Detection of Aerial Balls in Robotic Soccer Using a Mixture of Color and Depth Information

António J. R. Neves, Alina Trifan, Paulo Dias and José Luís Azevedo
Universidade de Aveiro
IEETA/DETI - IRIS Laboratory
3810–193 Aveiro, Portugal
Email: {an, alina.trifan, paulo.dias, jla}@ua.pt

***Abstract*—Detection of aerial objects is a difficult problem to tackle given the dynamics and speed of a flying object. The problem is even more difficult when considering a non-controlled environment, where the predominance of a given color is not guaranteed, and/or when the vision system is located on a moving platform. Taking as an example the game of robotic soccer promoted by the RoboCup Federation, most of the teams participating in the soccer competitions detect the objects in the environment using an omnidirectional camera. Omnidirectional vision systems only detect the ball when it is on the ground, and thus precise information on the ball position when in the air is lost. In this paper we present a novel approach for 3D ball detection in which we use the color information to identify ball candidates and the 3D data for filtering the relevant color information. The main advantage of our approach is the low processing time, being thus suitable for real-time applications. We present experimental results showing the effectiveness of the proposed algorithm. Moreover, this approach was already used in the last official RoboCup Middle Size League competition. The goalkeeper was able to move to a right position in order to defend a goal, in situations where the ball was flying towards the goal.**

***Index Terms*—Robotic Vision, Color Classification, 3D Ball Detection, Trajectory Calculation.**

I. Introduction

Autonomous robotic soccer represents a practical testing scenario for many research ideologies and it is greatly promoted through an international initiative called RoboCup[1]. Through the game of soccer, innovations in some of the most distinct research areas can be tested and shared. Such research areas are: artificial intelligence, computer vision, and multi-agents cooperation, among others.

The RoboCup competitions take place once a year and they are divided in different leagues, one of them being the Soccer League. Within the Soccer League, one of the most dynamic and challenging competition is the Middle Size League[2]. In this challenge, autonomous mobile robots compete in a game of soccer in which each team can be made of up to five robots. The robots are built by the participating teams, taking only into consideration size and weight constraints.

In this paper we present an algorithm for the detection of 3D flying balls using a mixture of color and depth information provided by a Kinect[3] sensor. The algorithm has been successfully used so far by the goalie of the CAMBADA[4] robotic soccer team during the most recent RoboCup MSL competitions (Fig. 1). The final objective is to equip all of the robotic players of our team with 3D sensors and use the approach presented in this paper to extend the current vision of an MSL robotic agent, so that it can cope with 3D object positions.

Figure 1: An image of the robotic soccer team CAMBADA in a game situation.

The paper is structured into 6 sections, first of them being this Introduction. In Section 2 we present related work, however limited until now, according to our knowledge. In Section 3 a short description of the vision sensor that has been used in the experiments is presented. Section 4 presents in more detail the approach that we are proposing for the detection of aerial balls. Experimental results are presented in Section 5. Finally, Section 6 concludes the paper.

II. Related work

To our knowledge, most of the RoboCup teams competing in the Middle Size League (MSL) have limited vision systems regarding the detection of the ball when it is in flight. Most teams use an omni directional camera on top of the robots that only detects the ball and its position when on the ground. This is due to the use of a single camera and the application of projective geometry calculations. Given that most of the robots shoot the ball through the air, the possibility to detect the ball when in flight is very relevant and so far, not carefully explored. The environment of this league is not as restricted as in the others and the pace of the game is faster than in

[1] http://www.robocup.org
[2] http://wiki.robocup.org/wiki/Middle_Size_League
[3] http://www.microsoft.com/en-us/kinectforwindows/
[4] http://www.robotica.ua.pt/cambada

978-1-4673-6991-6/15 $31.00 © 2015 IEEE
DOI 10.1109/ICARSC.2015.13

any other league (currently with robots moving with a speed of 4 m/s or more and balls being kicked with a velocity of more than 10 m/s), requiring fast reactions from the robots. In terms of color coding, in the fully autonomous MSL, the field is still green, the lines of the field and the goals are white and the robots are mainly black. The two teams competing are wearing cyan and magenta markers. For the ball color, the only rule applied is that the surface of the ball should be 80% of a saturated color, which is usually decided before a competition. The colors of the objects of interest are important hints for the object detection, relaxing thus the detection algorithms. Many teams are currently taking their first steps in 3D ball information retrieving [1], [2], [3].

Given that most of the robots shoot the ball through the air, the possibility to detect the ball when in flight is very relevant. Obvious solutions using more than a single camera (either using two additional cameras to provide stereo vision, or combining the information from the omni directional camera with additional cameras) can be considered, as in [4], [5], [6]. However they present some limitations: first these additional cameras may point outside the field and cannot use background or color information to simplify ball segmentation (since a flying ball might be in the air with any possible backgrounds - tribunes, chairs, spectators, etc.) or might have limited field of view (most omni directional camera point downwards, meaning a maximum height of around 60 cm).

In [7], an algorithm based on color and shape detection using a single perspective camera is presented. In [7], the above mentioned problems were also detected. Moreover, in the scenario of a real game, the ball moves at a speed up to 10 m/s, which leads to a considerable blur effect and the shape analysis is compromised.

More recently, [8] presents a different vision approach based on a depth sensor instead of an intensity sensor. As 3D sensor, a Kinect was chosen given its ability to directly provide 3D depth information and its refresh rate of 30 fps. The algorithm presented in [8], given the properties of a flying ball in the MSL environment, starts by voxelizing the space to work in an occupancy voxel space rather than considering the whole cloud of points. A flying object is defined as an object that occupies a given number of voxels with a minimum number of points and whose surrounding voxels are empty. It can be seen as a 3D mask inside voxels non empty (containing a minimum number of points) and the outside voxels empty. In [8] experimental results have been obtained with the robot stopped without any other process running on the computer and in a controlled environment without any other robots. However, after the full integration of the Kinect on the goalie, it has been observed that during a real game, a lower performance on the detection and higher processing time have been obtained.

In this paper we present a novel approach for 3D ball detection in which we use a single Kinect camera. The proposed algorithm uses in real-time the color information to identify ball candidates and the 3D data for discarding color information outside the soccer field. This filtering process reduces the detection of false positives.

III. Kinect Sensor

Kinect is a motion sensing input device developed by Microsoft and launched on November 2010. The sensor includes an RGB camera, an infrared laser projector, a monochrome CMOS sensor, and other components less relevant for our application. The field of view is 57 degrees horizontally, and around 43 degrees vertically. Acquisition of the 3D data from the Kinect is done using a C++ wrapper for libfreenect[5] that transforms depth images into an OpenCV[6] matrix.

In its original configuration, Kinect normal working range is from 0.8 meters to 4 meters. However, the sensor provides distance measurements for longer distances while suffering from additional errors and loss of precision, which cause a discretization effect to appear when distance increases.

In Fig. 2 we can see an example of the images acquired by a Kinect sensor.

a) b)

Figure 2: Images acquired by a Kinect sensor. In a) the color image in RGB color space and in b) the depth image as single channel grayscale image.

IV. Proposed approach

In this paper we present a novel approach for aerial ball detection using a mixture of color and depth information. For this, we use color information to detect blobs of the ball color and the depth information to filter the image in order to remove color information regarding objects that are found outside the soccer field limits. We present these two steps of our approach next.

A Kinect sensor has been mounted on the goalie of CAMBADA robotic soccer team (see Fig. 3) and the approach that we are proposing has been integrated with success in its vision system software (Fig. 4). The software has been used in the most recent RoboCup competition and the logs saved by the robots prove that this approach has been used successfully in several game situations, allowing the defense of several possible goals.

A. Colored blob detection

The UAVision library[7], a computer vision library developed within our research group for color object detection, has been successfully employed for the detection of aerial balls, based on color information, using a Kinect sensor. The

[5]https://github.com/OpenKinect/libfreenect
[6]http://docs.opencv.org
[7]http://sweet.ua.pt/an/uavision/

Figure 3: The goalie of the robotic soccer team CAMBADA with the Kinect sensor installed on its frontal part.

UAVision library [9] includes software for image acquisition from video cameras supporting different technologies, for camera calibration, color blob formation, which stands at the basis of the object detection, and image transfer using TCP communications.

The UAVision library has been used for detecting blobs of the color of the ball. After performing a color segmentation on the input image using a look-up table (LUT), we apply a filter based on depth information in order to remove the color classification of objects that are outside the soccer field. This step will be described next. Scanlines are used for searching for pixels of the color of interest (the color of the soccer ball). Different types of scan lines can be used: linear (horizontal and vertical), radial and circular. When scanning the image in search of the color of interest, the relevant found information is saved using a run-length encoding (RLE) approach. The run-length information is used for forming blobs or clusters of the color of interest. These blobs have to pass a validation process in order to establish if a given blob is a ball. The validation procedure is based on calculating different features for each of the found blobs, such as the bounding box area, the circularity, and width-height relation. The pipeline of the approach that we are proposing is presented in Fig. 4.

B. *Color filtering using depth information*

In this step of the processing pipeline, the depth information from the Kinect sensor is used for discarding the color of the objects that are found farther than a certain distance (in this case, 7m were considered). This complements the previous step by filtering possible objects of the ball color found outside the field. As stated before,this step is applied after the color classification.

First, a calibration between the RGB and depth images provided by the sensor has to be performed. The default calibration parameters available in the ROS package for Kinect

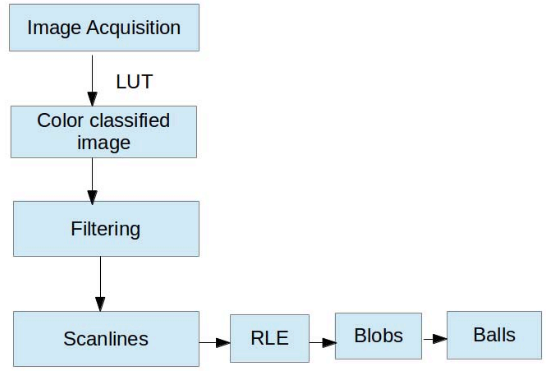
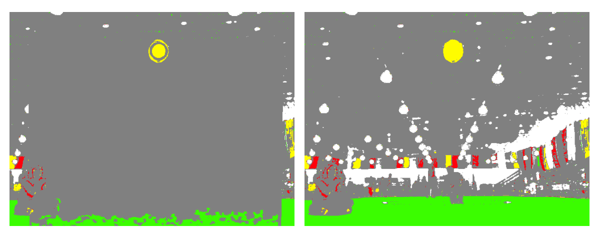


Figure 4: Pipeline of the proposed approach.

calibration [8] have been used for this purpose with some small experimental adjustments.

After having the correct calibration parameters, the distance in meters according to the pixel values of the depth image has been obtained. Two look-up tables relating the pixels of the depth image and the RGB image have been created. This allows filtering the RGB information as stated before. In Fig. 5 we present an example of the filtering step. As shown in the images, after filtering, the classified pixels that are farther than 7m are discarded. There are regions close to the limits of the image that are not filtered due to the fact that there is no correspondence between the depth and RGB pixels for those positions. These regions are not considered for processing regarding blob formation since we configured the limits of the scanlines to be inside the filtered area.



a) b)

Figure 5: On the left, the obtained color classified image after filtering. On the right, the original classified image.

Using the distortion coefficients of the RGB image and the intrinsic parameters of the depth camera, information from the depth camera can be projected to a metric 3D space. Having a valid ball, the next step is to calculate its 3D position relative to the robot, as described next. This step is necessary in order to estimate the direction towards which the goalie should move in order to defend a goal. This is done by establishing a relation between the coordinates of the center of the blob in the RGB image and in the depth image.

[8] http://wiki.ros.org/kinect_camera

C. Obtaining the 3D position of the ball

To obtain the correct 3D position of the ball relatively to the center of the robot, the calibration of the position of the Kinect sensor relative to its position on the robot has to be calculated. The coordinates of the detected ball on the field coordinate frame can then be obtained easily by applying the transformation from Kinect to robot system coordinates and then from robot to world coordinates. In the developed vision system, for the goalkeeper, the same calibration application as presented in [8] has been employed. This application (see Fig. 6) acquires on demand an image from Kinect and allows the user to pick some points on the 3D cloud of points. The chosen points correspond to points in the world whose relative position to the robot are known by the user. The software then evaluates the rigid body transform between the 2 coordinates systems corresponding to the position of the Kinect and its orientation relatively to the origin of the robot coordinates system.

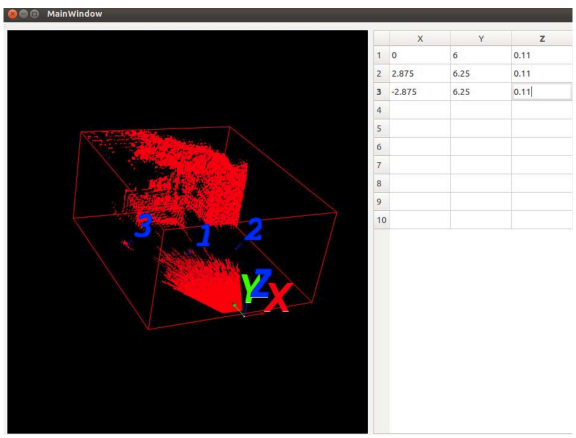


Figure 6: Application for the calibration of the position of the Kinect with 3 reference points on the Kinect cloud and the corresponding coordinates in robot coordinates system [8].

V. Experimental Results

To confirm the effectiveness of the proposed algorithm for aerial ball detection, several experiments have been performed in our laboratory and the obtained results are presented in this section. In these experiments, we calculated a correct detection of the ball in more than 90% of the frames acquired by the Kinect sensor, when the ball was in the field of view of the camera, up to a distance of 6 meters.

Moreover, the proposed algorithm has been already successfully used by the goalkeeper of the CAMBADA team in the last RoboCup event in Brazil, where the referred team won the third place in the competition.

In Figs. 7, 8 and 9 we present several examples of successful aerial ball detections based on the approach that we are proposing.

Figure 7 shows a ball correctly detected when sent by a human towards the robot. This experiment was performed during RoboCup 2014 in a real scenario. As we can see in the image, the environment is very complex and the ball color is also presented in the background.

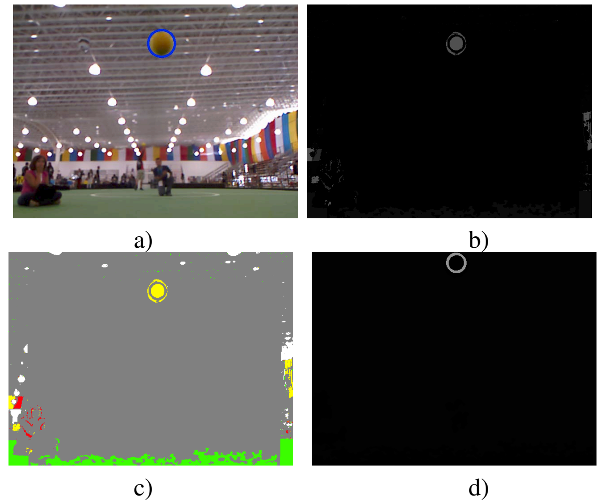


Figure 7: Results obtained when a human kicked the ball towards the goalkeeper: a) original image with the ball detected; b) index image; c) color classified image; d)depth image.

Figure 8 shows the ball being correctly detected when a robot kicked the ball towards the goalkeeper, a realistic situation during the game.

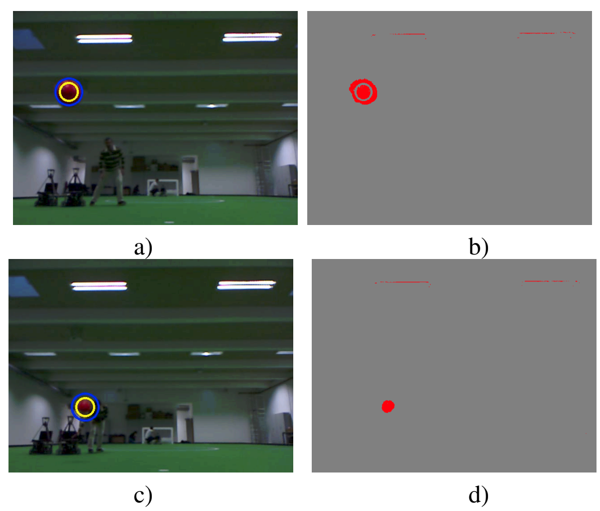


Figure 8: Results obtained using a robot kicking towards the goalkeeper: a) and c) original images with the ball detected; b) and d) corresponding index images.

Figure 9 shows more than one ball being correctly detected when kicked by humans towards the robot. These results prove the efficiency of the proposed approach, even in cluttered environments and with more than one object of interest.

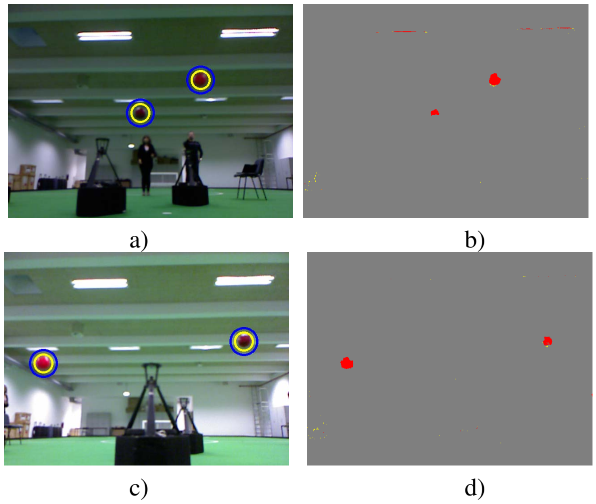


Figure 9: Results obtained using two balls: a) and c) original images with the balls detected; b) and d) corresponding index images.

A. Trajectory of the ball

Using the position of the detected ball in the environment, it is essential for the robotic goalkeeper to estimate the ball trajectory in order to predict the best position to receive it.

For flying objects, and considering that air resistance is negligible, the trajectory can be approximated by a simple ballistic trajectory. To perform this evaluation, we keep trace of the last ball positions and we use the algorithm described in [8] to estimate the trajectory.

Figure 10 shows an example of two calculated trajectories based on the same experiment of Fig. 7. The obtained trajectory is also a confirmation of the detection accuracy in real-time by the algorithm that we are proposing in this paper. The ball was correctly detected in all the consecutive frames acquired by the Kinect sensor.

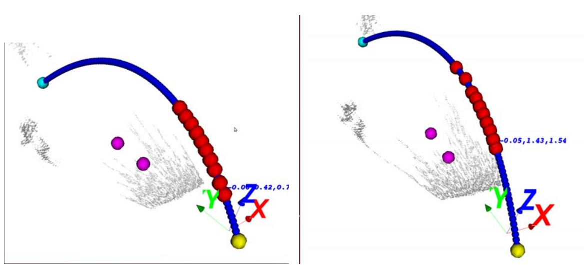


Figure 10: Trajectory of the ball calculated using the position of the ball obtained based on the algorithm proposed in this paper. The positions of the balls used for the computation are presented as large red spheres and the parabolic trajectory estimated is represented by the small blue spheres. The purple spheres represent the projection on the ground of the detected balls.

For flying balls, the best position for the robot to intercept the ball is over the goal line aligned with the ball trajectory. Taking this into consideration, having the trajectory calculated, the goalkeeper can estimate the best position to intercept the ball using its projection of the trajectory on the ground, determined by the two magenta spheres drawn on Fig. 10.

B. Processing time

Besides the detection accuracy, another advantage of the approach that we are presenting in this paper resides in the low processing time (Table I). All the algorithms behind our approach have been stripped to the essential, keeping the focus on real-time performance. The detection process takes an average time of 20ms, which is 10ms less than the approach presented in [8].

Operation	Time (ms)
Acquisition	1
Color classification	2
Filtering	16.5
RLE	0.2
Blob creation	0.04
Blob validation	0.02
Total	**20**

Table I: Average processing times measured for the proposed approach.

The code is written in C++ and the main processing unit available on the robots is an Intel Core i5-3340M CPU @ 2.70GHz 4 processor, running Linux (distibution Ubuntu 12.04. LTS Precise Pangolin). In the implementation of the vision system that we present in the paper, we did not use multi-threading, however both image color classification and the next steps can be parallelized if needed. Multi-threadind has not been used in the scenario we have presented for two major reasons. On one hand, none of the modules of our vision system has processing time higher than the scheduling interval configured in the Linux kernel installed on our computers. On the other hand, there are other processes running on the same computer, namely another vision process for the catadioptric vision system, AI agent, hardware communication, among others.

C. Competition results

This approach has been used in the most recent RoboCup competition by the robotic goalie of CAMBADA team. After analyzing the videos of the games and the log files recorded by the goalie during game time, we conclude that the position of aerial balls have been correctly detected in several game situations and the robot defended most of these balls. This was an important contribution since until now the robot was only able to detect balls on the ground plane.

Taking into consideration the last two games of the referred competition, the semi-final where the CAMBADA team lost the game by 2-1 and the 3rd/4th place game where the CAMBADA team won by 4-0, the ball was detected by the proposed algorithm in 110 images, on a total of 16 game situations where the ball was kicked by the opponent team towards the goalkeeper and traveled by air (see some game situations on Fig. 11). As the game results shows, our team only suffered two goals, being one of them achieved by the ball moving on the ground and after being kicked by one of our own robots.

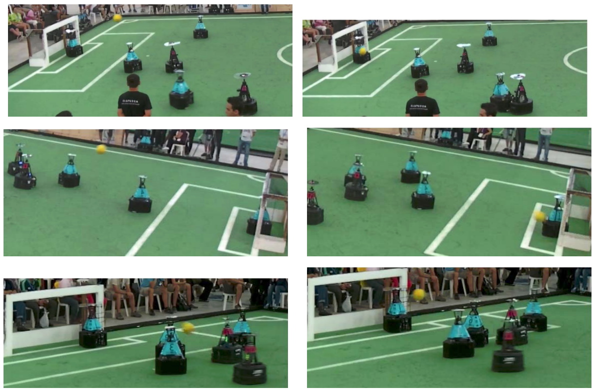


Figure 11: Three game situations during RoboCup where the goalie correctly defended the ball. On the left we can see the moment after the kick and on the right the goalkeeper correctly positioned on the goal line when the ball reaches it.

VI. Conclusions and Future Work

In this paper we have presented a fast approach for the detection of aerial balls in a game of robotic soccer, using a mixture of color and depth information. The experimental results prove that it is possible to have a fast detection of aerial objects in clustered environments by merging color and depth information. The simplicity of our approach, as well as the undeniable results that we present, prove that it is possible to use such an algorithm in real-time applications, such as the games of robotic soccer.

In what concerns future work, the next step will be to include this detection approach within the behavior of different robotic soccer agents such as the strikers. Moreover, we want to extend our algorithm for the detection of more aerial color coded objects that can be used in different robotic applications.

VII. Acknowledgements

This work was developed in the Institute of Electronic and Telematic Engineering of University of Aveiro and was partially supported by FEDER through the Operational Program Competitiveness Factors - COMPETE and by National Funds through FCT - Foundation for Science and Technology in a context of a PhD Grant (FCT reference SFRH/BD/85855/2012) and the project FCOMP-01-0124-FEDER-022682 (FCT reference PEst-C/EEI/UI0127/2011).

References

[1] F. Kanters, R. Hoogendijk, R. Janssen, K. Meessen, J. Best, D. Bruijnen, G. Naus, W. Aangenent, R. van der Berg, H. van de Loo, G. Heldes, R. Vugts, G. Harkema, P. van Brakel, B. Bukkums, R. Soetens, R. Merry, and M. can de Molengraft, "Tech United Eindhoven Team Description," RoboCup 2011, Istanbul, Turkey, 2011.

[2] U. P. Kappeler, O. Zweigle, H. Rajaie, K. Hausserman, A. Tamke, A. Koch, B. Eckstein, F. Aichele, D. DiMarco, A. Berthelot, T. Walter, and P. Levi, "RFC Stuttgart Team Description," RoboCup 2011, Istanbul, Turkey, 2011.

[3] A. Ahmad, J. Xavier, J. Santos-Victor, and P. Lima, "3d to 2d bijection for spherical objects under equidistant fisheye projection," *Computer Vision and Image Understanding*, vol. 125, pp. 172–183, 2014.

[4] A. Ahmad and P. Lima, "Multi-robot cooperative spherical-object tracking in 3D space based on particle filter," *Robotics and Autonomous Systems*, vol. 60, no. 10, pp. 1084–1093, 2013.

[5] A. Voigtländer, S. Lange, M. Lauer, and M. Riedmiller, "Real-time 3d ball recognition using perspective and catadioptric cameras," Tech. Rep., 2007.

[6] D.Scaramuzza, S.Pagnottelli, and P.Valigi., "Ball detection and predictive ball following based on a stereoscopic vision system," in *In IEEE International Conference on Robotics and Automation*, 2005, pp. 1561–1566.

[7] J. Silva, M. Antunes, N. Lau, A. J. R. Neves, and L. S. Lopes, "Aerial ball perception based on the use of a single perspective camera," in *Proc. of the 16th Portuguese Conference on Artificial Intelligence*, ser. LNAI, vol. 8154. Angra do Heroísmo, Azores, Portugal: Springer, September 2013, pp. 235–249.

[8] P. Dias, J. Silva, R. Castro, and A. J. R. Neves, "Detection of aerial balls using a Kinect sensor," in *RoboCup 2014 Symposium*, Joao Pessoa, Brazil, July 25 2014.

[9] A. Trifan, A. J. R. Neves, and B. Cunha, "UAVision: A modular time-constrained vision library for soccer robots," in *RoboCup 2014 Symposium*, Joao Pessoa, Brazil, July 25 2014.