TopLeague and Bundesliga Manager: New Generation Online Soccer Games

Ubbo Visser

Department of Computer Science
University of Miami
1365 Memorial Drive
Coral Gables, FL, 33146, USA
visser@cs.miami.edu

Abstract. This paper describes a new generation online soccer manager. More than 210,000 users operate TopLeague and the Official Bundesliga Manager, complex real-time soccer simulators that are based on actual data of real professional soccer players. The underlying technology is a hierarchical three-tier multiagent system that consists of autonomous BDI agents and allows dynamic group structures (e.g. an emergent situation for a wing attack). The German Bundesliga, one of the most prestigious soccer leagues in the world, adopted this AI system for their official web site. The online games run seamlessly in a web browser with a state-of-the-art 3D visualization engine. Fundamental research within the domain of the RoboCup simulation league is the basis for this technology. We will describe the architecture of the multiagent system (MAS) in this paper, discuss motion capture techniques for graphical animation, and reveal details about user acceptance of the games.

Keywords: Robotic Entertainment, Teamwork and Heterogeneous Agents, Motion Capture, Facial Expressions, Soccer Simulation, Manager Games.

1 Introduction and Related Work

A vast amount of games have been developed for the mass market over the past decade. Not only can we observe more than ten types of games (e.g. adventure games, casual games, sports games [1,22]), we can also state that most of the games have been produced for specific hardware (e.g. consoles, game consoles, Wii consoles). Online games have also been developed in great numbers [10], however, web browser restrictions prevented them from being as complex - especially in their graphical animation - as their console counterparts. One reason for this is the computer's configuration (e.g. real-time rendering, older computers have limited rendering capabilities). Some hardware related features (e.g. vertex shaders) may not work on older machines. Another reason is security (applets for instance cannot change the users configuration without prior permission). Our research reveals that a significant number of users (approx. 10%) have not installed their graphic drivers properly.

Nevertheless, online games have become more and more complex, especially in the last few years. We see a merge between the technologies, e.g. consoles have Internet

J. Ruiz-del-Solar, E. Chown, and P.G. Ploeger (Eds.): RoboCup 2010, LNAI 6556, pp. 230–241, 2011.

access now and can download the latest updates while still having an advantage of a defined hardware. On the other hand, the market share of online games (and mobile applications) are increasing [14]. We can even argue for a paradigm shift as nowadays computers and network bandwidth reach a point where complex real-time online games are possible for the mass market.

The RoboCup simulation league offers a number of research targets. Among them are complex simulation [16,15], real-time situation estimation [4,28,34,12] and behavior of agents [11,35,21]. Starting off as a simulation team in the early years of this decade we generated the idea to combine the new game paradigm shift with autonomous robots and sports data from real professional soccer players. We have developed a simulation engine where the agents comply to weekly updated performance indicators of real players (e.g. accuracy of shooting, tackle performance) and act according to tactical preferences of the user who operates the game.

Fig. 1. Autonomous agents in the Official Bundesliga Manager

Game play. The general idea for the user is to pick a virtual version of a professional soccer team and act as a coach. Games follow the regular season and the official game days follow the game days of the professional leagues. The user adds soccer knowledge for the next game while choosing formation, tactics, and training sessions for his team. Each game is simulated and a newly developed and high quality 3D graphics engine allows the user to see the match in a web browser thereafter (cf. figure 1). The user coaches a real team (e.g. Bayern Munich) and has the same problems to solve than the real coach in the real world (e.g. players that are suspended in the real world are also suspended in the virtual world). The games have the same rhythm than the reality, however, TopLeague games will be played one day earlier.

The paper is organized as follows: the architecture including simulator and graphics engine of the game is presented in the next section. We then describe our animation process discussing our latest research results. We include a section of user acceptance where we discuss feedback and give statistics of the users of this technology.

2 Architecture

All data are held in a relational database (MySQL). A number of modules as middleware have been developed ensuring the scalability of our game. A scheduler is responsible for the list of games that need to be evaluated (see figure 2a).

2.1 Simulator

The simulator is developed for a sports game in general. In this instance, however, it is used to simulate a soccer game. It consists of a soccer environment with the necessary agents (two teams, referee, substitute players). There is no physics involved in the game with one execption: the ball has an accurate model for gravity, air friction as well as bouncing effects on the ground and for the posts. The friction varies with weather conditions, a rainy playground has less friction for example. Each team has optimal vision and accurate information about the world. They can communicate to each other and have a 100% access to each others world model. Each team is organized in a hierarchy. Within the simulation, the team decides about the general strategy (e.g. offense play, wing attacks in general). The agents in the next level (stable team structures such as defense, middle field, forwards) balance out the given general strategy with their own goals (e.g. defense within an attack or distribution of the players within an attack). The lowest level consists of agents that act and react according to the given overall goals and the concrete play situation based on their personal skills.

All agents 'exist' within an agent environment (cf. figure 2b). The environment's changes are sensed by each agent with the help of sensors leading to a world model. A perception of the world induces the reflection model that updates the internal world

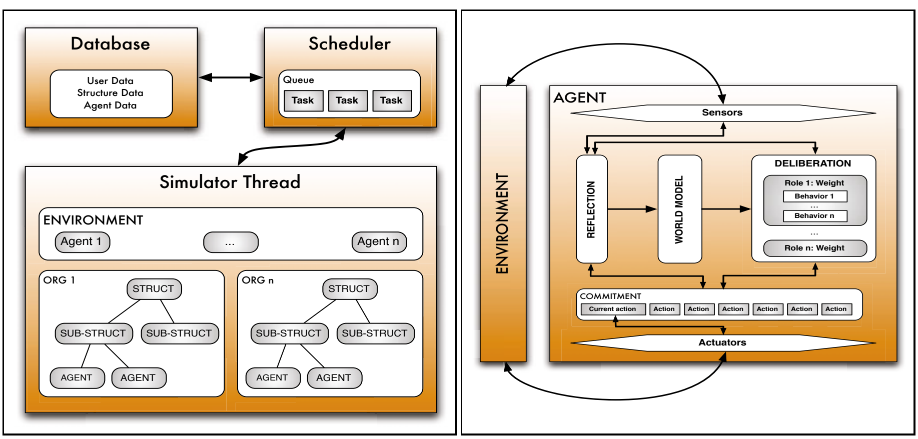

Fig. 2. a) Setup of MAS with MySQL DB and scheduler, b) Agent architecture

model of the agent with the actual perceptions and that validates the current action sequence taken by the agent (current commitment). A new deliberation cycle will be induced if the current action sequence is no longer valid (e.g. change of game situation). Within the deliberation cycle the various roles of an agent will be weighted based on the current situation evaluation. A role of an agent is directly linked to a list of behaviors, which can be used for the calculation of the action selection. A planning process is carried out that calculates the action sequence for the agent.

Input parameters. All real professional soccer players are grouped into five skill groups. The skills are calculated for all players of same position in same group based on Opta Sports Data statistics[1]. After this, we are ranking players within one group. If players don't play due to sick leave or suspension, minimum skills will be taken for those players. Often, bad player skills can be observed (e.g. player haven't played or played but did not score or perform well). In order to rectify this situation we perform small changes in skills randomly.

A general input factor is a random seed for the pseudo random number generator. Team settings include but do not exhaust home team/away team, formation, and heterogenous players. Tactics include team, shooting and passing behavior, side of attack, rules for ball holding and rules for tactical changes throughout the match. The user is also able to define certain roles such as captain, penalty scorer, free-kick and corner-kick player. Player settings includes the level of aggressiveness and deployment.

SPIELERINFO

Torsten Frings		
SA	DEU	
PO	Mittelfeldspieler	MF
SF	Rechtsfuß	
Spielerwerte		
GS		58,3
ZW		52,7
PS		69,0
KO		44,3
SG		54,9
DR		42,4
FL		70,2
ST		56,8
SK		82,0
MO		48,4
FI		35,8
FR		100,0
FO		37,3

SPIELERWERTE

POS	Spieler	GS	ZW	PS	FA	AB	AW	PA	ST	SK	MO	FI	FR	FO
TW	Hans-Jörg Butt	51,4	51,3	44,8	49,9	44,7	53,2	62,6	48,0	80,0	50,0	33,8	100,0	28,3
TW	Alexander Stolz	23,0	19,0	19,0	18,2	19,7	19,7	28,9	48,0	78,0	41,7	34,1	100,0	27,0
TW	Christian Vander	29,4	22,0	25,5	20,2	38,5	27,3	30,9	48,0	78,0	42,1	33,3	100,0	57,6
POS	Spieler	GS	ZW	PS	KO	SG	DR	FL	ST	SK	MO	FI	FR	FO
AW	Leon Balogun	30,6	29,2	18,7	34,1	19,8	20,2	21,1	27,0	79,0	44,1	40,4	100,0	42,6
AW	Jérôme Boateng	60,4	66,0	56,2	64,0	53,6	56,5	61,3	64,0	82,0	48,1	35,8	100,0	52,0
AW	Andreas Görlitz	45,4	50,6	38,1	44,1	35,2	33,7	38,4	48,5	80,0	45,0	35,9	100,0	35,8
AW	Kaká	38,0	38,4	27,9	39,0	29,1	24,6	28,1	44,0	77,0	48,6	34,9	100,0	34,5
AW	Nikola Petkovic	36,9	38,4	28,3	39,0	30,2	23,8	28,3	45,0	79,0	40,0	35,7	100,0	37,6
AW	Marco Russ	51,8	56,4	46,6	55,9	48,6	34,2	51,1	49,0	80,0	46,8	40,1	100,0	32,4
AW	Lukas Sinkiewicz	45,6	51,2	38,2	44,4	33,8	33,3	38,8	49,8	79,0	47,9	35,7	100,0	27,6
AW	Paul Stalteri	39,6	39,3	33,2	39,6	37,1	30,2	29,4	45,0	80,0	43,7	40,0	100,0	36,4
POS	Spieler	GS	ZW	PS	KO	SG	DR	FL	ST	SK	MO	FI	FR	FO
MF	Alexander Baumjohann	55,6	52,0	62,8	51,3	38,7	50,2	63,0	63,0	81,0	40,1	35,4	100,0	43,8
MF	Pal Dardai	52,0	51,8	59,7	50,3	39,6	43,2	54,0	44,0	77,0	40,2	35,8	100,0	35,4
MF	Torsten Frings	58,3	52,7	69,0	44,3	54,9	42,4	70,2	56,8	82,0	48,4	35,8	100,0	37,3
MF	Thomas Hitzlsperger	60,4	55,0	66,8	58,0	58,0	43,9	70,7	68,6	81,0	48,4	35,3	100,0	24,4
MF	Lennart Hartmann	31,1	19,0	30,3	19,0	20,0	29,3	28,4	20,4	78,0	39,8	35,9	100,0	43,0
MF	Mesut Özil	62,7	52,5	73,5	47,4	61,8	57,7	74,3	59,3	82,0	47,8	40,4	100,0	20,7
MF	Petit	57,8	54,8	67,5	53,0	51,1	45,3	69,9	55,4	83,0	47,8	36,3	100,0	35,5
MF	Kevin Pezzoni	44,2	37,7	47,1	39,9	37,4	32,2	47,5	34,3	80,0	39,7	35,6	100,0	20,3
POS	Spieler	GS	ZW	PS	KO	SG	DR	FL	ST	SK	MO	FI	FR	FO

Fig. 3. Example for skills, GS is overall strength; ZW is tackle performance etc. The numbers are in %, different colors (or grey-colors) show skill classes, green (light grey) is a good classification, red (dark grey) is a negative classification.

[1] http://www.optasportsdata.com

The player attributes/skills (cf. figure 3) are based on Opta Sports Data. Thirteen skills are used either internally or externally. An example for internal skills that are only used inside our simulator is $penalty\_kick$. An example for the an external skill that is also shown to the end user via web front-end is $duel\_strength$. Here is the equation for the penalty kick:

$$penalty\_kick = \left(\frac{p\_goal + p\_tar}{p\_miss + p\_post + p\_goal} + \frac{rf\_goal + rf\_post + rf\_tar}{rf\_tot} + \frac{lf\_goal + lf\_post + lf\_tar}{lf\_to} \right) / 3. \quad (1)$$

where p= penalty, tar = target, rf = right foot, lf = left foot, and tot=total.

2.2 3D Visualization Engine

The simulator produces a < 250K log-file that contains the positions of the players, the referees and the ball at each given time slot. The simulator runs with 10Hz. The log-file is then stored on one of our servers and can be requested from each user via web service. Technically, the log-file is requested by an applet on the client side that shows the game and various resolutions/options that are tailored for a variety of graphic cards.

All animation scenes are visualized in this graphics engine. It has been developed from scratch and guarantees high quality pictures in a web browser. The kernel is also domain independent and carries most of the features. Another module contains the 3D model for the stadium, scenes, textures, animations and interaction GUI for the soccer stadium. FaceIntegrator, developed in-house, is a tool that contains scenes and textures among 3D models. It also contains an implementation for the generation and integration of real soccer player images for our autonomous agents (cf. figure 8 on page 238).

The graphics engine contains features that include static and dynamic models, (hand-made) animations, skeletons (soon also bipeds), terrain description, object motions for animations, motion differences for animations, bezier curves, vertex-, fragment-, and geometry shader, potential fields, and some configuration files. A detailed description of features would be out of scope of this paper. However, we would like to point out two important features in this paper: the bone/joint system and the dynamic scene generation.

2.3 Quaternion Based Bone/Joint System

A quaternion-based bone/joint system allows a biunique interpolation of object rotations in 3D space. This is not possible within the Euler space because different object rotations can be combined in arbitrary orders and may end up in the same end-rotation position. Quaternions and their biunique interpolation characteristics allow us to fluently animate bone systems of dynamic objects with the help of a few key frames. An advantage is that we can animate objects at any given time independently from the frequency rates. The animated objects can be rendered in real time with software and hardware (shaders). Figure 4 shows our level-of-detail dependent on the use of the bone systems. Objects in the foreground show more bones as objects in the background. In this scene, players and banners are animated objects. Note that the level-of-detail does not show the hands of players in the background.

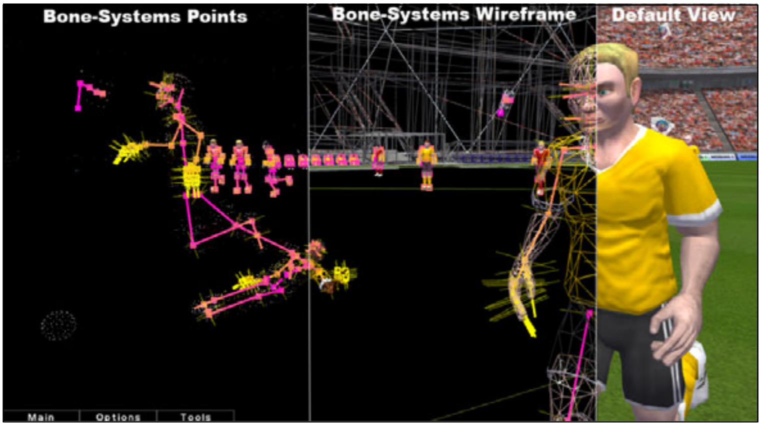

Fig. 4. Quaternion based bone/joint system with level-of-detail: the players in the background do not have as many bones as teh players in the foreground

2.4 Dynamic Scenes

Users have the option to differentiate between various camera settings. All cameras can be used with a zoom. The cameras focus on an area close to the ball. In addition to the ball position in the game, more ball positions are estimated for potential future ball positions. This gives an illusion that the 'camera man' would take the scene live and would not know future ball positions. All ball positions are smoothened by cubic curves (Bezier). Using curves allows us a 3rd degree interpolation, which is significantly more dynamic than a 1st degree interpolation (linear interpolation). The soccer simulator marks special events such as goals, fouls, offside situation etc. These marks are the basis for a game situation evaluation which creates a suitable camera path. Figure 5 shows two different camera paths. The upper row shows a foul situation, the lower row a goal situation. Each game situation is unique and creates a unique camera path.

Fig. 5. Dynamic camera scenes: the upper row shows a foul situation, the lower row a goal situation

3 Motion Capture for Body Motions and Facial Expressions

Character animation is an important part of a graphical game engine. The restriction of an applet (or better: the client machine and the applet) does not allow us to create high-resolution models that we can see in console based games. Motion capture techniques are well-known and can be used for character animation.

There are many different approaches in the area of motion capture. Approaches contain non-optical systems such as inertial systems (inertial sensors, biomechanical models and sensor fusion) [33], electromechanical motion [31], and electromagnetic systems [17]. Other techniques are optical. Among these are semi-passive imperceptible markers [29], pure vision-based, markerless techniques [18,9] as well as active markers [37].

The most popular approach to motion capture today is to use reflective markers to the body and/or face of an actor, and track these markers in images acquired by multiple calibrated video cameras [32,13,30]. The marker tracks are then matched, and triangulation is used to reconstruct the corresponding position and velocity information.

Furukawa & Ponce [8] state that any motion capture system is limited by the temporal and spatial resolution of the cameras, and the number of reflective markers to be tracked. Matching becomes difficult with too many markers that all look alike or are getting lost while labeling. They state further that on the other hand, although relatively few (say, 50) markers may be sufficient to recover skeletal body configurations, thousands of markers may be needed to accurately recover the complex changes in the fold structure of cloth during body motions, or model subtle facial motions and skin deformations (see also [36]).

Our system is a mixture of hand-made animations and those that are made with the help of motion capture techniques (reflective markers). Our Motion Capture Lab is equipped with a modern Vicon System. It consist of ten 4-megapixel cameras with 120 Hz, two video cameras for ground truth, four force plates to measure forces on ground, facilities for facial expressions, and an electromagnetic device capture voltage in max 16 muscle groups. Figure 6 outlines the various steps (workflow) that have been performed to get a motion onto animated characters that can be used for a web browser game.

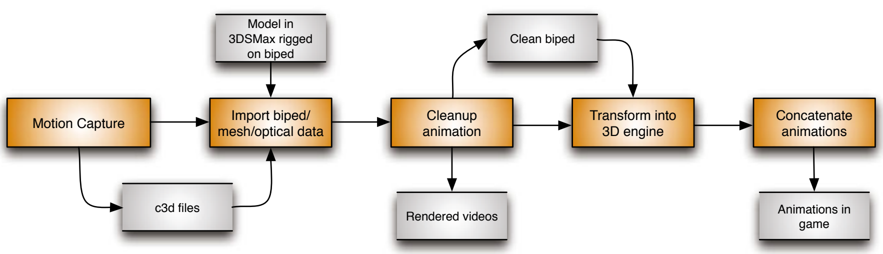

Fig. 6. Motion Capture Process from capturing to animation

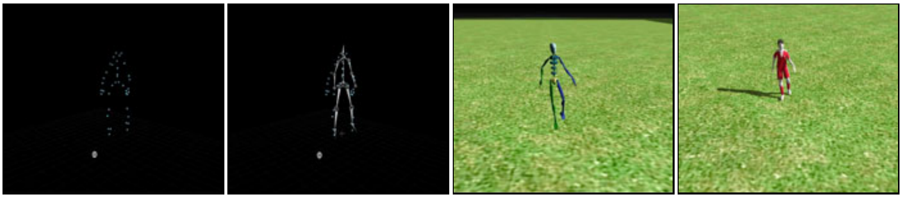

Fig. 7. Motion Capture Steps: a) raw markers in 3D space; b) data cleaned and assigned to bones; c) cleaned again and mapped onto biped; d) fully rendered video

3.1 Modeling Avatars

There are three main approaches to obtain facial models. Firstly, 3D scanning of faces. Face models can be complete representations of a polyhedron that is obtained with the help of the faces' surroundings scanning algorithm. These algorithms can use the viewing sphere and the perspective projection concepts (known as the K-M view space model) [20,19,3]. Secondly, available software products that come with pre-defined models such as FaceGen[2], Poser[3] or DAZ-Studio[4]. FaceGen has been created using statistical modeling of the shape and appearance of human faces. FaceGen uses a similar approach to Blanz's [2], although the implementation differs in the details, and much additional work has been done to add features and allow integration in the character-creation pipeline. Poser offers eight ready-to-pose and textured humans of different ethnicities. Lastly, hand-made models with other software products such as Poly Modeling with 3D Studio Max or ZBrush. The basic idea here is to create characters from scratch using generic spheres from which polygons are used to create the desired shape.

We use existing software tools (e.g. FaceGen) and develop necessary tools (e.g. FaceIntegrator) ourselves (see figure 8 and 9 for the outcome).

3.2 Facial Expression Generation

Much work is being performed to animate believably avatars [24,5,27,25]. To generate facial expressions, the approach taken by many research group has been to use the well known Facial Action Coding system [7] to extrapolate different facial areas that are then independently activated according to predefined pattern [24,23,26]. These approaches address the difficult notion of expressing mixed emotions, but because it is based on Ekman's static encoding of the face, no specific emphasis is given to the dynamics of the facial expression generation. However, as pointed out in a study on communicating facial affect in computer-mediated communication and in affective computing applications, if there is a tradeoff to be chosen between showing highly realistic facial images (i.e. high spacial resolution, full color) at the expense of motion (e.g. low temporal resolution, lags), the motion should be retained as it represents the most important factor in communicating facial affect [6].

[2] http://www.facegen.com

[3] http://my.smithmicro.com/

[4] http://www.daz3d.com

Fig. 8. A soccer player's picture projected on a morph with FaceGen

Fig. 9. Avatars with expressions of emotions within a browser applet

Morph target animation is a method that is used as an alternative to skeletal animation. Morph target animation is stored as a series of vertex positions which are moved to a different position in each keyframe.

Facegen has the ability to easily generate morphs of multiethnic avatars, which is an added advantage for our multi-cultural avatar dialog system. We generated a small number of dynamic facial expressions (see figure 9) using our simple player avatar model, following the dynamic principles for facial expression generation in humans.

4 Results and User Acceptance

This R&D project is ongoing for the last past couple of years. The game has been developed and is available online on *www.bundesliga.de* and *www.topleague.de*. We have currently more than 210,000 users that are signed up and play this game. Since launching in 2007, we have a steady and consistent growth of 500-1,200 users per week. Statistics also prove that a large part of users are coming back regularly. The activation level of our customers (i.e. unique user comes back at least once within that time frame) is shown for 30 and 90 days. The current numbers regarding returning users (30/90) days are: Official Bundesliga Manager (20%, 30%), TopLeague (15%, 40%).These numbers are very competitive for the free market.

To date (April 26, 2010), we have simulated and delivered nearly 9 million unique games (8,746,843 to be exact). 31,5 million page impressions (PIs) have been created by users with 617,162 visits each month for the last 12 months in average. This is a ration of approximately 1/20 visits/PI. In contrast, a regular news web site has a ration of 1/7 or 1/8. This indicates that the average user is staying on our web site quite long.

I detailed analysis with Webalizer[5] and Google Analytics[6] reveals that the average time on site per user is $>$ 12 minutes (time of analysis: Aug 1, 2009 - Nov 15, 2009, daily resolution). The bounce rate is below 8% and thus a very good rate according to Google Analytics ($<$ 20%). We also have approximately 25-30% new visits which indicates that 70-75% are returning users.

User support is an integrated piece of the game. Support is realized with tutorials, context-sensitive help functions as well as user forums. In addition, we have several godfathers, users, that take over work for other users voluntarily. We also have a group of editors that monitor the discussions in the forums and give feedback to the development team.

5 Conclusion

We have described a R&D project that has its roots and the spirit from the RoboCup simulation league. We have developed an online game that is available on the free market. Nearly three years after launching we can state that the proof of concept has been successful. The technical key features of our technology can be identified as a unique AI simulation engine with an authenticity module allowing same player skills in virtual world than in the real world. This is a new paradigm on online sports games and can be seen as the closest soccer simulation game to reality. Another key feature is the visualization which is a 3D state-of-the-art graphics engine for a web browser.

Acknowledgements. We would like to thank the following individuals and organization for their help in order to conduct this R&D project: Dennis Pachur, Dirk Lohwasser, aitainment GmbH, Bremen, Germany. Adam McMahon, Justin Stoecker, Armando Locay, AI & Games Group, Department of Computer Science, University of Miami. Lester Melendez, Florida International University Dr. Asfour, Moataz Eltoukhy, Biomedical Engineering Lab, University of Miami.

References

1. Bethke, E.: Game development and production. Wordware Publishing, Plano (2003)
2. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: SIGGRAPH 1999: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, pp. 187–194. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1999)

[5] http://www.mrunix.net/webalizer/
[6] https://www.google.com/analytics

3. Bowyer, K.W., Chang, K., Flynn, P.: A survey of approaches and challenges in 3d and multi-modal 3d + 2d face recognition. Comput. Vis. Image Underst. 101(1), 1–15 (2006)
4. Choppy, C., Bertrand, O., Carle, P.: Coloured Petri Nets for Chronicle Recognition. In: Kordon, F., Kermarrec, Y. (eds.) Ada-Europe 2009. LNCS, vol. 5570, pp. 266–281. Springer, Heidelberg (2009)
5. De Rosis, F., Pelachaud, C., Poggi, I., Carofiglio, V., de Carolis, B.: From greta's mind to her face: modeling the dynamics of affective states in a conversational embodied agent. International Journal of Human-Computer Studies 59(1-2), 81–118 (2003)
6. Ehrlich, S.M., Schiano, D.J., Sheridan, K.: Communicating facial affect: it's not the realism, it's the motion. In: CHI 2000 Extended Abstracts on Human Factors in Computing Systems, p. 252. ACM, New York (2000)
7. Ekman, P., Friesen, W.: Facial Action Coding System. Consulting Psychologists Press Inc., Palo Alto (1978)
8. Furukawa, Y., Ponce, J.: Dense 3d motion capture for human faces. In: CVPR, Miami Beach (2009)
9. Horaud, R., Niskanen, M., Dewaele, G., Boyer, E.: Human motion tracking by registering an articulated surface to 3d points and normals. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 31(1), 158–163 (2009)
10. Hsu, C.L., Lu, H.P.: Consumer behavior in online game communities: A motivational factor perspective. Computers in Human Behavior 23(3), 1642–1659 (2007)
11. Intille, S.S., Bobick, A.F.: Recognizing planned multiperson action. Comput. Vis. Image Underst. 81(3), 414–445 (2001)
12. Jenkins, O.C., Matarić, M.J.: A spatio-temporal extension to Isomap nonlinear dimension reduction. In: The International Conference on Machine Learning (ICML 2004), Banff, Alberta, Canada, pp. 441–448 (July 2004)
13. Kurihara, K., Hoshino, S., Yamane, K., Nakamura, Y.: Optical motion capture system with pan-tilt camera tracking andreal time data processing. In: ICRA, Washington DC, USA, vol. 2, pp. 1241–1248. IEEE, Los Alamitos (2002)
14. Sector Games Media Board Berlin-Brandenburg. Online study for sector games. Technical report, Media Board Berlin-Brandenburg (2007)
15. Miene, A., Lattner, A., Visser, U., Herzog, O.: Dynamic-preserving qualitative motion description for intelligent vehicles. In: IEEE Intelligent Vehicles Symposium, Parma, Italy, pp. 642–646. IEEE, Los Alamitos (2004)
16. Miene, A., Visser, U., Herzog, O.: Recognition and Prediction of Motion Situations Based on a Qualitative Motion Description. In: Polani, D., Browning, B., Bonarini, A., Yoshida, K. (eds.) RoboCup 2003. LNCS (LNAI), vol. 3020, pp. 77–88. Springer, Heidelberg (2004)
17. Mitobe, K., Kaiga, T., Yukawa, T., Miura, T., Tamamoto, H., Rodgers, A., Yoshimura, N.: Development of a motion capture system for a hand using a magnetic three dimensional position sensor. In: International Conference on Computer Graphics and Interactive Techniques, Boston, MA, p. 102. ACM SIGGRAPH (2006)
18. Moeslund, T.B., Hilton, A., Krüger, V.: A survey of advances in visionbased human motion capture and analysis. Computer Vision and Image Understanding 103(2-3), 90–126 (2006)
19. Mokrzycki, W.S., Salamonczyk, A.: Generating 3d multiview exact polyhedron representation by scanning faces surroundings. MG&V 15(3), 525–536 (2006)
20. Moreno, A.B., Sánchez, A., Frías-Martínez, E., Vélez, J.F.: Three-dimensional facial surface modeling applied to recognition. Eng. Appl. Artif. Intell. 22(8), 1233–1244 (2009)
21. Nakanishi, R., Maeno, J., Murakami, K., Naruse, T.: An Approximate Computation of the Dominant Region Diagram for the Real-Time Analysis of Group Behaviors. In: Baltes, J., Lagoudakis, M.G., Naruse, T., Ghidary, S.S. (eds.) RoboCup 2009. LNCS, vol. 5949, pp. 228–239. Springer, Heidelberg (to appear, 2010)

22. Parberry, I., Kazemzadeh, M.B., Roden, T.: The art and science of game programming. In: Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education, p. 514. ACM, New York (2006)
23. Pelachaud, C., Bilvi, M.: Computational model of believable conversational agents. In: Huget, M.-P. (ed.) Communication in Multiagent Systems. LNCS (LNAI), vol. 2650, pp. 300–317. Springer, Heidelberg (2003)
24. Pelachaud, C., Carofiglio, V., Poggi, I.: Embodied contextual agent in information delivering application. In: Proceedings of the First International Joint Conference on Autonomous Agents & Multi-Agent Systems, Bologna, Italy (2002)
25. Pelachaud, C., Maya, V., Lamolle, M.: Representation of expressivity for embodied conversational agents. In: Proceedings of the AAMAS, Embodied Conversational Agents: Balanced Perception and Action, vol. 4 (2004)
26. Poggi, I., Pelachaud, C., de Rosis, F., Carofiglio, V., de Carolis, B.: Greta. a believable embodied conversational agent. In: Stock, O., Zancanaro, M. (eds.) Multimodal Intelligent Information Presentation, vol. 27, pp. 3–27. Springer, Heidelberg (2005)
27. Predinger, H., Ishizuka, M.: Life-Like Characters. Springer, New-York (2004)
28. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition, pp. 267–296. Morgan Kaufmann Publishers Inc., San Francisco (1990)
29. Raskar, R., Nii, H., de Decker, B., Hashimoto, Y., Summet, J., Moore, D., Zhao, Y., Westhues, J., Dietz, P., Inami, M., Nayar, S., Barnwell, J., Noland, M., Bekaert, P., Branzoi, V., Bruns, E.: Prakash: Lighting-aware motion capture using photosensing markers and multiplexed illumination. ACM Trans. Graph. 26(3), 36 (2007)
30. Sementille, A.C., Lourenço, L.E., Brega, J.R.F., Rodello, I.: A motion capture system using passive markers. In: ACM SIGGRAPH International, Singapore, pp. 440–447. ACM, New York (2004)
31. Thomas, M.D., McGinley, J.A., Carruth, D.W., Blackledge, C.: Crossvalidation of an infrared motion capture system and an electromechanical motion capture device. In: Digital Human Modeling, Seattle, WA (2007)
32. Vicon (2009), `http://www.vicon.com`
33. Vlasic, D., Adelsberger, R., Vannucci, G., Barnwell, J., Gross, M., Matusik, W., Popović, J.: Practical motion capture in everyday surroundings. ACM Transactions on Graphics (TOG) 26(3), 35 (2007)
34. Warden, T., Lattner, A., Visser, U.: Real-time spatio-temporal analysis of dynamic scenes in 3D soccer simulation. In: Iocchi, L., Matsubara, H., Weitzenfeld, A., Zhou, C. (eds.) RoboCup 2008. LNCS, vol. 5399, pp. 366–378. Springer, Heidelberg (2009)
35. Wendler, J.: Automatisches Modellieren von Agenten-Verhalten. PhD thesis, Humboldt- Universität zu Berlin, Mathematisch-Naturwissenschaftliche Fakultät II (2003)
36. White, R., Crane, K., Forsyth, D.: Capturing and animating occluded cloth. ACM Transactions on Graphics (SIGGRAPH) (2007)
37. Yu, Q., Li, Q., Deng, Z.: Online motion capture marker labeling for multiple interacting articulated targets. Computer Graphics Forum 26(3), 477–483 (2007)