

with mipro to knowledge society ~ s miprom u društvo znanja

Lampadem tradere

mipro
2009

32nd international convention on
information and communication
technology, electronics and
microelectronics

32. međunarodni skup MIPRO

European Year of
Creativity and Innovativity

Europska godina
kreativnosti i inovativnosti

PROCEEDINGS
Vol. III.

CTS & CIS

COMPUTERS IN TECHNICAL
SYSTEMS

INTELLIGENT SYSTEMS



GRAND HOTEL ADRIATIC CONVENTION CENTRE AND HOTEL ADMIRAL
KONGRESNI CENTAR GRAND HOTELA ADRIATIC I HOTEL ADMIRAL

mipro

www.mipro.hr ~ mipro@mipro.hr

 **IEEE**

Celebrating 125 Years
of Engineering the Future

Synchronous Localization and Mapping (SLAM) used for Real-Time Autonomous Navigation

Helmut Angerer, Andreas Blassnig, Christopher Kaiser,
Markus Kandler, Florian Prieler, Christian Madritsch

Carinthia University of Applied Sciences
Europastraße 4, 9524 Villach, Austria
Phone: +43 4242 90500 E-mail: c.madritsch@fh-kaernten.at

In contrast to remote controlled vehicles, which need reliable visual and communication links between the operator and the craft, autonomous navigation excludes the operator and enables the vehicle to be used in dangerous or hostile environments. A group of students from the Carinthia University of Applied Sciences developed a full functioning Real-Time Autonomous Navigation prototype using a model-sized car. The autonomous navigation is based on the data acquired by the Light Detection and Ranging (LIDAR) sensor and the incremental encoders. The LIDAR data is processed using the Random Sample Extraction (RANSAC) Algorithm where as the incremental encoder information is processed using the Dead Reckoning Algorithm. An Extended Kalman Filter (EKF) is used to derive the actual position of the vehicle. Finally, an Expert System calculates the navigational commands in Real-Time.

I. INTRODUCTION

II. THE MODEL-SIZED CAR

The base plate of the car is a 30x60 centimeters aluminum plate and acts as the cassis for the whole car. The base plate gets stabilized by aluminum profiles affixed under the plate. The steering mechanism and the drive system are two front suspension units of a Carson 1:8 Specter 4WD buggy combined with two gearboxes of a Phaser 4 by Pro-tech. The usage of two front suspension units enabled the implementation of a four wheel steering and driving system where each wheel can be individually directed and driven. This system allows a very small turn radius which is very advantageously for indoor driving. The car body is divided into three levels the bottom-, the mid- and the top level (see Figure 1). The whole power supply unit is located at the bottom level with the DC/DC converters used to create the supply voltages. The total output power of the power supply unit is about 1kW. To avoid overheating, fans are installed which are controlled by the control system.

The control system for the hardware is situated on the next level. It consists of two printed circuit boards including three microcontrollers.

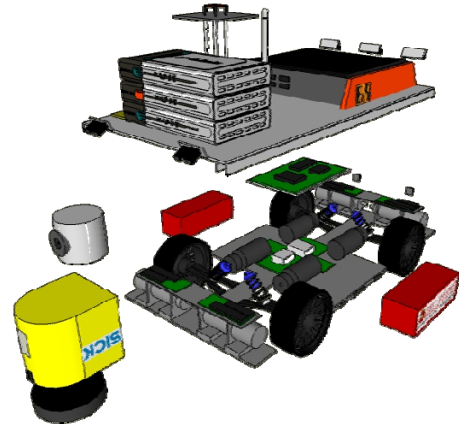


Figure 1: Block Diagram of the model sized car

The first part of the control system, the main board is the central access point for the control PC with the autonomous driving software to the hardware. To provide a preferably flexible interface, Ethernet is used for communications between the main board and the control PC. Furthermore the functional range of the main board contains the monitoring of the hardware and the control of steering and driving. Therefore different driving algorithms, which can easily be accessed by the control PC via Ethernet, have been implemented. These driving algorithms realize geometrical correct steering and the electrical differential. For that reason the steering angle and rotation speed of each wheel gets calculated in real time. The implemented self diagnostic system allows the engineer to locate broken hardware quickly.

The second part of the control system is the sensor board which is responsible for a simple collision avoidance function. If an obstacle gets detected an emergency brake will be initiated and the car stops. Therefore the sensor board uses the emergency and the alert field of the LIDAR sensor and five infrared sensors. The LIDAR is used for detecting obstacles in front of the car and the infrared sensors are necessary for lightening the dead angles of the LIDAR sensor. For detecting steps, the infrared sensors are also used since the LIDAR only delivers a 2D image of the room (see Figure 2).

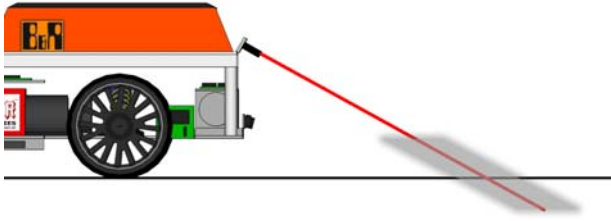


Figure 2: Infrared Sensor detecting steps

Due to the reflective surface of the objects to be detected, common measurement methods like triangulation using line lasers or the use of structured light cannot be used. Here, the measurement method has to ensure a contour resolution of $50\mu\text{m}$, therefore requiring a very precise method as well.

The final level is the mounting base of the control PC (see Figure 3) for the autonomous driving software, the LAN equipment, the compass and the infrared sensors. It also provides further stabilization of car body.



Figure 3: Fully assembled car

III. DEAD RECKONING

Dead Reckoning is based on the odometry data. Odometry is the determination of the position by monitoring the wheels of a vehicle. The dead reckoning position estimation is used to determine the position of the car, needed for the SLAM algorithm explained later.

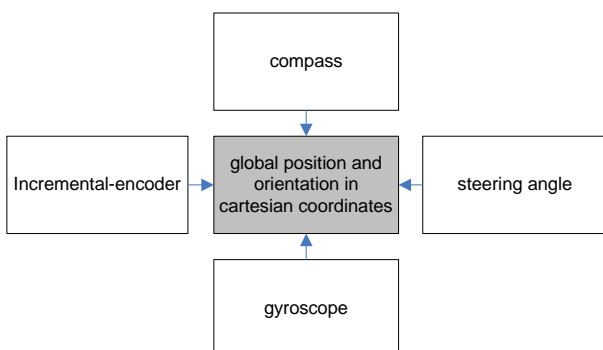


Figure 4: Inputs for the Dead Reckoning

In order to calculate the dead reckoning position, different sensors are used. The sensor system consists of four rotary potentiometers whose values define the steering angle of the four wheels, the incremental-encoders to obtain the driven distance, a compass for an absolute orientation of the vehicle in the room and three gyroscopes as supporting sensors for the measurement of the acceleration force (see Figure 4). Based on the measurement results of the incremental-encoders, and the rotary potentiometers detecting the steering angle, a position can be estimated. Due to the fact that each wheel of the concept car AICC is driven and steered individually, an electronic differential is implemented. This fact has to be considered for the calculations. The estimated position is defective and the error accumulates with increasing distance. In contrast to the outliers, which could occur using a compass, this method provides reliable measurement results concerning the orientation of the vehicle. For this reason, the estimated orientation is only based on the analysis of the incremental-encoders and the rotary potentiometers. The compass orientation gets incorporated, if inside a specified variance, to eliminate outliers. The value of the compass gets discarded if an outlier occurs, which can be caused by electro-magnetic interferences, typical for indoor environments. The estimated position and orientation get sent to the IPC via Ethernet by the main board. An Extended Kalman Filter receives the Dead Reckoning position of the prediction step (see SLAM).

IV. LIDAR AND RANSAC

LIDAR (Light Detection and Ranging) is an optical sensor which, similar to the radar technology, is determining the range to an object by measuring the time delay between the transmission of a pulse and the detection of the reflected signal.

The SICK LMS300 sensor works with an angle-resolution of 0.5 degrees and provides 541 distance-values in centimeters. The scan area is limited to 30 meters with accuracy of $\pm 28\text{ mm}$, depended on the remission value. The range of covered area is sufficient for indoor environment identification (see Figure 5, left part).

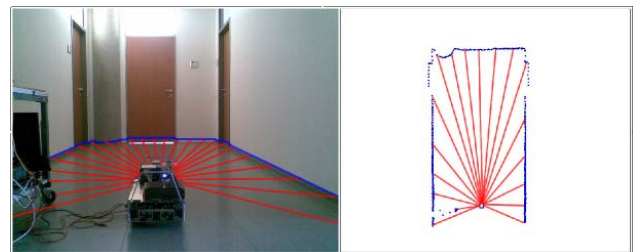


Figure 5: LIDAR scan

The polar coordinates get converted to Cartesian coordinates to visualize the data with a MFC-application see Figure 5, right part. The visualization makes it easier to interpret the abstract sensor data and supports the development of special algorithms. The red lines were drawn afterwards to demonstrate the functional principle of the sensor.

Amongst others, the data points get analysed by the RANSAC algorithm, which is an abbreviation for "RANDOM Sample Consensus". The RANSAC algorithm is suited for the detection of walls inside buildings using the values provided by a laser scanner.

Description of the process:

1. Choose any subset of a dataset
2. Constitute a model with the data points
3. Check how many points support model (inliers)
4. Repeat steps 1 to 3

The algorithm's were empirically adjusted to the application and optimized for certain environments. The consensus sets are models with a defined number of inliers. Optimized by the method of least square errors the consensus sets provide acceptable results (see Figure 6).

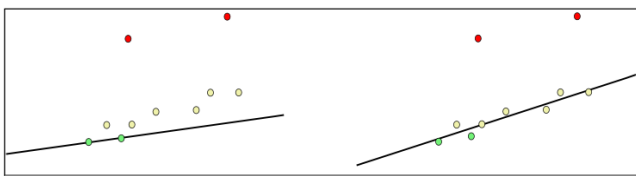


Figure 6: RANSAC simple line calculation

The intersection points of the lines (see Figure 7) provide the basis for the detection of landmarks in the data association process explained in the next section.

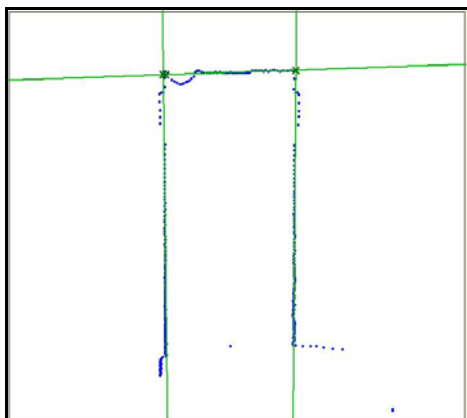


Figure 7: RANSAC intersection calculation

V. SLAM

In autonomous navigation without external information (GPS – positioning, predefined map data), the robot must be able to acquire information about its current position and its surroundings while normal operation. An accurate position is required to create and maintain a map, because the acquired environment data can only be inserted correctly into the map if the position of the observation is known.

Meanwhile the position is determined by comparing the observation with the map data. This leads to the “chicken – egg” problem of needing both, current position and map, at the same time gained from each other, which is commonly called “Synchronous Localization and Mapping” (SLAM).

There was a lot of research done on solving this central problem in robot navigation. Most of the found methods rely on the fusion of information gained from so called landmarks and from a prediction based on the dead reckoning method. Landmarks are static, re-observable patterns extracted from environment data provided by the sensors. Some different algorithms (RANSAC and SPIKES to name a few) can be used to do this trick of reducing the acquired sensor data to a few points of interest. The program flow of a SLAM – process is depicted in Figure 8.

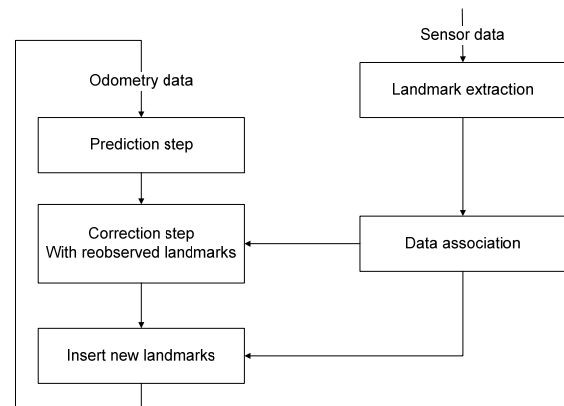


Figure 8: SLAM process diagramm

Basically, position and map, which is in fact a list of known landmarks, are updated in an iterative manner. After each time interval the new position is approximated using dead reckoning (prediction step) and then corrected using the information gained from comparing the extracted landmarks of the actual sensor data with the stored landmark positions from previous observations.

The extracted landmarks are associated with the stored ones using the predicted position to approximate their position on the map. If the distance between an observed and a stored landmark is within a certain threshold, the observed landmark is assumed to be present - observation of the stored landmark and the distance is related to the positioning error of the prediction step (data association). This difference between stored and observed landmark is sent to an adaptive filter algorithm, which weights it according to estimation errors, which are wielded within the filter. Therefore the Extended Kalman Filter is the first in line historically and in the ease of implementation. More complex algorithms like particle filters would offer greater performance and stability.

VI. EXPERT SYSTEM

The software system of AICC was planned and designed as a rule based expert system. The major task of this expert system is to generate the driving commands as efficient and intelligent as possible to arrive at the destination area. The intention to use rules was to get an easy way to add, modify and remove patterns of behavior in specific situations. On the basis of rules and recognized events in the nearby environment, the inference engine decides which driving command has to be executed next.

A programming tool which was designed to develop and to build up rule based expert systems is CLIPS (C Language Integrated Production System). It includes an editable knowledge base and an inference engine. The knowledge-base consists of rules and facts. If in the nearby environment an event takes place, this event is recognized by the sensor system of AICC, processed by various C – functions and finally an appropriate fact is generated for the fact list in CLIPS. These incoming facts of the environment help to take the decisions how to reach the goal.

An example how the car navigates to the desired position is shown in Figures 9 10. In Figure 9 there are three marked spots (red crosses) in the series of measuring points of the laser scan (blue points). These spots indicate passages like doors. The spots get prioritized; the higher their priority, the closer they are to the desired position. Around the spot with the highest priority a corridor (black rectangle) is created. This corridor is check whether it is free of obstacles. In this example the left side of the corridor is free. Finally, three waypoints are generated to drive around the highest prioritized spot on the side on which no obstacles were detected. A chronological sequence how the car is navigating through the passage to the goal is shown in Figure 10, steps 1 to 4.

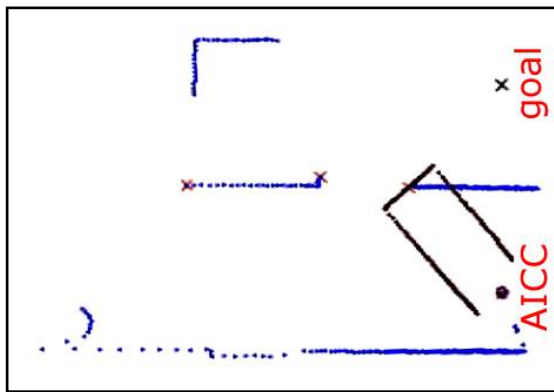


Figure 9: xxx

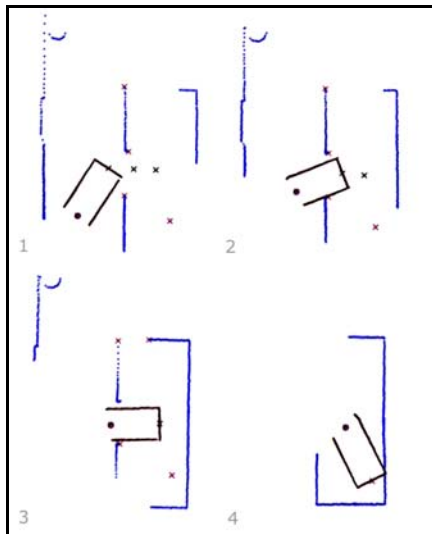


Figure 9: xxx

VII. CONCLUSION

REFERENCES

- [1] Thomas Klinger, *Image Processing with LabVIEW and IMAQ Vision*, Prentice Hall, 2003
- [2] Peter A. Blume, *The LabVIEW Style Book*, Prentice Hall, 2007
- [3] www.fh-kaernten.at/rts
- [4] *NI Vision Concepts Manual*, National Instruments, Document Number 372916F-01, July 2007
- [5] *GRS500 User Guide*, Polar Instruments, Manual Number 186-0803