Five Years of SSL-Vision – Impact and Development

Stefan Zickler[1], Tim Laue[2], José Angelo Gurzoni Jr.[3], Oliver Birbach[2], Joydeep Biswas[4], and Manuela Veloso[4]

[1] iRobot Corp.,
8 Crosby Dr., Bedford, MA, 01730, USA
szickler@irobot.com

[2] Deutsches Forschungszentrum für Künstliche Intelligenz GmbH,
Cyber-Physical Systems, Enrique-Schmidt-Str. 5, 28359 Bremen, Germany
{Tim.Laue,Oliver.Birbach}@dfki.de

[3] Centro Universitário da FEI, AI and Robotics Lab.,
Av. Humberto A.C. Branco, 3972, S. Bernardo do Campo, SP, 09850, Brazil
jgurzoni@ieee.org

[4] Carnegie Mellon University, Robotics Institute, Computer Science Department,
5000 Forbes Ave., Pittsburgh, PA, 15213, USA
joydeepb@ri.cmu.edu, mmv@cs.cmu.edu

Abstract. Since its start in 1997, the setup of the RoboCup Small Size Robot League (SSL) enabled teams to use their own cameras and vision algorithms. In the fast and highly dynamic SSL environment, researchers achieved significant algorithmic advances in real-time complex colored-pattern based perception. Some teams reached, published, and shared effective solutions, but for new teams, vision processing has still been a heavy investment. In addition, it became an organizational burden to handle the multiple cameras from all the teams. Therefore, in 2008, the league started the development of a centralized, shared vision system, called SSL-Vision, which would be provided for all teams. In this paper, we discuss this system's successful implementation in SSL itself, but also beyond it in other domains. SSL-Vision is an open source system available to any researcher interested in processing colored patterns from static cameras.

1 Introduction

The RoboCup Small Size League (SSL) has evolved to a fast-paced and dynamic environment for cooperative multi-robot research. In the SSL, robots are able to traverse the game field in merely two seconds, which demands algorithms capable of decision making, team coordination and motion control in fractions of a second. In part, this dynamism is due to the adoption of global vision systems in the league. Not having to be concerned with robot localization and mapping problems, teams can focus more on intelligent software algorithms and more precise hardware and control engineering.

S. Behnke et al. (Eds.): RoboCup 2013, LNAI 8371, pp. 656–663, 2014.

However, what should have been an advantage turned out to be a considerable bottleneck. Most of the time available before a competition was devoted to the assembly of multiple cameras brought by each team, and to the iterative adjustment of their vision algorithms. Also, any new team interested in joining the league had to develop its own computer vision system before they could place robots on the field.

An open-source software[1], SSL-Vision [17], changed this scenario five years ago by providing a shared vision system that could be used by all teams. SSL-Vision not only resolved the organizational obstacles and provided a high performance common ground teams could rely on, it exceeded and served purposes outside the SSL domain. In this paper, we review the ramifications of the introduction of SSL-Vision and the impact it had both within and outside of the SSL domain. We also look at the ongoing and future developments that can further improve SSL-Vision.

This paper is organized as follows: Section 2 provides a brief overview of the overall system. Its impact on the SSL and other domains is described in Sect. 3, followed by a summary of ongoing and future work on SSL-Vision in Sect. 4.

2 Structure of SSL-Vision

SSL-Vision [17] was designed with the goal of being flexible and robust enough to meet the demands of all SSL teams. The system's architecture employs an extendable design that reflects the collaborative spirit of the League and that aims to foster use of SSL-Vision as a research tool beyond the SSL and beyond RoboCup. SSL-Vision supports concurrent image processing of multiple cameras in a single integrated application, bringing together all of its functionality, including robot and ball detection, configuration, calibration, and visualization.

SSL-Vision's architecture is described in detail in a previous paper [17]. Briefly, SSL-Vision's processing flow is encoded in a *multi-camera stack* that defines how many cameras are used for capturing, and what particular processing pipeline should be executed. A multi-camera stack consists of several threads, each representing a *single-camera stack*, consisting of multiple *plugins* which are executed in order. The system allows developers to create different stacks for different application scenarios. All configuration parameters of the system are represented in a unified way through a variable management system called *VarTypes* [15], which allows real-time introspection, editing, and XML-based data storage of all stack and individual plugin parameters. Fig. 1 shows a snapshot of SSL-Vision, including the data-tree's visualization on the left-hand pane.

2.1 RoboCup SSL Image Processing Stack

SSL-Vision's default multi-camera stack implements a processing flow for solving the RoboCup Small Size League vision task using a dual-camera setup. In the

[1] `http://code.google.com/p/ssl-vision/`

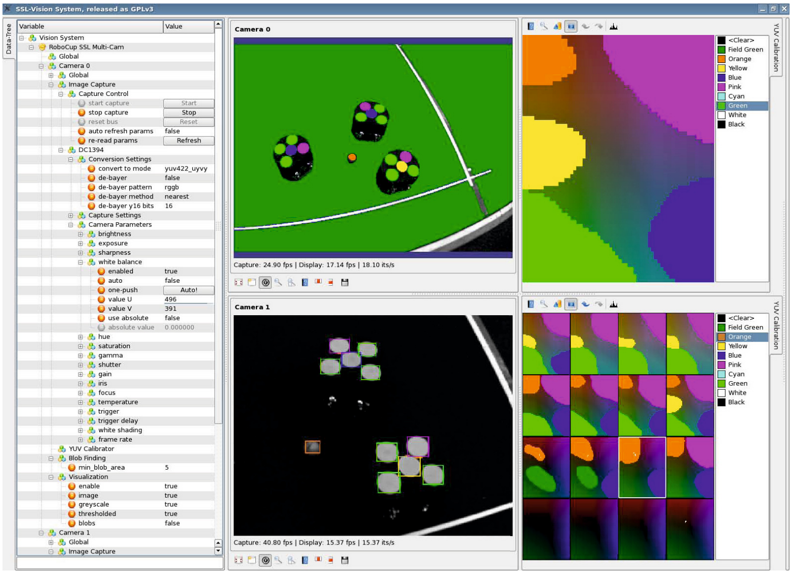


Fig. 1. Screenshot of SSL-Vision, showing the parameter configuration tree (left), live-visualizations of the two cameras (center), and views of their respective color thresholding YUV LUTs (right).

Small Size League, each robot is uniquely identifiable and locatable based on colored markers that are arranged on top of the robot in a standardized pattern. The processing flow of the Small Size League vision stack is as follows:

Image Capture
SSL-Vision supports capture from DC1394-based devices. In the Small Size League, the system typically captures two 780x580 YUV422 streams, each at 60Hz on a separate 1394B bus. We discuss ongoing community efforts to support other capture methods in Section 4.

Color Segmentation
The color segmentation process has been implemented by porting core algorithms of the CMVision library [1] to the SSL-Vision plugin architecture. CMVision's color segmentation is based on a lookup table (LUT) that maps the multi-dimensional camera color space (e.g. raw RGB values) into a discrete 1-dimensional color label (e.g. 'Pink', 'Cyan', 'Yellow', . . .). To ease the calibration of this LUT, the SSL-Vision system features a fully integrated GUI which is able to visualize the 3D LUT (Fig. 1) and which allows to directly pick calibration measurements and histograms from the incoming video stream. The plugin then computes the bounding boxes and centroids of all merged regions and finally sorts them by color and size.

Conversion from Pixel Coordinates to Real-World Coordinates
In order to deduce information about the objects on the field from the measurements of the cameras, a calibration defining the relationship between the

field geometry and the image plane is needed. SSL-Vision stands apart from many previous Small Size League vision systems, because it includes tools that significantly simplify the calibration process and that do not require any calibration accessories (such as checkerboard patterns). SSL-Vision's calibration routine consists of a few simple steps. First, it is assumed that the physical field dimensions (which are standardized by the League's rules) and the camera height (which can be measured easily) are known. Next, using the video image, the user has to manually annotate a rough estimate of the corners of the field and a rough search boundary for the field's line locations. Given these manually defined constraints, SSL-Vision automatically detects line locations as calibration reference points. It then uses the Levenberg-Marquardt algorithm [8] to find the optimal intrinsic and extrinsic model parameters.

Pattern Detection and Filtering

Once all colored markers have been segmented and all their real-world coordinates have been computed, the processing flow continues with the execution of the pattern recognition plugin that extracts the identities, locations, and orientations of all the robots, as well as the location of the ball. The pattern detection algorithm was adopted from the CMDragons vision system [2].

Delivery of Detection Results

The results of the pattern detection are delivered to participating teams via UDP Multicast. Data packets are encoded using Google Protocol Buffers [6], and contain positions, orientations, and confidences of all detected objects, as well as additional meta-data, such as a timestamp and frame-number.

3 Impact

3.1 Impact on the Small Size League

From the beginning, SSL-Vision was not only intended to be an alternative vision system that teams *could* use but a system that everyone *has* to use during official competitions since RoboCup 2010. Therefore, a significant impact on the league has always been foreseeable.

The process of its introduction, including a one-year trial period starting in 2009, can be considered as essentially flawless and appeared to be much appreciated by the community. The fact that there have been few improvements on SSL-Vision since its release as well as almost no complaints about its usage indicate the maturity of the already existing computer vision approaches for the RoboCup Small Size League domain. However, for anybody who wants to do vision algorithm research in the SSL context, SSL-Vision provides a modular framework to write, test and objectively evaluate new approaches against a well-tested baseline implementation.

One major motivation for the introduction of a shared vision system has always been the impact on the organizational realization of Small Size League competitions. Having each team set up its own vision equipment – as was the case before the official use of SSL-Vision– much of the valuable time during a

RoboCup was lost as teams blocked whole fields for setting up their equipment. This was the case during the setup days before the round robin as well as during the finals. In addition, the consequent field assignment made friendly matches against teams from other round robin groups almost impossible. Having the shared vision system, changing fields became as easy as it is in every other RoboCup robot soccer league.

Although successful in resolving the organizational issue during competitions, the improvements due to having an efficient shared vision system go well beyond that. The time a new team entering the League takes to have a functional system has significantly reduced, as the vision system can be executed on most modern computers and requires no knowledge in the computer vision field.

SSL-Vision has also become a foundation for the development of other long-term goals of the League. One of these is the SSL Autonomous RefBox[2], an initiative started in 2011 with the goal to create an autonomous assistant referee for the league. Another related open-source project used by some teams in the League is the grSim simulator [9], a 3D physics simulator developed on top of the SSL-Vision architecture and communication protocol.

The current vision equipment[3] has been purchased by the RoboCup Federation and is handed over to the following year's organizing committee after each competition. This procedure allows a setup of the whole equipment even before the first teams arrive at the competition site, as well as the configuration and calibration. To operate SSL-Vision, the League created the role of the *vision experts*, who are members of the teams who help in operating SSL-Vision in the competitions. Their task is to mark the field boundaries, perform the color calibration, as well as adjusting the settings if the field luminance conditions change. The stages described in Sect. 2.1 happen mostly in the background after the operator performed these few interactions.

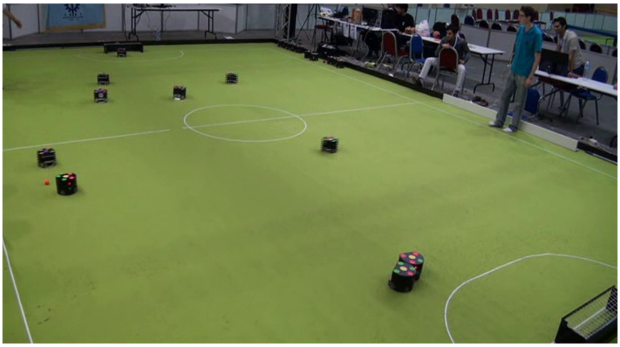
3.2 Impact beyond the Small Size League

As the system does not make too specific assumptions regarding the setup in detail, it can easily be applied to other scenarios, such as tracking objects that are carrying patterns on their tops. A currently popular application is the usage of SSL-Vision as a source for ground-truth data in RoboCup scenarios that demand local vision systems, such as the Standard Platform League (Fig. 2a) and the Humanoid Leagues. Some examples for SPL self-localization research using SSL-Vision for ground truth are [4], [13], and [3]. The two latter projects have been based on the B-Human software which is released as open source since several years and already includes an interface for SSL-Vision data [12]. SSL-Vision has also been selected and demonstrated as the vision system for a proposed new robot soccer league *SSL-Humanoid* that combines the Small Size League's global vision paradigm with a small humanoid robot platform [10]. In addition, SSL-Vision has been used to track robots and obstacles in real-time motion planning research [16] and in university courses such as *CMRoboBits* [14].

[2] `http://code.google.com/p/ssl-autonomous-refbox/`

[3] Four pairs of AVT Stingray F46C cameras with Tamron 12VM412ASIR lenses.

(a)



(b)

Fig. 2. (a) Using SSL-Vision as source for ground truth data in the RoboCup Standard Platform League. Image has been taken from [13].
(b) Demonstration game on a larger field during RoboCup 2011. The field was 9.4 x 6.15 meters and 4 cameras were used.

4 Ongoing and Future Work

The modular plugin-based architecture of SSL-Vision allowed the Small Size League to take important evolutionary steps, such as the addition of a sixth robot to teams without need for vision algorithm adjustments, and the convenient setup of the so called mixed-team games, where two teams play side-by-side, cooperating, each providing a half of the robots in play.

An important step for the League, made possible by SSL-Vision, is the ongoing work towards the creation of a larger field, of almost twice the current official size, for games with more robots playing, perhaps even an 11 vs. 11 soccer match. The first trial of the larger field setup occurred during RoboCup 2011 in Istanbul, and can be seen in Fig. 2b. Two fields were placed side-by-side, and the 4 cameras mounted were used to transmit vision information to the teams participating on the demonstration match. Although field geometry calibration and sensor fusion can become more complex as more cameras are added, and were manually adjusted for the demonstration in 2011, we believe that extending the current capabilities of the software to handle 4 or more cameras is feasible.

There are several extensions to SSL-Vision that are worth to be explored, among them the color calibration system. The current procedure is the most time-consuming configuration task in the software, and requires the operator to calibrate the colors for all different luminance conditions found on the field. The calibration is also static, so changes in the lighting of the field result in need for recalibration. Currently, a few plugins for automated color calibration are under development, one of them employing Artificial Neural Networks and another using Self-Organizing Map (SOM) networks.

An interesting extension would be the change in the robot detection plugin for probabilistic detection of partially detected patterns. This is the case, for instance, when the robot stops under the goal bar, occluding some of its pattern circles, or when a color segmentation error occurs in a small patch of the field. Several algorithms could be used, from naive Bayes classifiers to particle filters (i.e.: [7],[5]). This would improve teams' robot detection and could lead to a fully featured object tracking like [11].

The addition of a capture stack based on Video4Linux, to support USB cameras, as well as support for cameras using Gigabit Ethernet (GigE), would also considerably increase the compatibility and spread the use of the software both for new SSL teams and applications in other domains. A community-contributed stack to add Video4Linux support has been announced[4] as of the writing of this paper.

Another interest of the Small Size League is the distribution of game logs and their corresponding videos in a standard format, therefore a plugin to stream and record camera images via network could improve the qualify of the logs teams use for training and machine learning tasks.

5 Conclusions

In this paper we have shown the vast impact that SSL-Vision has had since its introduction five years ago. By replacing all teams' individual vision systems with a unified open architecture, SSL-Vision solved the biggest organizational challenge of the league and furthermore made it easier for new teams to get involved and immediately focus on the key research aspects of the league. Beyond the Small Size League, SSL-Vision has had significant impact as a research tool, providing a flexible and inexpensive ground-truthing solution for various applications. SSL-Vision continues to thrive on the strong momentum of its active developer community. Hence, looking forward, we expect SSL-Vision's capabilities to further grow and mature, and its applicability to increase to an even larger set of domains, both within and beyond the scope of robot soccer.

Acknowledgments. The authors would like to thank all persons who contributed to SSL-Vision.

References

1. Bruce, J., Balch, T., Veloso, M.: Fast and inexpensive color image segmentation for interactive robots. In: Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000), vol. 3, pp. 2061–2066 (2000)
2. Bruce, J., Veloso, M.: Fast and accurate vision-based pattern detection and identification. In: Proceedings of the 2003 IEEE International Conference on Robotics and Automation, ICRA 2003 (2003)
3. Burchardt, A., Laue, T., Röfer, T.: Optimizing particle filter parameters for self-localization. In: Ruiz-del-Solar, J., Chown, E., Plöger, P.G. (eds.) RoboCup 2010. LNCS (LNAI), vol. 6556, pp. 145–156. Springer, Heidelberg (2010)

[4] https://github.com/cktben/ssl-vision

4. Coltin, B., Veloso, M.M.: Multi-observation sensor resetting localization with ambiguous landmarks. In: Burgard, W., Roth, D. (eds.) Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011. AAAI Press, San Francisco (2011)
5. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003, vol. 2, pp. 264–271 (2003)
6. Google Inc.: Protocol Buffers, `http://code.google.com/p/protobuf/`
7. Isard, M., MacCormick, J.: Bramble: A bayesian multiple-blob tracker. In: Proceedings of the Eighth IEEE International Conference on Computer Vision, ICCV 2001, vol. 2, pp. 34–41 (2001)
8. Marquardt, D.: An algorithm for least-squares estimation of nonlinear parameters. SIAM Journal on Applied Mathematics 11(2), 431–441 (1963)
9. Monajjemi, V., Koochakzadeh, A., Ghidary, S.S.: grSim - RoboCup Small Size robot soccer simulator. In: Röfer, T., Mayer, N.M., Savage, J., Saranlı, U. (eds.) RoboCup 2011. LNCS, vol. 7416, pp. 450–460. Springer, Heidelberg (2012)
10. Naruse, T., Masutani, Y., Mitsunaga, N., Nagasaka, Y., Fujii, T., Watanabe, M., Nakagawa, Y., Naito, O.: Ssl-humanoid. In: Ruiz-del-Solar, J., Chown, E., Plöger, P.G. (eds.) RoboCup 2010. LNCS (LNAI), vol. 6556, pp. 60–71. Springer, Heidelberg (2010)
11. Nguyen, H., Smeulders, A.: Fast occluded object tracking by a robust appearance filter. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(8), 1099–1104 (2004)
12. Röfer, T., Laue, T., Müller, J., Bösche, O., Burchardt, A., Damrose, E., Gillmann, K., Graf, C., de Haas, T.J., Härtl, A., Rieskamp, A., Schreck, A., Sieverdingbeck, I., Worch, J.H.: B-human team report and code release 2009 (2009), only available online: `http://www.b-human.de/file_download/26/bhuman09_coderelease.pdf`
13. Seekircher, A., Laue, T., Röfer, T.: Entropy-based active vision for a humanoid soccer robot. In: Ruiz-del-Solar, J., Chown, E., Ploeger, P.G. (eds.) RoboCup 2010. LNCS (LNAI), vol. 6556, pp. 1–12. Springer, Heidelberg (2010)
14. Veloso, M., Lenser, S., Vail, D., Rybski, P., Aiwazian, N., Chernova, S.: CMRoboBits: Creating an intelligent AIBO robot. In: 2004 AAAI Spring Symposium on Accessible Hands-on Artificial Intelligence and Robotics Education (March 2004)
15. Zickler, S.: The VarTypes System, `http://code.google.com/p/vartypes/`
16. Zickler, S.: Physics-Based Robot Motion Planning in Dynamic Multi-Body Environments. Ph.D. thesis, Carnegie Mellon University, Thesis Number: CMU-CS-10-115 (May 2010)
17. Zickler, S., Laue, T., Birbach, O., Wongphati, M., Veloso, M.: SSL-vision: The shared vision system for the RoboCup Small Size League. In: Baltes, J., Lagoudakis, M.G., Naruse, T., Ghidary, S.S. (eds.) RoboCup 2009. LNCS (LNAI), vol. 5949, pp. 425–436. Springer, Heidelberg (2010)