

Introducing Image Processing to RoboCupJunior

PALB VISION – A First Implementation of Live Image Processing in RCJ Soccer

Christoph Siedentop, Max Schwarz, and Sebastian Pfülb

RoboCupJunior Team C-PALB
CJD-Christophorusschule, Königswinter, Germany*
christophsiedentop@gmail.com,
Max@x-quadraht.de,
mail@sebastianpfuelb.de
<http://www.c-palb.de>

Abstract. For a long time, teams in the RoboCup Junior competition have relied on the same basic sensorical appliances. It is time to evolve. We believe that, the introduction of image processing to RoboCupJunior will take Soccer to a new level of intelligent and less aggressive gameplay. We found it especially challenging to design a system that could not only successfully detect the goal, but also any obstructions, i.e. robots from the opponent's team, in order to score goals more precisely. This work aims to prove that the implementation of image processing does not need to be as much work as one might assume. Our project PALB VISION is designed to serve as an example for other teams in RCJ who plan to use camera-based detection software on their future robots.

We use a CMUcam3 with onboard processing connected to an ATmega 2560. The code for visual detection is written in C and runs directly on the CMUcam. The camera only takes into account the small area between the upper and lower edge of the wall and applies three simple filters for each pixel on a horizontal line.

We have created a quick and reliable vision system which processes ten frames per second and is very resistant to changing illumination. By implementing a special calibration mode, the pre-game setup is reduced to less than one minute. PALB VISION has turned out to be an improvement to the game of RoboCupJunior Soccer and may provide a robust framework for other teams who wish to adopt live image processing into their strategy.

* Special thanks go to Dr Winfried Schmitz a great mentor whose continuous commitment to all different kinds of Robotics activities are outstanding and unmatched. We are also grateful to our understanding parents for their support throughout the years we spent working on robots. Also we would like to thank those not mentioned but who helped us indirectly and expanded our minds in school and at the various RoboCup competitions.

1 Challenge

In the RCJ Soccer 2vs2¹ discipline the field spans 122 cm by 183 cm² and is surrounded by a wall of 14 cm in height. “The walls are painted matte black.”³ An opening of 45 cm in width and a depth of 7 cm in each of the two shorter sides of the field represent the goals. The entire goal is painted matte grey⁴. The robots are 22 cm in diameter and 22 cm high⁵. The weight limit in 2vs2-Secondary is 2.5 kg⁶ and the robot must act completely autonomously.

2 Setup

PALB VISION is based on the CMUCam3⁷ developed by Carnegie Mellon University. Before we opted for the CMU Cam, we used a simple USB webcam connected to an ARM 9-based board⁸ with software developed in C++. However, this setup was abandoned due to its insufficient processing speed.

The robot itself is powered by an Atmel ATmega2560⁹ to which the CMUCam is connected via RS232.

The camera is mounted just above the dribbler at a height of 12 cm from the ground, facing forwards. (See Figure 1)

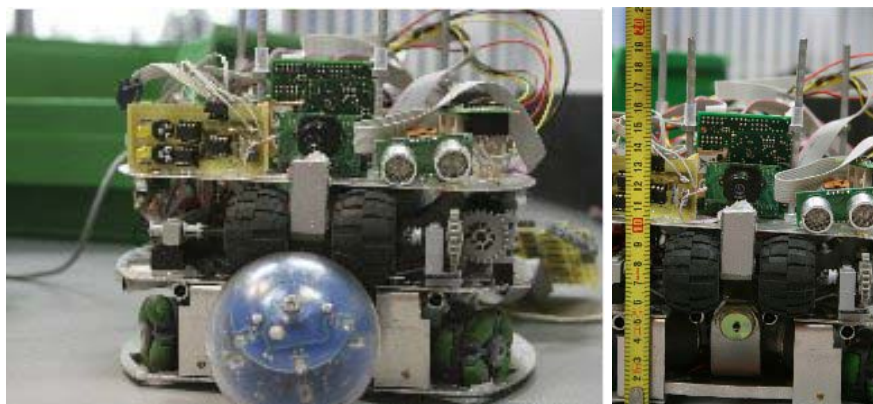


Fig. 1. Robot opened for camera to be visible

¹ Note: Since only our 2vs2 Striker is equipped with a camera all following information applies to 2vs2 soccer.

² [Rules, § 1.1.2]

³ [Rules, § 1.3]

⁴ “75% matte white and 25% matte black” [Rules, § 1.4.3]

⁵ [Rules, § 2.1]

⁶ Secondary refers to games where students are between 15 and 19 years old.

⁷ [CMUCam3]

⁸ portux920T

⁹ [ATmega 2560]

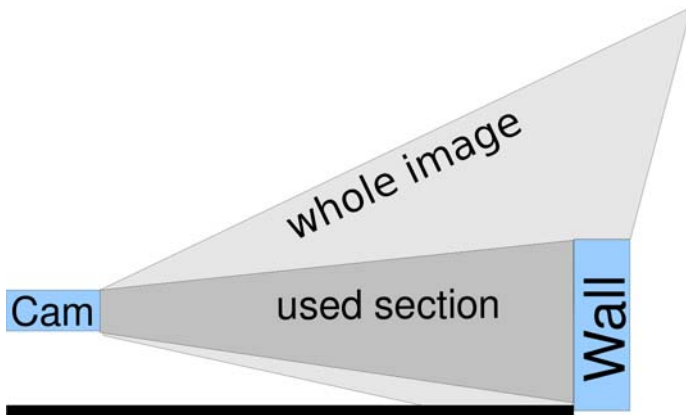


Fig. 2. Vision field

3 Filter

In order to make PALB VISION as simple and error-free as possible and to ensure that the wall is always visible, the CMUcam only processes a cropped version of the original picture (Figure 2).

The image data is loaded as a single line (color) and 14 lines (greyscale). Greyscale values are calculated with the formula of Listing 1.1.

There are five filter passes per frame, which are covered in the following paragraphs.

3.1 Pass 1: Goalie Detection

The first filter pass is the goalkeeper detection which consists of two sub-passes:

1. The filter renders the vertical greyscale variance for each horizontal position. It analyses the seven pixels above and below the middle line and checks whether the values vary. Goalies significantly vary vertically compared to the black wall.

2. The filter renders color information of the picture. Since the wall and the goal have virtually no saturation (grey / black), high saturation areas belong to an enemy goalie.

The results of this filter pass are stored and used by pass 2.

3.2 Pass 2: Division into Sections

For later use we define three types of areas in the image:

- 0: CLASS_WALL
- 1: CLASS_GOAL
- 2: CLASS_GOALIE

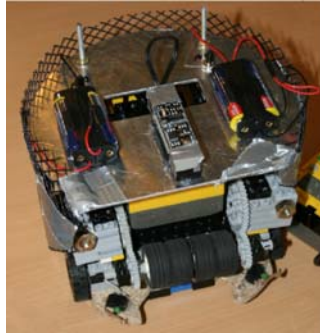


Fig. 3. Colorful robot with many edges

In the second pass, the entire horizontal line is divided into multiple sections by searching for edges in the image. The edge detection uses all of the 14 lines of greyscale and looks for significant changes of luminance in at least 10 lines. The previously calculated goalie areas (see Pass 1) are premarked as `CLASS_GOALIE` and no edge detection is rendered in those areas because goalies typically have many edges (See Figure 3).

3.3 Pass 3: Classification

Pass 3 classifies the remaining section by comparing their brightness level with previously calibrated reference luminances (Listing 1.2).

3.4 Pass 4: Simplification

Small (within a preset range) “gaps” between sections of the same class are erased. Sections which are considered too short, are also erased.

If neighboring sections are of the same class, they are merged. The result is that sequential sections always have different classes.

Listing 1.1. Calculating greyscale values

```
1  uint8_t pv_pixel_bw(uint8_t *pixel) //Calculates ←  
    the black/white (bw) pixel for PalbVision  
2  {  
3  return    ((double)0.299) * ((double)pixel[0])  
4           + ((double)0.584) * ((double)pixel[1])  
5           + ((double)0.114) * ((double)pixel[2]);  
6  }
```

3.5 Pass 5: Result

The fifth and final pass sends the width and position of the longest section which is classified as CLASS_GOAL to the robot’s controller. Additionally the “goalie” section closest to this “goal” section is transmitted. (Fig. 4)

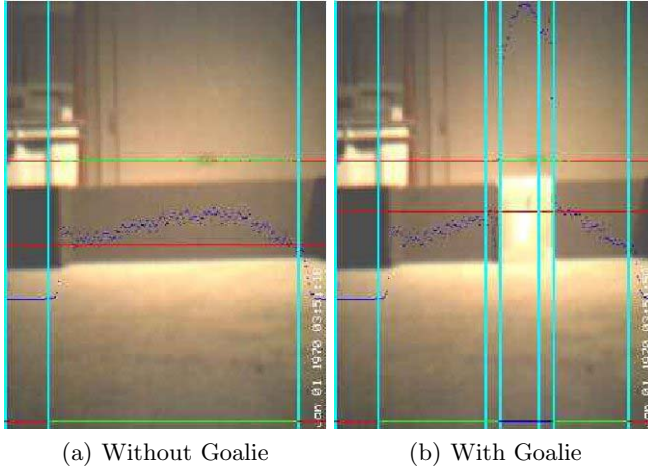


Fig. 4. The sections are distinguished by vertical lines. The overlaid scatter plot represent greyscale values. The photos are taken from an older version of the system but with the same algorithm. See Figure 6 to 7 for newer images.

4 Application in Strategy

The use of real-time image processing opens up many new ways for a more strategic play in the RCJ. At the moment, our robot acts according to the following principles:

Listing 1.2. Comparing average brightness of a section with goal and wall reference

```

1  if(abs(goal_ref - section->avg) < abs(wall_ref - ↵
    section->avg)) {
2      section->class = CLASS_GOAL;
3  } else {
4      section->class = CLASS_WALL;
5  }
```

Whenever the robot is in possession of the ball, the camera is used to aim towards the goal and, if necessary, evade the goalie. As soon as a sufficiently large section of goal is detected in the center of the visual field, the kicker is triggered.

If the goal is too narrow to aim at, PALB VISION decides whether this is due to a goalie or a mispositioning of the robot. In the first case, the robot moves sideways until the goal is wide enough. In the second case, the robot rotates until the goal is wide enough.

5 Calibration

There are six values which have to be calibrated in order to make PALB VISION operate properly. `goal_ref` and `wall_ref` define reference luminance for goal and wall (See Listing 1.3. Because lighting changes at every table, these two have to be recalibrated before every game. `goalie_sigma_thr` and `goalie_sat_thr` define the thresholds above a pixel is thought of as a goalie based on variance (σ)¹⁰ and saturation. However, other than the goal and wall references these two settings do not depend on the lighting. Saturation and variance are relatively consistent, even with different lighting conditions. The last values to calibrate are two values that are used in pass 4 (simplification). `goalie_min_size` describes the minimum size of a goalie in the picture. `goalie_max_gap` determines the maximal length of a goal section lying between two goalie sections may be before being merged.

Listing 1.3. Default values for calibration

```

1 // color values 0-255
2 int goal_ref = 50;
3 int wall_ref = 35;
4 int goalie_sigma_thr = 70;
5 int goalie_sat_thr = 70;
6
7 //in pixel, picture width 176
8 int goalie_min_size = 1;
9 int goalie_max_gap = 10;

```

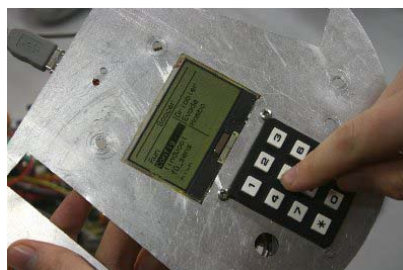
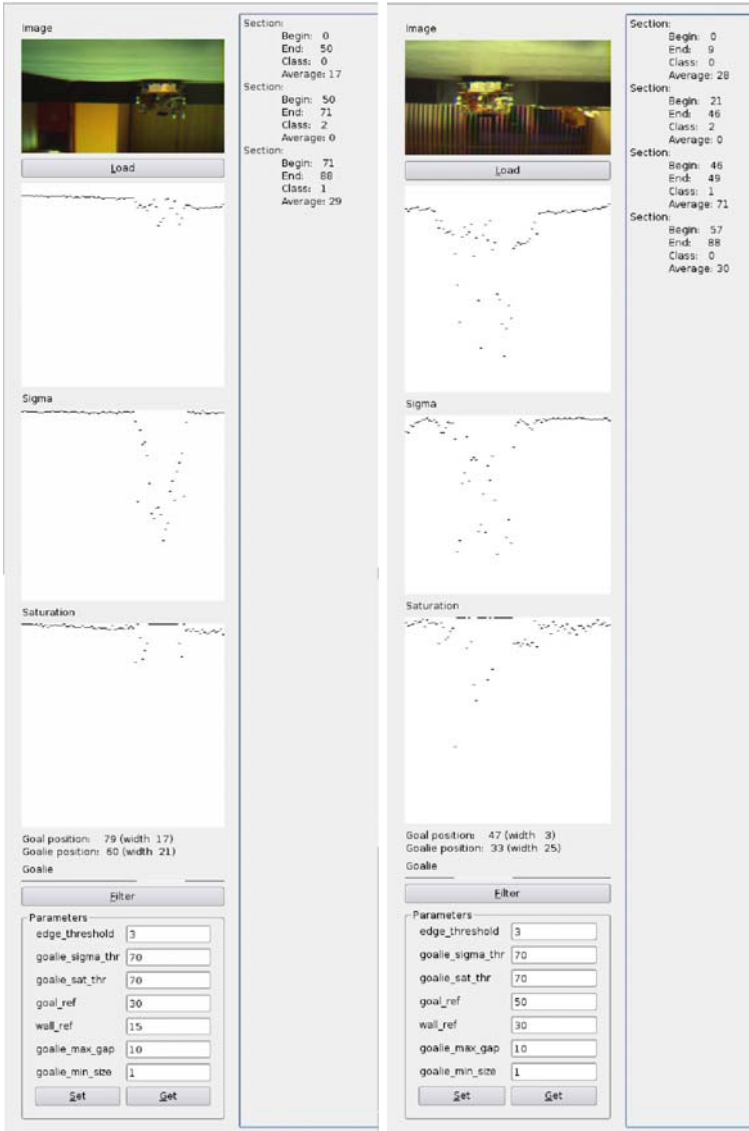


Fig. 5. Easy Calibration with external display

¹⁰ We initially worked with the standard deviation but used the variance later because it is simpler to calculate.

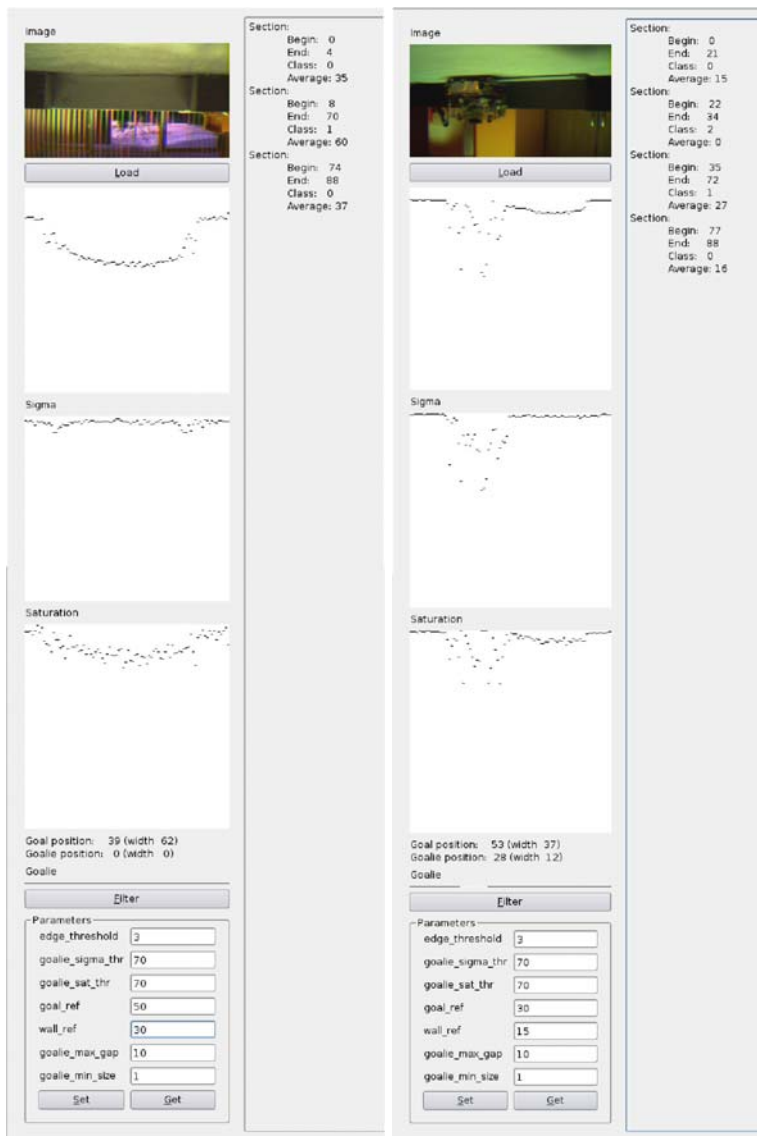


(a) Goal is off-centered (b) Robot is far away from goal

Fig. 6. Different results of the filter

5.1 Calibration Before Game Play

Since there is usually little time before and in between matches left for calibration, we had to make sure PALB VISION has a quick and easy to use calibration system. It is based on a graphic display together with a numpad connected to the main controller (Figure 5). By aiming the robot towards goal and wall and



(a) Without goalie

(b) From the side

Fig. 7. Different results of the filter

holding down a specified button, the controller reads several values from the camera, averages them and stores them as the new reference. This procedure takes less than a minute. It is also possible to adjust those values manually, e.g. for tweaking. All values are stored in the EEPROM of the main controller board.

For programing purposes visualisation and calibration over a computer interface is also possible (See Figure 6 - 7 on 314f)

6 Possible Errors

Errors can be caused by bad illumination, e.g.

- Shadows caused by humans
- Inhomogenous illumination of the field
- Insufficient lighting¹¹

Since these problems do not occur often at competition sites, we ignored them.

One obstacle that we were not able to overcome yet, is that greyscale values change depending on the distance to the wall. It might be possible to eliminate this problem by using a different goal filter, one that is more adaptive than the current static filter used in PALB VISION.

7 Possibilities for Adaptation

The main motivation we wrote this paper is because RoboCup – and RoboCupJunior – are about sharing ideas and research. This is something that could be endorsed more, especially in RCJ. PALB VISION is designed to be easily adoptable by other teams.

7.1 Hardware

Teams will be able to use any camera and processing system, although we recommend the CMU Cam 3. It costs €150 and can be obtained worldwide [CMUCam3]¹². The camera can be programed over a regular serial connection through any standard computer and can easily be connected to most μ -controllers through its serial connection. Alternatively there are also servo outputs (PWM) which can be used for various outputs.

7.2 Software

The software supplied with the CMU Cam is published under the Apache license [CMUCam3, CMUcam.org] and comes with some example programs of computer vision. RoboCupJunior teams will also be able to download the PALB VISION source code from C-PALB's website [www.c-palb.de]. The source code is licensed under GPL license.

¹¹ The system works fine with illumination over 290 lux. Best results are obtained at 580 lux.

¹² For countries outside Europe, North America or Singapore Elektronikladen [Elektronikladen] will send it.

8 Conclusion

In the future, our image processing system could be employed to detect where an opposing robot is moving. This along with features like detecting how far away the ball is might be very useful on the goalie as well.

We think that the entire RoboCupJunior community can profit from using real-time image processing and even though PALB VISION at the moment resembles a rather rudimentary system, the introduction of camera vision to RoboCupJunior in our opinion is a huge step forward.

References

- [Rules] RoboCupJunior Soccer Rules (2007),
<http://rcj.sci.brooklyn.cuny.edu/rcj2007/soccer-rules-2007.pdf>
- [CMUCam3] <http://www.cmucam.org>
- [Robotikhardware] <http://www.robotikhardware.de>
- [ATmega 2560] Atmega 2560 from Atmel, http://www.atmel.com/dyn/products/product_card.asp?part_id=3632,
http://www.atmel.com/dyn/resources/prod_documents/doc2549.pdf
- [www.c-palb.de] <http://www.c-palb.de>
- [Elektronikladen] <http://elmicro.com/de/cmucam3.html>