

Soccer Teleworkbench for Development and Analysis of Robot Soccer

Emad Monier, Ulf Witkowski, and Andry Tanoto
Heinz Nixdorf Institute, System and Circuit Technology
University of Paderborn, Germany
{monier, witkowski, tanoto}@hni.upb.de

Abstract

This paper describes a platform for assisting researchers and/or students to perform and validate experiments using real robots in robot soccer domain. Some important features are: live-video of the experiment, recording video for the experiment, data logging of exchanged messages, robots positioning system, downloading user-specific program to each robot, and post-experiment analysis tool. The platform can be used locally or remotely via the Internet.

1 Introduction

Robot soccer is an attractive domain for researchers and students working in the field of autonomous robots. However developing (coding, testing, and debugging) robots for such domain is a rather complex task, especially for Kheperasot which requires programs to run onboard the robots.

Robot simulator is commonly used during the development since it can help robot developer to analyze and debug the robot program. However, simulation, in most cases, just simplifies the real situation. Due to noises from real sensors and non-linearity in hardware components, programs that run well in simulator may fail in experiments using real robots. Thus, for validation purpose, experiments are still necessary.

However, doing experiments with real robots is a tedious task. Moreover, it is also repetitive. Before experiments, we need to prepare the robots, set up the environment, and so on. During the experiments, we need to record some important events and messages as well as the video of the experiments if necessary. After that, we need to analyze the data generated during the experiments. Although the result of an experiment is satisfying, we need to repeat it several times to make sure that the result is valid and not just by chance. For example, to ensure the developed strategy algorithm working robustly in

varying situations, we need to test the program by putting the robot at some different initial positions and orientations.

For these reasons there is a need for a tool that can complement robot simulator for developing robot programs in robot soccer domain which can ease and assist robot developers in analyzing and debugging experiments using real robot.

In this paper we present a tool called Soccer Teleworkbench which we will refer through out the paper as STWB, which is actually a smaller version of a system called Teleworkbech [1], developed in our research group at Heinz Nixdorf Institute. It has several features that can simplify the execution, analysis, and debugging of robot soccer experiment for Kheperasot. The features include web-based interface, live-video of the experiment, recording video for the experiment, data logging of all messages exchanged such as sensor data, communication message, etc., robots tracker to calculate position and orientation of the robots on the field, downloading user-specific program to each robot, and post-experiment analysis tool to help robot developer to analyze the experiment.

The rest of this paper is organized as follows. In the second section, we present more detailed information of our system. Afterward, an overview of the robot platform we are using will be described. Later on, we present an experiment to demonstrate STWB and other tools developed in our research group. At the end, a short conclusion will conclude this paper.

2 STWB System Description

The proposed system of the STWB, shown in Figure 1, comprises one field used in Kheperasot leagues, equipped with a webcam above it, a wireless communication system using Bluetooth and one computer (PC or Laptop) which performs tasks such as image processing, video broadcasting, and web hosting.

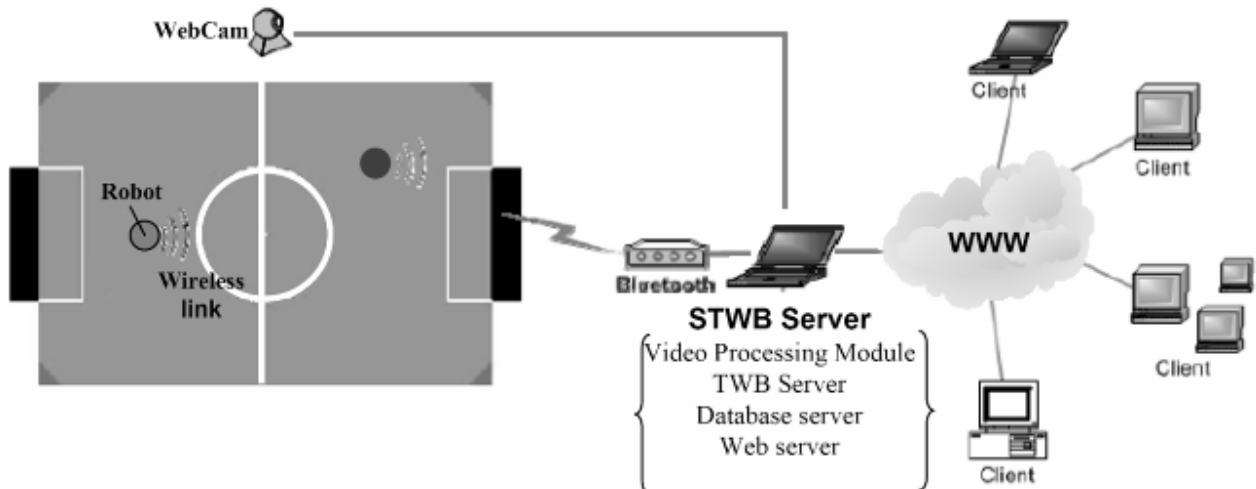


Figure 1: The Soccer Teleworkbench diagram.

The field of the STWB is the same as the fields used in the Kheperasot. Further details of the field as well as the rules of the game can be found in [2]. A webcam is fixed over the field to provide STWB with raw video data needed for image processing and live video. More details of the system will follow in the next subsections.

2.1 STWB server

The STWB server is responsible for downloading programs to robots, routing information between users through WWW server and robots, and logging all events as well as communication messages. The data in the log file will be used as inputs to the post-experiment analysis module.

2.2 Video Processing Module

The video processing module has two roles. First is to provide positions as well as orientations of the robots on the field. The second is to broadcast the video data to the streaming server. As the robot position provider, the video processing module captures the raw frame data from the camera and processes it by detecting objects on the image, and classifying them into two groups: robots and non-robots. To enable classification, we use an identifier which is unique so that it will get recognized easily. The identifier is of circle-form consisting of two colors, each of which covers a half-circle area, and a white box stretching from the center to the edge. This circle is surrounded by a black border for easier identification (See Figure 2). After the position and orientation information is calculated, the robot position extractor will send the information to the STWB server module.

As a video broadcaster, it encodes the raw video data and streams it to the streaming server. In our STWB system, we use open-source software called MP4LIVE which is part of the MPEG4IP project [3] as a broadcaster.

2.3 WWW Server

The WWW server is used to host the web interface for the STWB system. The web interface written in PHP is used to enable remote users to access the STWB. With this interface the users can do the following tasks: Watch the live video of the soccer field, command the STWB, and download the soccer program to a robot.

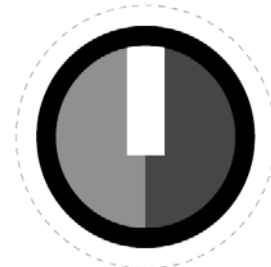


Figure 2: The color mark for Khepera identification.

2.4 Video Streaming Server

In our STWB, the heart of the video streaming server is software called Darwin Streaming Server (DSS), which is the open source version of Quick Time Streaming Server from Apple [4]. One important advantage of using this program is that we can stream hinted QuickTime, MPEG-4, and 3GPP files over the internet Real-Time Transport Protocol (RTP) and Real-Time Streaming Protocol (RTSP) protocols. The first standard is used to

transmit real-time video content and the latter is used to control the stream of real-time video content.

2.5 Post Experiment Analysis and Visualization Tool

The log files that contain all information of the experiment such as robot positions, sensor data, and messages between robots and so on, are used as an input to an experiment post-analysis and visualization tool developed in our group [5]. This tool uses MPEG-4 standard to visualize information acquired during an experiment. On top of the video recorded by the webcam, some computer generated objects representing the information needed to be visualized are overlaid and the appearance of these objects can be controlled by users.

3 Robot Platform

Robot soccer in the Khepera league uses Khepera robot base [6] that is equipped with a camera top module. The KheperaSot rules do not specify the type of the camera, but mostly a linear camera module with 64 pixel with 8 bit gray scale is used [7]. This type of camera limits the object recognition capability due to the missing 2D pixel array and the missing color information. For this reason we have developed a 2D color camera module for the Khepera robot with flexible on-board image processing features. Figure 3 depicts the Khepera robot equipped with the new camera top module and additional image processing module.

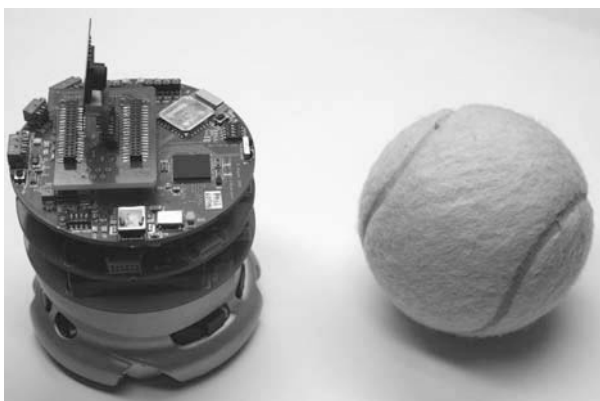


Figure 3: Khepera robot equipped with FPGA module and 2D color camera top turret

The camera is a miniature 2D color CMOS camera from Transchip, model TC5740 with VGA resolution. To control the camera we use an FPGA (field programmable gate array) module that additionally processes captured images. Results of the image processing are symbolic

data like position of the ball in the image plane, size of the ball, coordinates (left and right corner) of the goal (if visible), and coordinates of the opponent. In this way, the microcontroller of the robot's base module will not be burdened with image processing tasks. Thus the processing power can be used for realizing soccer strategy, planning, positioning, and drive control.

Besides the image processing capability, the FPGA module offers important features which are very useful during the development phase of the image processing and soccer algorithms. The module is equipped with a Bluetooth transceiver and USB 2.0 port. Via the Bluetooth link, status data of the robot can be transferred wirelessly for analyzing the robot's behavior. This feature is used in the soccer teleworkbench environment to record all actions of the robot along with the video captured by the camera over the pitch. Examples are presented in the experiment section. Via the USB port the programmer is able to see on a computer screen what the robot *sees*. In addition, all (intermediate) results of the image processing performed by the FPGA can be displayed as a real time video. For this, the robot is connected via a very flexible USB cable to a computer. Although the cable is limiting the area of operation of the robot, the video streaming is a very helpful feature during the development and calibration phase. For a soccer match, the USB cable will not be used. Features of the FPGA module (Figure 4) are:

- FPGA Virtex-E XCV300E; flexible programming by user via VHDL; supports dyn. reconfiguration
- Cypress EZ-USB FX2 USB 2.0 micro-controller
- Multi-chip memory package with 128 MBit Flash and 32 MBit SRAM
- Programmable clock (up to 60 MHz)
- ADC and DAC for sound in/output
- Communication links: Bluetooth (point-to-point, piconet scatternet), USB 2.0, RS232, I2C, IrDA

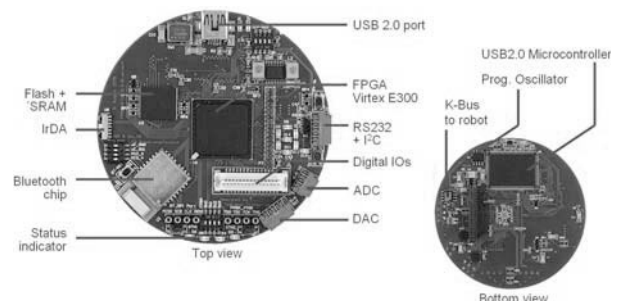


Figure 4: FPGA module for image processing tasks

4 Experiments

In this section, some features of the soccer teleworkbench are presented. The main aim here is not to demonstrate the optimal soccer algorithm, but the soccer teleworkbench facilities for analyzing and optimizing robot's behavior.



Figure 5: Robot close to the goal with ball ahead

We have to distinguish three different views to and from the robots on the pitch. The one a human usually have is shown in Figure 5. This is not the ideal view to analyze the strategy of a soccer robot because of 3-dimensional perspective projection of the scene. Distances, speeds and trajectories of the robots can not be registered correctly and, for example, the ball may be undetectable when it is masked by a robot. Thus, for match analysis the over-pitch cam view in Figure 6 is used to record a match. The robot soccer programmer gets this view from the soccer teleworkbench during a match, and the video recorded from this view is the basis for the post processing step to analyze a match. The third view is depicted in Figure 7. It shows the pitch from the robot's perspective through the camera on top of the robot. Usually, this view is not available during a match, but in the development phase the programmer may use this view to optimize image processing steps and robot's soccer strategy.

A successful robot soccer match is, on one side, based on good perception of the environment to get the current match situation. On the other side, the match strategy is important to be faster than the opponent, for example, and to shoot goals. Both steps are supported by our development environment. The FPGA module with the USB link provides images of the robot's cam to the developer in real time.

An exemplary captured scene is depicted in Figure 7; it can be displayed in color mode or in gray scale by interactive selection in the display window. In order to analyze and optimize the image processing steps the user can interactively choose filtered images to be displayed.

Depending on the user input, selected streams are transferred to the PC via the USB link without complicated set-up. Additionally, the user is also able to adjust filter parameter or thresholds online to optimize filter algorithms.

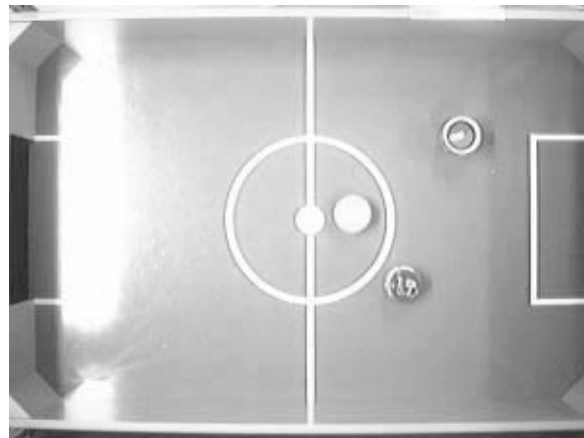


Figure 6: Cam view on the pitch. The color identifier is attached to the robot in the upper right area.

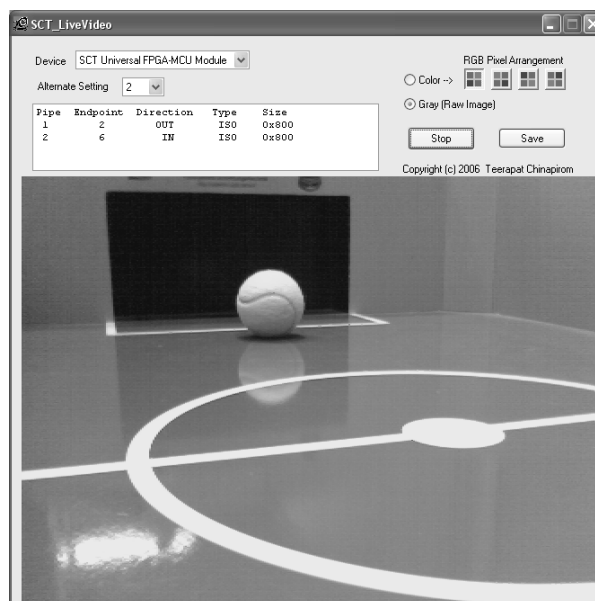


Figure 7: Scene captured by robot's camera

4.1 Development of Image Processing Algorithm

For image analysis and object detection we have implemented several filter algorithms in the FPGA hardware. One task for example is the detection of the yellow tennis ball.

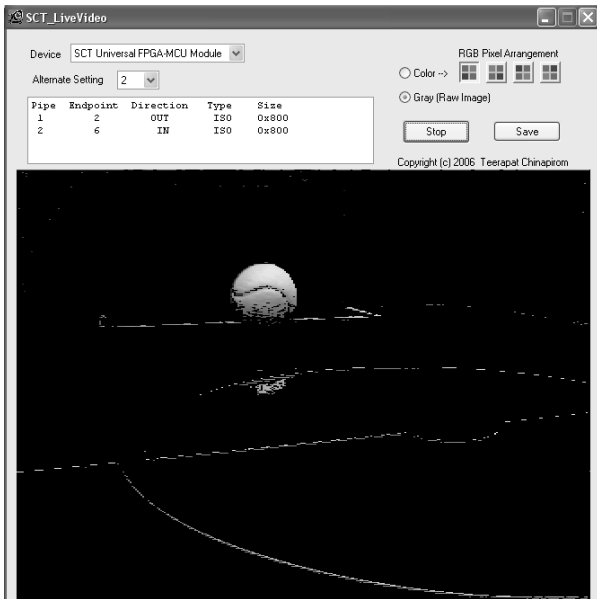


Figure 8: Image after applying yellow color filter to extract ball (source: image in fig. xx)

Therefore, the RGB raw image from the image sensor is converted into the HSV color space. Afterwards a segment of the color space cone is cut by defining minimum and maximum thresholds for the three parameters in the color space. An exemplary result of this processing is presented in Figure 8.

All pixels outside the decided cone segment are displayed in black; just the ball and some additional noise are left. By applying a mosaic filter the noise from the pitch lines can be suppressed and the yellow ball can be detected reliably [8]. All important filter parameters can be easily adapted via the graphical user interface to optimize the image processing to ensure robust object detection. The optimized parameters are stored in the non volatile FLASH memory of the FPGA module.

In order to analyze the robot's behavior and the soccer strategy in different situations a more global view of the robots on the pitch is necessary. For this analysis the cam view as depicted in Figure 6 is used within the soccer teleworkbench environment. From this top view all robots are visible and the ball can be detected for all possible soccer situations. Thus, this view is used for recording matches. Additionally, extracted positions of the robots as well as internal data of the robots are recorded. As described in Sub-Section 2.5, the video file and the additional data are encoded to an MPEG-4 video file. With a special MPEG-4 video player such as Osmo4 [9], the user can interactively select information to be displayed during playback, e.g. the trajectories of both robots can be switched on or off, internal states of the robots can be displayed together with sensor data or communication data – if used. An example image is shown in Figure 9. The video comprises the whole screen. The middle part shows the video with overlaid path of the robot (can be switched on and off arbitrarily

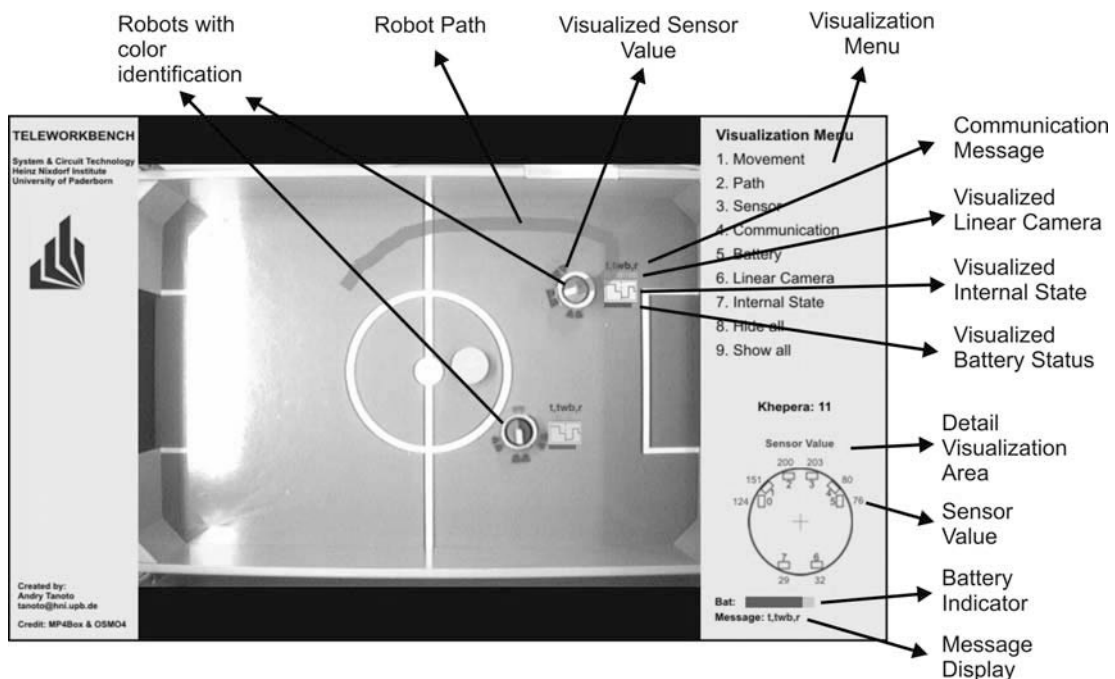


Figure 9: Post experiment analysis tool with superimposed robot's path.

by the user during playback). The right part of the window contains interactive buttons to switch on various display options like visualization of symbolic view of the robot, path of the robot, sensor data, communication data, battery status, etc. By clicking a robot in the middle part of the video, more detailed information of the robot, e.g. robot's infrared sensors, is displayed in the lower right symbolic robot view of the window. The playback can be done continuously, or step wise for detailed analysis of the match.

5 Conclusion

We have presented a test bed that can be used for debugging and analyzing the Kheperasot robot soccer experiments. It can also be used as a general purpose tool for robotic experiments. The components of the system which include webcam, kheperasot pitch, video processing and positioning module, STWB server, video streaming server, web interface have been introduced. By using the soccer teleworkbench environment, a match strategy can be developed faster and failures in robot's behavior can be analyzed precisely. Furthermore, the design cycle becomes faster because new algorithms can be downloaded wirelessly via the Bluetooth link onto the robot. The user can reprogram the robot's base as well as the FPGA by downloading a new bitstream to the FPGA module. The access to the robots is done via a web interface, thus the soccer teleworkbench can be used locally or remotely. The FPGA hardware of the Khepera robot allows detailed analysis of the image processing and interactive optimization of filter operations. For this, a video data stream is transferred via a USB 2.0 link to a Laptop or PC. This feature is available during the development phase because in a real match all cables have to be removed. All in all, the soccer teleworkbench can be used in the development phase and during match for recording purpose.

Acknowledgements

Part of this work is supported by the International Graduate School "Dynamic Intelligent Systems", Paderborn. We would like to thank the KheperaSot robot soccer project group with René Griessl, Christian Renneke, Arne Hilgenstein, and Marx Karl for programming and experiments.

References

- [1] Tanoto, Andry; Witkowski, Ulf; Rückert, Ulrich: Teleworkbench: A Teleoperated Platform for Multi-Robot Experiments. In: Proceedings of the 3rd International Symposium on Autonomous Minirobots for Research and Edutainment

- (AMiRE 2005), Awara-Spa, Fukui, JAPAN, 20 - 22 Sep 2005.
- [2] <http://www.fira.net/soccer/kheperasot/KheperaSot.pdf>
- [3] MPEG4IP. Mpeg4ip project - open source, open standards, open streaming. Available: <http://mpeg4ip.sourceforge.net>
- [4] Apple Computer Inc. Darwin Streaming Server - Administrator's Guide. Available: <http://developer.apple.com/darwin/projects/streaming/>.
- [5] Andry Tanoto, Jai Lie Du, Tim Kaulmann and Ulf Witkowski. MPEG-4-based interactive visualization as an analysis tool for experiments in robotics. In the 2006 Conf. on Modeling, Simulation and Visualization Methods, Las Vegas, USA, June, 2006.
- [6] K-Team S.A., Mobile Minirobots Khepera II, Lausanne, Switzerland, www.k-team.com, 2005.
- [7] Narongdech Keeratipranon and Joaquin Sitte, Beginners Guide to Khepera Robot Soccer, <http://www.fira.net/soccer/kheperasot/Guid.pdf>, 2003.
- [8] Kaulmann, Tim; Strümkamm, Marc; Witkowski, Ulf: FPGA-based Object Detection in Robot Soccer Application. In: Proceedings of the 3rd International Symposium on Autonomous Minirobots for Research and Edutainment (AMiRE 2005), Awara-Spa, Fukui, JAPAN, 20. - 22. Sep. 2005.
- [9] GPAC, GPAC project on advanced content. Available: <http://gpac.sourceforge.net/index.php>