The Dutch AIBO Team Report on RoboCup 2004

Stijn Oomes[1], Pieter Jonker[2], Mannes Poel[3], Arnoud Visser[4], Marco Wiering[5],
Wouter Caarls[2], Stefan Leijnen[5]; Silvain van Weers[5],
Niek Wijngaards[1] and Frank Dignum[5]

January 2005

[1] DECIS Lab, Delft Cooperation on Intelligent Systems
[2] Quantitative Imaging Group, Department of Applied Physics, Delft University of Technology
[3] Human Media Interaction, Department of Computer Science, University of Twente
[4] Intelligent Autonomous Systems, Department of Computer Science, University of Amsterdam
[5] Intelligent Systems Group, Department of Computer Science, University of Utrecht

http://www.dutchaiboteam.nl or http://aibo.cs.uu.nl/

Email: coach@dutchaiboteam.nl

1. Introduction

This document provides a brief report on the Dutch AIBO Team in 2004, its first year of existence. The team setup is described, as are the research plans for 2004. In addition, the results obtained and scores achieved are presented, together with an outlook for 2005.

2. Dutch AIBO Team

At the end of 2003, several institutes in the Netherlands have joined forces and formed *The Dutch AIBO Team.* We are a group of researchers and students from the universities of Amsterdam, Delft, Twente and Utrecht, and the DECIS Lab. Our goal is to stimulate research, teaching, and applications in the fields of artificial intelligence and collaborative robotics in the Netherlands by joining the international RoboCup community.

Our team combines a group of senior researchers with a strong research record and a long experience in robot soccer competitions, with a diverse group of the most talented students in the country. Since we are distributed across the Netherlands, we have chosen a modular architecture that allows us to develop our innovations in parallel. We believe we have chosen a promising approach to the AIBO team architecture which will position us as a serious competitor.

2.1 Team Partners

We consider our project to be a long term endeavor, so we have set up a professional project organization. The Dutch AIBO Team consists of different sub-teams that all have their specific goals and responsibilities. The **coordination** of all activities is in the hands of the people at the DECIS Lab. Since this lab is itself an open consortium consisting of multiple institutes and companies, there is a lot of experience and know-how on coordinating large multi-partner projects.

The **research** team consists of the authors of this paper and our job is to keep an eye on the long term research goals. We all have well established positions and research programs at our respective institutions. Though using the AIBO as a platform is new for most of us, we have a long joint experience in robot soccer competitions (see Appendix A). Here we summarize our research programs and relevant courses:

University of Amsterdam

The Intelligent Autonomous Systems group studies methodologies to create intelligent autonomous systems, which perceive their environment through sensors and use that information to generate intelligent, goal-directed behavior. This work includes formalization, generalization, and learning of goal-directed behavior in autonomous systems. The focus is on perception for autonomous systems, learning and neuro-computing, principles of autonomous systems and hardware and software systems. In this year’s Autonomous Systems course, 8 students divided into 2 teams worked full-time for 4 weeks on understanding and programming the AIBO ERS-7.

Delft University of Technology

The Quantitative Imaging Group (formerly known as Pattern Recognition Group) studies a wide variety of methodologies and applications in the field of image processing and pattern recognition. Within the fields of industrial inspection and robot vision, we focus on the themes "vision based motion & motion based vision", "sensor data fusion", and "hardware architectures for real-time imaging". Within the AIBO project we have assigned one MSc student (~1 year) on the subject of high speed robust color vision. Furthermore a second year BSc project is assigned to the project (4 months, 6 students) with the task to look at: inventory and testing of all available AIBO motions, dynamic system simulations of AIBO motions, and suitability of AIBO's for 1st year programming practical work.

University of Twente

The Human Media Interaction group studies the interaction between men and machines. Computers operate in every day life as universal media machines presenting multi-media information and as communication devices connecting people. The interface is what presents users with information and what allow users to manipulate and command the machine. This has become a prominent topic of concern to researchers and designers. One of our research streams is intelligent agents and collaborative autonomous systems. We use AIBOs in our courses on process control & robotics, on multi agent systems, and in thesis assignments.

University of Utrecht

The main focus of the research group is on logic and multi-agent systems. The group has developed its own agent-programming language, called 3APL, which provides a logical basis for building (communicating) agents. Other areas of research are reinforcement learning, neural networks, and genetic algorithms. Both approaches to AI are currently being tested in a number of different robots. This year's Software Project course has assigned 9 BSc students to work on the motion modules for 20 hours a week, from February until the end of June.

DECIS Lab

DECIS Lab is an open research consortium involving multiple partners and was founded by the universities of Delft and Amsterdam, the Netherlands Organization for Applied Research, and Thales Research & Technology. The focus of DECIS Lab is on Actor-Agent Communities, in which humans and artificial (robotic and software) systems collaborate in complex environments. Our research results not only include theoretical knowledge and models, but also their transfer into actual prototypes and demonstrations. An important objective for the DECIS Lab, therefore, is to situate these research challenges in real-world domains (including dynamic or even chaotic environments). Two large research programs, COMBINED and ICIS are currently active, addressing collaboration among actors and agents in chaotic environments such as crisis management.

2.2 Student participation

Those of us with appointments at universities use the AIBOs as a robot platform in our courses and thesis assignments. This allows us to easily recruit talented students for *The*

Dutch AIBO Team. Our **development** team consists of these (groups of) students that, due to the modular architecture approach, focus on different aspects of the system (see the detailed description in the next section). Students therefore compete against each other to come up with the best solution for a particular problem, e.g. dribbling with the ball. The best code will end up in the next competition version of our overall implementation.

The **competition** team, the people that are actually taking part in RoboCup 2004, is to consist of 9 students and one team leader. The selection of students from the development team is based on the premise that everyone who is joining the team has to bring something special to the team; we use the available slots as a motivator for the students. In this way, we aim to build a proper team of developers that will be able to truly collaborate during the tense days of the competition.

In summary, the strength of our team is that we combine the experience and stability of senior researchers with the enthusiasm and creativity of students. Because of multiple institutions we are able to tap into a large pool of bright students. The activities of our team are embedded both in our research & development programs and our academic curricula.

3. Our Approach

In the spirit of RoboCup, we take full advantage of the availability of the code of last year's contestants. Because of the similar organization of the GermanTeam 2003 (Röfer *et al.*, 2003), we decided to use their approach in our first year to allow us a kick start. Since we have acquired 14 ERS-7 AIBO robots, we had to port the code that was developed for ERS-210 AIBOs to the new platform. In order to gain some wider experience, we also ported the code from CMUPack 2003 (CMU, 2003) to the ERS-7.

After that we have successfully ported the code to the ERS-7, we concentrated on improving the behaviors of the AIBO's). The behavior-tree was simplified, and only tested behaviors were preserved. One of the main tests was attending the German Open 2004 , which allowed to evaluate and benchmark the performance of the available solutions. After optimizing the code to the maximal performance possible without complete redesign, we decided to start early with the technical challenges, because these challenges are the most interesting from a scientific point of view.

The strength of our approach is that we combine the available implementation of the GermanTeam 2003, the experience that we have gained in other leagues, and a pool of talented students that are already well informed about the ins- and outs of both the German and the CMU code. The modular approach allows us to develop and improve components of the architecture in different labs simultaneously.

4. Our Results

In 2004, the Dutch AIBO Team competed at the German Open 2004 in Paderborn, Germany, and in RoboCup 2004 in Lisbon, Portugal.

When we arrived at the German Open 2004, our robots had never played any official soccer game ever. We nevertheless managed to win one game; we won 5-1 against the Italian *S.P.Q.R. Legged.* We therefore ended 3rd out of the 4 teams in our group. We lost the quarter finals against the *Hamburg Dog Bots* with 6-0. In the final of the German Open 2004 the *Aibo Team Humboldt* beat the *Darmstadt Dribbling Dackels* with 2-1 to become the champion.

Our performance at RoboCup 2004 turned out to be somewhat better. In the soccer competition, we came out 4th in our group of 6 teams with a positive goal difference of 5 (16 goals scored, 11 against us). We lost 3 games and managed to win 2. In the end the *German Team* won the final match from *UTS Unleashed!,* a team that had beaten us with a mere 4-2.

We did even better in the Challenges; we ended 6th out of 19 teams. In the Open Challenge we impressed the judges by showing our AIBO's performing very robust sound localization [note: the original algorithm was developed by TU Delft students Marlon Richert, Tijmen Roberti, and Wouter de Vries, and was adapted for demonstrations by Wouter Caarls]. We came in 2nd after the German Team with 22 points. The Landmark Challenge gave us no points, although we came near most of the required locations. Only 5 teams scored any points at all in this challenge. In the final Variable Lighting Challenge we won 19 points which positioned us at the 5th place. Overall we earned 41 points which gave us the aforementioned 6th position.

The rules for pre-qualification of the next RoboCup state that all the quarter finalists are pre-qualified plus the 4 runners-up in the Challenges. Because *UTS Unleashed!* and *rUNSWift* were both quarter finalists and 1st and 4th in the Challenges, they made our prequalification possible for RoboCup 2005 in Osaka, Japan.

Figure 1: The Dutch Aibo Team in Lisbon, 2004.

Overall, we are quite pleased with our accomplishments, especially given the fact that our robots arrived just before Christmas 2003 and we had only 6 months to get ready for the competition and challenges at RoboCup 2004. These results depend on the progress achieved by our partners, as described below. The finishing touch to the code happened during the competitions in Paderborn and Lisbon, in which students from multiple partners were involved (see figure 1).

4.1 University of Amsterdam

The research at the University of Amsterdam is focused on the different aspects of distributed cooperating systems, like fusing distributed sensor information, representation issues such as shared dynamic world models, and the optimization of the action planning. Because of noise, uncertainty and time dependency, which characterizes real sensor information, computer science methodologies are investigated, which can deal with that adequately. A case study is Robot Soccer, as for instance the 4-Legged League (Groen, Spaan & Kok, 2004).

System Architecture Selection

In January 2004 eight Master students of the UvA were assigned the task to select the most promising System Architecture for the Dutch Aibo Team for the coming years. Half of them concentrated their efforts on the German Team (Ottens, Abbo, van der Meer & Stienstra, 2004). The other four analyzed the team that beat the German Team in the quarter final in Padova, CMU (de Oude, van Erven, Liem & van Kasteren, 2004). This was partly a literature study, but also the software architecture's flexibility was tested by modifying both code bases to facilitate the new Aibo ERS-7 model. This was a thorough study, as can be seen for the comment of Scott Lenser (2004), one of the Ph.D. students of the Carnegie Mellon University:

> "*Overall, I think the evaluation is a very fair representation of the capabilities of the 2 teams.*
>
> *At RoboCup we had a penalty shootout with the German Team. One thing we noticed is that we were able to see the ball from a little further away than they were. I believe the CMUPack vision system is slightly better but having not run the German code I can't be sure.*
>
> *You missed some trade offs in the motion systems for the two teams. The German kick system encodes a state machine for the transitions between motions which allows it to represent a few things the CMUPack code cannot. The CMUPack code, though, allows motions to be developed in terms of either angles or positions.*"

Based on the following arguments, the system architecture of the German Team was chosen. The code is clean and is highly modular. Graphical simulators are included that can be used for monitoring and debugging purposes. Elaborate documentation is written and the code itself is well documented also, allowing a fast learning curve for people unfamiliar with the project. XABSL facilitates creating complex extensible agent behavior solutions instead of just using C++.

Color invariance

Figure 2: the official RoboCup ball under variable lighting conditions.

In June 2004 one Bachelor student started a study on color invariance algorithms for the Variable Lighting Challenge (Pieterse, 2004), see Figure 2. In this study a comparison was made between algorithms based on a Gaussian color model (Geuzenbroek, 2000) or a color-histogram (Zivkovic & Kröse, 2004). The conclusion of this study was that the algorithm based on color-histograms is applicable on the Aibo platform. The computational resources on the ERS-7 are clearly not enough to facilitate an algorithm based on a Gaussian color model.

4.2 Delft University of Technology

We have researched determining the azimuth of an observed sound source (Datcu, Richert, Roberti, de Vries & Rothkrantz, 2004). For sound source localization, the azimuth is defined as being a horizontal direction expressed as the angular distance between the observer's orientation (the ERS-7's, in this case) and the direction of the sound source. This azimuth is determined by using the inter-aural time delay (ITD). This is the difference in arrival time of the sound source's signal between the left ear and the right ear. To be able to discern multiple sound sources from one another and facilitate the distinction between signal and noise, we perform the sound source localization on a number of separate, predefined frequencies only.

The frequencies are isolated using a fast Fourier transform, and the inter-aural time delay is calculated using a cross-correlation on those frequencies. A point-base sound source model is then used to convert the ITD to the azimuth, and a final deconfusion step solves the 180 degree ambiguity by referencing the difference in azimuth to the movement of the head.

The theory requires that the robot does not move during a sample period, but in practice a small movement (in our case, walking toward or away from the sound source) is compensated by the fact that in a coupled system, the introduced errors are more or less filtered out by the slow reaction time of the walking engine.

4.3 University of Twente

This year's contribution to the software used in the RoboCup competitions was minimal.

4.4 University of Utrecht

The main points of focus of the Utrecht University research group are agent cooperation and agent architectures. They have designed their own agent programming language, called 3APL, to implement logic based architectures. Eventually, this language might be used for playing soccer with the AIBO's. A group of students investigated the current cognitive architecture, which is written in XABSL, an XML-based agent programming language (see Röfer et al., 2003). The students worked on extending the model with additional states as the current model did not dynamically switch between different playing-modes (e.g. defensive play) when the score changed. Corresponding meta-states were added to the model. Finally the efficiency of the existing decision tree was addressed. This inefficiency was probably due to the fact that the software had been ported from the old ERS-210 robot to the new ERS-7. The decision tree was restructured in order to reduce computational resources.

Adding new states

When a human soccer team plays a game, its behavior changes according to the current score on the board. If the match is nearing the end and the team needs a goal to stay in the tournament, they usually switch to an offensive stance, e.g. the goalie becomes a striker. If the team leads by one goal, they might switch to defense. If the lead is already decisive and the time is almost up, they can even decide to entertain the public with some gallery play. As usual, these examples from the real world can give us ideas for improving our robot soccer players. The XABSL state machine that is currently used to describe behaviors allows for these kinds of meta-states to be added. Parameters like 'time left', 'score' and 'match type' (e.g. friendly match, tournament final) may influence the transition between states. In the RoboCup 2004 software, we have implemented several of these new behaviors.

Dynamical Landmark Challenge

This challenge entails self location using randomly placed and colored landmarks. In presence of the original and new landmarks the aibo is given one minute to recognize and store the new landmarks. Then the original landmarks and goals are covered or removed, and the aibo has to visit five random positions on the field relying on the new flags for self localization.

Because of the fact the not all information about the dynamical landmark challenge was available during the time of development we made some assumptions about the landmarks. The landmarks

- are rectangles or near rectangles,
- have two colors,
- the colors are oriented the same way as the default landmarks: a top color and some bottom color, and
- they look the same no matter under which horizontal angle they are examined.

The development of the challenge was implemented in two tasks:

- Flag finding: finding potential flags in the environment
- Flag recognition: recognition and storage of determined flags

Flag Finding

The *FlagFinder* is contained in the *DAT3ChallengeImageProcessor* together with the *DT2004ImageProcessor*. During the first part of the challenge both implementations of the image processors are actually executed. The FlagFinder uses roughly the same approach as the original *DT2004 Imageprocessor*.

Scanning is done in two phases using scanlines perpendicular to the horizon. During the first phase widely separated scanlines are used to determine the extent of the white boarding around the field. This is done to prevent the robot from trying to recognize flags in the background or on the field itself. The goal is to try to find two points on every scanline which correspond to the top of the boarding and the bottom of the boarding. The scanline is scanned from top to bottom and a toppoint is found when at least *10* consecutive white pixels are found. A bottompoint is found when a toppoint has been found before and when for the last time the white area ends with at least *3* non-white pixels below it or when the white area extends all the way to the bottom of the image.

At the end of the first phase these top- and bottompoints have to be processed to account for outliers. When only a toppoint is found the point is ignored. We expected that the points found all lie on a relatively straight line so points which do not lie on this line could be considered outliers. There are however exceptions to this rule: the goals have to be disregarded so bottompoints lying on the top of a goal have to be regarded as correct even when they don't lie on a straight line with the rest of the points. The solution for this problem is to cluster the points based on the direction from the current point to the next. The largest cluster is taken to be the correct slope, which should be roughly parallel to the horizon but this is not required. Points which, together with their neighbor, lie in this cluster are considered correct while points which do not are disregarded. These outliers as well as missing bottompoints are replaced with calculated points by interpolating points on nearby scanlines.

During the second phase more scanlines are used which lie between the scanlines from the first phase which are also scanned again, this time only from toppoint to bottompoint. The toppoints and bottompoints on the secondary scanlines are calculated by interpolating the points found during the first phase. We make the assumption that one scanline crosses only one flag. So for every scanline, if a possible landmark is found, only one top- and bottompoint pair is stored. We cope with noise by requiring that landmarks be at least 5 pixels high and have a white area above and below them.

During the next step we try to find groups of adjacent scanlines which run through a possible landmark. Such a group has to span at least *3* scanlines. Also, both the toppoints and bottompoints are required to lie within a certain range from a line parallel to the horizon. The median point is determined for both the top- and bottompoints, which in their turn form the base of a line parallel to the horizon. If more than *70%* of the points lie more than one sixth of the flagheight away from this line, the flag is disregarded. The average *y* value of the bottompoints together with the *x* value of the leftmost bottompoint form the leftbottom corner of a rectangle which is then considered to define the rough rectangular outline of the landmark found. The topright point is calculated in a similar fashion. The center of this rectangle is also determined.

Finally the found rectangles are cross-checked with rectangles circumscribing landmarks found by the old *ImageProcessor* by checking if the two rectangles intersect or if the center of our rectangle lies within the other rectangle. When no such intersection is found it is considered to be a new valid landmark and the *FlagSpecialist* is called by supplying the centerpixel.

Flag Recognition

The *DATChallenge3FlagSpecialist* receives a point in an image from the *FlagFinder*. First the *FlagSpecialist* calculates the dimensions of the flag the received point is part of. The *FlagFinder* has the responsability to let the *FlagSpecialist* calculate the dimensions for every found flag in the image. If all the dimensions have been calculated we can go on with a color calculation. We made the assumption that every flag has two colors, a top- and bottom color. Based on this assumption we calculate two colors for a flag. This is done by calculating the greatest difference (calculated by taking the sum of the absolute differences of the three color channels) between the average color of two successive rows of the flag. After the calculation of this separating line we can calculate the average of the two colors present in the flag. The top and bottom color of the flag are calculated by taking the median of every color channel and calculating the average of these three colors. In our view it is best to take the median instead of some average for two reasons. First, outliers are automatically filtered out and second, if a flagside contains more than one color we believe we have a larger chance that we can recognize the same flag later (for example a single flagside contains three colors, the median will be one of the three colors the first time and hopefully the second time about the same median will be calculated. On the other hand if a flag as a whole contains more than two colors we can have the problem the border between top and bottom will be calculated differently in two different images). Recognition of a flag is based upon the two calculated colors. If the distance of both colors is below a certain threshold we deduce that the flags are the same.

Results

The method described gave good results in a lab environment with the given assumptions. At the RoboCup event, the procedure gave also quite satisfactory results but of course the assumptions were pretty oversimplified because there were multi-color star-like landmarks and they were positioned outside the border. Though the vision system was proven quite good we missed the 3rd landmark with only 2cm resulting in the 4th place. Though given the execution of the challenge during the RoboCup of 2004 some assumptions about the landmarks should be reconsidered and refined to let the algorithm achieve better results.

4.5 DECIS Lab

DECIS Lab offered organizational support by means of our coach for 2004, Stijn Oomes. His efforts inspired the team and facilitated the coming about of our results. Niek Wijngaards is to fulfill his role in 2005.

5. 4-Legged RoboCup 2005

The Dutch AIBO Team looks back on a fruitful pilot year in which much experience is gained. Our intent is to continue our efforts of conducting research on intelligent autonomous robots and applying our results in future RoboCup Soccer games. Our team description paper for 2005 is to outline our joint research plan, plus our intended modifications to our code from 2004.

Acknowledgements

We wish to thank our RoboCup 2004 team, consisting of Wouter Caarls, Jan Peter Jansen, Andrew Koster, Stefan Leijnen, Floris Mantz, Silvain van Weers, and team leader Stijn Oomes for their achievements. In addition, we are grateful to the support received from our (research) partners, students, sponsors and other enthusiasts.
This report is based on the Dutch AIBO Team description paper of 2004, by Oomes, Jonker, Poel, Visser and Wiering.

References

CMU (2003), *RoboCup 2003 Code*, http://www.cs.cmu.edu/~coral-downloads/legged/code/CMPack03_Release.tar.gz

Datcu, D., Richert, M., Roberti, T., de Vries, W. and Rothkrantz, L.J.M. (2004). AIBO Robot as a soccer and rescue game player, *Proceedings of GAME-ON 2004*, ISBN 90-77381-15-5, pp. 45-49, November 2004. http://mmi.tudelft.nl/pub/dragos/GAME-ON.pdf

de Oude, P., van Erven, T., Liem, J. and van Kasteren, T. (2004). Evaluation of CMPack 2003, University of Amsterdam, 29 January 2004. http://www.science.uva.nl/~arnoud/education/DOAS/2004/Project2004/FinalReport2.pdf

Geuzenbroek, J.M. (2000). *Color and Geometrical Structure in Images*, PhD-thesis University of Amsterdam, November 2000.

Groen, F.C.A., Spaan, M.T.J. and Kok, J.R. (2004). Real world multi-agent systems: information sharing, coordination and planning. In *Proc. of the International workshop Military Applications of Agent Technology in ICT and Robotics*, The Hague, The Netherlands, November 2004. ftp://ftp.science.uva.nl/pub/computer-systems/aut-sys/reports/Groen04wicta.pdf

Lenser, S. (2004), private communication to Tim van Erven and Jochem Liem, 11 June 2004.

Ottens, B., Abbo, A., van der Meer, P.J. and Stienstra, M. (2004). Aibo Project 2004 - German Team Report, University of Amsterdam, 29 January 2004. http://www.science.uva.nl/~arnoud/education/DOAS/2004/Project2004/FinalReport1.pdf

Röfer, T., Burkhard, H-D., Düffert, U., Hoffmann, J., Göhring, D., Jüngel, M., Lötzsch, M., von Stryk, O., Brunn, R., Kallnik, M., Kunz, M., Petters, S., Risler, M., Stelzer, M., Dahm, I., Wachter, M., Engel, K., Osterhues, A., Schumann, C. and Ziegler, J. (2003). *GermanTeam RoboCup 2003*, http://www.germanteam.org/GT2003.pdf

Pieterse, C.A.M. (2004). *Kleur invariantie voor de RoboCup challenge - onderzoek naar een belichtingsinvariante methode voor object herkenning*, Bsc-thesis University of Amsterdam, in Dutch, June 2004. http://www.science.uva.nl/research/ias/research/multi_agents/robocup4legged/KleurInvariantieRobocupChallenge.pdf

Zivkovic, Z. and Kröse, B. (2004). An EM-like algorihm for color-histogram-based object tracking, In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2004. ftp://ftp.wins.uva.nl/pub/computer-systems/aut-sys/reports/Zivkovic04cvpr.pdf

Appendix A – Competitions

University of Amsterdam

- Soccer Simulation League (team: UvA Trilearn)
  RoboCup 2003, 1st place
  American Open 2003, 1st place
  German Open 2003, 1st place
  RoboCup 2002, 4th place
  German Open 2002, 1st place
  RoboCup 2001, 4th place
  German Open 2001, 5th place
  RoboCup 1999, 9th place
  RoboCup 1998, 3rd place
- Rescue Simulation League (team: UvA C2003)
  RoboCup 2003, 16th place
  German Open 2003, 2nd place

Delft University of Technology & University of Amsterdam

- Soccer Middle Size League (team: Clockwork Orange)
  German Open 2003, quarter final
  German Open 2002, 4th place
  RoboCup 2001, quarterfinal
  German Open 2001, quarterfinal
  European Championship 2000, quarterfinal

University of Twente

- Mirosot *Middle Size League* (team: MiroSot?)
  FIRA World Championship 2004
  FIRA European Championship 2003, 4th place

University of Utrecht

- Rescue Simulation League (team: BanzAI)
  RoboCup 2003, 18th place

DECIS Lab & University of Amsterdam

- Robot Rescue League (team: Zeppelins)
  RoboCup 2003, Round Robin

Appendix B – Relevant Publications

J. Kok (2003), Multi-robot decision making using coordination graphs, *Proceedings of the 11th International Conference on Advanced Robotics*, Coimbra, Portugal, 2003.

F.C.A. Groen, M.T.J. Spaan, and N. Vlassis (2002). Robot soccer game or science. In M. Ivanescu, editor, *Proceedings CNR-2002*, p. 92-98. Editura Universitaria Craiova, October 2002. ISBN:973-8043-165-5.

Nikos Vlassis and Matthijs T. J. Spaan (2004). A fast point-based algorithm for POMDPs. In *Benelearn 2004: Proceedings of the Annual Machine Learning Conference of Belgium and the Netherlands*, pages 170-176, Brussels, Belgium, January 2004.

Roland Bunschoten and Ben Kröse (2003). Visual odometry from an omnidirectional vision system. In *Proceedings of the International Conference on Robotics and Automation ICRA'03*, pages 577-583, Taipei, Taiwan, 2003. ISBN 0-7803-7737-0.

J. Caarls, P.P. Jonker, and S. Persa (2003), Sensor Fusion for Augmented Reality, in: Emile Aarts, Rene Collier, Evert van Loenen, Boris de Ruyter (eds.), *Ambient Intelligence (Proc. 1st*

European Symposium EUSAI 2003, Veldhoven, Netherlands, Nov.3-4), Lecture Notes in Computer Science, vol. **2875**, Springer Verlag, Berlin, 2003, 160-176.

W. Caarls, P.P. Jonker, and H. Corporaal (2003), Benchmarks for SmartCam Development, *Proceedings of Acivs 2003, Advanced Concepts for Intelligent Vision Systems* (Ghent, Sep.2-5), Ghent University, Ghent, B, 2003, 81-86.

P.P. Jonker and W. Caarls (2003), Application Driven Design of Embedded Real-Time Image Processors, *Proceedings of Acivs 2003, Advanced Concepts for Intelligent Vision Systems* (Ghent, Sep.2-5), Ghent University, Ghent, B, 2003, 1-8.

P.P. Jonker, S. Persa, J. Caarls, F. de Jong, and R.L. Lagendijk (2003), Philosophies and technologies for ambient aware devices in wearable computing grids, *Computer Communications*, **26**(11):1145-1158, (Special Issue on Ubiquitous Computing, Edited by T. Pfeifer), 2003.

Jelle R. Kok, Matthijs T.J. Spaan, and Nikos Vlassis (2003). Multi-robot decision making using coordination graphs. In A.T. de Almeida and U. Nunes, editors, *Proceedings of the 11th International Conference on Advanced Robotics, ICAR'03*, pages 1124-1129, Coimbra, Portugal, June 30-July 3 2003. IEEE Press. ISBN 972-96889-9-0.

S. Persa and P.P. Jonker (2001), Real-time computer vision system for mobile robot, in: David P. Casasent, Ernest L. Hall (eds.), *Intelligent Robots and Computer Vision XX: Algorithms, Techniques, and Active Vision* (Proc. Conf. Boston, USA, Oct.28-Nov.2), Proc. SPIE, vol. **4572**, 2001, 105-114.

J.M. Porta and B.J.A. Kröse (2003). Vision-based localization for mobile platforms. In E. Aarts, R. Collier, E. van Loenen, and B.D. Ruyter, editors, *Proceedings of the First European Symposium on Ambience Intelligence (EUSAI)*, pages 208-219, Eindhoven, The Netherlands, November 2003. Springer. ISBN 3-540-20418-0.

Josep M. Porta, Bas Terwijn, and Ben Kröse (2003). Efficient entropy-based action selection for appearance-based robot localization. In *Proceedings of the International Conference on Robotics and Automation ICRA'03*, pages 2842-2847, Taipei, Taiwan, 2003. ISBN 0-7803-7737-0.