A New Video Rate Region Color Segmentation and Classification for Sony Legged RoboCup Application

Aymeric de Cabrol[1], Patrick Bonnin[2,1], Thomas Costis[2], Vincent Hugel[2], Pierre Blazevic[2], and Kamel Bouchefra[2,⋆]

[1] Laboratoire de Transport et de Traitement de l'Information L2TI
Institut Galilée, Av JB Clément, 93430 Villetaneuse, France
mrik@l2ti.univ-paris13.fr
[2] Laboratoire de Mécatronique et Robotique de Versailles
10-12 Av de l'Europe, 78 140 Vélizy, France
{bonnin, thcostis, hugel, pierre}@lrv.uvsq.fr

Abstract. Whereas numerous methods are used for vision systems embedded on robots, only a few use colored region segmentation mainly because of the processing time. In this paper, we propose a real-time (i.e. video rate) color region segmentation followed by a robust color classification and region merging dedicated to various applications such as RoboCup four-legged league or an industrial conveyor wheeled robot. Performances of this algorithm and confrontation with other existing methods are provided.

1 Introduction: Motivations and Constraints

Our motivation is to find a segmentation method that is robust to changes of lighting conditions as well for the RoboCup challenges in the four-legged league, as for other applications such as the vision system of the wheeled industrial conveyor robot of the CLÉOPATRE project [1]. RoboCup challenges we want to deal with are:

- the Vision Challenge,
- the ability to tune vision parameters quickly.

These robotics applications require to get several kinds of information: on the one hand they need to identify colored areas of the image that are supposed to be the ball, the goals, the landmarks, the players and the soccer field for the RoboCup application. On the other hand they have to extract the edges that are supposed to be the white lines on the field.

The constraints are stringent in term of available computing power. The proposed segmentation will be performed at video rate on AIBO ERS-7 for the

⋆ Thanks to the RNTL. This work is partly supported by the French research office, grant number 01 K 0742, under the name of CLÉOPATRE project.

A. Bredenfeld et al. (Eds.): RoboCup 2005, LNAI 4020, pp. 436–443, 2006.

RoboCup Challenge, and on the laptop of the vision system embedded on the wheeled robot of the CLÉOPATRE project.

As we want the algorithms to be usable for both applications and others, they must not be dedicated to a specific application.

2 RoboCup Related Works

2.1 Different Directions

As most teams (UNSW [2], CMU [3], UChile [4], etc ...), we use since our first participation in Paris in 1998 [5] a color classification based on a look-up table, followed by a blob detection as low level vision processes. The look-up table is generated by taking a large number of color samples from images of the game field.

Unfortunately color labeling is sensitive to changes in illumination, and manual calibration is time consuming. So autonomous and dynamic color calibration methods have been proposed [6, 7]. The latter paper underlines that the effects of changing lighting conditions are the displacement of the position of the colors in the color space, but the relative position of colors to each other does not change. So the green of the soccer field is taken as a reference color.

Another direction is to use a specific segmentation [8]. Vision processing speed can be increased by avoiding processing all pixels of the image. Even though this specific algorithm is very efficient, we do not want to go that way because we would like to implement a more general purpose segmentation.

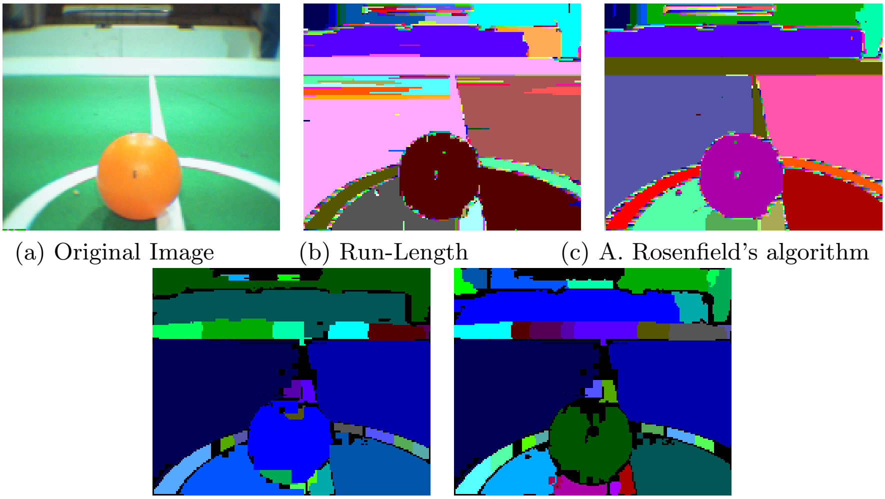
2.2 Proposed Direction

Our key idea is to use a color region segmentation on the whole image, step which is widely independent of lighting conditions, followed by a region color classification. Such a classification, based on the average intensity values of the pixels belonging to the regions, is more robust than color pixel classification. Unlike [9], the growing procedure must be fast enough to guarantee the independent whole segmentation of every image.

Outside of the RoboCup framework, many mobile and autonomous robots integrate vision systems for navigation, obstacle avoidance, and for other robotics tasks [10, 11, 12]. Vision systems are generally based on edge or optical flow segmentations, or color classification of pixels, or neural networks. But few are based on region segmentation [13, 14], even if the pieces of information extracted by region image features are useful for the accomplishment of the robotic task. In addition Priese and Rehrmann implemented their algorithm on a dedicated hardware (Transputer Image Processing System) [14].

3 Region Segmentation Algorithms

Firstly, we adapted two *blob detection* methods (the A. Rosenfield, JL. Pflaz algorithm [17], and a run-length algorithm [3]) to color region growing. The performances are slightly better with A. Rosenfield's algorithm. Temporal results



(a) Original Image (b) Run-Length (c) A. Rosenfield's algorithm

(d) Parallel Isotropic Growing (e) Cooperative Segmentation

Fig. 1. Segmentation results with different algorithms

will be given in §. 5. But in both cases the quality of the segmentation and the processing time are too dependent of the image and of the values of the control parameters.

Secondly, we adapted our *Gray Level Cooperative Segmentation* [16] to color segmentation. The first step is a splitting implementation adapted from the first step of the Horowicz and Pavlidis's algorithm. A region is settled if no edge point is found in it and if an homogeneity criterion is verified. The second step is a parallel and isotropic growing from embryo regions. The quality of this segmentation is quite independent of the image, and does not vary in dramatic proportions with the values of the control parameters. But the processing time is too long! In removing the cooperation with the edge detection, the quality of the results decreases without increasing time performances! It can be noticed from the segmentation results of figure 1 that an isotropic and parallel growing procedure (see (d) and (e)) gives better quality results than a line scan procedure (see (b) and (c)).

4 Proposed Segmentation

Our color region segmentation method is composed of three main steps:

- a *Hierarchical and Pyramidal Merging*, initialized from the pixels,
- a *'Video Scan' (or 'Data Flow') Merging*, adapted for the pyramidal regions,
- a *Color Merging*, merging step based on a color classification of regions.

Each of these two previous main steps considers the operation of merging of each kind of regions separately as a sub-step. Regions are 3×3, 9×9 and 27×27 pixels. Be aware that all the pixels of these regions are not gathered in the square region.

For the first step, as in [13], we use a hierarchical and pyramidal merging method which takes advantage of the connexity of the current pixel neighbourhood, except that we work with an orthogonal topology, and not an hexagonal one. This merging is more efficient than using the quad tree structure. The order of the merging is the following: 3×3 regions from image pixels, then 9×9 regions from 3×3 and finally 27×27 regions from 9×9. The germ is the central pixel or the central region. For the initial step of this fusion, each pixel belonging to the 3×3 neighbourhood is merged to the central pixel (germ) if their intensity values on the three image planes (RGB or YUV) are not too different (see Fig.2 (b)). This sub-step requires a first control merging parameter based on the difference of adjacent pixel intensity. Then, successively, 9×9 (see Fig.2 (c)) and 27×27 (see Fig.2 (d)) regions are obtained in quite a same manner. A neighbour 3×3 region (resp. 9×9) is merged into the 3×3 (resp 9×9) central region (germ) if they verify the connexity criterion, and if the intensities of the adjacent 3×3 regions (in both cases) are not too different. The connexity criterion is the following: the two adjacent pixels must belong to the regions, one for each.

Since the extraction of edge point information is also needed for the localization of the robot, and as shown by the German Team [8] the localization does not need the computation on each pixel of the high resolution image, we combine an adaption of Kirsh 4 gradient operator [18] and an edge thinning step with the initial 3×3 pyramidal gathering. The Kirsh 4 operator is applied on the Y plane only for a pixel every 3 lines and every 3 columns. The thinning step is applied simultaneously on the reduced image.

Associated with the initial 3×3 pyramidal gathering, this processing takes 0.11 ms more than the gathering alone. This additional time is only the computation time. Alone, this edge point detection takes 0.55 ms.

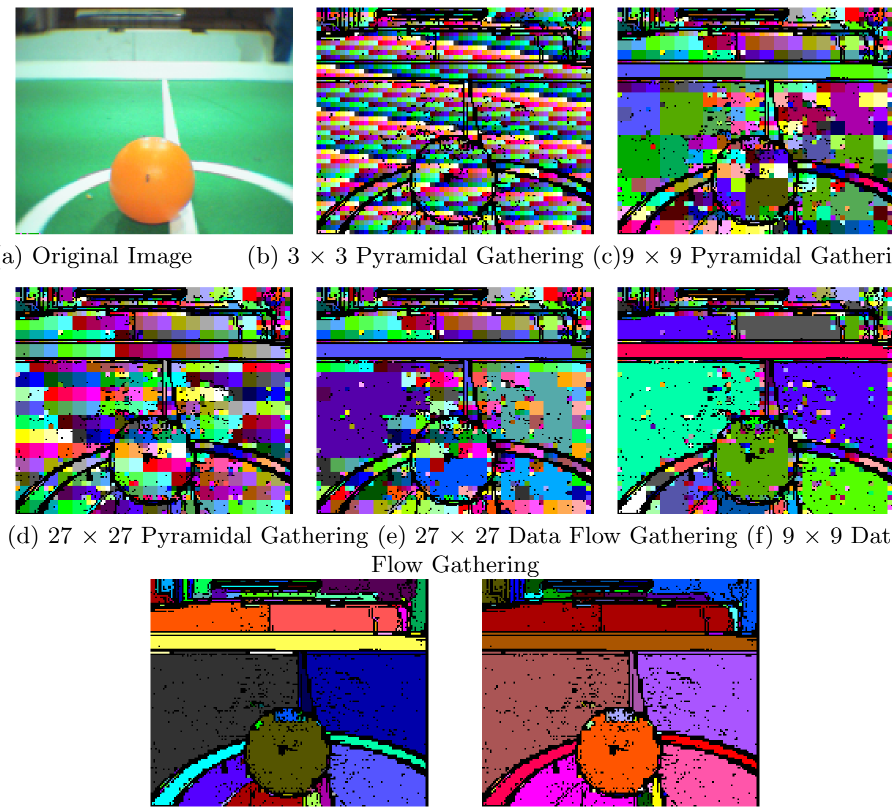
For the second main step, the regions are considered in the opposite order: 27×27, then 9×9 and finally 3×3. The principle of gathering of the 27×27 (see Fig.2 (e)) regions is exactly the same as those used for the pixels in the A. Rosenfield & JL. Pflatz's algorithm, except that it is applied on 27×27 regions rather than on pixels, and therefore the connexity must be verified. Only the two past regions (relatively to the video scan) are taken into account: the left and the upper ones. The gathering of the 9×9 (see Fig.2 (f))and 3×3 (see Fig.2 (g)) region is slightly more complex, because it takes into account the two previous past regions and two next regions: the right and the bottom ones. The merging criteria are the same as for the first step: related to the connexity and to the difference of intensity of 3×3 regions. The information related to the topology of regions (e.g. gathered points and then 3×3 and 9×9 regions) is stored in the data structure of regions. So our algorithm requires only one examination of each image pixel value during the initial 3×3 gathering. All previous steps are quite independent of the changes of the lighting conditions.

The third main step is constituted by two sub-steps:

– *Color classification of regions*
– *Region merging based on Color Classification*

For the moment, the *color classification of regions* consists of finding the best color classification in the YUV space for each region, and to verify the coherence of this classification in the UV space. The results seem to be stable, because it is more robust to classify mean intensity values on a given region rather than pixel intensity values as for a pixel classification. But we are looking for another color space as HSV. As for the 9×9 pyramidal gathering, the *region merging based on color classification* consists of considering one 3×3 every 3 according to the lines, and every 3 according to the columns, and to look at its upper, lower, right and left 3×3 neighbour regions. The regions are merged if they are classified with the same colour. The good quality of the segmentation results can be noticed. In fact the pyramidal gathering main step followed by the data flow gathering procedure simulates a parallel and isotropic growing (see Fig. 2). Two different kinds of parallelism may be underlined:

– all adjacent regions are *simultaneously* merged into the growing region,
– several regions are growing simultaneously.



(a) Original Image (b) 3 × 3 Pyramidal Gathering (c) 9 × 9 Pyramidal Gathering

(d) 27 × 27 Pyramidal Gathering (e) 27 × 27 Data Flow Gathering (f) 9 × 9 Data Flow Gathering

(g) 3 × 3 Pyramidal Gathering (h) Color Identification Gathering

Fig. 2. Segmentation Results of the Different Steps of our Algorithm

Table 1. Detailed processing times of algorithms steps. *Processing time* column contains duration of each step, whereas *total* shows the total duration up to this step. *Percentage* is the contribution of the step to the whole process duration. Steps are in bold font, while sub-steps are in normal one. The first sub-step is shown as the more time consuming, that underlines the efficiency of the following gatherings.

Algorithm	Processing Time	Total	Percentage
Pyramidal Gathering	**4.44 ms**	**4.44 ms**	**64%**
3 × 3	3.95 ms	3.95 ms	57 %
Edge Point Detection	0.11 ms	4.06 ms	1.6 %
9 × 9	0.31 ms	4.37 ms	4.4 %
27 × 27	0.07 ms	4.44 ms	1 %
Data Flow Gathering	**1.45 ms**	**5.89 ms**	**20.8 %**
27 × 27	0.01 ms	4.45 ms	0.15 %
9 × 9	0.31 ms	4.76 ms	4.4 %
3 × 3	1.13 ms	5.89 ms	16.5 %
Color Classification Gathering	**1.06 ms**	**6.95 ms**	**15.2 %**
Color Classification	0.06 ms	5.95 ms	0.9 %
Color Gathering	1.0 ms	6.95 ms	14.3 %

This last property is due to the fact that the merging criterion between two given regions is based on the mean intensity values (on Y, U and V planes) of the initial and connected 3×3 regions (one for each given region). These parameters do not vary during the growing of regions and are independent of the order of the merging between regions.

5 Results and Comments

The temporal results are obtained with an ultra light notebook DELL X300 (Intel Pentium M 378, 1.4 GHz, 2 MB L2 cache, 400 MHz FSB) dedicated to the embedded vision system of a robot. Approximately 10 images are used for testing. The image size is 176×144, with 3 bytes per pixel. The performances of the Sobel's and Kirsh4's operators for edge detection are given for comparison. The processing time of a given algorithm must be at most twice the one of Kirsh's operator to be of interest for the following of our studies. Results are presented in table 2.

Though the cooperative segmentation extracts edge points also, this algorithm is faster than its version without the cooperation. In fact, the edge points make the isotropic and parallel region growing faster. But we are surprised at the bad performances of the adaptation of the run-length algorithm. The explanation is that too many segments are generated, and merging them takes a long time. We are also surprised at the good performances of our method, compared to Kirsh's operator and to the adaptation of A. Rosenfield's algorithm. The explanation is the reduced number of pixel access: 3 pixels for the adaptation of Rosenfield's algorithm, 9 for the Kirsh's, and 1 for our algorithm and the pyramidal approach.

Table 2. Comparison between several algorithms duration

Algorithm	Processing Time	Speed Up
Adapt. of A. Rosenfield's Algo	13.6 ms	2.3
Adapt. of Run-Length	28 ms	4.75
Cooperative Segmentation	72.5 ms	12.3
Parallel and Isotropic Growing	106 ms	18
Proposed Method	5.89 ms	1 : reference
Color Kirsh 4 operator	12.8 ms	2.17
Color Sobel operator	17.6 ms	2.99
Kirsh 4 on Y plane	4.45 ms	0.755
Color Classification and Blob Extraction	2.57 ms	0.436

All source, binaries and tested images will be available on the web site of the CLÉOPATRE project. The quality of the resulting segmentation is suitable for robotics applications. The different regions of the color image are correctly separated. Some points are missing inside regions. Though this is penalizing for a nice looking segmentation, all needed region attributes (gravity center, including boxes etc..) are correct.

Taking into account the processing times of table 2, the proposed segmentation will run at video rate on the robot of the CLÉOPATRE project. For the moment, the *Color Classification and Blob Extraction* (2.57 ms on our test computer) and the *Kirsh4 on Y plane* (4.45 ms) are still used on the AIBO ERS-7 for low level vision processing. The processing time of our new segmentation is similar (6.95 ms compared with 7.02 ms). During RoboCup 2005 we implemented the new segmentation algorithm on AIBO ERS-7. It ran at 15 Hz together with all other modules such as locomotion, localization and behaviours.

6 Conclusion

We have proposed a general-purpose robust real-time region color segmentation and classification, and shown that this was more efficient than pre-existing methods.

The swiftness of this algorithm is mainly due to the reduced number of pixel access, to the bottom-up and then top-down hierarchical merging. Its robustness is the consequence of the region color classification based on mean value for the area.

References

1. M. Silly-Chetto, T.Garcia, F.Grellier, Open source components for embedded real time applications, IEEE Int Conf on Control and Automation, June 2002, Xiamen, China.
2. A.Olave, D.Wang, J.Wong, T.Tam, B.Leung, MS.Kim, J.Brooks, A.Chang, N.Von Huben, C.Sammut, B.Hengst, The UNSW RoboCup 2002 Legged League Team, 6th Int. Workshop on RoboCup 2002 Fukuoka, Japan.

3. J.Bruce, T.Balch, M.Veloso, Fast and inexpensive color image segmentation for interactive robots. Proceedings of the 2000 IEEE / RSJ Int Conf on Intelligent Robots and Systems (IROS'00) 3 (2000) 2061-2066.
4. JC.Zagal, R del Solar, P.Guerrero, R.Palma, Evolving Visual Object Recognition for Legged Robots, 7th Int. Workshop on RoboCup 2003 Padova, Italy.
5. V.Hugel, P.Bonnin, JC.Bouramoué, D.Solheid, P.Blazevic, D. Duhaut, Quadruped Robot Guided by Enhanced Vision System and Supervision Modules, RoboCup 1998, Springer Verlag, M. Asada and H. Kitano ed.
6. D.Cameron, N.Barnes, Knowledge based autonomous dynamic colour calibration, 7th Int. Workshop on RoboCup 2003 Padova, Italy.
7. M.Jüngel, J.Hoffmann, M.Lötzch, Real Time Auto Adjusting Vision System for robotic soccer 7th Int. Workshop on RoboCup 2003 Padova, Italy.
8. T.Röfer, M.Jüngel, Fast and Robust Edge - Based Localization in the Sony four-legged robot league, 7th Int. Workshop on RoboCup 2003 Padova, Italy.
9. F.Hundelshausen, R.Rojas, Tracking Regions, 7th Int. Workshop on RoboCup 2003 Padova, Italy.
10. G.N.DeSouza, A.C.Kak, Vision for Mobile Robot Navigation: A Survey, IEEE Trans. on Pattern Analysis and Machine Intelligence PAMI, vol.24 n2, February 2002.
11. Proceedings of IROS 2001, IEEE Int Conf on Intelligent Robots and Systems, October 2001, Hawai USA.
12. Proceedings of ROMAN 2001, IEEE Int Workshop on Robot and Human Communication, September 2001, Bordeaux - Paris FRANCE.
13. L. Priese, V. Rehrmann, A Fast Hybrid Color Segmentation Method, Proceedings Mustererkennung 1993, pp 297-304, DAGM Symposium, 1993 Lübeck.
14. L. Priese, V. Rehrmann, R. Schian, R. Lakmann, Traffic Sign Recognition Based on Color Image Evaluation, Proceedings IEEE Intelligent Vehicles Symposium '93, pp. 95-100, July 1993, Tokyo, Japan.
15. S.Horowitz, T.Pavlidis, Picture segmentation by a direct split-and-merge procedure, Sec Int Joit Conf on Pattern Recognition, pp 424-433, 1974.
16. P.Bonnin, J.Blanc Talon, JC.Hayot, B.Zavidovique A new edge point / region cooperative segmentation deduced from a 3D scene reconstruction application, SPIE 33rd Ann Int Symp on Optical & Optoelectronic Applied Science & Engineering, vol Application of digital image processing XII, August 1989, San-Diego California.
17. A. Rosenfeld, JL Pfalz, Sequential operations in digital picture processing, Journal of ACM, vol 13, no 4 1966.
18. Kirsh, Computer determination of the constituent structure of biological images, Computer & biomedical research, USA, 1971 vol 4 no 3 pp 315-328.