A talking head architecture for entertainment and experimentation

Kim Binsted
Sony Computer Science Lab
3-14-13 Higashigotanda
Shinagawa-ku, Tokyo 141

Abstract

Byrne is a talking head system, developed with two goals in mind: to allow artists to create entertaining characters with strong personalities, expressed through speech and facial animation; and to allow cognitive scientists to implement and test theories of emotion and expression. In this extended abstract, we emphasize the latter aim. We describe Byrne's design, and discuss some ways in which it could be used in affect-related experiments. Byrne's first domain is football commentary; that is, Byrne provides an emotionally expressive running commentary on a RoboCup simulation league football game. We will give examples from this domain throughout this paper.

Background

Contributing technologies, such as speech synthesis and facial animation, are now sufficiently advanced that talking heads — systems which combine facial animation with speech synthesis — can be built. To date, most have been built from scratch, with the aim of demonstrating some particular theory (e.g. (Pelachaud, Badler, & Steedman 1994)), or supporting some particular application. These systems have often been effective for the purpose for which they were designed; however, few have been reusable.

Since the range of potential applications for a talking head system is so broad, it seems a waste for every researcher to have to reimplement it. However, for a talking head system to be useful to a wide range of researchers, it would have to be:

- **Modular:** There should be clean boundaries between the various subsystems (text generation, speech synthesis etc.) and widely accepted inter-system standards should used. Modularity is important for two reasons: so that researchers can substitute their own modules with relative ease; and so that improvements in the contributing technologies can be taken advantage of quickly.
- **Expressive:** The animated face and synthesized voice should as expressive as the human face and voice, if not more so. That is, every facial and vocal expression possible for a human should also be possible for the talking head. Moreover, these expressions should be controllable at a high level, so that researchers can get a reasonable results without knowing a great deal about prosody, face musculature, etc.
- **General-purpose:** The system should be, as much as it can be, application and theory neutral. Of course, any talking head system is going to have certain assumptions about the relationships between sub-systems implicit in its structure, and this would make it incompatible with certain models. However, we must try to avoid hard-wiring theoretical features into the system.

Here we describe work in progress on Byrne, an expressive, modular and general-purpose talking head system, and discuss a few of its potential uses. We also describe its first application: as a commentator for the RoboCup simulation football league.

Related work

Recently there has been a great deal of interest in the design and implementation of characters with personality. For example, the Virtual Theatre Project at Stanford (Hayes-Roth, van Gent, & Huber 1997) is working on a number of animated virtual actors for directed improvisation, basing their efforts on Keith Johnstone's theories of improvisational theatre (Johnstone 1992). They make use of character animations developed as part of the IMPROV project (Goldberg 1997), which can take fairly high-level movement directions and carry them out in a natural, expressive manner. Related work on agent action selection and animation has been done as part of the ALIVE (Blumberg & Galyean 1997) and OZ projects (Loyall 1997) (Reilly 1996).

Although these projects have similar goals and assumptions to ours, our approach differs from theirs on several points. First, our focus on talking heads (rather than fully embodied agents in virtual environments) leads to a stronger emphasis on face- and language-related behaviours. Also, we do not attempt to have the personality of the character control content selection, or the selection of any non-communicative actions, for that matter. Although this sharp distinction be-

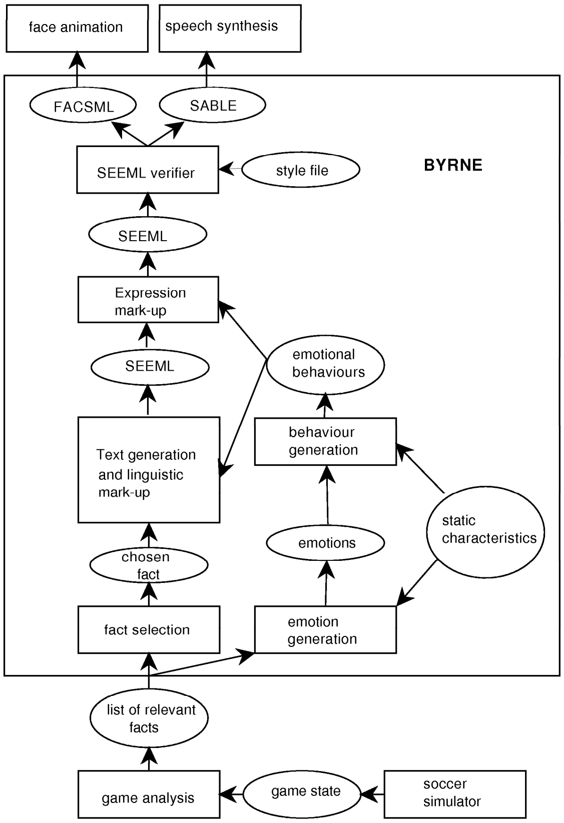

Figure 1: The Byrne system architecture

tween content and expression might negatively affect the consistency of the character, a clear separation between content and expression allows the character to be portable across content generators. For example, you could have essentially the same character (an aggressive older Scottish man, for example) commentate your soccer games and read your maps. In the first case, the content-generating application is the RoboCup soccer simulator, and in the second case it is an in-car navigation system — but the character remains the same.

There are two other research groups looking at the automatic generation of commentary for RoboCup, one in Japan(Tanaka-Ishii *et al.* 1997) and one in Germany(Andre, Herzog, & Rist 1998). Their emphasis, however, is more on game analysis and text generation than on expression.

Byrne

See Figure 1 for an outline of the Byrne architecture. The modules are described below. In the current implementation, none of the modules are very sophisticated; however, we expect that they are independent enough to be improved or replaced without large adjustments to the rest of the system.

Text mark-up

We have made heavy use of inter-system standards, especially SGML-based (Goldfarb 1991) mark-up systems, in an attempt to make Byrne as modular as possible. This is both so that researchers working on related technologies can make use of Byrne to demonstrate their systems with a minimum of fuss, and so that Byrne can be improved as its supporting technologies advance.

The ‘SEEML’ mark-up language referred to in Figure 1 is, at present, little more a concatenation of three other SGML-based mark-up languages: GDA (Global Document Annotation) (Nagao & Hasida 1998), which indicates the linguistic structure of the text; Sable (Sable 1998), which adds information relevant to speech synthesis (rate, prosody etc.); and FACSML, which is a FACS-based (Ekman & Friesen 1978) mark-up for control of facial expressions. These have been supplemented with a shorthand for some of the more common combinations of tags. Here is an example of SEEML text:

> <CHARACTER ID = ”Tammy”> <FROWN> I am so tired of this. <SHOUT> Why dont you ever <EMPH> listen </EMPH> to me! </SHOUT> <WHISPER> I tell you, <BREAK> I could just <EMPH> scream </EMPH>. </WHISPER> <AUDIO SRC=”scream.au”> </LANGUAGE> <LANGUAGE ID=”FRENCH”> <GRIMACE>Ah, mince alors! </GRIMACE> </LANGUAGE> </CHARACTER>

Input

Although the talking head can work directly from SEEML text, here we are more interested in having Byrne react emotionally to events in some domain. The system takes as input an ordered list of ’observations’ about the domain. In the soccer commentary case, this list is the output of a domain analysis system, and contains observations about the state of play, the positions of the players, the scores, etc. The analysis system must maintain this list, reordering the observations as their importance changes, and removing them when they become irrelevant or untrue. Here is an example of such a list:

- (goal player: a2 forteam: a fromloc: (20 10) begintime: 130 endtime: 135 importance: 10)
- (pass from: a1 to: a2 fromloc: (30 10) toloc: (20 10) begintime: 120 endtime: 125 importance: 8)
- (has-ball player: a2 location: (20 10) time: 125 importance: 5)
- (move player: b1 fromloc: (5 10) toloc: (10 10) begintime: 115 endtime: 120 importance: 3)
- . . .

The automatic analysis of the RoboCup simulation league is an interesting research area in and of itself, and the generation and maintenance of such a list of observations is not a trivial task. However, football analysis is not our main goal here. We have implemented a simple analyser which produces fairly low-level observations about the game (kicks, passes, goals and so on), but we do not consider this to be a part of the Byrne system.

Text generation

Character has a role to play in natural language generation. For example, a character from a particular country or region might use the dialect of that area, or a child character might use simpler vocabulary and grammar than an adult. The current emotional state of the character would also have an effect: an excited, angry character might use stronger language and shorter sentences than a calm happy one, for example. Loyall's work in the OZ project (Loyall 1997) discusses some of these issues.

Text generation is an area in which Byrne could be very sophisticated, but is not. At present, text generation is done very simply through a set of templates. Each template has a set of preconditions which 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. Pointers to the observations which caused the text to be generated are maintained, to aid later text mark-up (see section).

Byrne's text generation module does not generate plain text, but rather text marked up with SEEML (see section). Although the speech synthesis system we use can generate reasonable speech from plain text, it is helpful to retain some phrase structure and part of speech (POS) information from the natural language generation process to help the speech sythesis system to generate appropriate prosody.

Moreover, linguistic information embedded in the text also helps determine appropriate interruption points, should a more important fact need to be expressed. At present, we assume that Byrne should finish the phrase it is currently uttering before it interrupts and starts a new utterance. This is a simplistic approach, and may not be adequate.

Finally, linguistically-motivated facial gestures and speech intonation are now hard-coded into the templates. If the natural language generation system were more sophisticated, then a post-generation gesture and intonation system might be necessary, but with simple template generation this is the most effective method.

Domain abstraction

Although the football commentary domain is interesting, producing a soccer commentator is not our main goal. For this reason, we include a simple domain abstraction module. At present, this simply matches the each incoming observation to a set of templates, putting it into a less domain-specific form. For example, a goal observation might become:

> (event type: goal benefits: TeamA begintime: 130 endtime: 135 cause: (action type: kick actor: a2))

Of course, we could have simply written the analysis system to generate observations in this form. However, we cannot assume that we will always have access to the internals of the various domain analysis systems. The domain abstraction module serves as an interface between the domain analyser and Byrne.

Emotions and emotion generation

Athough Byrne's initial, very simple, emotional model is loosely based on (Ortony, Clore, & Collins 1988), we hope that the system is modular enough, and that the inter-module communication is flexible enough, that researchers in emotional modelling could usefully substitute their own model.

Emotion generation is rule based. The rules take domain states and events, static characteristics of the talking head personality, and the current emotional state of the character, as preconditions. Emotion structures have a type, an intensity, a target (optional), a cause, and a decay function. When the intensity of an emotion structure decays below one, it is removed from the pool.

So, if a character is very sad about Team A having just scored, the relevant emotional structure might be:

> (emotion type:sadness, intensity:10, target:nil, cause:(event type: goal benefits: TeamA ...) decay:1/t)

An emotion structure generation rule consists of a set of preconditions, which are to be filled by matching them on the current domain observations, static facts about the character, and currently active emotion structures. For example:

> **{Preconditions**:
> (appealingness object: ?team score: >10)
> (event benefits: ?team . . .)
> **Emotional structures to add**:
> (type: pleasure intensity: 5 target: nil cause: (event benefits: ?team . . .) decay: 1/t)
> **Emotional structures to delete**:
> none}

This rule indicates that, if the team that the commentator supports scores, a pleasure structure should be added to the emotion pool.

There are only two ways for an emotion structure to be removed from the emotion pool: it can be explicitly removed by an emotion structure update rule, or its intensity can decay to below one, in which case it is automatically removed.

Both emotion generation and behaviour generation are influenced by the **static characteristics** of the commentator character. This is a set of static facts about the character, such as gender, nationality, static prejudices, and so on. It is used to inform emotion and behaviour generation, allowing a character to react in accordance with his preferences and biases. For example, if a character supports the team which is winning, his emotional state is likely to be quite different than if he supports the losing team.

Behaviour generation

Byrne's face animation and speech synthesis systems are controlled via text marked up with SEEML tags. For this reason, Byrne's lowest level emotional behaviours are defined in terms of these tags, and describe how these tags are to be added to the text.

Emotion structures and static characteristics are preconditions to the activation of high-level emotion-expressing behaviours. These in turn decompose into lower-level behaviours. All emotionally-motivated behaviours are organized in a hierarchy of mutually inconsistent groups. If two or more activated behaviours are inconsistent, the one with the highest activation level is performed. This will usually result in the strongest emotion being expressed; however, a behaviour which is motivated by several different emotions might win out over a behaviour motivated by one strong emotion.

It is entirely possible for mixed emotional expressions to be generated, as long as they are not inconsistent. For example, a happy and excited character might express excitement by raising the pitch of his voice and happiness by smiling. However, it is less likely that a character will have a way to express, say, happiness and sadness in a consistent manner. At present, expressions are considered to be inconsistent if they use the same muscle group or speech parameter.

Here is an example of a simple expressive behaviour, which makes the character sneer every time a disliked object is mentioned:

```
{Preconditions:
(emotion type:dislike, intensity:>5, target: (object
name: ?X))
(text-contains ?X)
Actions:
(text-replace ?X "<SNEER> ?X </SNEER>")}
```

A single piece of text might contain tags motivated by any number of different behaviours, as well as linguistically-motivated tags added during text generation.

SEEML verifier and style file

The SEEML verifier parses the marked-up text 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 (supplemented with phrase structure and part of speech tags) to the speech synthesis system.

Although sophisticated lip-synchronization algorithms have been developed (e.g. in (Waters & Levergood 1993), they are not necessary for our purposes. Instead, we use a simple 'cartoon style' lip animation, which only shows the more obvious phoneme-viseme matches, as described in (Parke & Waters 1996).

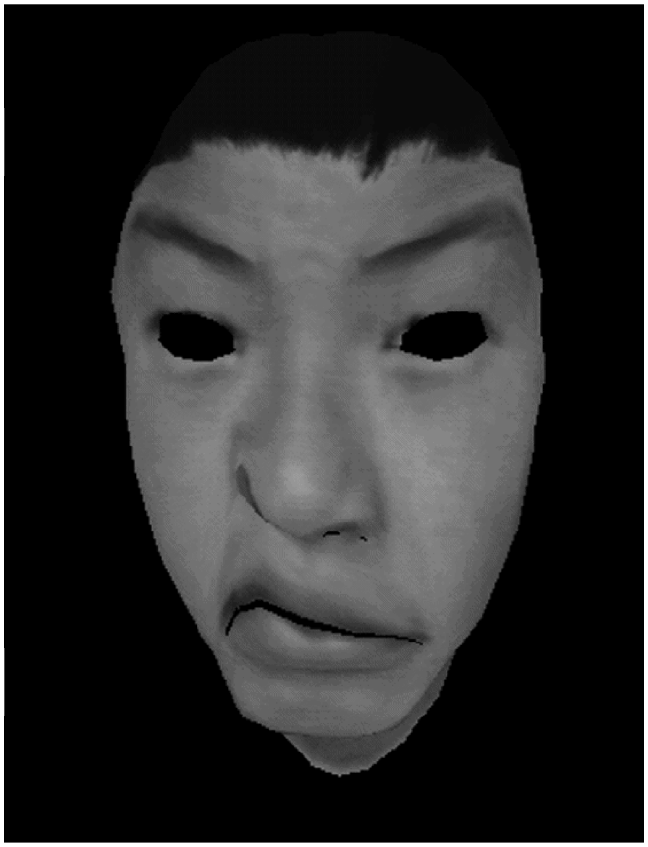

Figure 2: Byrne sneering.

The style file contains speech and animation system specific interpretation rules. For example, it would determine the FACS which are to be used to indicate a **smile** for this particular face model, and the sound file to be used for a **hiccup**.

Implementation

In the first implementation, Byrne uses Franks and Takeuchi's facial animation system (Takeuchi & Franks 1992) and the Festival speech system (Black, Taylor, & Caley 1997). We hope that the standardized mark-up of the output will allow the use of other face and speech systems as well.

Our first prototype was demonstrated at RoboCup98 (Binsted 1998), an official part of the *human* 1998 World Cup. Although the implemented emotional behaviours were few and simple, Byrne managed to generate rudimentary expressive commentary in real time on a number of simulation league games.

Talking heads as experimental apparatus

Byrne has three main applications in research. First, as a whole system for demonstrating advances in component technologies, such as speech synthesis, face animation etc. Not only is a demonstration with a full talking-head system generally more appealing than that of a sub-system, but it shows a certain amount of robustness and standardization. For example, if a speech synthesis system works well with Byrne, we know that it is relatively fast, doesn't rely on system specific controls, responds gracefully to interruption, etc.

Second, Byrne could be used as a character in simulated interactions, particularly when the researcher wishes to be able to have fine control over the parameters of the character. For example, a psychologist investigating aggression might wish to expose subjects to a simulated interaction with Byrne, varying only the simulated character's tone of voice and level of formality.

Finally, Byrne provides a full system (and context, if the football commentator is used) within which researchers can implement their models of natural language generation, emotion and emotional expression. Of course, Byrne's architecture will make some models easier to implement than others; nonetheless, we believe that a reasonable range of ideas can be accommodated with relatively minor modifications to Byrne.

In all three cases, Byrne's usefulness would come from its being a whole system, going from domain information to expressive output. Researchers would only have to modify those modules which were directly relevant to their own work.

Drawbacks

Byrne is only a talking head. It does not have a body, so would not be useful to researchers for whom locomotion, non-facial gestures and other body actions are important.

Byrne is modular, so would not be very useful to a researcher whose model does not fit neatly into those modules.

Byrne has no language understanding abilities, so would not be able to interact directly with another simulated or real agent. However, Byrne could be used as the expressive face for a more complete system with understanding.

Future work and conclusions

We are presently improving our implementation of Byrne as a football commentator. With RoboCup as the domain, we are exploring control issues for expressive and personality-driven speech and face animation. We also plan to look into how dominance affects turn-taking between two talking heads.

We hope that, as Byrne becomes more sophisticated, and as the inter-system standards it uses become more widely accepted, that other researchers will also find it useful in their work.

Acknowledgements

We would like to thank Hiroaki Kitano for encouragement and support on this project, Alan Black and Steve Franks for their patient responses to my endless questions, and Sean Luke for his work on the domain analysis module.

References

Andre, E.; Herzog, G.; and Rist, T. 1998. Generating multimedia presentations for robocup soccer games. Technical report, DFKI GmbH, German Research Center for Artificial Intelligence, D-66123 Saarbrucken, Germany.

Binsted, K. 1998. Character design for soccer commentary. In *Proceedings of the Second International Workshop on RoboCup*, 23–35.

Black, A. W.; Taylor, P.; and Caley, R. 1997. *The Festival Speech Sythesis System.* CSTR, University of Edinburgh, 1.2 edition.

Blumberg, B., and Galyean, T. 1997. Multi-level control for animated autonomous agents: Do the right thing... oh, not that... In Trappl, R., and Petta, P., eds., *Creating Personalities for Synthetic Actors.* Springer-Verlag Lecture Notes in Artificial Intelligence. 74–82.

Cahn, J. 1989. Generating expression in sythesized speech. Master's thesis, Massachusetts Institute of Technology Media Laboratory, Boston.

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

Goldberg, A. 1997. IMPROV: A system for real-time animation of behavior-based interactive synthetic actors. In Trappl, R., and Petta, P., eds., *Creating Personalities for Synthetic Actors.* Springer-Verlag Lecture Notes in Artificial Intelligence. 58–73.

Goldfarb, C. 1991. *The SGML Handbook.* Clarendon Press.

Hayes-Roth, B.; van Gent, R.; and Huber, D. 1997. Acting in character. In Trappl, R., and Petta, P., eds., *Creating Personalities for Synthetic Actors.* Springer-Verlag Lecture Notes in Artificial Intelligence. 92–112.

Johnstone, K. 1992. *Impro.* Routledge Theatre Arts Books.

Loyall, A. B. 1997. Some requirements and approaches for natural language in a believable agent. In Trappl, R., and Petta, P., eds., *Creating Personalities for Synthetic Actors.* Springer-Verlag Lecture Notes in Artificial Intelligence. 113–119.

Nagao, K., and Hasida, K. 1998. Automatic text summarization based on the global document annotation. Technical report, Sony Computer Science Laboratory.

Ortony, A.; Clore, G.; and Collins, A. 1988. *The cognitive structure of emotions*. Cambridge University Press.

Parke, F. I., and Waters, K. 1996. *Computer Facial Animation*. Wellesley, MA: A K Peters Ltd.

Pelachaud, C.; Badler, N.; and Steedman, M. 1994. Generating facial expressions for speech. In *Cognitive Science*.

Reilly, W. S. N. 1996. *Believable social and emotional agents*. Ph.D. Dissertation, School of Computer Science, Carnegie Mellon University.

1998. Draft specification for sable version 0.1. Technical report, The Sable Consortium.

Takeuchi, A., and Franks, S. 1992. A rapid face construction lab. Technical Report SCSL-TR-92-010, Sony Computer Science Laboratory, Tokyo, Japan.

Tanaka-Ishii, K.; Noda, I.; Frank, I.; Nakashima, H.; Hasida, K.; and Matsubara, H. 1997. MIKE: An automatic commentary system for soccer. Technical Report TR-97-29, Electrotechnical Laboratory, Machine Inference Group, Tsukuba, Japan.

Waters, K., and Levergood, T. M. 1993. DECFace: An automatic lip-synchronization algorithm for sythetic faces. Technical Report CRL 93/4, Digital Cambridge Research Laboratory.