig. 2(a). Another possibility is to use the method of an intersection of three circles, where the values of three points with the smallest differences are taken and their coordinates are used as the centres of circles. The values of the differences in distances are used as their radiuses.

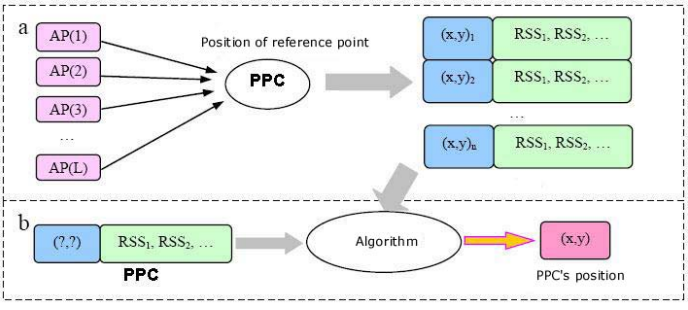


Fig. 1. Two stages of location marks: a) training stage; and b) locating stage

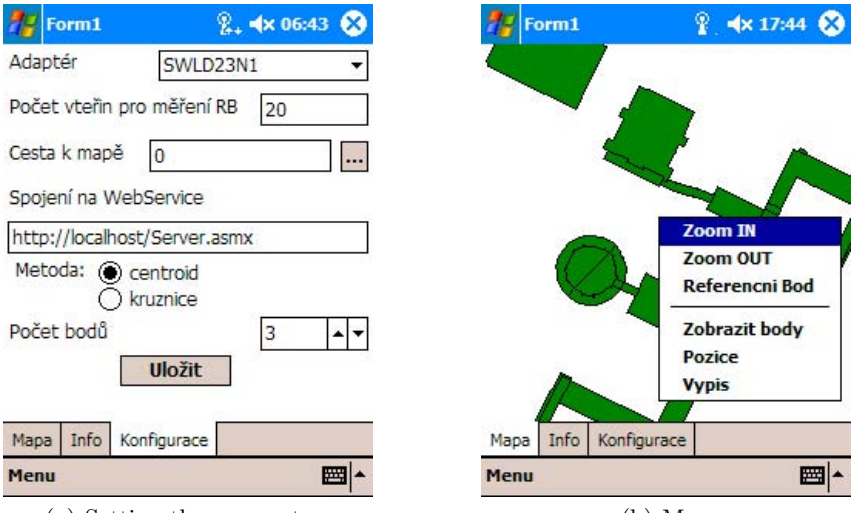
After the calculation we get a single point determining the location. The calculation result is displayed on a map and sent to the server via a web service. If the server is not available, the result is stored in a database in the mobile device and it is sent later. If the repeated sending is not successful, the values in the Pocket PC are deleted. The user can move on the map using the keys and zoom the map in or out using menu items, as shown on Fig. 2(b).

1.2 Server Application

Another part of the whole system is the server application consisting of two parts:

1. The first part is formed by the web services which handle the communication between the mobile device and the server. It includes two methods. One method receives data from the devices that are registered in the system and stores the data in a database. The second method synchronizes the reference points between the server and the mobile device. Synchronization is performed upon starting the mobile client which sends its reference network to the server and when the data are compared they are sent back.

2. The second part of the server is the client which serves for monitoring the positions of individual devices on the map based on the selected data. This client also serves for registering the devices from which data can be stored in the server.



(a) Setting the parameters (b) Map

Fig. 2. Application screenshots (for simplicity GUI is in Czech language)

The whole project is based on .NET 2.0 technology and MS SQL Server 2005 is used for storing data. Due to the limitations of .NET Compact Framework it is necessary to use the Smart Device Framework which allows us to obtain the values of signals from a Wi-Fi card. The SharpMap library is used for displaying the map and working with it.

2 Monitoring the Position by Means of GPS

The aim of this work was to design and implement and application for locating a mobile device by means of GPS and for displaying the location on a map.

The mobile device receives data about its position by means of an attached GPS receiver. The read data (sentences) are processed and divided into required values for determining the position (latitude, longitude, altitude, speed and direction of movement). The processed information is stored in a database in the mobile device where it can also be displayed on a map. When the application is launched, a string is created the task of which is to attempt to access a web service at predefined intervals. If the web service is not available, the string is temporarily paused. After some time the string attempts to contact the web

service again. If the web service is available all data from the mobile device database are sent to the server. After successful sending of data, all data in the database are deleted.

The web server serves as a data storage system. The server records the data received from the mobile device in its database and when requested by an application running on a desktop PC sends the data to the application. The desktop PC application allows us to display the recorded data and plot the data on a map using defined filters.

The whole application is built on .NET technology. The data storage system in the server is Microsoft SQL Server 2005. The data storage system in the Pocket PC is Microsoft SQL Server 2005 Compact Edition. Communication between the Pocket PC and the server, and also communication between the server and the desktop application is provided by a web service. The user interface of the Pocket PC application as well as the desktop application is based on Windows Forms. The simple-to-use SharpMap library is used in web and desktop applications for working with the map.

3 Future Work

We plan to develop the current results to form a shared database for data determined both by means of GPS and by means of Wi-Fi, which will result in a more precise position displayed on the map. We also want to work with better maps. We will also test other algorithms for determining the position by means of Wi-Fi. The aim of testing other algorithms is to obtain as many experimental results as possible for determining the suitability of using Wi-Fi for locating the position of a given mobile device.

Embedded Vehicle Systems and Networks

Michal Krumnikl, Jan Martinovič

Department of Computer Science,
Faculty of Electrical Engineering and Computer Science,
VŠB – Technical University of Ostrava,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
{michal.krumnikl,jan.martinovic}@vsb.cz

Abstract. This paper is dealing with online communication among embedded systems in vehicles and data acquisition from on-board controlling systems and auxiliary sensors. The goal of the conducted research is to develop embedded system capable of interconnecting with standard vehicle systems and communicating via wireless networks. Several wireless technologies were presented and evaluated, considering the usability in traffic environment. These types of network architecture were considered - adhoc, infrastructure and piconet networks. The concept of the system is set to allow the basic functionality without the central node/server, by spreading information across the scattered network of mobile clients.
The proposed system is composed of embedded computers inside the vehicles and remote central storage – the informational system consisting of data storage, geographic systems, statistical and analytical modules. Such models can be used in the future for preventing and forecasting traffic jams, accidents, etc.

1 Introduction

Embedded vehicle systems are subject of the research currently carried out in many laboratories of car manufacturers. The goal of conducted research is to provide safety for driver and expand potential of on-board electronics.

Despite the development of safety features, car accidents kill an estimate 1.2 million people worldwide each year, and injure about forty times this number. This means that every one and half second one man is killed or injured. Projections indicate that these figures will increase by about 65% over the next 20 years unless there is new commitment to prevention [1]. Although many accidents are caused by human factor or unpredictable situations that are difficult to alter (mechanical failures, or bad road conditions), there are already available technical solutions preventing some circumstances. Despite the fact that today's cars are equipped with anti lock braking system (ABS), electronic stability control (ESP) and many other systems enhancing the stability of vehicle, there is still a lack of more sophisticated systems actively avoiding dangerous situations.

Knowing these facts, we are trying to develop an embedded vehicle computer, interconnected with already existing electronic networks in the vehicle. It would

be responsible for gathering data from on-board systems, analysing them and communicating with other vehicles in the range of wireless network. The analyses done by the system are aimed at providing more details concerning the traffic situation and possible threats.

2 Embedded System Overview

Several types of embedded systems were developed during the last year. The first version, using mobile phones was part of M–Net project lead by Roman Szturc. The second version was still based on M–Net protocol, but used it's own hardware consisting of Siemens TC65 modules and embedded processor controlling the communication with vehicle control units. The third version is based on PXA processors and was supported by Microsoft Academy.

The system can be divided into two parts – a computer and wireless interfaces. Each vehicle should be equipped with an embedded computer – a mobile client responsible for data processing and analysis. The solution used during the experiments consisted of these parts : embedded computer based on PXA270, vehicle diagnostics interface, GPS unit, auxiliary sensors (digital accelerometer, camera etc.) and wireless communication devices.

Data transmitted by the mobile clients are either sent to the central server or spread across the network made by vehicles – dynamical scattered networks.

2.1 Embedded Computer

The platform used in the last revision of the embedded system is Toradex Colibri - development board for Intel PXA processors. It provides serial and CAN interfaces as well as connection for LCD display used for communication with the driver.

PXAs are wide spread processors used in the majority of mobile devices, such as mobile phones, PDA etc. ARM core is fully supported by vendors, several development toolsets are available. The controlling software is running on Windows CE, providing support for real-time applications. The application itself uses .NET Framework for compatibility with other platforms supported by the framework.

Control application has modular design, each module is used for a specific purpose: program logic is stored in the core module, data acquired from local sources or remote clients are processed in input modules. The system is designed to use universal message passing between the modules, allowing to separate module specific behaviour from the rest of the application. Communication with the central server and remote clients is held by the communication modules which will be described in the detail in the following section.

2.2 Diagnostics Interface

Embedded computer is connected with the car via OBD-2 interface. This is a standard interface providing access to electronic control units (ECU) of cars and

small trucks. Modern cars usually have several control units linked together, providing a variety of functions ranging from engine control to anti-theft protection.

The specification of OBD-2 was introduced in the United States in 1990s to simplify and standardize the ECU connection with the emission testing hardware. The EOBD (European OBD) is mandatory in Europe by European Directive for all petrol vehicles manufactured since 2001 [2, 3].

The OBD-2 interface provides access to diagnostic trouble codes (DTC) stored in the control units, which have been stored during the failure, as well as real–time data gathered by on-board sensors. Additionally the interface provides functions to erase the DTC codes after the car was fixed.

The interface used in this project is using auxiliary Atmel microcontroller maintaining real time access to vehicle bus. This device was developed according to SAE [3, 4] and ISO [2] standards and should be able to communicate with all EOBD cars equipped with the K-line. Additionally, experimental implementation of Volkswagen proprietary protocol was added, based on experiments with VAG 1552. Unfortunately this protocol is no longer used by Volkswagen.

2.3 Wireless Interfaces

The crucial parts of the system are wireless parts that provide connection to other vehicles on the road. Several types of wireless technologies were used to determine its suitability for transmitting information among vehicles.

Despite the fact that packet radio is obsolete technology used by radio amateurs in the past, it is interesting to compare it with today's wireless networks. The tests were done with radio modem using Bell 202 modulation – 1200 bps. The system was sufficient only for transmitting the GPS position of the car. Similar system is used for Automatic Packet Reporting System (APRS), real time tactical digital communication protocol for exchanging information between stations covering large area. It is used for transmitting reports of the exact location of an object via amateur radio frequencies.

More effective wireless technology, designed for low power consumption, high resistance to interference and low price is Bluetooth. It supports several types of configurations – point-to-point connection and point-to-multipoint. The latter is also referred as a piconet. It supports sharing the channel among all devices in the piconet, which can be composed of seven slaves and one master. The Bluetooth specification [6] provides also scenario of two piconets that have overlapping coverage areas. Each of the piconets are operating in its own frequency hopping channel while any of the device can participate in multiple piconets. In this way information can be transmitted from one piconet to the other. Such networks are called scatternet.

The Bluetooth communication among the vehicles was realized by using Bluetooth Network Encapsulation Protocol (BNEP), encapsulating Ethernet frames to Bluetooth and vice versa. This ensures transparent operation for TCP/IP protocol used in applications. The ability of creating scatternets seemed to be perfect for vehicle networks but appeared to be ineffective for quickly changing

environment. The technology does not have effective mechanisms for fast reconfigurations – nodes entering and leaving the network. Further research will be done in terms of altering Bluetooth for such environment.

The most promising technology appeared to be well known ad-hoc mode of WiFi, which allows to form a wireless network spontaneously. It is able to cope with rapid changes in network topology and provides fast channel for data transmissions. The IEEE 802.11 standard [7] defines this peer-to-peer mode for small networks, but for larger networks its effectiveness rapidly falls due to common clock synchronization problem. Several solutions [8] were introduced to solve this task. The IEEE 802.11 networks are easy for implementation in applications and are supported in the majority of embedded systems.

The last used communication device was GPRS module. Key features, pros and cons of this technology are generally known and are well described in many sources. As an alternative to embedded PXA270 computer, we have developed smaller device based on Siemens TC65 module which acts as a thin client in our network. It is sufficient to send data to the central server, but it does not provide any user interface because of its simplicity.

The solution used at the end of this phase of development is the combination of IEEE 802.11 networking and GPRS connection. The part of application responsible for the connection is selecting the communication channel based on the message priority. Data that are not essential and are not changing very often are send via the slower channel if both connections are available. The main emphasis was put on applicability of the idea of inter-vehicle networks and sharing the information of vehicles among them.

2.4 Data Types

Dealing with multiple sources of various information raises the question of the demands for their processing. The system deals with data that need to be processed in the term of time variance more than actual value, e.g. accelerometer readings reflect the actual acceleration vector, but for motion analysis we need its development in time. On the other hand some information are valid for longer period due to physical laws, e.g. engine coolant temperature cannot change rapidly in a few seconds.

The first type of information we need to process are data from on-board electronic control units. Nowadays we use diagnostics interface running OBD–2 or CAN bus. The former was developed mainly for diagnostics purposes, the latter is used as the main bus for communication among the control units. The communication can be generally described as testing device interrogation. After the initialisation phase, testing device sends requests for services (vehicle identification, diagnostic trouble codes, actual measured values etc.) and the vehicle unit responds with appropriate messages. The other approach that can be used to obtained vehicle data is to connect directly to CAN bus lines in the vehicle. There are usually two independent CAN buses, usually called the "fast" and "slow". The fast CAN bus is used between engine, brakes, steering assistant etc. The slow, also called the comfort, among other systems, such as air conditioning,

car security system, navigation. The embedded computer acts in this situation as the CAN bus message analyser. Intercepted known messages are decoded and sent to the core module.

The second type of data are geographic information obtained from GPS units. Almost all commercial low cost units support NMEA data exchange format. This is a standard that permits marine electronics (including GPS) to send information to computers and other marine equipment. This standard was developed in 1983 and has undergone several revisions [5].

At last we should mention the sensor data and camera images that must be processed as well.

Vehicle data from two independent sources were used for analysis and model calculations. The first source of information were embedded devices as described in this paper, but because of insufficient records from multiple vehicles only a part of analysis can have been done. The other source of data was obtained from the NAM a.s. [9], company cooperating with the university. The first dataset contains the position obtained by GPS, engine telemetry from OBD2 interface, acceleration vector measured by digital accelerometer and images taken by mounted camera. The second dataset contains only the data from GPS.

3 Usability of Acquired Data

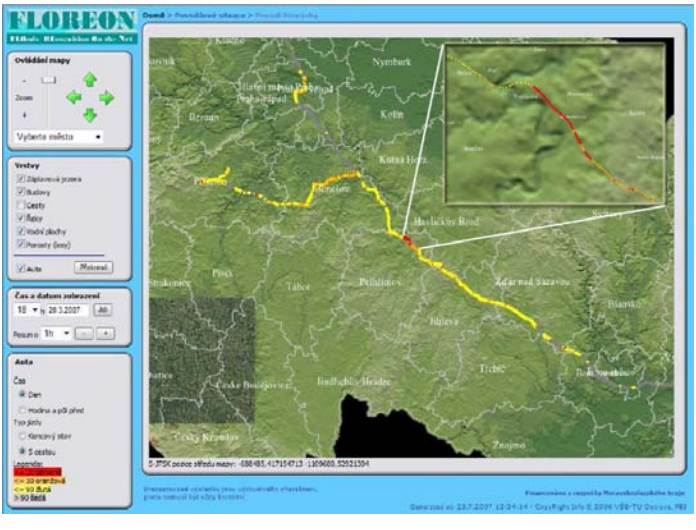


Fig. 1. FLOREON Visualisation of Traffic Jam (GUI is in Czech language)

All incoming data are stored in the spatial database on the central server for further analysis. Geographic information system is used to connect the positions

of cars with proper road segments and points of interests. All statistical values are connected with road segments.

Data produced by our system can be used in various informational systems such as FLOREON [10] (FLOods Recognition on the Net) [1]. In the Figure 1 we can see the FLOREON interface. The map is showing the traffic jam near Havlickuv Brod. The red dots are representing cars moving with avarage speed below 30 kmph.

4 Conclusion and Future Work

The system presented in this chapter is the combination of the Internet technologies, embedded systems and visualisation tools. The purpose is to provide all available traffic information in one place and one common interface. Moreover the system can be used as a part of larger informational system such as FLOREON. In contrast to the current available systems, we can exploit the potential of embedded computers mounted in the vehicles involved in this project. Extra data provided by this system may be found useful for the future research involving the intelligent traffic systems. Research teams of major vehicle manufacturers have already announced several solutions involving embedded computers and mobile networks that will be released in oncoming car models.

References

1. World Health Organization, *World report on road traffic injury prevention*, 2004
2. International Organization for Standardization, *ISO 9141–2*, 1994
3. SAE International, *Surface Vehicle Standard J1979*, 2002
4. SAE International, *Diagnostic Connector J1962*, 2002
5. National Marine Electronics Association, *NMEA 0183 Standard*, 2002
6. Bluetooth SIG *Bluetooth Protocol Specifications, Profile Specifications*, 2003–07
7. IEEE Standards Association, *IEEE 802.11 Standard for Information technology*, 1999
8. Huang H., Lai T., *On the Scalability of IEEE 802.11 Ad Hoc Network*, MOBIHOC, 2002
9. NAM system a.s., Orlova, www.nam.cz
10. FLOREON, floreon.vsb.cz

[1] The main concern of project is to proof that the combination of newest information technologies and results from the disciplines that are somehow connected to the solving of floods simulations and predictions can be very helpful. The list of the disciplines whose results must be combined comprises hydrology, meteorology, integration of prediction models, development of new models, etc. The information system itself is focused on the development of the reliable architecture and is also concerned of the graphical user interface (2D, 3D) that should be able to satisfy all types of users that can interact with the system.

VŠB – Technical Univesity of Ostrava Faculty of Electrical Engineering and Computer Science

1 Introduction

The Faculty of Electrical Engineering and Computer Science, with more than 3000 students, is one of the largest faculties in VŠB-Technical University Ostrava. Since its founding in 1991, the Faculty has been linked to the tradition of teaching electrical engineering and computer science stretching back to 1970, when the first independent study field concentrated in this rapidly developing direction was established at the university.

Today, a feature of the Faculty of Electrical Engineering and Computer Science is the great interest of students from secondary schools in studies focused on the most advanced technology in the computer science field, communications technology, electronics and electrical engineering.

The Faculty has become a significant factor in the restructuralization of the Ostrava region, which today is significantly concentrated on the automotive industry, electrical engineering, and computer technology.

The Faculty provides a quality education corresponding to European standards. It accepts the structured model of studies divided into the levels of Bachelor, Master and Doctor in the form of presentation and combined studies.

The Faculty has been successful in acquiring financing for large and interesting projects in science and research. It has been distinctly oriented toward cooperation in practice and actively seeking contacts with companies offering application for graduates both in the context of the Czech Republic as well as abroad.

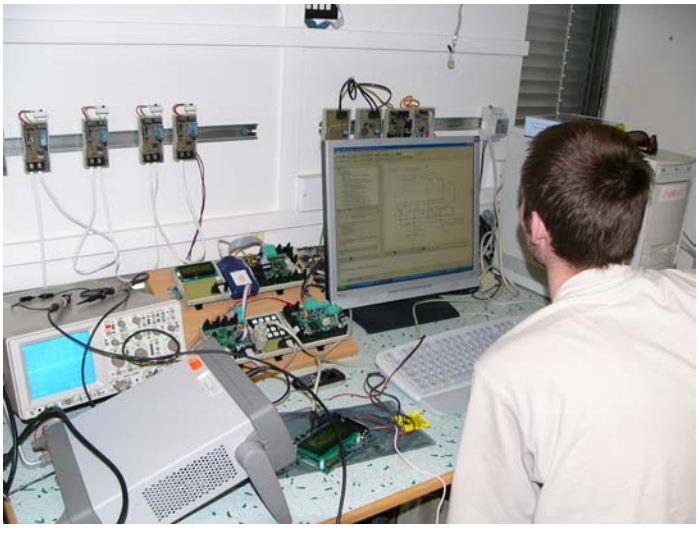
2 Departments

2.1 Department of Measurement and Control

The Department sponsors the following study fields: Measurement and control and Biomedical Technology.

The Department has laboratories available that are equipped with modern measuring and control technology, computer technology and program facilities. One of the most advanced worksites is the Laboratory of Programmable Automated Units and Distribution Systems Management, which is equipped predominantly with products by the company Siemens. Currently, not only does instruction take place, students resolve their Bachelor's and thesis work, and courses are beginning to be arranged for workers in companies and professional seminars for teachers at secondary schools and vocational institutes.

- *Control and information systems, distribution management systems, PLC, industrial networks, built-in control systems, robot control systems.* They use primarily 8 and 16 bit microprocessors by Freescale and Atmel and 32 and 64 bit processors by Freescale, Intel and AMD, industrial communications bus bars CAN and LIN, PC and PLC systems, industrial networks and visualization systems – automated unit programming – PLC Siemens, S7-214, S7-300, Soft PLC , Coros OP5 and OP7, PLC Bernecke & Reiner, Power Panel PP41 (DI/DO, AI/AO, Ethernet, CAN), System 2005, PowerPanel PP21, operator panel PP100, PLC SLC - 500, TC600 and TC 500, Modicon Micro 612 - 03, etc. Industrial networks - ASI, Profibus, Industrial Ethernet, CAN, Epsnet and LonWorks. Visualization systems - WinCC, Intouch 7.0, Factory link 7, Promotic, Control Web. For programming smaller 8 and 16 bit microprocessors, company programming tools are used primarily for language C and C++ , e.g. Codewarrior IDE. For programming higher processors, operational systems are used for work in real time such as QNX, RT Linux and uCLinux.
- *Control technology, analysis and design of control systems, information technology management.* Laboratories are oriented on measurement and control in real time with the help of program environments Matlab & Simulink + Real Time Toolbox. They are furnished with computers and measurement cards MF624 by the company Humusoft. The laboratory tasks for students – Magnetic Levitation, Levitating Balls, Regulation of Elevation of Surfaces in Tanks, Marbles on a Rod, Marbles on a Plane, Classifying Conveyor System.



- *Processing signals, processing multimedia signals.* This uses modern HW and SW environments for processing multimedia signals and data of top companies such as M-Audio, Soundcraft, Behringer, Audiowerk, Adobe, Steinberg, Pinnacle, among others.

- *Electronic measurement, measuring instruments, developmental sensors, EMC* – measurement technology from the companies Agilent, Tektronix, Rohde & Schwarz, among others - digital memory oscilloscopes, spectral and logical analyzers, signal generators etc., special PULSE system used for measuring and analysis of noise and vibration, artificial single-phase and triple-phase LISN network for measuring disruptive signals generated in the feeder network
  With the help of top spectral analysers, together with special measuring antennae and power probes, it is possible to measure and analyse electromagnetic signals in the range from 0 Hz to 6 GHz
- *Measurement and control in biomedicine, processing signals in biomedicine, biomedical instrument technology.* The Biomedical Laboratory, in cooperation with the hospital facilities and clinical worksites, is engaged in projects focusing on medical device technology, sensors, measurement, processing and visualization of biomedical data.
  Trends in resolving projects in laboratories are inquiring into modern technologies and designs, directed primarily into applied research of real problems.

Some other projects – Telemonitoring of life functions, Measuring glycemia of the blind, Reconstruction of 3D image from ultrasonograph, Impact of extremely low temperatures on the functionality of devices and sensors, Processing ultrasonographic data (Reconstruction of 3D image from ultrasonograph), Processing bio-signals with the help of FPGA, Rehabilitation management of drive simulators for traumatology, Software application records of abused children, Measurement of biological signals

Another top worksite of the Department is the Laboratory for Fuel Cells, the working content of which is directed into two areas

- Mobile means using hydrogen as energy carrier
- Stationary units for micro-cogeneration for small buildings and residential houses

Activities of this laboratory appear in the following areas

- motivation of secondary school youth and publicity of alternative sources of energy in the context of the "Napájeni Sluncem" (Powered by the Sun) Program
- motivation of students in the Bachelor's, Master's and Doctoral fields to further study, research and development in the areas of alternative energy sources
- active cooperation with industry (Siemens, Hoerbiger, LabTech, UTC, Ballard, Heliocentris, BMW, CompLet, Remerx, Tufo) and universities (TU Liberec, VŠCHT Praha, VUT Brno, Univ. Bratislava, TU Košice).

Other activities of the Department

- Competition for robots powered by solar energy "Napájeni Sluncem" (Powered by the Sun)

- Robot Soccer – 3rd place at the European Championship and 5th place at the World Championship
- Installation of solar collectors at VŠB - TU Ostrava with maximum output of 20 kW
- HydrogenIX – Steering system for hydrogen vehicle – participation at Shell Eco-Marathon France
- Strategic management with multi-agents
- Architecture for network of built-in systems management

Projects

Interesting projects

- Telemonitoring of life functions;
- Measuring glycemia for the blind;
- Reconstruction of 3D image from ultrasonograph;
- Impact of extremely low temperatures on the operational ability of instruments and sensors;
- Processing ultrasonographic data (Reconstruction of 3D image from ultrasonograph), processing of bio-signals using FPGA;
- Rehabilitation governed by traumatology simulator;
- Software application for records of abused children;
- Measurement of biological signals;
- Processing signals, processing multi-media signals;
- Electronic measuring, measuring instruments, developed sensors, EMC.

Popular activities of the Department

- Solar powered robot competition "Powered by the Sun"
- Robot Soccer – 3rd place at the European Championship and 5th place at the World Championship
- Installation of solar collectors at VŠB - TU Ostrava with maximum output of 20 kW
- HydrogenIX – steering system for hydrogen vehicle – participation in Shell Eco-Marathon France
- Strategic management with multi-agents
- Architecture of networks for embedded management systems

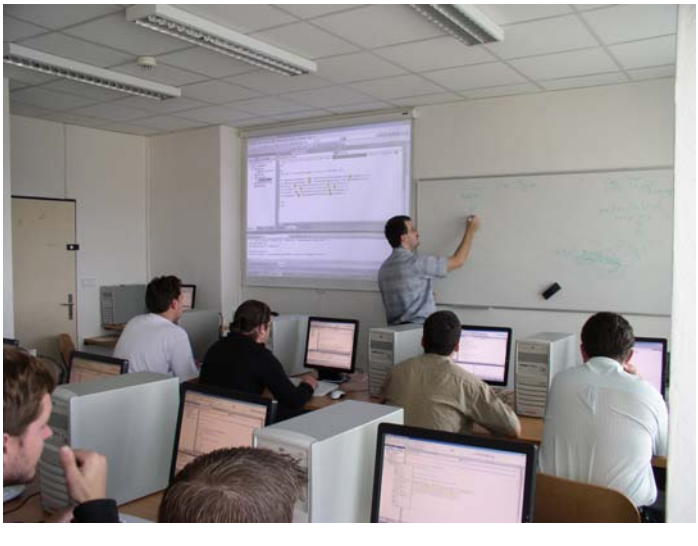
More information about the Department of Measurement and Control — http://www.fei.vsb.cz/kat455

2.2 Department of Computer Science

With the growing proliferation of computers as standard parts of many facilities (even at those which until recently did not anticipate them), demand is also growing for professionals with knowledge of programming, computer networks, hardware and also mathematics.

Instruction. Parties interested in "penetrating the mysteries" of computer science have had help from the Department of Computer Science at FEI since 1990. The effort by members of the Department is to make available to students as much as possible a broader circle of current knowledge in the fields of Computer Science, Computer Technology and Mobile Technology. The basic level of this knowledge is provided in a three-year Bachelor's study and gradually is expanded through more specialised two-year Master's studies through studies at the Doctoral level. Knowledge acquired is both of a theoretical character – from the fields of logic, algorithms and their complexities, as well as of a practical character – from the fields of creating applications of the most diverse types, computer networks, chip designs, digital processing and analysis of images, technical and geographical visualization, as well as, for example, from the fields of project solutions. The study program is offered in presentation or in combined form with support from high quality study materials (e.g. animation) as well as virtual laboratories .

Practical instruction takes place in computer or hardware classrooms, in computer network classrooms or, for example, in the laboratories for visualization, virtual reality and digital image processing. Under its administration, the Department has several servers, which are operated under the MS Windows or Linux operating systems. For study of mobile technology, there is an entire array of mobile equipment available on various platforms. A matter of course is connection of all computers to the university network and through it to the Internet. All classrooms are covered by the WiFi network, during instruction students may use their own computers.



Study abroad. During studies, students may take advantage of the offer of various study programs and stipends and complete part of their studies at some foreign university. Annually, numerous students spend one semester or an entire

year abroad. Some return to defend their Bachelor's or thesis work. Traditional locations for study abroad are Leoben in Austria and Jyväskylä in Finland.

Cooperation in practice. The Department successfully cooperates with a number of companies, in particular in the fields of software development (e.g. information systems and databases), visualization of scientific-technological data, digital processing and analysis of images, development of computer system units, etc. Among the most well-known companies with which the Department works include: TietoEnator, Cisco Systems, Microsoft, Hewlett Packard, Texas Instruments, Sun Microsystems, StoraEnso, Schneider Electric, T-Mobile, Tele Data Systems, NAM Karviná, VÚKROM Kromeríž, ITA Ostrava, Akord software Ostrava, Medium soft, K2atmitec Ostrava, Mareti CZ Ostrava, the Faculty Hospital with Polyclinic in Ostrava, the Faculty Maternity Ward in Brno. Cooperation of a somewhat special character is carried out with the companies Cisco Systems and Microsoft.

Further education. As a result of cooperation with the companies Cisco Systems and Microsoft, the educational academies Regional Cisco Networking Academy and Microsoft IT Academy were established. Their goal is to help students prepare for acquiring industrial certificates in the fields of computer networks and technology of the Microsoft Corporation. Evidence of the quality is the fact that the first of these has already received two awards as Academy of the Year in the Czech Republic.

Competitions. Students may diversity their studies by participating in a number of competitions. This involves, for example, the competition Imagine Cup, a competition in programming under the auspices of the ACM organisation, competitions for Bachelor's and thesis work etc.

Science. The content of work by members of the Department of Computer Science is not only instruction, but the majority of members of the Department are involved in various scientific research projects. This concerns both projects from grant agencies and development funds, as well as projects financed by private subjects. Not only are nearly all students of Doctoral studies linked to these projects, but so are some students in Master's and Bachelor's studies. Joining some of the existing teams gives students the chance to contribute to resolving non-trivial problems, to meet with interesting people, and to gain access to unique technology.

Projects

The content of work for members of the Department of Computer Science is not just teaching, but the majority of members of the Department are also linked to scientific research projects. This involves projects from grant agencies

and development funds, as well as projects financed by private entities. Joined to these projects are nearly all Doctoral candidates as well as some Master and Bachelor students. Joining some of the existing teams gives students the opportunity to contribute to the solution of non-trivial problems, to meet with interesting people, and to gain access to unique technology. Of all the projects, we have selected the following:

- *Floreon* — Modelling, simulation and monitoring of crisis situations caused by unfavourable natural phenomena, part of which is also monitoring the traffic carrying capacity of the transportation infrastructure[1].
- *iParking* — The goal of the project is implementation of a system enabling drivers to find and select available parking places, navigation in parking lots. The system also enables administrators of parking lots to have an overview of the actual status of parking places and supervision over the movement of vehicles in the parking area.
- *Intelligent camera* — The results of projects in the field of hardware for processing images are, in particular, the diverse construction of intelligent cameras (cameras equipped with built-in computer, which use in part standard processors (Intel), as well as special digital signal processors).
- *Robot Soccer and its simulation* — Students also share in the resolutions of the project in the context of resolving their thesis work.
- *Equi-thermal regulation of residential home heating* — with Siemens a Papouch s.r.o. Technology in cooperation with the Department of Electrical Power Engineering
- *Modern methods of synthesizing numerical systems* — design and personification of programmable type FPGA and CPLD numerical circuits
- *On-line arrangement of loans at Czech Post Office counters for ČSOB*
- *E-learning teaching system Barborka* — http://www.unob.cz/
- *Information systems and Internet presentations of companies and offices*
- *Virtual laboratory for computer networks* — The idea of the Virtlab project is to provide remote access to laboratory components for practical instruction in computer networks via the Internet. With the help of WWW, students may reserve laboratory components for certain time intervals and subsequently access them with the help of a standard WWW browser with the support of Java applets. Connection of laboratory components is implemented automatically according to the selection of specific tasks from the set of offered laboratory tasks, or a student may request his/her own topology, http://www.cs.vsb.cz/vl-wiki.

More information about the Department of Computer Science — http://www.fei.vsb.cz/kat456

[1] http://arg.vsb.cz/msk/

VŠB – Technical Univesity of Ostrava
Faculty of Mining and Geology

1 Institute of Geoinformatics

Institute of Geoinformatics has been established at 2002 by transforming of former section GIS of Institute of Economics and Control Systems, *Faculty of Mining and Geology*, VSB - Technical University of Ostrava. This was a result of eight year development starting at the end of 1993. We have started development of new curricula focused on education of specialists on Geographic Information Systems. This Curricula has been accepted at the beginning of 1994 and first group of students started their studies on September 1994.

Today the Institute is running complex system of geoinformatics education, which is composed of wide variety of different modules taught in different forms. The structure of complex system is as follows (full time and part time forms): pregraduate courses, bachelor, masters, postgraduate courses and Ph.D. studies in geoinformatics and geoinformation technologies. Significant part of the study is dedicated to student projects, which are solved in close cooperation or on demand of private and state companies and institutions.

Besides education, our institute is focused to research and development in the sphere of geoinformatics, geoinformation technologies mobile geoinformation technologies and in the fields of their applications. For example we can list metainformation systems, mobile GIT in the sphere of emergency management and civil response , 3D models and 3D data gathering and processing , GPS applications for mapping and mobile means tracking or meteorology, application of (mobile) GIT in intelligent transport systems (ITS).

We have participated and still participate in many national and international projects as like TEMPUS, PHARE, UN-SDI, WELL-GIS, PANEL-GI, E-Content, TRANSCAT, INSPIRE, METROPOLIS, GINIE. A list of research and applied engineering projects is available at our web[1]. Students and employees are taking placements at projects in the framework of LEONARDO and ERASMUS programmes, etc. Our institute is a member of Geographic Information Systems International Group (GISIG).

We should not forget to mention our cooperation with government institutions like Ministry of Labour or Czech Police on solution of geoinformation related tasks and also education of state employees.

Institute of Geoinformatics maintains several laboratories used both for education and research. These are Laboratory of geographic information systems, Laboratory of geoinformation technologies and Laboratory of mobile geoinformation technologies (MGIT). First two laboratories are equipped with GIS work-

[1] http://gis.vsb.cz/Vyzkum/veda_a_vyzkum.htm

stations and specialized software for data processing and analysis. Special remote sensing and photogrammetry tools are available as well.

Laboratory of MGIT is equipped with mobile and telecommunication technologies (PDA's, notebooks, GPS receivers of many types, embedded computers and development kits, wireless accesspoints, routers, GPRS phones and modules, etc.). Wi-Fi coverage available for students is a matter of course.

Our institute is running large annual conference "GIS... Ostrava" from 1994 and annual student conference "GISacek" from 1998.

More information about the Institute of Geoinformatics — http://gis.vsb.cz

Author Index

Augusta, Jan, 39

Fialík, Ondřej, 32

Gacek, Michal, 25

Hurych, David, 15

Janckulík, Dalibor, 5

Kárník, Ladislav, 32
Krejcar, Ondřej, 5
Krumnikl, Michal, 43

Martinovič, Jan, 5, 43

Ochodková, Elišška, 39

Pešat, Michal, 39

Stankovič, Jan, 32

Vala, David, 32
Vašíček, Petr, 5

Walder, Lubomír, 1