CM-Dragons'01 - Vision-Based Motion Tracking and Heteregenous Robots

Brett Browning, Michael Bowling, James Bruce, Ravi Balasubramanian, and Manuela Veloso

School of Computer Science, Carnegie Mellon University, Pittsburgh PA 15213, USA
{brettb, mhb, jbruce, bravi, mmv}@cs.cmu.edu,
http://www.cs.cmu.edu/~robosoccer/

1 Introduction

At Carnegie Mellon, we have developed several small-size robot teams that have helped us to investigate a variety of aspects of the small-size RoboCup competition. The CM-Dragons'01 is our new team complete with new hardware and sensing and behavior-processing algorithms. Although still in the development phase, a number of modules have been developed that we feel can contribute to the RoboCup community. In this paper, we briefly describe our vision and tracking modules, the new robot hardware and our new communications modules. Our primary interest is on presenting our advances in modeling and prediction using an Extended Kalman-Bucy Filter (EKBF) that tracks the ten robots and the ball through vision. We identify that Kalman-Bucy filters are susceptible to white noise caused by misidentifications. Within CM-Dragons'01, we developed a new approach, Improbability Filtering, that addresses this problem in a computationally efficient yet principled manner.

Throughout this paper, we assume that the reader is familiar with the general structure of small-size robot soccer applications (for further details, please see other articles in this book or for example [6]).

2 Vision

The image processing module consists of the low-level CMVision color segmentation library and high-level vision algorithms that analyze the colored regions reported by CMVision to identify robots, their orientation, and the ball. CMVision is public software available under GPL [2]. The vision algorithms work as follows. CMVision segments pixels using a 3D lookup table, which is a subsampled version of the color space with 8 levels for Y, and 32 levels each for U and V. The YUV table is generated off-line using a GUI editor. Identically segmented pixels are collected into horizontal runs, which in turn are combined into regions. The run unification stage calculates statistics for use by the high-level vision algorithm.

Identifying robots is a two-stage process, namely: (i) suitable colored regions, as determined by the bounding box, are mapped into world coordinates using a

A. Birk, S. Coradeschi, and S. Tadokoro (Eds.): RoboCup 2001, LNAI 2377, pp. 567–570, 2002.

precalculated camera model, and given confidence ranking based on their area and field position; (ii) robot identification and orientation is accomplished by scanning for the closest orientation markers (pink patches) and identity markers (white squares) within a suitable distance. The orientation markers are offset forward on the robot so that the orientation has only one solution.

Running at 60Hz with interlaced 640x240 YUV 4:2:2 images, the whole vision module takes about 40% of the processor on an Athalon 1.3GHz PC. With the loop closed, considering vision to action, the system has a latency of about 100ms corresponding to 6 to 7 fields. Additionally, vision suffers the usual problems of intermittency, noise due to pixelization, and occasional misidentifications. Tracking such data over time to produce robust position and velocity estimates is the job of the tracking software discussed next.

It is our experience, that global vision processing, as permitted in this league, offers difficult challenges. Robot identification and orientation need to be particularyly robust to false positives. We describe next an algorithm to achieve this robustness. The behaviors can only produce successful results if the state of the world can be processed and modeled with reliability and accuracy.

3 Tracking

To track the ten robots and ball, and to make predictions regarding the future state of these objects, we use multiple independent EKBF's combined with Improbability Filters (ImpF). EKBF's are widely used state estimators that provide optimal state tracking for non-linear systems with Gaussian noise components [4], [7]. We used independent EKBF's, based on the assumption that tracking and dynamics noise between objects was negligible, to track the robots and the ball. Due to space considerations we will not show the full equations that describe the tracking module. Instead we refer the reader to [1] and our earlier work [5]. In short, the EKBF's are a straight implementation of the algorithm with a dynamical model of the robots that reflects their kinematic motions, acceleration limits and velocity commands. Similarly the ball EKBF uses a dynamics model that reflects the physics of its motion including the friction of the field and the motion along the inclined walls.

EKBF's provide an optimal estimator of the system state provided the tracked system has only Gaussian noise components. With the appropriate choice of parameters, EKBF's are robust to moderate violations of the Gaussian noise assumption. However, they are not robust in the short term to white noise in the form of misidentifications. In such cases the filter jumps across the field causing havoc to the robot behaviors and ball prediction mechanisms. We devised and implemented a new approach, called Improbability Filtering (ImpF) [1], to reject false-positives thereby overcoming the white noise problem.

The ImpF rejects false-positive, cases where the vision module reports confidently but incorrectly, by determining the likelihood of observing the reported observation given the current model as specified by $\widetilde{x}_k$ and P_k the estimated state and covariance, respectively. The observation likelihood is just the conditional probability density function (pdf) $P\left[z_k|\widetilde{x}_k, P_k\right]$ evaluated for the reported observation. Given the Gaussian model stored by the EKBF we have :

$$C_k = H_k^T P_k H_k + R_k \tag{1}$$

$$P\left[z_k|\widetilde{x}_k, P_k\right] = \frac{1}{\left(2\pi|C_k|\right)^{\frac{n}{2}}} e^{\frac{-1}{2}\left(z_k - H_k\widetilde{x}_k\right)^T C_k \left(z_k - H_k\widetilde{x}_k\right)} \tag{2}$$

All terms are identical to their use in [7] and are described more fully in [1]. We have H_k as the Jacobian of the observation function $z_k' = h\left(\widetilde{x}_k\right)$ with respect to the state variables, z_k is the observation, R_k is the observation noise covariance, $\widetilde{x}_k$ and P_k are the state estimate and covariance, k is the time step, and n is the number of state variables (4 for the ball and 6 for the robots).

Observations that are sufficiently likely, as determined by a threshold set to reject a majority of false-positives, are accepted and incorporated into the EKBF estimates while low likelihood observations are ignored. This approach operates efficiently to remove white noise from the vision output and produces good, robust tracking behavior.

We use the EKBF's for both filtering and prediction. Clearly, the EKBF will filter small Gaussian noise from the vision output and with the ImpF it will also remove false-positives. In addition to filtering, we use the EKBF dynamics equations to create future state estimates and their associated covariances. Essentially, the dynamics update is applied repeatedly corresponding to the number of time steps into the future the estimate is required. The prediction capability allows us to overcome latency issues. In short, we predict ahead the estimated latency and use the predicted values for the robot behaviors and motion control.

4 Hardware

The CM Dragons are a heterogeneous team consisting of two different types of hardware platforms; a fast differential drive robot and an omni-directional robot. Both robots use identical electronics consisting of a processor board with associated power circuitry and a communications board. The differential drive robot, hereafter the diffbot, is capable of high acceleration, enough to break stiction while in motion, with a top speed of 2.5m.$^{-1}$. Given the dimensions of the field and the limited acceleration enforced by wheel slip, the robot can barely approach its top speed in normal play. The omni-directional robot, hereafter omnibot, uses three roller wheels from North American Roller Products in a Y arrangement. The omnibot has lower acceleration capabilities but a higher top speed with maximum figures of 3m.s^{-1} and 3m.s^{-2}, respectively.

The robots use identical electronics; a processor board and communications board. The processor board, the brain of the robot, is based on a TMS320LF2407 16-bit, 30 MIPS DSP from Texas Instruments. The DSP offers both computational power, low power consumption, and lots of functionality via its on-chip peripherals. In particular, the on-chip timer module facilitates generation of dead-band PWM signals to drive the motors with minimal shoot-through current losses. The communications board is based around the 50kbps commercially available lynx^{TM} HP-II modules. These dedicated transmitters and receivers operate in the International Science and Medicine (ISM) band from 900MHz-1GHz with eight selectable operating channels. These devices provide two key advantages over the common Radiometrix module; a) the ISM band is away from the heavily used 418/433MHz band and b) the modules offer multiple channels on a single chip. The reliability of the modules is excellent with our testing reporting packet losses of 1% or less at 38.4kbps where a packet consisted of a start byte, id byte, frame byte three data bytes and a stop byte. Software wise, the robots currently act as dumb slaves and operate velocity PID loops where the set point for the PID loops is obtained from the received RF packets.

5 Conclusions and Future Directions

CM-Dragons'01 represents a new research effort on teams of heterogeneous robots. We briefly presented our EKBF-based tracking system and the novel ImpF algorithm used to overcome white noise caused by misidentifications.

At RoboCup-2001, the different efforts within CM-Dragons'01 were in a preliminary phase of development and their integration was not fully operational. Much work remains to develop the CM-Dragons platform into a truly integrated and competitive multi-robot system. In particular, we are finishing the robot platforms and high-level vision primitives. Finally, our future aims are to step towards developing automated strategies where the team changes playing mode as a function of the overall state of the game.

References

1. Browning, B., Bowling, M., Veloso, M. "Rejecting false-positives using Improbability Filtering and the Kalman Filter", International Conference on Robotics and Automation 2002 (ICRA 2002), submitted.
2. Bruce, J., Balch, T., and Veloso, M. "Fast and Inexpensive Color Image Segmentation for Interactive Robots", Proceedings of IROS-2000, October, Japan, 2000.
3. D'Andrea, R, Kalmar-Nagy, T, Ganguly, P, Babish, M. "The cornell robocup team", Stone, P, Balch, T., Kraetzschmar (Eds), RoboCup-2000: Robot soccer World Cup IV, Springer Verlag, Berlin, 2001, 41–51.
4. Kalman, R. E., Bucy, R. S. "New results in linear filter and prediction theory", Journal of Basic Engineering, March, 1961, 95–108.
5. Kwun, H, Veloso, M "Reactive visual control of multiple non-holonomic robotic agents", Proceedings of the International Conference on Robotics and Automation ICRA-98, Belgium, 1998.
6. Veloso, M., Bowling, M., Achim, S., and Stone, P. "The CMUnited-98 champion small robot team", In Asada, M, and Kitano, H. (eds), RoboCup-98: Robot Soccer World Cup II, Springer Verlag, Berlin, 1999, 77–92.
7. Welch, G, Bishop, G, "An introduction to the Kalman filter", TR 95-041, Technical Report, Department of Computer Science, University of North Carolina, NC, USA, 2001. Available online at *http://www.cs.unc.edu/˜welch*