robOTTO RoboCup Festo Simulation League Team Description - 2013

Mustapha Abdallah, Magnus Hanses, Stefan Wilske, Nils Harder, Erik Sommer, Alexander Ratai, Christian Deppe, Christoph Lerez, Carl Luecke, Georg Jaeger, Jakob Katz, Julia Schwarzer, Juliane Hoebel, Kai Seidensticker, Kristin Schroeder, Martin Roedel, Stefan Kannewurf, and Thomas Tessmer

Otto-von-Guericke University Magdeburg, Germany
tavitac@hotmail.com, Lloyd0409@hotmail.com, stefan.wilske@gmx.de, otto_der@yahoo.de, erik.sommer@st.ovgu.de, alexander.ratai@googlemail.com, cdeppe@st.ovgu.de, chlerez@googlemail.com, luecke.carl@gmail.com, gjaeger@st.ovgu.de, jakob.katz@st.ovgu.de, jkschwarzer@aol.com, JulianeHoebel@gmx.de, Kai.Seidensticker@st.ovgu.de, kristin-ina.schroeder@gmx.de, martin.roedel@st.ovgu.de, Stefan.Kannewurf@st.ovgu.de, thomas.tessmer@gmx.net

Abstract. In this paper three main techniques added to robOTTO team for 2013 will be presented. The first technique is the spline motion control mechanism that provides smooth, fast and efficient motion for the robots helping them reach their destinations faster without any stopping points on the way. The second technique is the two-level motion planning technique that caches motion paths for instant retrieval and still able to find motion paths in real-time when needed. Finally is the image analysis technique where we developed a fast 1-path algorithm for both segmentation and detection along with hierarchal multi-resolution in order to have control over the run-time of the algorithm.

1 Introduction

robOTTO was founded in 2010. In 2010 the robOTTO Festo Logistics League team won the **2nd** place in the RoboCup finals in Singapore. In 2011 the robOTTO Festo Logistics League team won the **2nd** place in the RoboCup German Open in Magdeburg. In 2012 robOTTO Festo Logistics League and Soccer Simulation 2D teams placed 4th and 8th respectively in the RoboCup finals in Mexico City.

As a part of the continuous applied research efforts robOTTO has worked on three new techniques for RoboCup 2013. The new spline motion control technique is better than the older in speed, smoothness and precision regardless of how difficult and edgy the path is. Instead of using straight lines to move from one path point to another, the whole path is converted in a continuous smooth spline with real-time error monitoring and correction. A new motion planning technique was developed in a way that it caches motion paths in the memory while keeping the ability to calculate paths in real-time when cached paths are blocked. The new motion planning technique also uses path points in a way that help maximizing the utilization of the whole field while keeping the robots from colliding with each other. For image analysis a new technique was developed to do only one path on the image in which it segments the image and classifies these segments into the objects of interest. Image analysis is also hierarchal multi-resolution in order to have control over the run-time of it. Since the robot's original gripper imposes various restrictions on its usability, a new gripper was designed in order to resolve those restrictions. The new gripper is smaller, fits better for the puck, doesn't let the puck go away as the robot rotates and has multiple sensor mounting points. On the hardware level, one of the robots was upgraded by mounting a powerful computer to it and replacing the current batteries with better ones. That upgrade aims at simulating the power specs of the new Robotino.

The remainder of this paper is organized as follows. Section 2 discusses our efforts in the context of similar scientific efforts. Section 3 explains the spline motion control technique. Section 4 demonstrates the caching motion planning technique. Section 5 explains the 1-path image analysis technique. Section 6 introduces the new gripper. Section 7 introduces the hardware enhancements applied. Finally, section 8 concludes the paper with a summary of possible future work to enhance the current performance.

2 Related Work

The spline motion control algorithm is mainly based on the Navigation, Guidance and Control approach [1]. For the special conditions of the Festo Logisitcs League and the limitations of the robot (Robotino) the motion control algorithm is designed to suit them. Our waypoints generation algorithm inspired some ideas from on-line waypoints generation algorithms [5] [6] while it's different in terms of being offline and focusing on space utilization and spline compliance than finding waypoints for a real-time destination, with prior knowledge of the environment rather than depending on sensory real-time data. The presented image analysis algorithm combines labeling and segmentation methodologies [7] with hierarchal multi-resolution [8]. Some - up to the best of our knowledge genuine - ideas are introduced to image analysis like feature enabling/locking and the way levels are defined for hierarchal multi-resolution where the main focus was efficiency (benchmarking showed the ability to get the same results $\approx$100 times faster with hierarchal multi-resolution).

3 Spline Motion Control

Using natural cubic splines, a continuous path reconstructing a set of discrete waypoints can be generated. Every path between two waypoints is described by a 3rd order polynomial. Those polynomials are connected by boundary and matching conditions. Main advantage of natural cubic spline is its minimal condition against curvature. This makes it the optimal path concerning centrifugal forces. In order to control the robot along the path, it is essential to determine its desired trajectory. Therefor a velocity profile is applied. Caused by its omnidirectional drive [3], the robot is very vulnerable against slippage. To reduce slippage we use a sinoidal velocity profile as shown in figure [1].

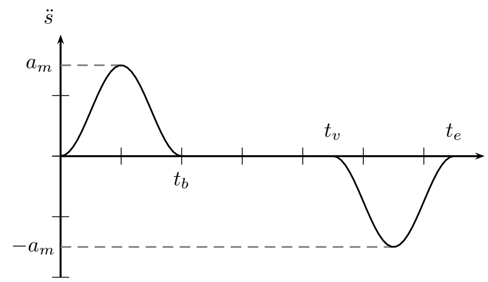

Fig. 1. Sinoidal velocity profile

As shown in figure [1] the acceleration behaves smooth. That means the jerk is limited which makes it very gentle for mechanical components and transmission.

The actual Motion Control is based on the well know *Navigation, Guidance and Control* approach. This approach solves the fundamental problems of mobile robotic.

- Navigation ⇒ Where am I?
- Guidance ⇒ Where do I want to go?
- Control ⇒ How will I get there?

The Navigation is splitted into two parts. Determine the absolute pose using a 270 degree Laser-Rangefinder and determine the relative pose using robot's odometry. There is a need to use the odometry caused by the time the Laser-Rangefinder takes to select sensor data. The odometry is fast enough to fit into the control loop. Still it is very inaccurate therefor the Laser-Rangefinder is needed to correct the odometry.

To answer the question *Where do i want to go?* the actual position of the robot is compared with the desired position extracted from the calculated trajectory. Using the direct Kinematic of the robot, the speed vector can be used as an input for the motor controllers [4]. In order to receive a better result the differential of the desired trajectory is used as pilot control for the motor controllers.

Combining Navigation,Guidance and Control leads to a cascade controller [2] as shown in figure [2].

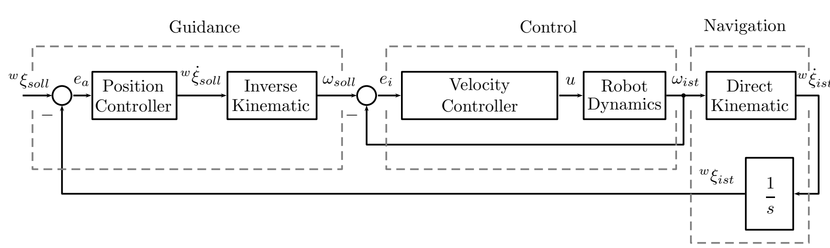

Fig. 2. Cascade controller

4 Caching Motion Planning

Waypoints and edges are the main building units of the motion graph. Given the dimensions of the field and machines' information, an algorithm generates waypoints and edges and store them in a config file. Waypoints are classified into normal waypoints that can act as start points, end points and intermediate points and other waypoints that can act only as start and end points. Waypoints positions and edges are designed in a way that maximizes the efficiency of spline motion control by making it easy to have smooth curves on the way from one waypoint to another. Distances between waypoints are close enough to maximize the utilization of the field's space allowing the robots to work together within the minimum possible area while they are far enough to avoid collisions between robots.

During the initialization process before the robot's system start, shortest paths between all the pairs of waypoints are calculated using dijkstra's algorithm and cached in a hash map keyed by start and end waypoints. When a robot asks motion planning for a path, it first checks if the shortest path in the hash map is unblocked, otherwise it calculates a new path (if there's one) in real-time. When a robot reserves a path is applies two types of blocks on waypoints, hard block and soft block. Hard block is applied on the waypoints that the robot will pass through and it means that no other robot can hard or soft block these waypoints. Soft block is applied on waypoints with direct edge to hard blocked waypoints where other robots can't directly pass by them but can soft block them for their paths.

When the needed start or end points aren't exact waypoints, temporary way points and temporary edges are created to connect these points to the predefined waypoints. Figure [3] shows an overview of the field with waypoints, edges and example path (from "S" to "D") showed on it.

5 1-Path HMR Image Analysis

5.1 Mapping from RGB to HSV

One of the parameters of features is color where HSV is used since it's easier to use in color classification compared to RGB. Since the input image from the Robotino API is RGB, the following mapping technique is used to make the conversion from RGB to HSV efficient. The ranges of HSV colors that represent the objects to be detected are defined in a map with an index for each. When an image is converted, it's converted to indices of the corresponding HSV ranges to make feature matching faster. The following steps and figure [4] demonstrates the mapping process.

- RGB values map to binary values where 1s are present corresponding to matching ranges indices
- Binary values are anded to find the common index for the H, S and V ranges

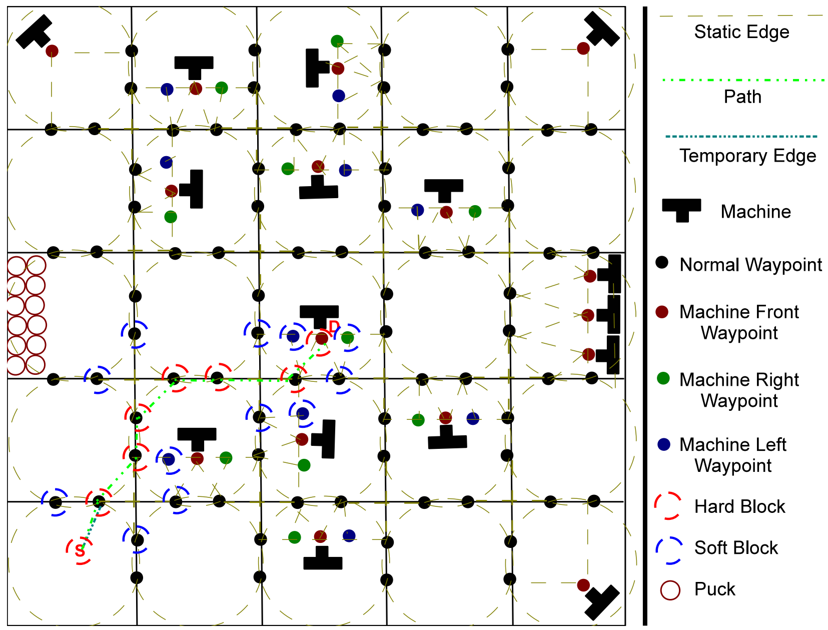

Fig. 3. Motion Graph

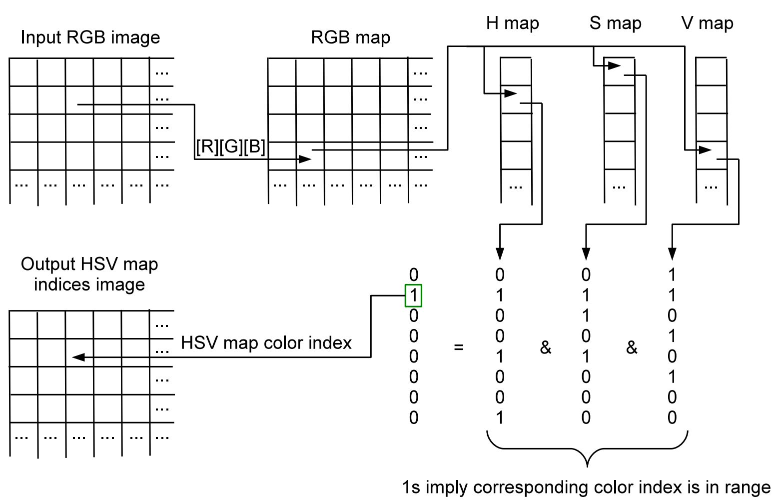

Fig. 4. Mapping RGB to HSV index

5.2 Segmentation

A one path algorithm is implemented for segmentation (using 4-way connectivity). Created segments are stored in a hash map for constant time access using an ID where each pixel in the image points to the ID of the segment that it belongs to. For each pixel there is one of three possibilities as follows.

- If the current pixel doesn't belong to the same color range of either the pixel above it or the pixel to the left of it, a new segment is created.
- If the current pixel belongs to only one of the preceding pixels (above and left), the pixel gets merged in the corresponding segment.
- If the current pixel belongs to both of the preceding pixels' color ranges, the pixel joins one of the them and both segments are merged if they aren't already merged.

5.3 Detection

Detection is classifying some segments into features. The user initializes a feature map that's used both for detection and objects retrieval. The feature map has a per-feature mutex to synchronize the r/w access between objects retrieval and detection. Each feature can be deactivated when there is no need to detect it to save processing power. And each detected object has a time stamp for detection time so users can retrieve objects detected within the last N milli seconds. A feature is a tuple of (α, β, γ).

- $\alpha \Rightarrow$ set of relevant indices in the color map.
- $\beta \Rightarrow$ range of area (min, max) in terms of the number of pixels in the segment.
- $\gamma \Rightarrow$ ranges of width and height.

After an object is detected, the distance from the object is detected using the vertical field of view of the camera, the tilt of the camera, the distance of the camera from the ground, the length of the image and the cut line of the detected object with the ground. And the relative orientation of the detected object is calculated using the horizontal field of view of the camera, the width of the image and mid-x point of the segment.

5.4 Hierarchal Multi-Resolution Architecture

Basically the architecture is to analyze the image level by level giving an independent result for each level until a time-out is hit. The main difference between levels is in the count of pixels per level, and the more pixels a level has the more precise its result is. The step between pixels on each level defines the amount of pixels to be used on that level. In HMR-IA analysis can start from any level, end at any level and possibly skip levels in between using a tuple of $(\varsigma, \varepsilon, \zeta)$.

- $\varsigma \Rightarrow$ start level [end level, ∞]
- $\varepsilon \Rightarrow$ end level [0, start level]
- $\zeta \Rightarrow$ level step [1, ∞]

Start and end level parameters (η) are converted into pixel step (ρ) using $\rho = 2^{\eta}$.

6 New Gripper

The design of the new gripper is the result of the following requirements.

- slim and small gripper to give give better freedom for motion
- pucks aren't pulled back when the robot moves backward
- fitter for the puck to make it easier to center it under the machines' RFID r/w
- when the robot goes in sharp curves or rotates the puck shouldn't be lost
- easy to mount to the robot
- easy to mount sensors on it
- complies with the rule book dimensions

Figure [5] shows a top-view diagram for the new gripper after testing multiple designs versus each other.

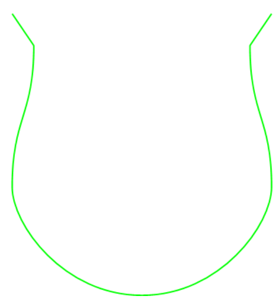

Fig. 5. New gripper (contour)

7 New Hardware

With the current robot's processing specs, it's very difficult to develop complex planning and image analysis algorithms. Alternatively using a PC just moves the delay from processing power to wireless communication. So Intel NUC is mounted on the robot with fast ethernet connection in order to be able to provide the needed processing power. Intel NUC is a small 12cm x 12cm desktop computer with a VESA-Mount and a single-rail power-supply, that made it easy to mount it on the robot. A a/b/g/n W-LAN with 5Ghz W-LAN is also installed for better communication between the robots and for communication with the referee box. As the NUC has a x86-architecture and is running a normal Linux distribution the code doesn't need to be ported. In order to cover the power requirements of the NUC, the original lead-acid batteries were replaced with lithium-polymer batteries, nearly doubling the capacity. Also a small switching power-supply was developed, as the original power-supply of the NUC is 100-240V AC.

8 Conclusion and Future Work

For 2013, robOTTO is trying to restructure its whole system in a modular way that should survive for years and be easily maintainable. As discussed in this paper, many vital parts of the robot's system were developed for 2013 and other parts are currently in development process for 2013 and 2014. For the future, robOTTO targets more hardware additions/development, development of the current work for more modularity and efficiency and adding new ideas and more external tools. Following are some of the ideas that may be developed for 2014.

– real-time adaption for spline motion paths which is mainly useful for real-time obstacle avoidance without having to stop the robot
– in motion planning, only likely-to-be-used paths should be cached into the memory in order to save RAM space
– image analysis should be able to dynamically decide which hierarchal multi-resolution level will be the best for each image

References

1. Naeem W. et al, *The design of a navigation, guidance, and control system for an unmanned surface vehicle for environmental monitoring*. Journal of Engineering for the Maritime Environment 222, 2008.
2. El-Shenawy A., *Motion Control of Holonomic Wheeled Mobile Robot with Modular Actuation*. Mannheim University, Germany, 2010.
3. Liu Y. et al, *Omni-directional mobile robot controller design by trajectory linearization*. American Control Conference, 2003.
4. Li X. et al, *Dribbling control of omnidirectional soccer robots*. Robotics and Automation, IEEE International Conference, 2007.
5. Sanjeev Sharma and Matthew E. Taylor, *Autonomous Waypoint Generation Strategy for On-Line Navigation in Unknown Environments*. Washington State University, 2012.
6. Shannon M. Farrell, *Waypoint Generation based upon Sensor Aimpoint*. Air Force Research Laboratory Munitions Directorate, Eglin Air Force Base, FL, USA and Air Force Institute of Technology, Wright-Patterson Air Force Base, OH, USA, 2009.
7. Peter Kontschieder et al, *Structured Class-Labels in Random Forests for Semantic Image Labelling*. Institute for Computer Graphics and Vision, Graz University of Technology, Austria and Dipartimento di Scienze Ambientali, Informatica e Statistica Universita Ca Foscari Venezia, Italy, 2011.
8. Lifshitz L.M. and Pizer, S.M., *A multiresolution hierarchical approach to image segmentation based on intensity extrema*. Massachusetts Univ. Med. Sch., Worcester, MA, USA, 1990.