An improved version of the Generalized geometric triangulation algorithm

Adriano Carvalho
University of Porto

Carlos Couto
University of Minho

An Improved Version of the Generalized Geometric Triangulation Algorithm

João Sena Esteves
Department of Industrial Electronics
University of Minho
4800-058 Guimarães, PORTUGAL
sena@dei.uminho.pt

Adriano Carvalho
Dept. of Electrical Engineering and Computers
Faculty of Engineering, University of Porto
4200-465 Porto, PORTUGAL

Carlos Couto
Department of Industrial Electronics
University of Minho
4800-058 Guimarães, PORTUGAL

Abstract – **Triangulation with active beacons is widely used in the absolute localization of mobile robots. The original Generalized Geometric Triangulation algorithm suffers only from the restrictions that are common to all algorithms that perform self-localization through triangulation. But it is unable to compute position and orientation when the robot is over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2. This paper presents an improved version of the algorithm that allows self-localization even when the robot is over that line segment.**

I. INTRODUCTION

Localization is the process of finding both position and orientation of a vehicle in a given referential system [1], [2], [3], [4], [5], [6], [7], [8], [9]. Triangulation with active beacons is a robust, accurate, flexible and widely used method of absolute localization [1], [10]. Self-localization through triangulation is based on the measurement of the bearings of the robot relatively to beacons placed in known positions. When navigating on a plane, three distinguishable beacons are required - and usually enough - for the robot to localize itself. In Fig. 1, λ_{12} is the oriented angle "seen" by the robot between beacons 1 and 2. It defines an arc between these beacons, which is a set of possible positions of the robot [11]. An additional arc between beacons 1 and 3 is defined by λ_{31}. The robot is in the intersection of the two arcs. Several algorithms of self-localization through triangulation are described in [1], [2], [5], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21] and [22].

Two restrictions are common to all algorithms that perform self-localization through triangulation [2]:

1. The robot must "see" at least three distinguishable beacons to localize itself in a plane. All areas of the plane with less than three visible beacons are unsuitable for robot localization;
2. Localization is not possible if the robot is over the circumference defined by three non-collinear beacons (the intersection of the arcs shown in Fig. 1 is another arc, not a point) or over the line defined by three collinear beacons.

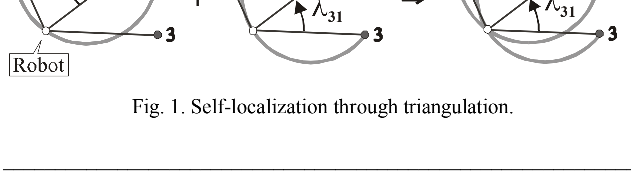


Fig. 1. Self-localization through triangulation.

The Geometric Triangulation algorithm described in [17] uses three distinguishable beacons that must be ordered in a particular way. According to the authors of that paper, *"the algorithm works consistently only when the robot is within the triangle formed by the three landmarks[1]."*

The Generalized Geometric Triangulation algorithm [2] does not require beacon ordering and suffers only from the two restrictions that are common to all algorithms that perform self-localization through triangulation. But it is unable to compute position and orientation when the robot is over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2. It was assumed that, when a beacon becomes between the robot and another beacon, the closest beacon hides the farther one or else the goniometer is not able of simultaneously detecting more than one beacon. Any of those situations prevents self-localization. However, the impediment is due to the technology used, not triangulation itself.

Section II presents an improved version of the Generalized Geometric Triangulation algorithm [1]. It works over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2. Position and orientation errors are defined on Section III. Simulation results [1] that validate the improved version of the Generalized Geometric Triangulation algorithm are shown in Section IV. Conclusions and references are presented in Section V and Section VI, respectively.

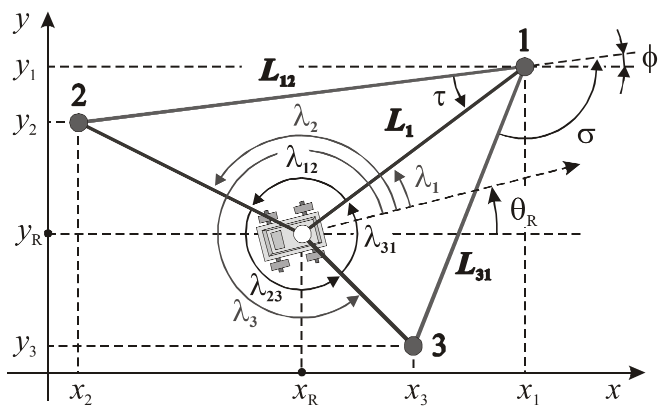
II. THE IMPROVED GENERALIZED GEOMETRIC TRIANGULATION ALGORITHM

The Generalized Geometric Triangulation algorithm uses (Fig. 2) three distinguishable beacons, randomly labeled 1, 2 and 3, with known positions (x_1, y_1), (x_2, y_2) and (x_3, y_3). L_{12} is the distance between beacons 1 and 2. L_{31} is the distance between beacons 1 and 3. L_1 is the distance between the robot and beacon 1. In order to determine its position (x_R, y_R) and orientation θ_R, the robot measures - in counterclockwise fashion - the angles λ_1, λ_2 and λ_3, which are the beacon orientations relative to the robot heading.

Line 14 of the algorithm was not present in the original version of the algorithm. It is required only when the robot is over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2.

---

[1] Beacons are also called *landmarks* by some authors.



<u>Generalized Geometric Triangulation algorithm:</u>

1. If there are less than three visible beacons available, then return a warning message and stop.
2. $\lambda_{12} = \lambda_2 - \lambda_1$
3. If $\lambda_1 > \lambda_2$ then $\lambda_{12} = 360^\circ + (\lambda_2 - \lambda_1)$
4. $\lambda_{31} = \lambda_1 - \lambda_3$
5. If $\lambda_3 > \lambda_1$ then $\lambda_{31} = 360^\circ + (\lambda_1 - \lambda_3)$
6. Compute L_{12} from known positions of beacons 1 and 2.
7. Compute L_{31} from known positions of beacons 1 and 3.
8. Let ϕ be an oriented angle such that $-180^\circ < \phi \leq 180^\circ$. Its origin side is the image of the positive x semi-axis that results from the translation associated with the vector which origin is (0, 0) and ends on beacon 1. The extremity side is the part of the straight line defined by beacons 1 and 2 which origin is beacon 1 and does not go by beacon 2.
9. Let σ be an oriented angle such that $-180^\circ < \sigma \leq 180^\circ$. Its origin side is the straight line segment that joins beacons 1 and 3. The extremity side is the part of the straight line defined by beacons 1 and 2 which origin is beacon 1 and does not go by beacon 2.
10. $\gamma = \sigma - \lambda_{31}$
11. $\tau = \tan^{-1}\left[\dfrac{\sin\lambda_{12}\cdot(L_{12}\cdot\sin\lambda_{31} - L_{31}\cdot\sin\gamma)}{L_{31}\cdot\sin\lambda_{12}\cdot\cos\gamma - L_{12}\cdot\cos\lambda_{12}\cdot\sin\lambda_{31}}\right]$
12. If $\begin{cases}\lambda_{12} < 180^\circ \\ \tau < 0^\circ\end{cases}$ then $\tau = \tau + 180^\circ$
13. If $\begin{cases}\lambda_{12} > 180^\circ \\ \tau > 0^\circ\end{cases}$ then $\tau = \tau - 180^\circ$
14. If $\tau = 0^\circ \wedge \left[\begin{cases}\sigma > 0^\circ \\ \lambda_{31} > 180^\circ\end{cases} \vee \begin{cases}\sigma < 0^\circ \\ \lambda_{31} < 180^\circ\end{cases}\right]$ then $\tau = 180^\circ$
15. If $|\sin\lambda_{12}| > |\sin\lambda_{31}|$ then $L_1 = \dfrac{L_{12}\cdot\sin(\tau + \lambda_{12})}{\sin\lambda_{12}}$
16. else $L_1 = \dfrac{L_{31}\cdot\sin(\tau + \sigma - \lambda_{31})}{\sin\lambda_{31}}$
17. $x_R = x_1 - L_1\cdot\cos(\phi + \tau)$
18. $y_R = y_1 - L_1\cdot\sin(\phi + \tau)$
19. $\theta_R = \varphi + \tau - \lambda_1$
20. If $\theta_R \leq -180^\circ$ then $\theta_R = \theta_R + 360^\circ$
21. If $\theta_R > 180^\circ$ then $\theta_R = \theta_R - 360^\circ$

Fig. 2. Generalized Geometric Triangulation.

III. POSITION AND ORIENTATION ERRORS

In general, the *computed position* does not coincide with the *true position*[2] of the robot and the *computed orientation* also does not coincide with its *true orientation*. There are a *position error* and an *orientation error*, which – in this paper – have the following definitions:

- the *position error* ΔP_R (Fig. 3) is the distance between the *computed position* P_{Rc} and the *true position* P_R;
- the *orientation error* $\Delta\theta_R$ is the modulus of the difference between the *computed orientation* θ_{Rc} and the *true orientation.*

Measurement errors constitute the main source of position and orientation errors, which magnitude also depends on the position of the robot relatively to the beacons.

IV. SIMULATIONS RESULTS

In order to validate the improved version of the Generalized Geometric Triangulation algorithm, five tests were made in a simulation environment. The code was written in Java 2. It was used the *Java 2 SDK, Standard Edition* (version 1.3 for *Windows*)*,* upgraded with *Java 3D* (version 1.2.1 Beta, for *Win32/DirectX*), on a personal computer equipped with a *Intel Pentium III* (995MHz) processor and running *Windows XP* (version 5.1.2600). Graphics were plotted with *Matlab* (version 5.2).

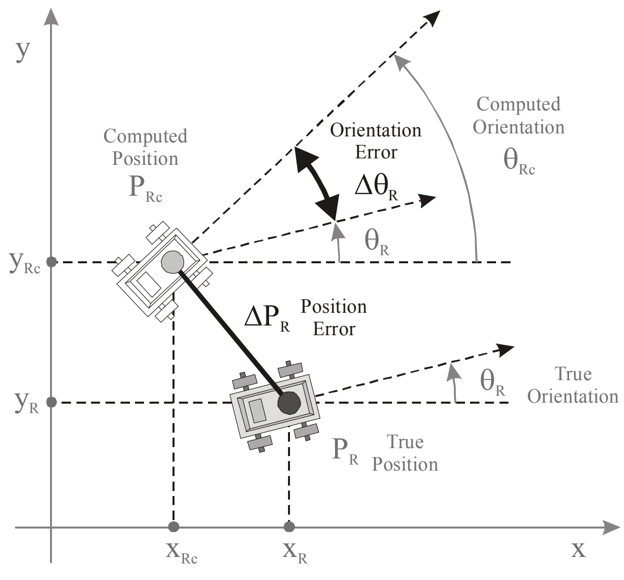


Fig. 3. Position error and orientation error.

[2] The position of a robot with non-negligible dimensions is the position of one of its points.

Simulations are performed in a square shaped area of the navigation plane, using the following beacon configurations:

- non-collinear beacons ordered in counterclockwise fashion;
- non-collinear beacons ordered in clockwise fashion;
- collinear beacons, beacon 1 between beacons 2 and 3;
- collinear beacons, beacon 3 between beacons 1 and 2;
- collinear beacons, beacon 2 between beacons 1 and 3.

In each test, three beacons labeled 1, 2 and 3 are placed in known positions of a Cartesian plane. Beacon positions are printed close to the results of each test. A robot is placed at the origin of the referential system. Its orientation is arbitrarily set to a value between -180º and 180º. Then, a four-step sequence (Fig. 4) is performed:

I. Angles λ_1, λ_2 and λ_3 are computed from *a priori* known beacons and robot positions.

II. Angles λ_1, λ_2 and λ_3 are rounded to integers, simulating the outputs of a digital goniometer with resolution ρ equal to 1º (measurement uncertainty $\pm\Delta\lambda$ equal to ±0,5º).

III. *A priori* known beacons positions and the rounded values of λ_1, λ_2 and λ_3 are used as inputs of the Generalized Geometric Triangulation algorithm, which computes both position and orientation of the robot.

IV. Position and orientation errors are computed from *a priori* known robot position and orientation and their computed values.

The four steps are repeated for robot positions covering a 100 x 100 square. Position increments of 0.1 are made in both *x* and *y* directions. In each point, robot orientation is arbitrarily set to a value between -180º and 180º. Double precision (64 bits) has been used in all four steps.

Position and orientation errors obtained in each position are plotted in 2D graphics (Fig. 5) using a grayscale such that a point becomes darker as the error verified in that position increases. The axes of position error graphics are labeled in the same length units. The *z*-axes of orientation error graphics are labeled in degrees.

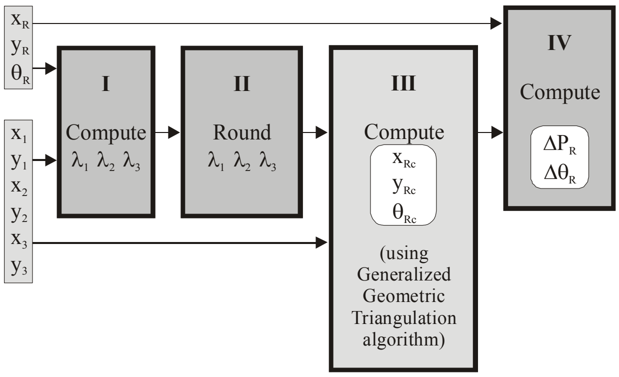


Fig. 4. Four-step sequence used in the five tests.

Simulations results show that the improved version of the Generalized Geometric Triangulation algorithm works with the five beacon configurations used. As expected, significant position and orientation errors occur when the robot is over or close to the circumference defined by three non-collinear beacons or the line defined by three collinear beacons.

V. CONCLUSIONS

Two restrictions are common to all algorithms that perform self-localization through triangulation:

1. The robot must "see" at least three distinguishable beacons to localize itself in a plane. All areas of the plane with less than three visible beacons are unsuitable for robot localization;
2. Localization is not possible if the robot is over the circumference defined by three non-collinear beacons or over the line defined by three collinear beacons.

The Generalized Geometric Triangulation algorithm suffers only from the two restrictions mentioned above. But it is unable to compute position and orientation when the robot is over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2. It was assumed that, when a beacon becomes between the robot and another beacon, the closest beacon hides the farther one or else the goniometer is not able of simultaneously detecting more than one beacon. Any of those situations prevents self-localization, but the impediment is due to the technology used, not triangulation itself.

An improved version of Generalized Geometric Triangulation algorithm was presented. It works over the segment of the line that goes by beacons 1 and 2 whose origin is beacon 1 and does not contain beacon 2.

Simulations were performed using non-collinear beacons ordered in counterclockwise fashion, non-collinear beacons ordered in clockwise fashion, collinear beacons with beacon 1 between beacons 2 and 3, collinear beacons with beacon 3 between beacons 1 and 2, and collinear beacons with beacon 2 between beacons 1 and 3. Results show that the improved version of the Generalized Geometric Triangulation algorithm works with the five beacon configurations used. Significant position and orientation errors only took place when the robot was over or close to the circumference defined by three non-collinear beacons or the line defined by three collinear beacons.

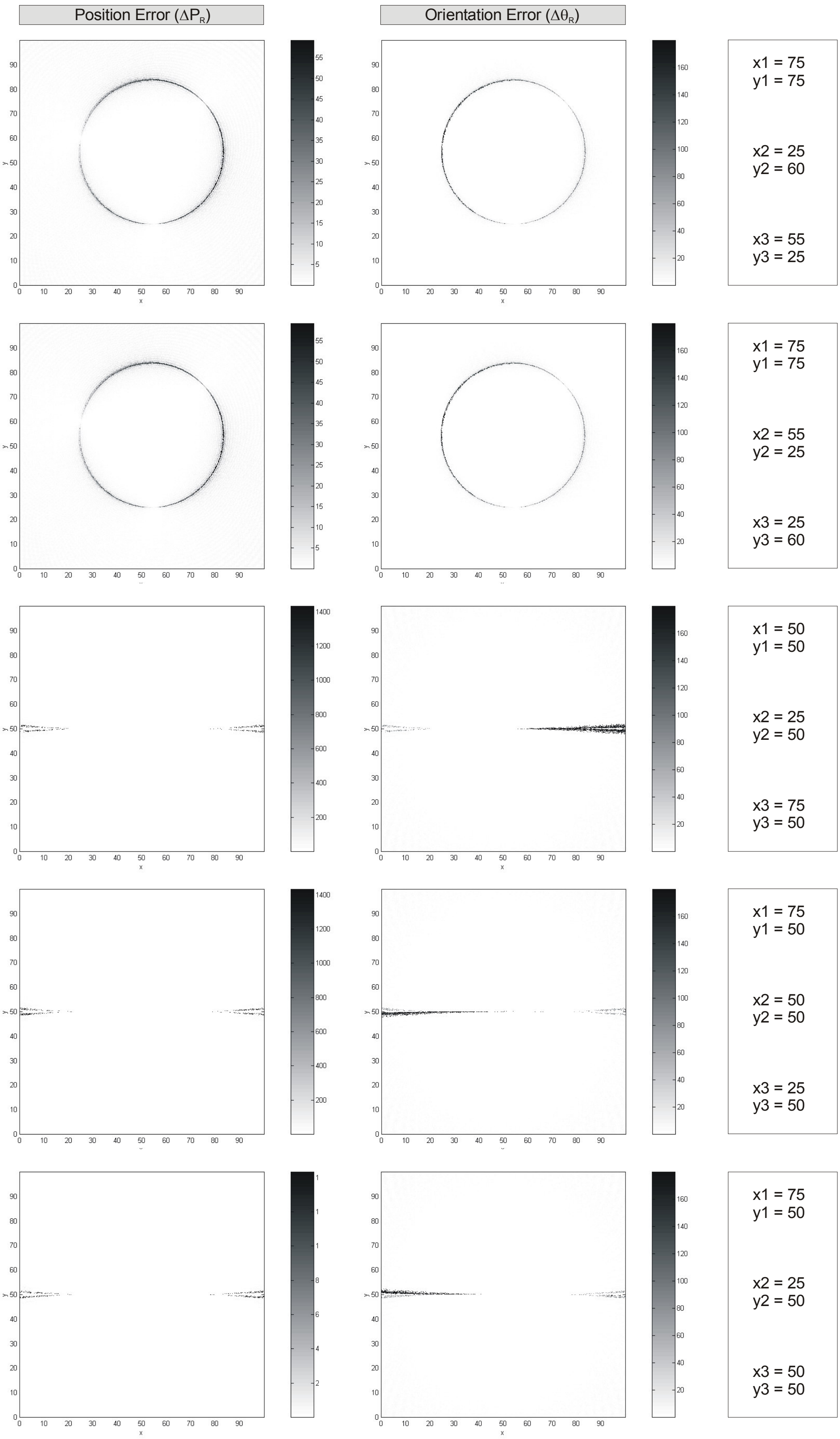


Fig. 5. Position and orientation errors.

VI. REFERENCES

[1] João Sena Esteves, *Metodologia de Autolocalização Absoluta em Ambientes Quase-Estruturados (PhD Thesis),* University of Minho, July 2005.

[2] João Sena Esteves, Adriano Carvalho and Carlos Couto, *Generalized Geometric Triangulation Algorithm for Mobile Robot Absolute Self-Localization;* ISIE 2003 – 2003 IEEE International Symposium on Industrial Electronics**,** Rio de Janeiro, Brazil, June 9–12, 2003.

[3] P. J. McKerrow, *Introduction to Robotics,* Addison-Wesley Publishing Company, Inc., 1991.

[4] John J. Leonard and Hugh F. Durrant-White, *Directed Sonar Sensing for Mobile Robot Navigation,* Kluwer Academic Publishers, 1992.

[5] J. Borenstein, H. R. Everett and L. Feng, *Where am I? Sensors and Methods for Mobile Robot Positioning* (Technical Report), The University of Michigan, 1996.

[6] Cyril Cauchois, Eric Brassart, Bruno Marhic and Cyril Drocourt, *An Absolute Localization Method using Synthetic Panoramic Image Base,* Proceedings of the OMNIVIS'02 - Third Workshop on Omnidirectional Vision, p. 128-135, June 2, 2002.

[7] Jens-Steffen Gutmann, *Markov-Kalman Localization for Mobile Robots,* ICPR'02 – 16th International Conference on Pattern Recognition, Vol. 2, Quebec, Canada, August 11 – 15, 2002.

[8] T. Venet, T. Capitaine, M. Hamzaoui, and F. Fazzino, *One active beacon for an indoor absolute localization of a mobile vehicle,* Proceedings of the 2002 IEEE International Conference on Robotics and Automation, p. 1-6, Washington, DC, May 2002.

[9] K. Briechle, and U. D. Hanebeck, *Localization of a Mobile Robot Using Relative Bearing Measurements,* IEEE Transactions on Robotics and Automation, Vol. 20, Nº. 1, p. 36-44, February 2004.

[10] NDC Automation Inc., *Advantage Lazerway™! - Laser Guidance vs. Inertial Guidance* (Technology Brief), USA (980129), 1998.

[11] Paulo Ventura Araújo, *Curso de Geometria,* Gradiva, 1998.

[12] Heinrich Dorrie, *100 Great Problems of Elementary Mathematics: Their History and Solution* (trad. David Antin), Dover Publications, New York, 1965.

[13] Lélis Espartel, *Curso de Topografia (7ª ed.),* Editora Globo, 1980.

[14] Raymond E. Davis, Francis S. Foote, James M. Anderson and Edward M. Mikhail, *Surveying: Theory and Practice (6ª ed.),* McGraw-Hill, 1981.

[15] Francisco Domínguez García-Tejero, *Topografia Abreviada (5ª ed.),* Dossat, Madrid, 1981.

[16] Clare D. Mcgillem and Theodore S. Rappaport, *A Beacon Navigation Method for Autonomous Vehicles,* IEEE Transactions on Vehicular Technology, Vol. 38, nº 3, August 1989.

[17] Charles Cohen and Frank V. Koss, *A Comprehensive Study of Three Object Triangulation,* Mobile Robots VII, SPIE Vol. 1831, 1992.

[18] H. R.Everett, *Sensors for Mobile Robots,* A K Peters, June 1995.

[19] O. Fuentes *et al.*, *Mobile Robotics 1994* (Technical Report 588), The University of Rochester Computer Science Department, Rochester, New York 14627, June 1995.

[20] E. Stella *et al., An Autonomous System for Indoor Structured Environment,* Intelligent Autonomous Systems, U. Rembold *et al.* (Eds.), IOS Press, 1995.

[21] Margrit Betke and Leonid Gurvits, *Mobile Robot Localization Using Landmarks,* IEEE Transactions on Robotics and Automation, Vol. 13, Nº 2, April 1997.

[22] Junhong Ji, Giovanni Indiveri, Paul-Gerhard Ploeger and Ansgar Bredenfeld, *An Omni-Vision based Self-Localization Method for Soccer Robot,* Proceedings of the IEEE IV2003, Intelligent Vehicles Symposium, Columbus, Ohio, USA, June 9 – 11, 2003.