AI Magazine Volume 21 Number 1 (2000) (© AAAI)

Three RoboCup Simulation League Commentator Systems

Elisabeth Andrè, Kim Binsted, Kumiko Tanaka-Ishii, Sean Luke, Gerd Herzog, and Thomas Rist

■ Three systems that generate real-time natural language commentary on the RoboCup simulation league are presented, and their similarities, differences, and directions for the future discussed. Although they emphasize different aspects of the commentary problem, all three systems take simulator data as input and generate appropriate, expressive, spoken commentary in real time.

Here we present three RoboCup simulation league commentator systems: (1) ROCCO from DFKI, (2) BYRNE from Sony CSL, and (3) MIKE from ETL. Together, the three systems won the scientific award at RoboCup-98 (Asada 1998) for making a significant and innovative contribution to RoboCup-related research.

Soccer is an interesting test domain because it provides a dynamic, real-time environment in which it is still relatively easy for tasks to be classified, monitored, and assessed. Moreover, a commentary system has severe time restrictions imposed by the flow of the game and is thus a good test bed for research into real-time systems. Finally, high-quality logs of simulated soccer games are already available and allow us to abstract from the intrinsically difficult task of low-level image analysis.

From Visual Data to Live Commentary

The automated generation of live reports on the basis of visual data constitutes a multistage transformation process. In the following subsections, we describe how the maintainable subtasks transform the input into the final output.

The Input

All three commentary systems concentrate on the RoboCup simulator league, which involves software agents only (as opposed to the real robot leagues). Thus, the soccer games to be commented are not observed visually. Rather, all three systems obtain their basic input data from the SOCCER SERVER (Kitano et al. 1997).

All three commentator systems use as input the same information that the monitor program receives for updating its visualizations. This information consists of player location and orientation (for all players), ball location, and game score and play modes (such as throw ins, goal kicks).

Game Analysis

Game analysis provides an interpretation of the input data and aims at recognizing conceptual units at a higher level of abstraction. The information units resulting from such an analysis encode a deeper understanding of the time-varying scene to be described. They include spatial relations for the explicit characterization of spatial arrangements of objects as well as representations of recognized object movements. Scene interpretation is highly domain specific. Because we are dealing with soccer games, it is often possible to infer higher-level concepts from the observed motion patterns of the agents.

Topic Control and Content Selection

The next stage is to select the information to be communicated to the user. This selection depends on a number of constraints, such as the game situation, the purpose of the comment, the user's information needs, and the available presentation media. Because the commentary output is basically speech, we also have to consider that only one thing can be said at a time. In addition, the whole utterance should be consistent and well planned.

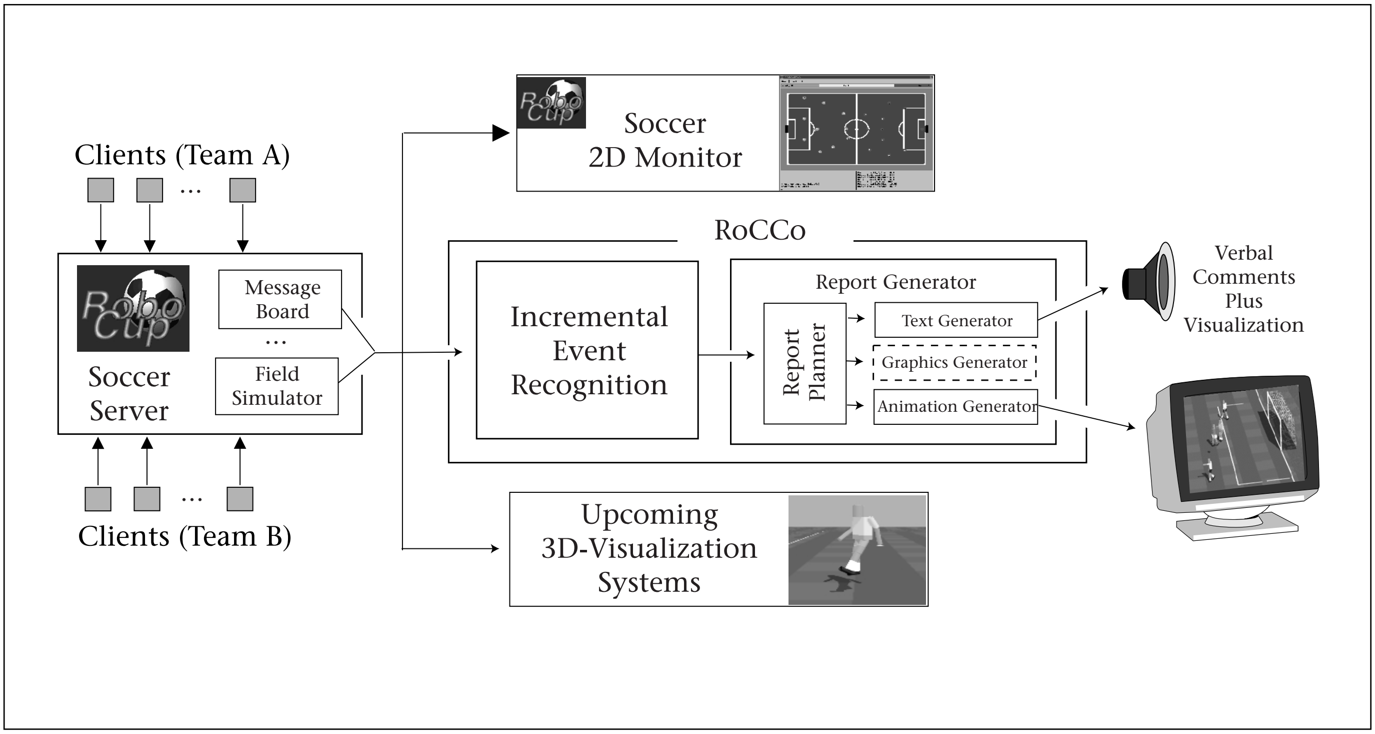


Figure 1. Connecting SOCCER SERVER and ROCCO.

Natural Language Generation

Once the content of the next utterance is decided, the text realization is performed, which comprises grammatical encoding, linearization, and inflection. Commentary is a real-time live report—information must be communicated under time pressure and in a rapidly changing situation. The system should choose whether to make short telegram-style utterances or grammatically complete and explanatory utterances according to the situation. Also, it can sometimes be necessary for the commentator to interrupt itself. For example, if an important event (for example, a goal kick) occurs, utterances should be interrupted to communicate the new event as soon as possible.

The Output

Finally, the natural language utterances are piped to a speech synthesis module. To get more natural speech output, these systems do not rely on the default intonation of the speech synthesizer but generate specific synthesizer instructions for expressing the corresponding emotions. These emotions, whether hard wired into the templates or generated in accordance with an emotional model and the state of the game, can also be shown in the facial expressions of an animated commentator character.

Although MIKE and ROCCO produce disembodied speech, BYRNE uses a face as an additional means of communication.

ROCCO

The ROCCO commentator system is a reincarnation of an early research prototype—called SOCCER—that was built by Andrè, Herzog and Rist (1988) in the late 1980s for the automated interpretation and natural language description of time-varying scenes. At that time, short sections of video recordings of soccer games were chosen as a major application domain because they offered interesting possibilities for the automatic interpretation of visually observed motion patterns in a restricted domain. Later, the work on incremental scene interpretation (Herzog and Rohr 1995; Herzog and Wazinski 1994; Herzog et al. 1989) was combined with an approach for plan-based multimedia presentation design (Andrè et al. 1993; Andrè and Rist 1990) to move toward the generation of various multimedia reports, such as television-style reports and illustrated newspaper articles (Andrè, Herzog, and Rist 1994). Part of this extension was a generic architecture for multimedia reporting systems that describes the representation formats and the multistage transformation process from visual data to multimedia reports. Both systems, the early SOCCER as well as the new ROCCO system, can be seen as instantiations of this architecture. In the following discussion, we briefly sketch ROCCO's main components and the underlying approaches for incremental event recognition and report generation.

In contrast to the early SOCCER system, ROCCO does not have to tackle the intrinsically difficult task of automatic image analysis. Instead of using real image sequences and starting from raw video material, it relies on the real-time game log provided by the SOCCER SERVER (figure 1). Based on these data, ROCCO's incremental event-recognition component performs a higher-level analysis of the scene under consideration. The representation and automatic recognition of events is inspired by the generic approach described in Herzog and Wazinski (1994), which has been adopted in ROCCO according to the practical needs of the application context. Declarative event concepts represent a priori knowledge about typical occurrences in a scene. These event concepts are organized into an abstraction hierarchy. Event concepts, such as locomotions or kicks, are found at lower levels of the hierarchy because they are derived through the principle of specialization and temporal decomposition of more complex event concepts such as a ball transfer or an attack. Simple recognition automata, each of which corresponds to a specific concept definition, are used to recognize events on the basis of the underlying scene data.

The recognized occurrences, along with the original scene data, form the input for report generation. To meet the specific requirements of a live report, ROCCO's report planner relies on an incremental discourse planning mechanism. Assuming a discrete-time model, at each increment of a time counter, the system decides which events have to be communicated next. Thereby, it considers the salience of events and the time that has passed since their occurrence. If there are no interesting events, the system randomly selects background information from a database.

The *text generator* is responsible for transforming the selected information into spoken utterances. Because it is rather tedious to specify soccer slang expressions in existing grammar formalisms, we decided to use a template-based generator instead of fully fledged natural language generation components, as in the earlier SOCCER system. That is, language is generated by selecting templates consisting of strings and variables that will be instantiated with natural language references to objects delivered by a nominal-phrase generator. To obtain a rich repertoire of templates, 13.5 hours of television soccer reports in English have been transcribed and annotated. Templates are selected considering parameters such as available time, bias, and report style. For example, short templates, such as "Meier again," are preferred if the system is under time pressure. Furthermore, ROCCO maintains a discourse history to avoid repetitions and track the center of attention. For example, a phrase such as "now Miller" should not be uttered if Miller is the current topic. For the synthesis of spoken utterances, ROCCO relies on the TRUETALK (Entropic 1995) text-to-speech software. To produce more lively reports, it annotates the computed strings of words with intonational markings considering the syntactic structure and the content of an utterance as well as the speaker's emotions. Currently, it mainly varies pitch accent, pitch range, and speed. For example, excitement is expressed by a higher talking speed and pitch range.

Figure 2 shows a screen shot of the initial JAVA-based ROCCO prototype that was taken during a typical session with the system. The monitor program provided with the RoboCup simulation environment is used to play back the previously recorded match in the upper right window. The text output window on the left-hand side contains a transcript of the spoken messages that have been generated for the sample scene. ROCCO does not always generate grammatically complete sentences but primarily produces short telegram-style utterances that are more typical of live television reports. The test-bed character of the ROCCO system provides the possibility of experimenting with various generation parameters, such as language style, which can be set in the lower right window.

… excitement is expressed by a higher talking speed and pitch range.

BYRNE

Now we present early work on an animated talking head commentary system called BYRNE (figure 3). This system takes the output from the RoboCup soccer simulator and generates appropriate affective speech and facial expressions (figure 4) based on the character's personality, emotional state, and the state of play (Binsted 1998).

BYRNE can use any modular game-analysis system as its input module. For RoboCup-98, however, BYRNE used a simple but effective play-by-play game-analysis system designed for the competition. This input module produces remarks much faster than BYRNE is capable of saying them. To compensate for this overabundance of things to say, the input module feeds its remarks into a priority queue. Each remark has a *birthday* (the time when it was entered into the queue), a *deadline* (a time beyond which it is "old news"), and a priority. When

Figure 2. The Basic Windows of the Initial ROCCO *System.*

BYRNE requests a new fact to say, the queue returns a fact using a simple priority-scheduling algorithm.

The information provided by the analysis system also drives the emotional model. The *emotion-generation module* contains rules that generate simple emotional structures. These structures consist of a type, for example, happiness, sadness; an intensity, scored from 1 to 10; a target (optional); a cause, that is, the fact about the world that caused the emotion to come into being; and a decay function, which describes how the intensity of the emotion decays over time.

An emotion structure–generation rule consists of a set of preconditions, the emotional structures to be added to the emotion pool, and the emotional structures to be removed. The preconditions are filled by matching on the currently true facts about the world and the character.

Both emotion generation and behavior generation are influenced by the static characteristics of the commentator character. This is a set of static facts about the character, such as its nationality and the team it supports. It is used to inform emotion and behavior generation, allowing a character to react in accordance with its preferences and biases. For example, if a character supports the team that is winning, its emotional state is likely to be quite different than if it supports the losing team.

Emotion structures and static characteristics are preconditions to the activation of high-level emotion-expressing behaviors. These, in turn, decompose into lower-level behaviors. The lowest-level behaviors specify how the text output by the text-generation system is to be marked up.

Emotionally motivated behaviors are organized in a hierarchy of mutually inconsistent groups. If two or more activated behaviors are inconsistent, the one with the highest activation level is performed, which usually results in the strongest emotion being expressed. However, a behavior that is motivated by several different emotions might win out over a behavior motivated by one strong emotion.

Emotions are expressed by adding mark up to already-generated text. Text generation is done simply through a set of templates. Each template has a set of preconditions that constrain the game situations they can be used to describe. If more than one template matches the chosen content, then the selection is based on how often and how recently the templates have been used. BYRNE's text-generation module does not generate plain text but, rather, text marked up with SEEML (speech, expression, and emotion mark-up language). Linguistically motivated facial gestures and speech intonation are currently hard coded into the templates.

SEEML is a slightly supplemented superset of three different mark-up systems, namely FACSML (a variant on FACS [Ekman and Friesen 1978]), SABLE (Sproat et al. 1997), and GDA (Nagao and Hasida 1998). GDA is used to inform linguistically motivated expressive behaviors and also to aid the speech synthesizer in generating appropriate prosody. FACSML is used to add facial behaviors, and SABLE is used to control the speech synthesis.

The expression mark-up module adds emotionally motivated mark up to the already marked-up text from the text-generation system. Conflicts are resolved in a simple (perhaps simplistic) manner. Unless identical, tags are assumed to be independent. If two identical tags are assigned to a piece of text, the one with the smaller scope is assumed to be redundant and removed. Finally, if two otherwise identical tags call for a change in some parameter, it is assumed that the change is additive.

The marked-up text is then sent to the SEEML parser. The parser interprets SEEML tags in the context of a style file, adds time markers and lip-synching information, and sends appropriate FACS to the facial-animation system and SABLE to the speech-synthesis system. The result is an emotional, expressive talking-head commentary on a RoboCup simulation league soccer game.

MIKE

MIKE (multiagent interactions knowledgeably explained) is an automatic real-time commentary system capable of producing output in English, Japanese, and French.

One of our contributions is to demonstrate

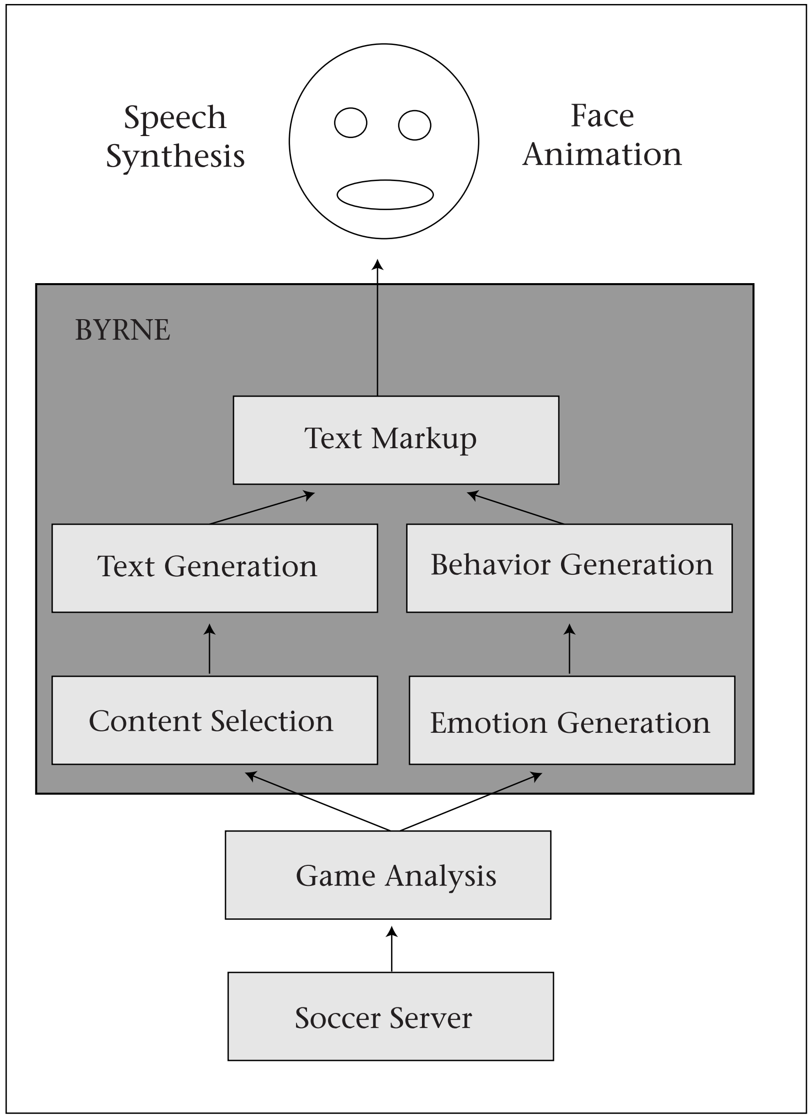


Figure 3. The BYRNE System Architecture.

Figure 4. BYRNE Sneering.

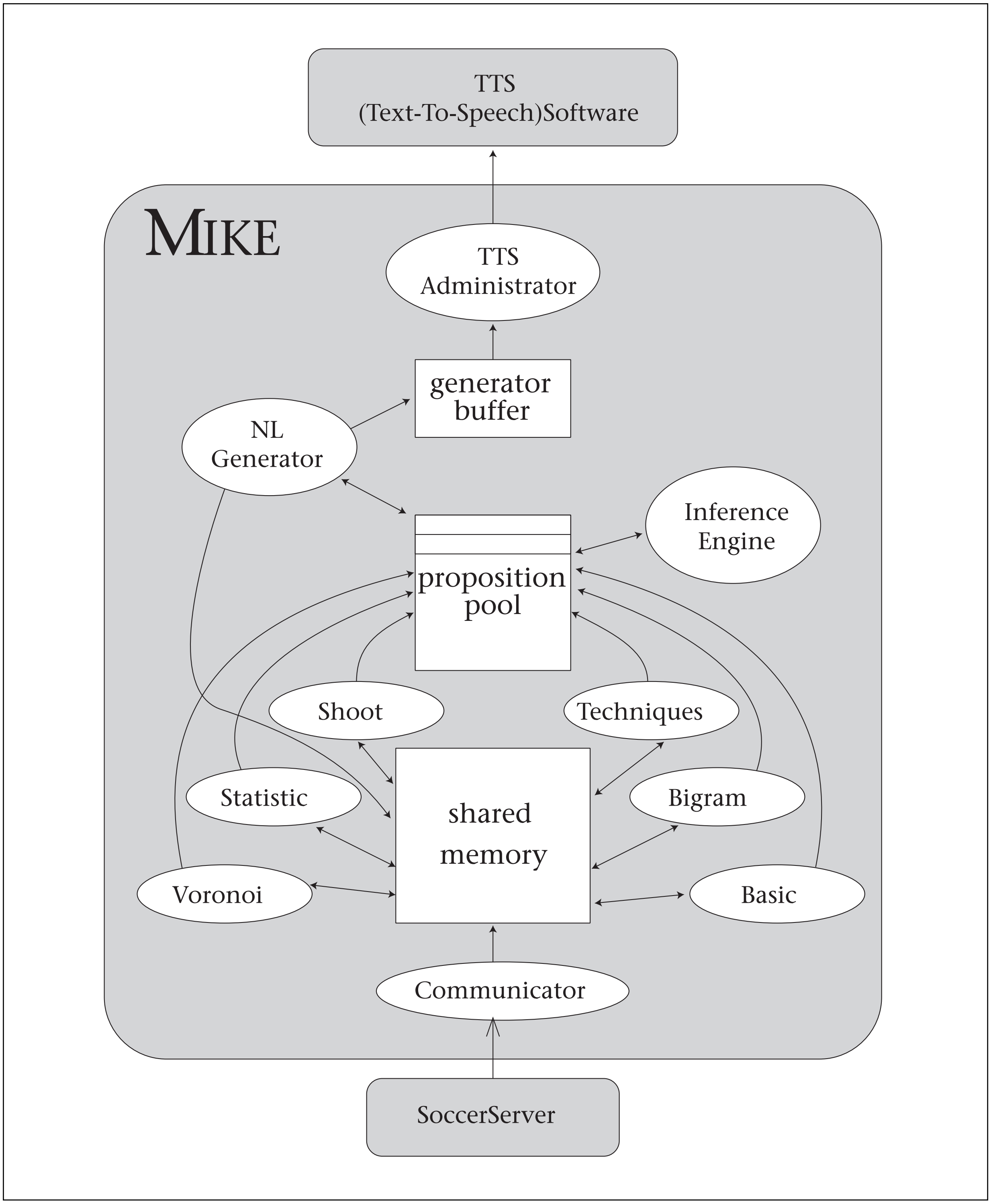


Figure 5. MIKE*'s Structure.*

that a collection of concurrently running analysis modules can be used to follow and interpret the actions of multiple agents (Tanaka-Ishii et al. 1998). MIKE uses six SOCCER ANALYZER modules, three of which carry out high-level tasks. Notably, these modules demonstrate the general applicability of analyzing the focus of a multiagent system and examining the territories established by individual agents.

Another technical challenge involved in live commentary is the real-time selection of content to describe the complex, rapidly unfolding situation (Tanaka-Ishii, Hasida, and Noda 1998). Soccer is a multiagent game in which various events happen simultaneously in the field. To weave a consistent and informative commentary on such a subject, an importance score is put on each fragment of commentary that intuitively captures the amount of information communicated to the audience. The inference and the content-selection modules are both controlled by such importance scores.

From the input sent by the SOCCER SERVER, MIKE creates a commentary that can consist of any combination of the possible repertoire of remarks. The commentary generation is coordinated by the architecture shown in figure 5, where the ovals represent processes, and the rectangles represent data. The repertoire is shown in figure 6; an English example of MIKE's commentary is shown in figure 7.

There are six SOCCER ANALYZER modules, of which three analyze basic events (shown in the figure as the basic, techniques, and shoot processes), and the other three carry out more high-level analysis (shown as the bigram, Voronoi, and statistic processes). These six processes analyze the information posted to the shared memory by the communicator, communicate with each other by the shared memory, and also post propositions to the proposition pool.

The *bigram module* follows and analyzes the ball plays as a first-order Markov chain that is a focus point in soccer game. A 2×24 ball-play transition matrix (22 players and 2 goals) is automatically formed during the game and used to describe the activity of each player and identify successful passwork patterns. Using this matrix, MIKE calculates and examines the players' ball-play performance, such as pass success rates, winning passwork patterns, and number of shots.

The Voronoi module calculates Voronoi diagrams for each team every 100 milliseconds. Using these partitions, one can determine the defensive areas covered by players and also assess overall positioning. Figure 8 shows an example of such a Voronoi diagram (+ and diamond indicate players of each team; box shows the ball.).

In the future, these rich state-analysis modules will be separated from the rest of the system into a "preMIKE" module so that simulation soccer teams or other commentary systems can dynamically refer to the analysis results.

All interprocess communication is done by handling an internal representation of commentary fragments, represented as propositions. These propositions consists of a tag and some attributes. For example, a kick by player 5 is represented as (Kick 5), where Kick is the tag, and 5 is the attribute.

Explanation of Complex Events: This concerns form changes, position changes, and advanced plays.

Evaluation of Team Plays: This concerns average forms, forms at a certain moment, players' locations, indications of the active or problematic players, winning passwork patterns, and wasteful movements.

Suggestions for Improving Play: These concern loose defense areas and better locations for inactive players.

Predictions: These concern passes, game results, and shots at goal.

Set Pieces: These concern goal kicks, throw ins, kick offs, corner kicks, and free kicks.

Passwork: This tracks basic ball-by-ball plays.

Figure 6. MIKE's Commentary Repertoire.

Red3 collects the ball from Red4, Red3, Red-Team, wonderful goal! 2 to 2! Red3's great center shot! Equal! The Red-Team's formation is now breaking through enemy line from center, The Red-Team's counter attack (Red4 near at the center line made a long pass towards Red3 near the goal and he made a shot very swiftly.), Red3's goal! Kick off, Yellow-Team, Red1 is very active because, Red1 always takes good positions, Second half of Robocup-97 quater final(Some background is described while the ball is in the mid field.) . Left is Ohta Team, Japan, Right is Humboldt, Germany, Red1 takes the ball, bad pass, (Yellow team's play after kick off was interrupted by Read team) Interception by the Yellow-Team, Wonderful dribble, Yellow2, Yellow2 (Yellow6 approaches Yellow2 for guard), Yellow6's pass, A pass through the opponents' defense, Red6 can take the ball,because, Yellow6 is being marked by Red6, ... The Red-Team's counter attack, The Red-Team's formation is (system's interruption), Yellow5, Back pass of Yellow10, Wonderful pass,

Figure 7. Example of MIKE's Commentary.

To establish the relative significance of events, importance scores are put on propositions. After being initialized in the ANALYZER MODULE, the score decreases over time while it remains in the pool waiting to be uttered. When the importance score of a proposition reaches zero, it is deleted from the pool.

Propositions deposited in the pool are often too low level to be directly used to generate commentary. The INFERENCE ENGINE MODULE processes the propositions in the pool with a collection of over–forward-chaining inference rules. The rules can be categorized into four classes, based on the relation of consequences to their antecedents: (1) logical consequences, (2) logical subsumption, (3) second-order relations, and (4) state change. For example, (HighPassSuccessRate player) (PassPattern player Goal) -> (active player) is a rule to make a logical consequence.

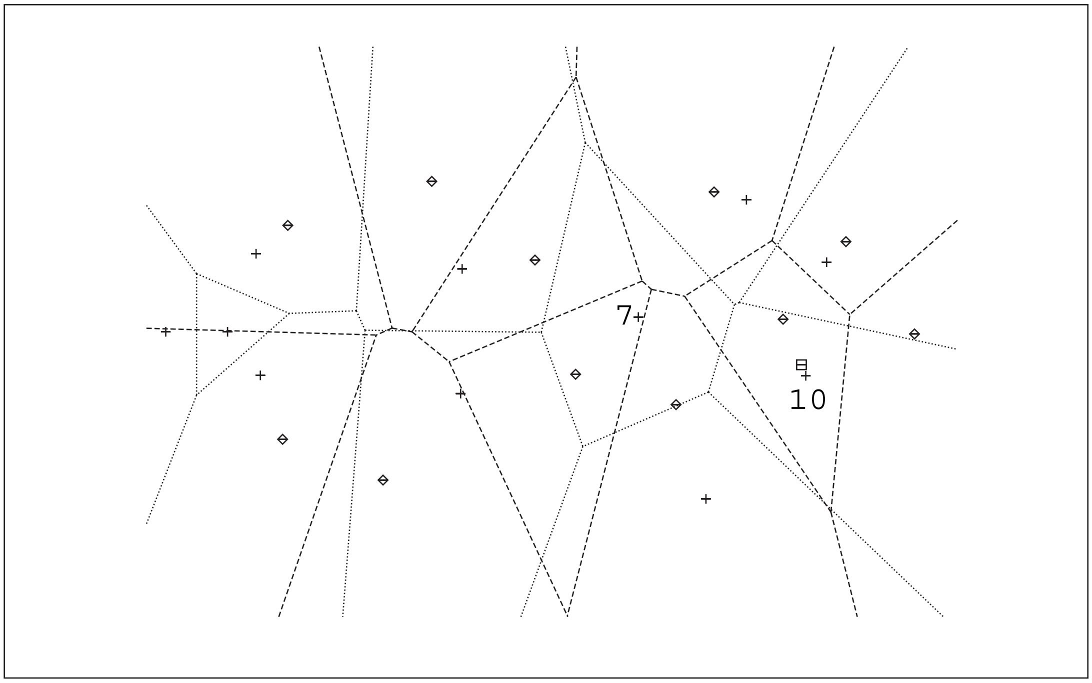


Figure 8. A Voronoi Diagram Example.

	Analysis	Methods for Topic Control	Natural Language Generation	Output
ROCCO	Recognition automata	Ranking based on salience, information gain, and age of events	Parameterized template selection + noun phrase generation in real time	Expressive speech
BYRNE	“Observers“ recognize events and states	Prioritized by birthday, deadline, and priority scores	Templates, marked up for expression and interruption in real time	Expressive speech and facial animation
MIKE	Events and states (bigrams, Voronoi)	Importance scores and inference	Templates, interruption, abbreviation using importance scores	Expressive speech

Table 1. A Comparison of the Features of the Three Commentator Systems.

Finally, the NATURAL LANGUAGE GENERATOR selects the proposition from the proposition pool that best fits the current state of the game and then translates the proposition into natural language. It broadcasts the current subject to the analyzers, so that they assign higher initial importance values to propositions with related subjects. Content selection is made with the goal of maximizing the total gain of importance scores during the game. Thus, the content selection is integrated in the surface-realization module, which accounts for interruption, abbreviation, and so on.

Discussion and Future Work

Even though our focus has been on the automatic description of soccer matches, none of the presented approaches is restricted to this class of application. For example, the group at DFKI originally addressed the dynamic description of traffic scenes (cf. Herzog and Wazinski [1994]). To port our approach to the domain of soccer, we didn't have to change a single line in the processing algorithms. Only our declarative knowledge sources, such as the event-recognition automata and the natural language templates, had to be adapted.

At Sony CSL, we are currently looking at adapting our systems to commentate other domains, in particular, other sports (for example, baseball, American football). Also, in real sports commentary, commentators usually work in pairs. One is the play-by-play commentator, and the other provides "color" (that is, background information and statistics about the players and teams). We are interested in having our commentator systems mimic these roles, taking turns providing relevant information in context.

DFKI is exploring new forms of commentary as well. We are currently working on a three-dimensional visualization component to enable situated reports from the perspective of a particular player.

As an application of MIKE, ETL is now developing a real-time navigation system for pedestrians, especially the blind, and also for automobiles. This system, like MIKE, senses the position of the target to be navigated, analyzes the situation using static information as maps, controls topic, and then outputs the navigation as expressive speech.

Conclusions

Although there are some differences in emphasis (table 1), all three systems provide real-time, expressive commentary on the RoboCup simulation league. All were demonstrated at RoboCup-98, and we hope that improved versions will be shown at future RoboCup events. The success of these systems shows that RoboCup is not just a robot competition—it is a challenging domain for a wide range of research areas, including those related to real-time natural language commentary generation.

Acknowledgments

We would like to thank Dirk Voelz for his work on the implementation of ROCCO.

References

Andrè, E., and Rist, T. 1990. Toward a Plan-Based Synthesis of Illustrated Documents. Paper presented at the Ninth European Conference on Artificial Intelligence, 6–8 August, Stockholm, Sweden.

Andrè, E.; Herzog, G.; and Rist, T. 1994. Multimedia Presentation of Interpreted Visual Data, Report, 103, SFB 314–Project VITRA, Universitat des Saarlandes, Saarbrüken, Germany.

Andrè, E.; Herzog, G.; and Rist, T. 1988. On the Simultaneous Interpretation of Real-World Image Sequences and Their Natural Language Description: The System SOCCER. Paper presented at the Eighth European Conference on Artificial Intelligence, 1–5 August, Munchen, Germany.

Andrè, E.; Finkler, W.; Graf, W.; Rist, T.; Schauder, A.; and Wahlster, W. 1993. WIP: The Automatic Synthesis of Multimodal Presentations. In *Intelligent Multimedia Interfaces,* ed. M. T. Maybury, 75–93. Menlo Park, Calif.: AAAI Press.

Asada, M., ed. 1998. RoboCup-98: Robot Soccer World Cup II. Paper presented at the Second RoboCup Workshop, 2–3, 9 July, Paris, France.

Binsted, K. 1998. Character Design for Soccer Commentary. Paper presented at the Second RoboCup Workshop, 2–3, 9 July, Paris, France.

Ekman, P., and Friesen, W. V. 1978. *Manual for the Facial-Action Coding System.* Palo Alto, Calif.: Consulting Psychologists.

Entropic. TrueTalk Reference and Programmers' Manual, Version 2.0, Entropic Research Laboratory Inc., Washington, D.C.

Herzog, G., and Rohr, K. 1995. Integrating Vision and Language: Toward Automatic Description of Human Movements. In *KI-95: Advances in Artificial Intelligence. Nineteenth Annual German Conference on Artificial Intelligence,* eds. I. Wachsmuth, C.-R. Rollinger, and W. Brauer, 257–268. Berlin: Springer.

Herzog, G., and Wazinski, P. 1994. Visual Translator: Linking Perceptions and Natural Language Descriptions. *AI Review* 8(2–3): 175–187.

Herzog, G.; Sung, C.-K.; Andrè, E.; Enkelmann, W.; Nagel, H.-H.; Rist, T.; Wahlster, W.; and Zimmermann, G. 1989. Incremental Natural Language Description of Dynamic Imagery. In *Wissensbasierte Systeme* (Knowledge-Based Systems). 3. Int. GI-Kongre, 153–162. Berlin: Springer.

Kitano, H.; Asada, M.; Kuniyoshi, Y.; Noda, I.; Osawa, E.; and Matsubara, H. 1997. RoboCup: A Challenge Problem for AI. *AI Magazine* 18(1): 73–85.

Nagao, K., and Hasida, K. 1998. Automatic Text Summarization Based on the Global Document Annotation. Technical report SCSL-TR-98-021, Sony Computer Science Laboratory, Tokyo, Japan.

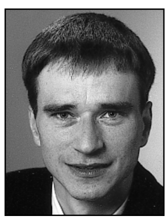
Sproat, R.; Taylor, P.; Tanenblatt, M.; and Isard, A. 1997. A Markup Language for Text-to-Speech Snythesis. Paper presented at EUROSPEECH'97, 22–25 September, Rhodes, Greece.

Tanaka-Ishii, K.; Hasida, K.; and Noda, I. 1998. Reactive Content Selection in the Generation of Real-Time Soccer Commentary. Paper presented at COLING-98, 10–14 August, Montreal, Canada.

Tanaka-Ishii, K.; Noda, I.; Frank, I.; Nakashima, H.; Hasida, K.; and Matsubara, H. 1998. MIKE: An Automatic Commentary System for Soccer. Paper presented at the 1998 International Conference on Multiagent Systems, 4–7 July, Paris, France.

Elisabeth Andrè is a principal researcher at the German Research Center for Artificial Intelligence in Saarbrüken, Germany. She is the chair of the ACL Special Interest Group on Multimedia Language Processing. She is on the editorial board of *Artificial Intelligence Communications* and is the area editor for Intelligent User Interfaces of the Electronic Transactions of Artificial Intelligence. Her research interests include multimedia communication, synthetic agents, and natural language interfaces. Her e-mail address is andre@dfki.de.

Kim Binsted (Ph.D., AI, University of Edinburgh, 1996; B.Sc., physics, McGill University, 1991) has been an associate researcher at the Sony Computer Science Laboratories in Tokyo, Japan, since January 1998. Her focus is on responsive entertainment and expressive characters. Previously, she was a Science and Technology Agency Postdoctoral Fellow at the Kansai Advanced Research Center (KARC-CRL) in Kobe, where she worked on computational humor in English and Japanese.



Gerd Herzog has been affiliated with the Intelligent Multimedia User Interfaces Laboratory at the German Research Center for Artificial Intelligence (DFKI) in Saarbrüken since 1996. Before joining DFKI, he was a project manager in the Computer Science Department at Universit des Saarlandes, where he was engaged in the long-term basic research project VITRA (visual translator) of the SFB 314, a special collaborative program on AI and knowledge-based systems, funded by the German Science Foundation. His research interests focus on intelligent user interfaces, in particular, vision and language integration. His e-mail address is herzog@dfki.de.

Thomas Rist is a senior scientist in the Department of Intelligent User Interfaces at the German Research Center for Artificial Intelligence, with a research focus on adaptive multimedia systems and multimodal interaction. He is currently a member of the European i3 (intelligent information interfaces) Coordination Group and is in charge of several managing tasks. His e-mail address is rist@dfki.de.

Kumiko Tanaka-Ishii (1991, University of Tokyo, B.Sc., applied mathematics; 1993, University of Tokyo, M.Sc., concurrent programming; 1997, University of Tokyo, Ph.D.) is currently a researcher at the Electrotechnical Laboratory, Japan, where her main research interests are natural language interface and concurrent systems. Her e-mail address is kumiko@etl.go.jp.

Sean Luke is a research assistant in the Department of Computer Science at the University of Maryland. Luke received a B.S. in computer science from Brigham Young University and is currently finishing his Ph.D. at the University of Maryland. His interests include cooperative multiagent behavior, massively distributed knowledge representation, robotics, and the evolution of neural systems. His e-mail address is seanl@cs.umd.edu.