Qualification Document for RoboCup 2016

Finn Poppinga*, Patrick Göttsch*, Arne Hasselbring*,
Finn Linzer†, Pascal Loth*, Thomas Schattschneider*,
Erik Schröder*, and Oliver Tretau*

*Hamburg University of Technology, †HafenCity University Hamburg
RobotING@TUHH e.V., roboting@tuhh.de

***Abstract*—In this Qualification Document, the work and approaches for robot soccer of the RoboCup SPL team HULKs is described. General information regarding the team's composition and equipment is given and a description of current implementations and software design decisions made for RoboCup 2016. The impact on the SPL is outlined by referring e.g. to an annual international workshop, that unites teams from different RoboCup leagues.**

I. Introduction

We, the Hamburg Ultra Legendary Kickers (HULKs), are a department of the organization RobotING@TUHH e.V. and were established in 2013. The organization holds 57 members, of which 31 are direct members of the HULKs team, of which again 21 are contributing to the development of the code base. The others are members of the marketing and organization department or passive members. Most of them are or were affiliated with Hamburg University of Technology (TUHH), one member is affiliated with HafenCity University Hamburg (HCU). Detailed team information is given in Tab. I.1.

The remainder of this Qualification Document is structured as follows. Sec. II describes our previous work, robot information, the team's preferences, and a general description of our code base. In Sec. III we outline our current developments and provide an introduction in our pursued coding implementations. Sec. IV closes with a short view on the HULKs impact within the RoboCup Standard Platform League (SPL).

II. General Information

1) Past Work: The HULKs participated in the RoboCup Major SPL competition from 2014 onwards. In 2014's competition we were pre-qualified for RoboCup 2015, due to a third place in the technical challenges. We participated in all RoboCup German Open tournaments since 2014 and plan to take part in the German Open tournament 2016. In 2014 and 2015 we have held the annual international Robotic Hamburg Open Workshop (RoHOW) engaging the SPL and Humanoid Kid-Size League (HKSL), which will be described in more detail in Sec. IV-2. Game Results from RoboCup competitions from 2013 onward can be reviewed in Tab. II.1.

2) Robot Information: The HULKs currently own eleven NAO robots. Five robots have a version 3.3 body and a version 4 head, the remaining six robots have version 5 bodies and heads. All NAOs are H25 models of the specified version.

3) Preference: We, the HULKs, aim to partici-

General Information	
Team Name	HULKs
Team Leader	Patrick Göttsch, M.Sc.
University	TUHH
Team Constitution	
Undergraduate Students	11
Graduate Students	15
External Members	4

Table I.1
HULKs Team Information

Competition	Opponent			
RoboCup German Open 2014	Nao-Team HTWK	RoboEireann	Bembelbots	
	2:4	0:5	0:0	
RoboCup 2014	rUNSWift	DAInamite	SPQR	UPennalizers
	0:9	0:0	0:2	0:5
RoboCup German Open 2015	Nao-Team HTWK	Berlin United	Z-Knipsers	
	0:6	0:4	0:2	
RoboCup 2015	B-Human	NTU RoboPAL	UPennalizers	TJArk
	0:8	0:1	0:2	0:6

Table II.1
HULKS GAME RESULTS FROM 2013 ONWARD, SPLIT BY COMPETITION. HULKS SCORE IS LISTED FIRST.

pate in three competitions, ordered by preference: the indoor tournament, the technical challenges, and in the *DropIn* games.

4) Code Usage: Our code base is an independently developed system and does not use code of any other SPL team. However, we link against third party libraries as e.g.`boost`, `libjpeg`, `libpng`, `eigen` [**?**], and `flann` [**?**]. The 2016 code no longer depends on `opencv`.

III. Current Development

As stated in previous Qualification Documents, the HULKs' code base is divided into three major fields of development: the *Brain*, the *Vision*, and the *Motion* departments. Since the focus of the first mentioned department was set on general and miscellaneous development, the major contribution within the last year is depicted in Sec. III-A. The other research field's current progress is described in Sec. III-B and Sec. III-C, respectively.

A. General Progress

1) Interprocess Framework: Within the last year, we finished the implementation of a framework for an interprocess communication allowing the major code base to disengage from the *NAOqi* framework. During this process it was possible to develop an own toolchain as well, providing `C++14` support for our code base.

B. Vision

1) Software Architecture: As stated in [1], we redesigned our whole vision system for RoboCup in China. This architectural redesign has been under heavy development during the preparation for RoboCup in China and has been continued ever since. The new framework speeds up the development process rapidly and allows for an easy to modify vision pipeline. Module execution is ordered automatically, depending on their requirements and productions, respectively.

2) Ball Detection: One of the major challenges resulting from the 2016 rule changes will be the change in ball color from orange to mostly white [3]. As all other objects in the field (except team jerseys) are white as well, all color based approaches to ball detection are now obsolete. To address this problem we evaluated different algorithms for ball detection, e.g. the *pixel2pixel* algorithm presented by Scaramuzza et al. [4] which uses the shape of the ball to detect it. This algorithm is based on binary edge detection. A basic problem is the performance of the *pixel2pixel* algorithm. It is not sufficient for real time RoboCup applications. We obtained the binary image by the Canny-Algorithm by John Canny [**?**] which we managed to make real-time capable. The Canny-Algorithm has great robustness properties for varying light conditions but doesn't provide usable binary images of the realistic ball.

In this year's Realistic Ball Challenge we used another ball detection algorithm that is based on a segmentation of the image. From a preprocessing stage we get lists of segments detected on vertical scanlines. The segments that are not field colored are clustered and their bounding boxes are computed. After collecting the clusters, basic dimension checks are performed. A result can be seen in Fig.

Figure III.1. The result of the ball detection as seen in the HULKs debug tool. Clustered regions are highlighted white, detected balls are circled red. Note, that the 2016 ball candidate is detected.

III.1.

3) Goal Detection: The new white goals were one of the challenges that resulted from the new rules for RoboCup 2015. A problem arising with white goal posts instead of the old yellow ones, is that it is more likely to find similar light-colored regions in the background of the image.

For this reason we use the brightness channel of the image, which is scanned on some pixels below the field border to find edges. We assume that the field is darker than the goal posts. By using this information we match rising and falling edges that have an appropriate distance and save these pairs including their center as goal post candidates.

Currently, there is no attempt to find the crossbar. The goal posts are passed to the brain module which checks whether there are exactly two goal posts and their distance matches the one specified in the *SPL* rules. The two goal posts can then be used for a position estimation.

4) Self-Localization: A good pose determination of the NAO provides additional information to improve most modules. Therefore we use a sensor fusion model working with odometry and gyroscope data, combined with our Iterative Closest Point (ICP) algorithm [**?**]. The algorithm builds up a transformation matrix using `eigen`, which describes the NAO to world coherence by aligning potential relations between identified field points of an image, according to the standardized appearance of the soccer field.

Odometry and gyroscope data drift increase with time and moving distance, but keep a good estimation over a short period. ICP is able to calculate an absolute coordinate near ground-truth, given the point to point correspondence using `flann`. With a good estimation of the alignment, less iterations are needed. Additionally we are able to reset the given drift error with every cycle. This synergy results in a continuous, fast, and robust pose estimation while playing.

5) Outlook: The current approach that is used for the image processing is completely analytical. In existence there are different modules which are dedicated to a certain task. The disadvantage of this vision pipeline is that the errors of each module add up and therefore lead to higher number of false detections.

It is hard to face these problems in a strict analytical fashion. Therefore the team now concentrates on a statistical machine learning approach for different tasks. It is expected that machine learning algorithms yield better outcomes.

C. Motion

1) Configuration: As of now, we still use the walking engine design from RoboCup 2014 [1]. However, some improvements on the structure of its code have been done lately. The biggest issue with the walking engine is the fact, that one needs to adopt eight parameters to get optimal results for different underground situations. For the any-carpet challenge in 2015 we needed to speed up this tuning process rapidly. The complete structure of how the team saves and edits configuration files has changed by introducing a *JavaScript* based web-application that can read and write configuration values on the NAO. The configuration is stored in a `JSON` format. To ease the tuning of parameters, this tool can bind configuration values to sliders on a generic MIDI-controller. This way, walking parameters can be tuned online without the need to enter numerical values. This improved our walking capabilities by a great amount and reduced the number of fallen robots per game. The new tuning capabilities allow us to walk on different surfaces, e.g. artificial turf or hard floor.

2) Push Recovery: The capabilty to recover from pushes is an important trait for soccer-playing robots, as the risk of collisions with other robots in the game is very high [**?**]. We use a three-step approach inspired by [5] and [2] to handle pushes and other disturbances of the robot's standing stability. All three steps observe the speed and position of the robot's center of mass as controller input. The actual control strategy is chosen using precalculated state space boundaries that are dependant on the geometry of the robot's feet and the height of it's center of mass above the ground (see Fig. III.2).

For small disturbances of the robot's center of mass a linear controller applies a moment in the ankle joint to bring the state of the center of mass back to the origin. This is called the *Ankle-Strategy*. The geometry of the support polygon limits the amount of torque that can be exerted by the robot's ankle motors. Thus, for larger disturbances a nonlinear control of the robots upper body, the *Flywheel-Strategy* is used. If all of theses countermeasures fail, the robot will use the *FallManager* we presented in RoboCup 2014's open challenge, to ensure falling in a posture that speeds up the process of standing up dramatically.

IV. Impact

1) SPL: Since our foundation, we develop and maintain our own code base independently. It will provide an alternative to the other team's framework (e.g. [**?**]), once it meets our requirements regarding stability and performance. The framework is designed to be robot platform independent and encourages the usage of `C++14` paradigms. Another main goal for the framework is, to allow for a short training period of programmers new to the code. We hope to be able to release our code base in the near future. It will furthermore broaden the range of implementations available for new and existing SPL teams.

2) RoHOW: In 2014 we held the first annual RoHOW[1], together with the HKSL-Team Hamburg Bit-Bots. This workshop has two main goals. On the one hand, we attempt to bring international teams from both leagues together, to foster scientific exchange and to allow students and researchers to utilize synergetic effects between both leagues. In 2014, about 100 members from ten different RoboCup teams came to Hamburg for this event. They participated in ten different workshops and presented their work in short lightning talks or longer presentations.

On the other hand, the RoHOW aims to present RoboCup and robot soccer to the public, hence one day of the three-day workshop is open for public visitors. This public day had a great response in 2014, when more than 400 people visited the RoHOW to watch the games, listen to talks presented by SPL and HKSL team members, and get into touch with students and experts from different universities world wide.

In 2015, RoHOW took place from November 27 to November 29 engaging 65 students and researchers from seven teams. It has become an established event within the European SPL and HKSL community.

[1] www.rohow.de

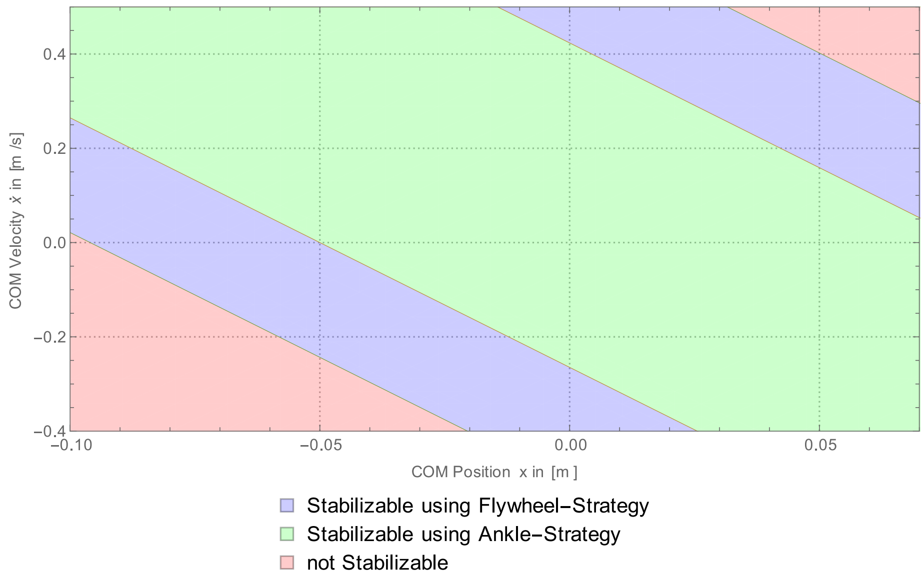

Figure III.2. Stabilization strategy decision surface in the state space.

REFERENCES

[1] Poppinga, F., Bergmann, F., Kaufmann, S.: HULKs-Team Research Report 2014 (2015)

[2] Pratt, J., Carff, J., Drakunov, S., Goswami, A.: Capture point: A step toward humanoid push recovery. In: Humanoid Robots, 2006 6th IEEE-RAS International Conference on. pp. 200–207. IEEE (2006)

[3] RoboCup Technical Committee: Summary of Major Rule Changes for 2016 (2015), http://www.informatik.uni-bremen.de/spl/bin/view/Website/MajorRule2016

[4] Scaramuzza, D., Pagnottelli, S., Valigi, P.: Ball detection and predictive ball following based on a stereoscopic vision system. In: Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on. pp. 1561–1566. IEEE (2005)

[5] Stephens, B.: Push Recovery Control for Force-Controlled Humanoid Robots. Ph.d. dissertation, Carnegie Mellon University, Pittsburgh, Pennsylvania USA (2011)