Informatik aktuell

R. Dillmann J. Beyerer
C. Stiller J. M. Zöllner
T. Gindele (Hrsg.)

Autonome Mobile Systeme 2009

Springer

Informatik aktuell

Herausgeber: W. Brauer
im Auftrag der Gesellschaft für Informatik (GI)

Rüdiger Dillmann Jürgen Beyerer
Christoph Stiller J. Marius Zöllner
Tobias Gindele (Hrsg.)

Autonome Mobile Systeme 2009

21. Fachgespräch
Karlsruhe, 3./4. Dezember 2009

Springer

Herausgeber

Rüdiger Dillmann
Tobias Gindele
Karlsruher Institut für Technologie - KIT
Institut für Anthropomatik
Adenauerring 4
76131 Karlsruhe
dillmann@kit.edu
gindele@kit.edu
http://his.anthropomatik.kit.edu/

Jürgen Beyerer
Karlsruher Institut für Technologie
Lehrstuhl für Interaktive Echtzeitsysteme
Fraunhofer-Institut für Informations- und
Datenverarbeitung IITB
Fraunhoferstr. 1
76131 Karlsruhe
juergen.beyerer@iitb.fraunhofer.de
http://www.ies.uni-karlsruhe.de

Christoph Stiller
Karlsruher Institut für Technologie
Institut für Mess- und Regelungstechnik
Engler-Bunte-Ring 21
76131 Karlsruhe
stiller@mrt.uka.de
http://www.mrt.uni-karlsruhe.de

J. Marius Zöllner
FZI Forschungszentrum Informatik
Technisch kognitive Assistenzsysteme
Haid-und-Neu-Str. 10-14
76131 Karlsruhe
zoellner@fzi.de
http://www.fzi.de

Fachgesprächsbeirat/Komitee

Bibliographische Information der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.ddb.de abrufbar.

CR Subject Classification (1998): I.2.9, I.2.10, I.2.11, I.4.7, I.4.8, J.7

ISSN 1431-472-X
ISBN-13 978-3-642-10283-7 Springer Berlin Heidelberg New York

Springer Berlin Heidelberg New York
Springer ist ein Unternehmen von Springer Science+Business Media

springer.de

Printed in Germany

Satz: Reproduktionsfertige Vorlage vom Autor/Herausgeber
Gedruckt auf säurefreiem Papier SPIN: 12796007 33/3142-543210

Vorwort

Das 21. Fachgespräch „Autonome Mobile Systeme (AMS)“ findet am 3. und 4. Dezember 2009 in Karlsruhe statt und wird vom Institut für Anthropomatik, Lehrstuhl Prof. Rüdiger Dillmann, dem Fraunhofer-Institut für Informations- und Datenverarbeitung IITB, Lehrstuhl Prof. Jürgen Beyerer, dem Institut für Mess- und Regelungstechnik, Lehrstuhl Prof. Christoph Stiller, und dem FZI Forschungszentrum Informatik, Abteilung Technisch kognitive Assistenzsysteme, Dr.-Ing. J. Marius Zöllner, ausgerichtet.

Das Fachgespräch stellt seit über 25 Jahren ein Forum für Wissenschaftler aus Forschung und Industrie dar, die auf dem Gebiet der autonomen und teilautonomen mobilen Roboter forschen und entwickeln. Es stellt besonders für Nachwuchswissenschaftler eine besondere Möglichkeit dar, die Ergebnisse aktueller Forschungsarbeiten auf diesem Gebiet zu diskutieren. Der Dialog zwischen Grundlagenforschern und der Robotikindustrie sorgt für einen bereichsübergreifenden Wissenstransfer und stellt für beide Seiten eine fruchtbare Bereicherung dar. Die ursprünglich auf den deutschsprachigen Raum ausgerichtete Konferenz findet mittlerweile auch im europäischen Raum regen Anklang. Diese Entwicklung wird durch den wachsenden Anteil englischsprachiger Veröffentlichungen begünstigt.

Inhaltlich finden sich dieses Jahr aktuelle Beiträge, die neueste Ergebnisse und Trends aus dem weiten Bereich der autonomen mobilen Robotik präsentieren. Zu den großen Themengebieten zählen Wahrnehmung und Sensortechnik, Regelung und Robotersteuerung, Lokalisierung und Kartierung, Navigation und Systemarchitekturen sowie Anwendungen von autonomen mobilen Systemen. Speziell die Bereiche humanoide Roboter und Laufroboter sowie Flugmaschinen und intelligente Automobile sind durch eine Vielzahl von Beiträgen vertreten.

Insgesamt wurden für die AMS 48 Beiträge eingereicht. Der Fachgesprächsbeirat hat aus diesen Arbeiten 34 ausgewählt, die in einem Vortrag präsentiert werden. Zusätzlich zu den wissenschaftlichen Präsentationen werden in einer begleitenden Ausstellung Exponate aus den Forschungs- und Anwendungsbereichen vorgestellt. Diese sollen den hohen Leistungsstand solcher Systeme demonstrieren und laden zur persönlichen Diskussion mit den Entwicklern ein.

Die Organisatoren der AMS 2009 möchten sich zunächst beim Fachgesprächsbeirat für die Begutachtung und Auswahl der Beiträge bedanken. Unser herzlicher Dank gilt auch den Autoren für die Einreichung der wissenschaftlichen Arbeiten, die wie in den letzten Jahren von hoher inhaltlicher Qualität sind.

Weiterhin sei auch Prof. Dr. Dr. h.c. Brauer, dem Herausgeber der Buchreihe „Informatik aktuell“, sowie dem Springer-Verlag für die erneute Bereitschaft, das Buch herauszugeben, und Frau Glaunsinger für ihre Unterstützung bei der Erstellung des Manuskripts gedankt. Für den unermüdlichen Einsatz möchten wir den beteiligten Mitarbeitern und Studenten sowie dem Sekretariat des Lehrstuhls für Industrielle Anwendungen der Informatik und Mikrosystemtechnik hiermit herzlich danken. Ohne sie wäre diese Veranstaltung nicht möglich gewesen.

Karlsruhe, im September 2009

Die Herausgeber
Rüdiger Dillmann, Jürgen Beyerer,
Christoph Stiller, J. Marius Zöllner

Inhaltsverzeichnis

Perzeption

Regelung

Lokalisation und Kartierung

Navigation

Architekturen und Anwendungen

Advanced Data Logging in RoboCup

Andreas Koch, Adam Berthelot, Bernd Eckstein, Oliver Zweigle,
Kai Häussermann, Uwe-Philipp Käppeler, Andreas Tamke,
Hamid Rajaie and Paul Levi

Institute of Parallel and Distributed Systems,
Universität Stuttgart, 70569 Stuttgart, Germany
`kochas@informatik.uni-stuttgart.de`,
WWW home page: www.informatik.uni-stuttgart.de

Abstract. In this work an advanced data logging approach for the RoboCup domain using an autonomous camera man is presented. It includes approaches from the field of real-time robot message logging, situation based camera control and 3D video visualisation. Based on the implementation of this work the RoboCup team of the IPVS of the University of Stuttgart won the first price in the scientific challenge of the RoboCup World Championship 2009 in Graz.

1 Introduction

In this work an approach for advanced data logging in the RoboCup domain is presented which includes real-time robot message logging linked to video recording and situation based camera control in combination with 3D rendering algorithms. The RoboCup domain has become an important testbed for robotic applications during the last years. The department Image Understanding of the IPVS - University of Stuttgart is participating with its team 1. RFC Stuttgart in the robotic soccer middle-size league. In this league a team of five autonomous robots is playing together. The robots share and merge information in order to show real team play and cooperative actions. As the robots are communicating data over a wireless network there is the possibility to log, visualize and analyze that data during matches for debug purposes. Due to the rapidly growing complexity of the robots software system, which acts in the real world, it becomes more difficult to debug the system in real-time. As a consequence powerful tools for collecting and analyzing data are very important. Those tools have to work in real-time during a match as well as in the post game analysis. In order to improve the general logging capabilities of such systems and visualize the corresponding data in a way that is easy to understand for humans, the following approach has been implemented and tested. We call the system the *autonomous camera man*. The system consists of a camera, a pan-tilt unit (PTU) and a software for controlling both. The software is connected to a message channel which the robots are using during a match. On this channel they communicate all relevant data like ball positions, strategical decisions etc. The program now uses this data to automatically steer the pan-tilt unit where the camera is mounted on. As a consequence a whole match can be filmed autonomously without human interference. Furthermore the system is rendering all the information gained from the robots into the live image of the camera. This allows for interpreting all scenes in a much better way. Moreover it is possible to save the whole video stream and data stream from the robots. Consequently the data and video stream can be replayed and even the kind and amount of information rendered into the video stream can be changed and adapted. Such a system is based on different approaches from autonomous systems, 3D visualization and vision. This paper should give a first overview how such a complex system is working.

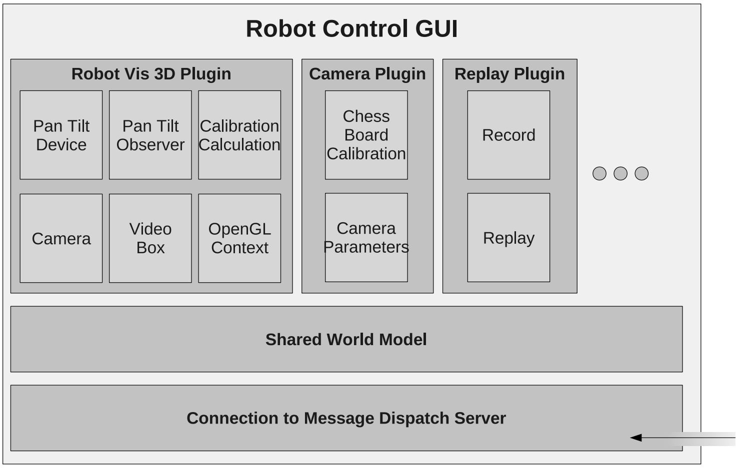


Fig. 1. Architecture of the software.

The algorithms and concepts described in this paper have been adapted to a specific software framework, namely the robot control graphical user interface (RCG). The overall architecture is shown in Figure 1. The software allows to connect to all of the robots directly and gather their shared world model information with read access.

To mix reality with the communicated data an *OpenGL scene graph* was used. Four main components are used to bring information together to one view. Firstly to mention is the *Shared World Model* (Section 3). The second one is the so called *Action Replay Component* that enables the framework to store any information including the video frames in a way that allows for very fast read and write processes. The third part is a *camera framework* (Section 4) that supports a vast range of different camera types. The fourth component is an *observer* (Section 4.1) that keeps track of the movement of the PTU. All gathered information is handed to the rendering framework (Section 4.2).

2 Related Work

As the autonomous camera man includes techniques from a lot of different fields, we want to give a short overview over some articles which mainly influenced the development. The information that is exchanged between the robots to generate a common world model [1] is merged to reduce noise in the measurements as described in [2]. The team behavior of the cooperative robot system is a result of negotiations, based on a multi agent system (MAS) [3] and can be compared to [4]. The behaviour can be configured by XABSL [5] and uses methods like selection equations and interaction nets that are described in [6] and [7]. The *autonomous camera man* enables an analysis of the team behavior by visualizing the information that is exchanged between the robots and which leads to the decisions in the teams strategy. The basic approach for the camera location determination problem is based on the work of Bolles and Fischler [8].

3 Shared World Model

All information for the augmented scene is gathered via the communication of the robots using a message dispatch server. The robot control graphical user interface (RCG) holds a copy of all transfered data. Usually the ball is used as an indicator for the most important view frustrum. But in some cases another object might be of interest, e.g. when the ball is shot with high velocity so that it is impossible for the robots to track it, the goal, the attacker or the keeper are chosen as center for the camera direction. The RCG does not only support messages that have been sent by the robots. The same messaging framework can be used internally, when sending information would mean unnecessary network traffic. The flow of information is shown in Figure 2.

4 Camera Framework and PTU

In order to use the same software framework for the camera man and the robots the whole camera framework of the robots was changed, adapted and extended. Since e.g. two cameras of different types should be used to acquire images, but the current filter graph uses a singleton model for each step, a redesign has become inevitable. The new framework consists of three parts. First of all different cameras can be obtained by factory classes and can be decorated by those. All camera parameters are modeled as properties so that a graphical user interface can be generated automatically at runtime. The second essential part is a new filter library, in which the old filters have been reimplemented in a way that also makes use of properties to enable easy generation of user interfaces and serialization of the filters parameters. Filters can be grouped together in one *VisionPipe Object*, that is attached to a specific camera. The distribution of vision results is done via the signal/slot architecture of Qt(c) [9]. The third part of the framework consists of widgets for an easy way of reusing the above mentioned features such as a property editor to parameterize cameras and vision pipes, windows to

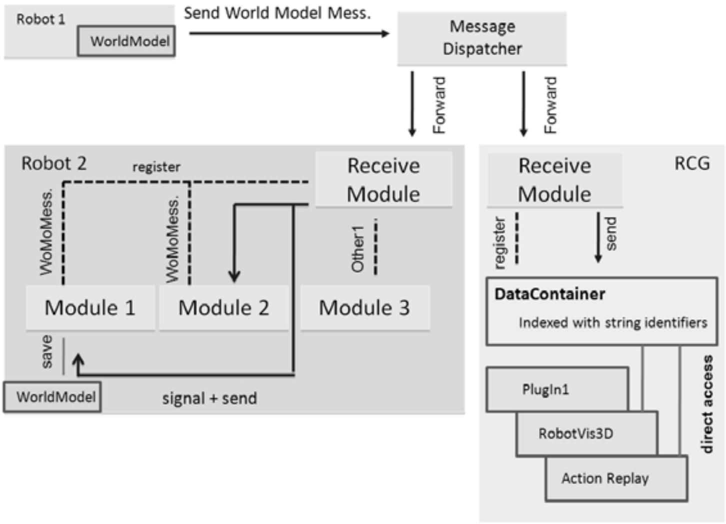


Fig. 2. Flow diagram of information propagated by the shared world model.

display live images from specific cameras and the intermediate results of vision pipes for debugging or parameter adaptation.

4.1 Observer

The PTU is controlled via a serial device, which makes it impossible to steer the device in real-time. The only information available consists of the commands that have been sent to the device. In order to merge the coordinate systems of the video and augmented scene, we have to know precisely the current orientation of the camera. To solve that problem an observer tries to track the angles of the PTU. The update cycle of the observer is depicted in Figure 3. The coordinates of the object of interest are first chosen due to the current state of the shared world model. Those are transformed to the camera frame and the corresponding pan- and tilt angles are calculated. They are sent to both the device and the observer. The visualization process always uses the values of the observer. Correction of the estimated position of the observer is done via *"bang-bang control"*, because the only fact known when reading the values from the device is, whether the observer has already been in position or not. The step of adjustment can be chosen by the time difference between PTU and observer in reaching the wanted position. Experience showed that the acceleration phase of the stepper engines is negligible small. So the observer equation can be obtained as

$$\begin{aligned} \theta_{n+1} &= \theta_n + \operatorname{sgn}(\theta_n - \theta_C)(t_{n+1} - t_n) v_C \delta v_e \ , \\ \delta v_e &= t_f^{obs} - t_f^{pt} \end{aligned} \tag{1}$$

where θ_n is the pan-, resp. tilt-, angle internally calculated in the observer, variables with subscript. C are the values of the latest command that has been

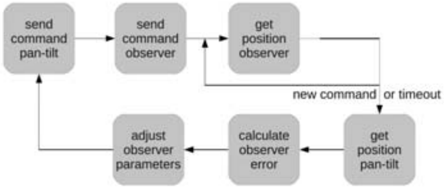


Fig. 3. Update cycle of the observer.

sent to the real PTU. As the communication with the device is done via serial ports, the reading process takes much time, so it is not done in each command cycle.

4.2 Rendering Framework

The grabbed image from the camera is directly mapped on a GL texture, the so called *videobox*, which is displayed as any other ordinary node in the scene graph to enable any 3D mapping of the image. Therefore the field of vision is not limited to the view of the camera. It can be extended to include more information that is exchanged between the robots, which extends the possibilities of analyzing the scene. The GL Thread directly controls the camera device, orders frames and copies them to graphic ram after removing distortions from the image.

5 Camera Pose Framework

In order to overlay the abstract information with the video stream a basic necessity is to find the camera position with respect to the RoboCup game field. The calibration steps have to be done each time the camera is moved. For example, the camera is shifted for a small degree (1 or 2 cm). The calibration is not accurate anymore and has to be done again. The Location Determination Problem is a common problem in image analysis and can be formulated as follows [8] : "Given a set of m control points, whose 3-dimensional coordinates are known in some coordinate frame, and given an image in which some subset of the m control points is visible, determine the location (relative to the coordinate system of the control points) from which the image was obtained". A complete solution to this problem is described by Bolles and Fischler [8]. The problem is simplified into having a tetrahedron, whose three base vertices coordinates are known (3 control points), and where each angle to any pair of the base vertices from top are also known. These angles are computed using the image properties and the location of the control points on this image. Knowing these values, it is possible to solve a biquadratic equation which can give up to four possible solutions. The camera is located near to the field, which makes it impossible to have a sufficient amount of the control points on a single image. Therefore a different way for obtaining the needed angles has to be found. The camera is

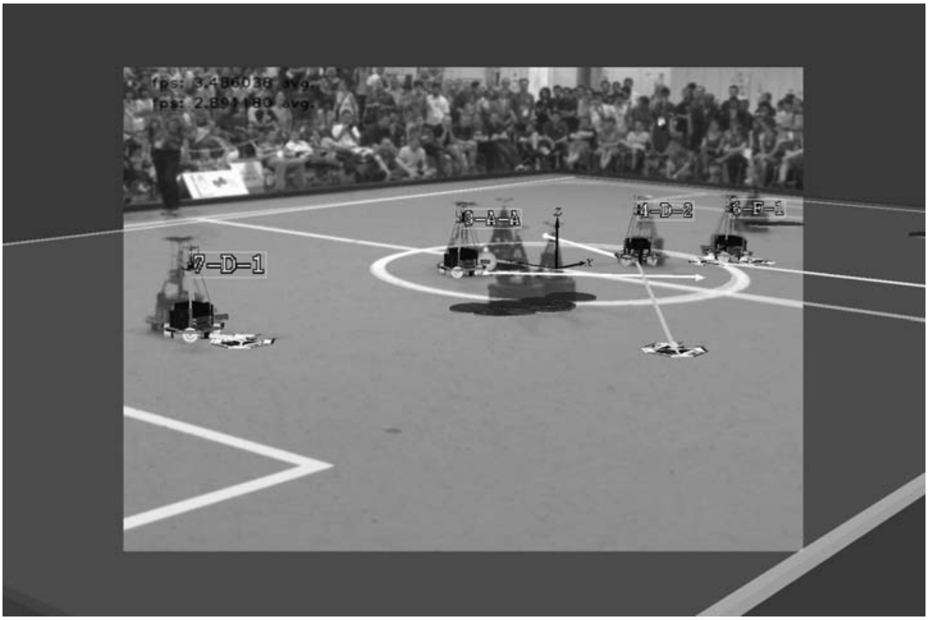
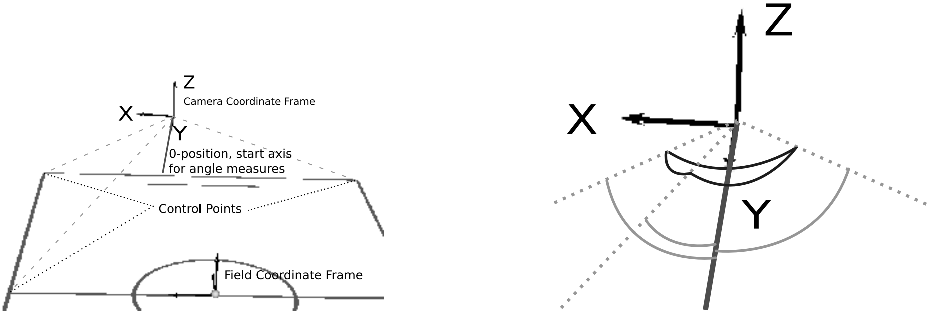


Fig. 4. Screenshot of the videobox in the scene, when the view matrix is fixed to the estimated camera position.

attached to a PTU, whose coordinates frame is not known in the soccer field's frame. We know the $3D$-coordinates of the control points, but it's impossible to determine the necessary angles from a single picture. As the camera is attached to the PTU, the idea is to focus the control points step by step, and to measure the angles between the origin position (also called 0-position) and the position heading to each control point as shown in Figure 5(a) and 5(b). Afterwards the camera location can be determined as described by Bolles and Fischler [8].

5.1 Camera Pose Calculation

In order to compute the angles offset of the camera, a new approach is used which differs from the one described by Bolles and Fischler [8]. This new approach consists in comparing the angles used by the pan-tilt unit to focus the control points from the 0-position as shown in orange in Figure 5(b), and the theoretical angles it should have used if there was no angle offset as shown in blue in Figure 5(b). The computation of the theoretical angles is only a geometrical problem: the location of the camera is known since the previous step (see 5) and the coordinates of the control points are known. Assuming that the the sytem is perfectly aligned to the coordinate frame (i.e. the 0-direction perfectly paralllel to one main axis of the coordinate frame), the angles between the 0-direction and the lines formed by the location point and each control point can be easily



(a) Schematic of the sytem Camera, Field, Control points.

(b) Angles from 0-position up to control points (orange), and angles from center of perspective to any pair of control points(blue)

Fig. 5. Computation of angles for Bolles and Fischler's method.

computed. There are three types of angle offsets for the camera which have to be determined: *pan*, *tilt* and *roll*. The angles previously measured and computed are split for the analysis in the *pan* and *tilt* directions, and we consider these angles as *2*-dimensional points with *pan* as *x*-coordinate and *tilt* as *y*-coordinate, so that the pan offset and tilt offset are equivalent to a translation (*pan* and *tilt* are the only two angles which are driven by the PTU).

The roll offset is equivalent to a rotation with the origin of the frame as center of rotation (the PTU is mounted in a way that the 0-position is also the axis of rotation for *roll*). Then the roll offset has to be compensated for computing the other offsets (which are simple translations) as depicted in Figure 6 and equation (2).

$$\begin{pmatrix} cos(\phi_{roll_{offset}}) & -sin(\phi_{roll_{offset}}) \\ sin(\phi_{roll_{offset}}) & cos(\phi_{roll_{offset}}) \end{pmatrix} \cdot \begin{pmatrix} \theta_p \\ \theta_t \end{pmatrix}_{measured} + \begin{pmatrix} \phi_{pan_{offset}} \\ \phi_{tilt_{offset}} \end{pmatrix} = \begin{pmatrix} \theta_p \\ \theta_t \end{pmatrix}_{theoretical} \tag{2}$$

6 Summary and Future Work

In this work we presented a first introduction into the basic techniques of the autonomous camera man as a powerful data logging tool in the RoboCup environment. Experimental results showed that the approach for logging game data in a RoboCup match makes it much easier for the developers to debug the robot software as the visualization is easy to interpret and much more user friendly than a simple graphical representation of the actions taking place on the field. This approach can not only be used for the RoboCup environemnt but also for a lot of other domains where the systems can provide relevant information about

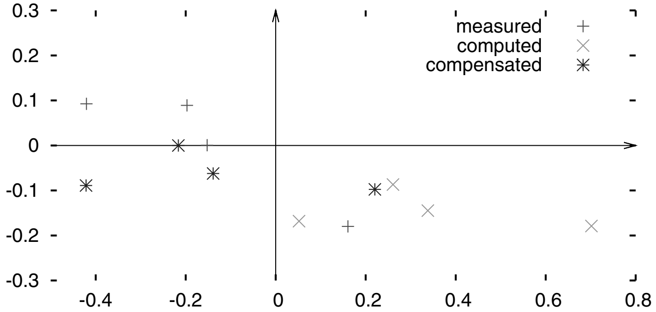


Fig. 6. Example of angles (pan,tilt) in radians. The difference from measured to theoritical angles is a rotation and a translation.

their state. Merged with a camera stream and a 3D visualisation, debugging in all kind of robotic applications becomes much easier and intuitive. As a further step the system should use the panorama camera LadyBug II, which is able to create 360degree images. Transfering the camera location determination problem to such a system will be a new challenge.

References

1. T. Buchheim, G. Kindermann, and R. Lafrenz. A dynamic environment modelling framework for selective attention. In *IJCAI Workshop: Issues in Designing Physical Agents for Dynamic Real-Time Environments: World modeling, planning, learning, and communicating*. IJCAI, 2003.
2. Benkmann et al. Resolving Inconsistencies using Multi-agent Sensor Systems. In *Proceedings of the 8th Conference on Autonomous Robot Systems and Competition: Robotica 08; Portugal*, pages 93–98. Universidade de Aveiro, April 2008.
3. K.M. Muscholl. *Interaktion und Koordination in Multiagentensystemen*. Dissertation, Universität Stuttgart, 2000.
4. M. Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7:83–124, 1997.
5. M. Lötzsch, J. Bach, H.-D. Burkhard, and M. Jüngel. Designing agent behavior with the extensible agent behavior specification language xabsl. In *7th International Workshop on RoboCup 2003*, 2003.
6. Lafrenz et al. Evaluating coupled selection equations for dynamic task assignment using a behavior framework. *Autonome Mobile Systeme 2007*, 1(1), 2007.
7. Zweigle et al. Cooperative agent behavior based on special interaction nets. In *Intelligent Autonomous Systems 9*, 2006.
8. Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.
9. Qt web page http://doc.trolltech.com.

Data Association for Visual Multi-target Tracking Under Splits, Merges and Occlusions

Michael Grinberg, Florian Ohr and Jürgen Beyerer

Universität Karlsruhe (TH), Lehrstuhl für Interaktive Echtzeitsysteme
`grinberg@ies.uni-karlsruhe.de`

Abstract. In this contribution we present an algorithm for visual detection and tracking of multiple extended targets which is capable of coping with merged, split, incomplete and missed detections. We utilize information about the measurements' composition gained through tracking dedicated feature points in the image and in 3D space, which allows us to reconstruct the desired object characteristics from the data even in the case of detection errors due to limited field of view, occlusions and sensor malfunction. The proposed feature-based probabilistic data association approach resolves data association ambiguities in a soft threshold-free decision based not only on target state prediction but also on the existence and observability estimation modeled as two additional Markov chains. This process is assisted by a grid based object representation which offers a higher abstraction level of targets extents and is used for detailed occlusion analysis.

1 Introduction

Most of the vision-based vehicle detection and tracking systems presented in the field of driver assistance systems in the last few decades were focusing on the front-looking applications and particularly on highway driving applications. Many of them make use of various presumptions and restrictions regarding possible objects' motion profiles and their lateral position (e.g. relative to the tracked lane markings), as well as assumptions about symmetrical appearance, shadows etc. In many other applications such as intersection assistance systems or side-looking pre-crash systems, most of those restrictions and assumptions do not apply any more. Many different object orientations have to be taken into account. Combined with a large variety of object types and large region of interest, this makes it extremely challenging to detect and track objects based on their appearance in real time. For the realization of such applications that are capable of a robust and reliable object detection, a generic approach has to be chosen. Object hypotheses are often generated from the range data that are the result of binocular stereo or motion stereo processing. Finding corresponding structures in two video images and reconstructing their depth using knowledge of mutual orientation of both viewpoints, delivers range data, the so-called depth maps. After extracting a depth map, road location is estimated and points belonging to the ground plane are removed. Spatial clustering of the remaining three-dimensional points delivers point clouds which are used as an input in the data association step.

Due to the noisy range estimation process, combined with the well known problems of stereo vision systems such as difficulties of depth estimation in homogeneous image regions and gross depth errors in regions with regular patterns, the results of clustering may vary from frame to frame leading to incomplete, split, merged or missing object measurements as well as to phantom objects. Further problem are incomplete measurements due to partial occlusions and limitations of the field of view (FoV) of the sensors. Visible dimensions of objects entering or leaving FoV or becoming occluded may change rapidly, which combined with a centroid-based object tracking approach leads to strongly biased object position and dynamics estimation.

In this contribution we propose a data association scheme which provides and utilizes information about the affiliation of stably tracked points to the tracked objects and allows for an effective handling of such cases. Using a six-dimensional state estimation of dedicated feature points and point-to-object affiliation probabilities together with knowledge about the spatial relationship between tracked points and the objects' centroids we can perform correct update of the object's position and dynamics in spite of partial occlusions, splits and merges. Image data based track existence probability modeling along with the Integrated Probabilistic Data Association Scheme (JIPDA) [1] and point-to-object affiliation based composition of measurements from point clouds allows for a globally optimal threshold-free probabilistic data association without hard decisions. Additionally, we propose an observability treatment scheme utilizing a grid-based object representation with occupancy and occlusion modeling for each cell. This allows for a dedicated observability and occlusion handling.

2 Overall System Description

The video-based object tracking system used in this work has been built in the course of the EU funded project APROSYS [2]. The goal was detection of imminent side collisions to enable timely activation of novel occupant protection systems [3]. The sensor system under consideration here is a side-looking stereo video camera. Ego-motion estimation and compensation is done using vehicle odometry data. Detection and tracking of the feature points is realized similar to the system proposed in [4]. Up to 3000 feature points are tracked simultaneously in 3-D space using Kalman Filters. Their six-dimensional state vectors $[x, y, z, v_x, v_y, v_z]^T$ are estimated from their image coordinates and displacement (optic flow) as well as stereo depth measurements. The measurement vector is thus $[u, v, d]^T$ with image coordinates (u, v) and feature depth d.

After elimination of the ground points, the remaining points are clustered to the point clouds which give measurements for object tracking. For performing the association between the point clouds and the tracks and for the track state propagation we propose to use the Feature-Based Probabilistic Data Association algorithm (FBPDA) which is described in the following sections. FBPDA