

Customisation and Context for Expressive Behaviour in the Broadband World

M. Gillies¹, I. B. Crabtree², D. Ballin²

¹University College London, Adastral Park Campus
Orion Building GND pp13, Adastral Park, Ipswich IP5 3RE, UK
email: m.gillies@ucl.ac.uk

²BT plc.
Ross Building pp 4, Adastral Park, Ipswich IP5 3RE, UK
email: {barry.crabtree,daniel.ballin}@bt.com

May 19, 2004

Abstract

The introduction of consumer broadband makes it possible to have an emotionally much richer experience of the internet. One way of achieving this is the use of animated characters endowed with emotionally expressive behaviour. This paper describes Demeanour, a framework for generating expressive behaviour, developed collaboratively by University College London and BT plc. The focus of this paper will be on two important aspects; the customisation of expressive behaviour and how expressive behaviour can be made context dependent.

Customisation is a very popular feature for internet software, particularly as it allows users to present a specific identity to other users; the ability to customise behaviour will increase this sense of identity. Demeanour supports a number of user friendly methods for customising behaviour, all of which use a character profile that ultimately controls the behaviour of the character.

What counts as appropriate behaviour is highly dependent on the context, where you are, who you are talking to, whether you have a particular job or role. It is therefore very important that characters are able to exhibit different behaviours in different contexts. Demeanour allows characters to load different profiles in different contexts and therefore produce different behaviour.

1 Introduction

One of the core advantages of consumer broadband is the ability to provide the end user with a far richer experience of the internet. User can go beyond terse textual communication and interaction to new forms of audio-visual experience. One particularly exciting possibility is the increase of potential for emotion and other forms of self expression within interactive media. It has often been noted that emotion and other forms of expression are severely lacking in internet

communication. For example, simple textual interactions are almost completely lacking in the cues we normally use to infer the emotional content of a statement and other similar features that are vital to the interpretation of communication. This often leads to confusion and misinterpretation which is often problematic, for example, misinterpreting a humorous comment as a hostile one. Another problem is that the lack of emotional feedback can lead to a breakdown in communication and often excessive emotion on either side as is seen in the phenomenon of “flaming”.

There are many ways in which expression can be added to broadband internet interaction. Simple text can of course be expressive; emotion, interpersonal attitude and personality can all be expressed both in the explicit content and the phrasing of text. Some of the techniques described here could be used for expressive text generation. However, the true potential of broadband internet for emotional expression comes when more direct modalities are available. Audio can be a very powerful emotional cue, both in the form of music and in non-verbal aspects of speech. The use of emotional music and speech has great potential for changing the way the internet works. Visual communication is, however, likely to make the most important emotional impact as audio over the internet is currently less popular, due to problems of quality. There are many ways to express emotion visually, for example by colour or abstract motion, however, we believe that the most natural and powerful way is to reproduce the same cues that are used in face-to-face communication between people. This involves using a human-like animated graphical character that uses gesture, posture, gaze, facial expression and other cues to express emotion. In this paper concentrates on animated characters though many of the techniques described would be applicable to other forms of expression.

There are many applications where expressive, networked communication is important, for example:

Social Software. Expressive communication can be used purely for communication and therefore be applied to new social technologies. Instant messaging applications can be enhanced with expressive characters providing an emotional channel. Social virtual worlds and graphical chat rooms provide another natural extension. A more business oriented application is to conferencing[24]. Expressive behaviour is particularly important here in presenting a favourable impression and in negotiation.

Computer Games. With the advent of massively-multiplayer networked computer games it is not just the impression the game presents to the player that is important but also the impression that players present to each other. Expressive behaviour is vital to this.

Education. Good quality social interaction can help learning and expressive behaviour can improve the quality of social interaction. There are a number of specific applications in education. Expressive feedback from learners to teachers can help signal whether the learners are understanding or having difficulties. One particularly useful application for education, particularly for children, is virtual theatre. Children are able to role play educational scenarios on-line and thus enhance their learning experience. Expression can be an important addition to this kind of role play.

Retail and customer relationship management. It is vital in retail to present a good image to customers. Having appropriate expressive behaviours can greatly enhance this, particularly when feedback appropriate to the customers' wishes and concerns is given.

Telecare. Ill health can be emotionally difficult, so it is very important to take the emotional aspects of healthcare into account when creating health care software. One example is the use of various types of internet forum to enable patients, particularly those with limited mobility, to meet others with their condition[10]. These fora help patients deal with the emotional aspects of illness and therefore emotional expression would be a useful enhancement. There has also been work on interactive drama specifically aimed at helping people cope emotionally with illness in their family[21].

This paper presents the Demeanour framework for the generation of expressive behaviour for animated virtual characters. In particular it focuses on two important and connected features: customization of characters, and context dependent expressive behaviour.

Customization is vitally important for the internet in general. Each individual wants to interact in their own personalised way. This becomes especially important when people are not merely gathering information, e.g. web surfing, but are interacting and presenting themselves to others as is the case in social software. Users of social virtual worlds and multi-player games are very keen to personalise their characters and therefore present a particular identity to others[9]. The nature of this identity can vary greatly, from wacky in a teenage chat-room to businesslike in a virtual conference, and the virtual identity is very often different from a persons' real identity. The important point is that personalisation of characters must be possible and must be under the control of the user. Up to now, virtual worlds have extensive capabilities for customising the graphical appearance of characters but few for the expressive behaviour which is equally important.

Expressive behaviour is highly context dependent; we do not behave the same way when we are at work as we do in a nightclub, nor do we behave the same way with our friends as with our parents. It is therefore very important that if characters are to be used in a variety of different contexts they should be able to vary their expressive behaviour as required. For example, a customer relations agent should be able to present different behaviours to a domestic or business client or when in a sales or customer support role. In educational environments it might be useful to define different behaviours for different roles, for example, teacher or learner. A doctor in a healthcare environment should be able to behave differently depending on the seriousness of an illness. Context dependent behaviour is therefore important to a number of applications but is probably most useful across applications as it enables a user to keep the same character across a wide variety of applications while being able to produce behaviour appropriate to each, for example, between a chat room, a computer game and a teleconferencing system. One final use of context dependence is dealing with cultural issues. Expressive behaviour such as gesture is highly culturally dependent; a gesture which might be polite in one culture can be highly rude in another. It is therefore important in international dealings to use appropriate expressive behaviour. A context dependent model of expressive

behaviour would make this possible by generating different behaviour (with the same meaning) in different cultural contexts.

2 Related Work

Our work builds on a body of work on autonomous characters for virtual environments, for example, Blumberg and Galyean[5]; Badler, Phillips and Webber[3]; Tu and Terzopoulos[30]; Perlin and Goldberg[26], and Rickel and Johnson[27]. There has been extensive research on autonomously producing expressive behaviour of a number of types including facial expression (Pelachaud and Poggi[25]), eye gaze (Cassel et. al. [6], Rickel and Johnson[27] and Gillies and Dodgson[17]), gesture (Cassell et. al.[6]), style of motion (Chi et. al.[11]) and, like our current implementation, posture (Cassell, Nakano and Bickmore[7]), Bécheiraz and Thalmann[4]).

Maya, Lamolle and Pelachaud[22] have investigated how to create variation between animated characters. They use XML based profiles which are merged with an XML based specification of the affective content of a particular piece of speech, using an XSLT based system, to produce a final piece of behaviour. However, they do not provide any user friendly system for customising characters, nor does their system work in real time. The use of profiles and context dependence has also been used in other types of agent technology, for example, Soltysiak and Crabtree[28].

3 Demeanour

Demeanour is a framework for the generation of non-verbal communication in animated virtual characters (figure 2 shows an overview of the system). It is able to generate non-verbal behaviour in real-time based on a user definable behaviour model (currently we are working with psychologically based models). Characters react appropriately to each other's body language, making it unnecessary for the user to explicitly react to the actions of others. Non-verbal communication expresses itself in a number of ways, or modalities, we are currently interested in three: posture, gesture and eye gaze. Figure 1 shows an example of a group of three characters interacting and displaying expressive behaviour generated by Demeanour.

3.1 Different levels of control

The aim of this research is to find ways in which end-users can influence the behaviour of characters without causing an excessive overhead on their other activities. One method would be to measure the affective state of the user, e.g. using computer vision based analysis of facial expression, and map it onto the character. Unfortunately such methods are currently difficult or unreliable. Also, users often want to hide their real feelings. Therefore our aim is to develop user interfaces based on direct control but make them as unintrusive as possible. There are two types of control: users can give commands to the character in real time while using the virtual world and interacting with other characters (real time control). It is also possible to perform customizations on the character's behaviour before using the virtual world or between sessions(off-line

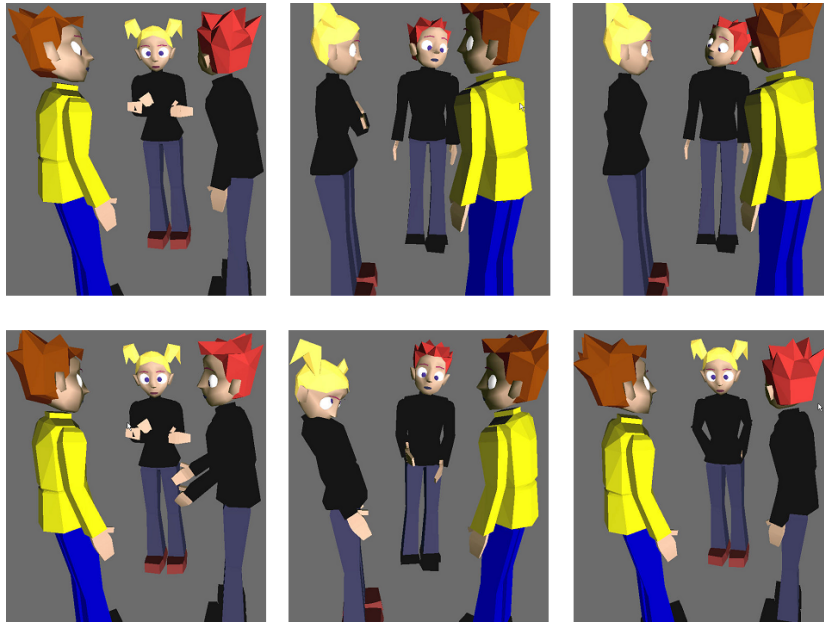


Figure 1: Three characters displaying expressive behaviour. The characters display gaze, posture and gestures behaviour and different attitudes to each other. The female character displays a negative attitude (low affiliation) whereas the male character dressed in yellow displays more positive attitude and the male character dressed in black has a mixed attitude.

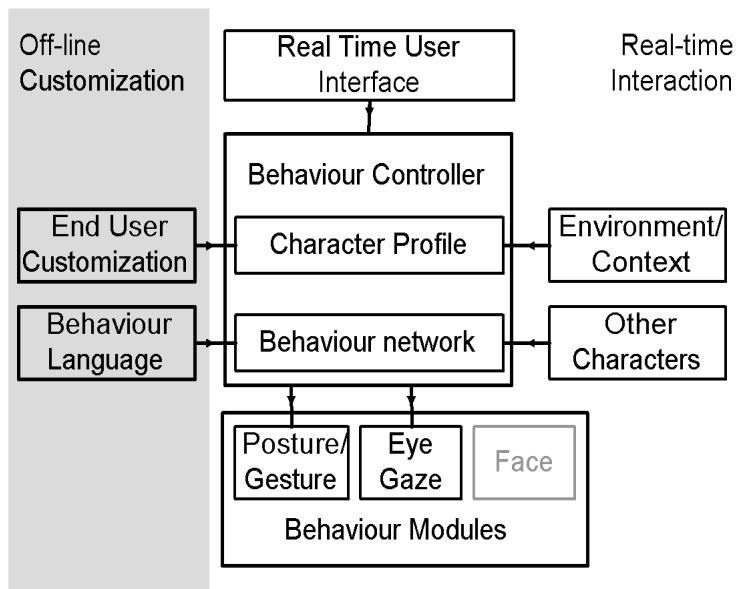


Figure 2: An overview of the Demeanour Framework

customization). In order to minimize overhead while using the virtual world it is desirable to have as much control as possible provided through off-line customization. We therefore propose a methodology for user control of characters that divides control and customisation into three levels:

Behaviour Language. The behaviour of the character should not be controlled by a fixed program but be entirely definable by the designers of a virtual world. This is important as the behaviour of characters is likely to vary considerably between different worlds used for different purposes (e.g. between a business conferencing environment and an adventure game). This capability is provided via a definition language.

Profiles. Demeanour contains a set of tools with which end users can define a profile for their character and provide context dependent behaviour. This is the main focus of the paper and described in more detail in section 5.

Real-time Control. It is also necessary for a user to provide some control of their character while interacting on-line. It is important that this is as unobtrusive as possible and well integrated with the other tasks to be performed in the world. This interface is described in more detail in Gillies, Crabtree and Ballin[18]

3.2 The Demeanour Architecture

The core of Demeanour is a behavioural controller that determines the type of behaviour to be performed at a given time. This contains a behaviour network

which gives the structure of the controller and is defined using the behaviour language. The details of the behaviour can vary between characters and is determined by character profiles, which are described in section 5. Characters react to the behaviour of other characters and the end-user can also control the character’s behaviour using a text-chat based interface, as described in [18]. Finally the character is actually animated by behaviour modules which interface with the underlying graphics API (the current implementation uses TARA, BT Exact’s scene-graph based graphics API). Currently Demeanour supports posture, gesture and eye gaze and we are considering adding a facial animation module.

As shown in Figure 2 the Demeanour framework combines a number of factors such as user input, context, and the behaviour of other characters in order to generate appropriate expressive behaviour. The outputs of the architecture are a number of parameters passed to behaviour generating modules. Input factors are mapped to outputs via of a number of terms which are intermediary values calculated from other terms, including input factors. For example, terms can represent the attitude values of a character such as “affiliation” (see below). Attitude values are calculated from internal parameters of the character plus input from other characters and are themselves used to calculate the term values that are used to directly generate behaviour. When there are multiple characters interacting then input from each character will be different. Rather than requiring a different set of terms for each character, Demeanour maintains a single set of terms, each of which can be evaluated differently for each character. Terms can have a number of types. The simplest are parameters which are single values. These can provide input from other characters, or they can provide a means for users to control the behaviour of their character, either through off-line customisation or real-time control. Terms can also be the combination of a number of other terms, this combination can be done in a number of ways:

- Sum of product terms combine their inputs via multiplication and addition of their values.
- Switch terms choose one of their inputs based on the value of another term.
- Random group terms map their input to a number of outputs. The outputs values are each a proportion of the input, the proportions are chosen at random.

4 Non-verbal behaviour

This section describes a behaviour network for non-verbal communication that we have developed. It models the way people relate to each other or their attitude to each other and is based on the work of Argyle[1]. In our model the attitude of one person to another is expressed through posture and, to a more limited degree, gesture. It is discussed in more detail in Gillies and Ballin[16].

Though there is an enormous variety in the way that people can relate to each other Argyle identifies two fundamental dimensions that can account for a majority of non-verbal behaviour, affiliation and status. Affiliation can be

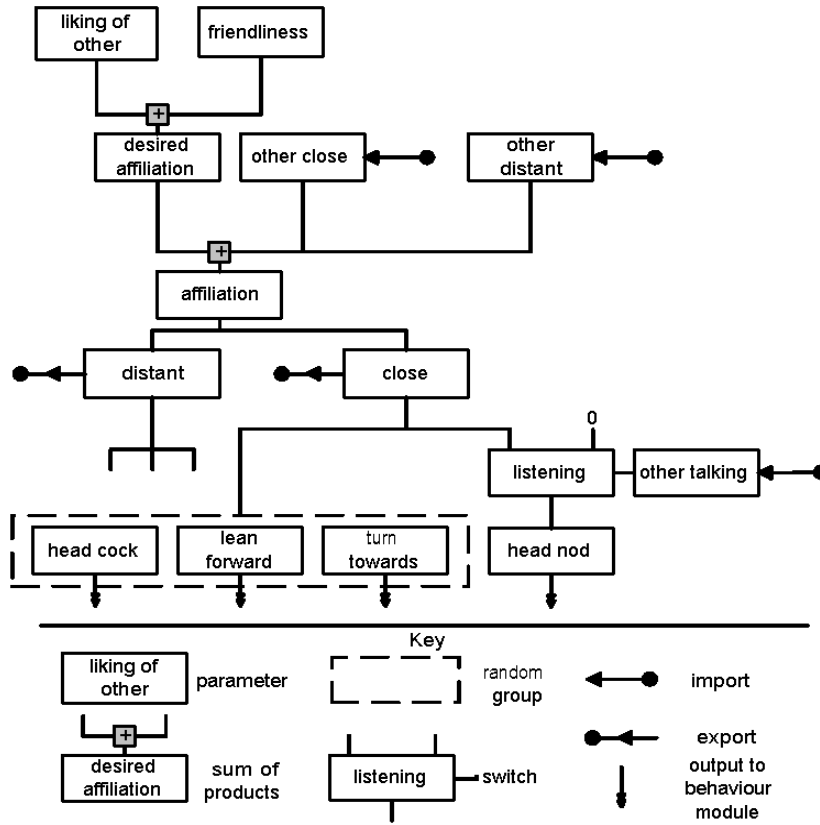


Figure 3: A section of a behavioural controller.

broadly characterised as liking or wanting a close relationship. It is associated with high levels of eye gaze and close postures, either physically close such as leaning forward or other close interaction such as a direct orientation. Low affiliation or dislike is shown by reduced eye gaze and more distant postures, including postures that present some sort of barrier to interaction, such as crossed arms. Status is the social superiority (dominance) or inferiority (submission) of one person relative to another, we will not discuss it directly in our examples.

Figure 3 shows in diagrammatic form a fragment of the attitude network that deals with affiliation (status is calculated in a similar way) and posture (eye gaze is discussed in section 4.2). At the top of the diagram the actual value for affiliation is calculated as a weighted sum of a number of factors (for the sake of clarity not all the factors used are actually shown). This is done in two stages, firstly the factors depending on the character itself are calculated. These factors are represented as parameters (here ‘liking of other’ and ‘friendliness’ are shown). Then factors depending on the other character’s behaviour (‘close’ and ‘distant’) are added in. These are import terms and are therefore taken directly from the controller of the other character. As the behaviours associated with positive and negative affiliation are very different it is split into two terms, ‘close’ which is equal to the affiliation and ‘distant’ which is its negation. Both of these



Figure 4: Examples of body language generated by the Demeanour framework to reflect different attitudes between the characters. Clockwise from top left: mutual gaze; close and relaxed postures; the male character is gesturing while talking; the female character has a distant, hostile posture; the female character has a high status, space filling posture the male character has a low status, submissive posture; the male character is relaxed (a high status posture) and the female character has a close posture.

terms are constrained to be greater than 0. The ‘close’ term is then mapped into actual behaviour (as is ‘distant’ but it is not shown in the diagram). In order to vary the behaviour produced a random group is used. At semi-regular intervals a new combination of the various behaviours (‘head cock’, ‘lean forward’ and ‘turn towards’) is produced, this combination is always proportional to the value of ‘close’. These behaviour types are output terms and are passed as parameters to the underlying animation system. Another affiliative behaviour is head-nodding, but this is only shown when the other person is talking. This behaviour is controlled by a switch node (‘listening’), based on a boolean import term which specifies if the other character is talking. If ‘other talking’ is true then ‘head nod’ is proportional to ‘close’ otherwise it is zero. Figure 4 shows examples of body language generated by the Demeanour framework.

4.1 Posture and Gesture

Human bodies are highly expressive; a casual observation of a group of people will reveal a large variety of postures. Some people stand straight, while others are slumped or hunched over; some people have very asymmetric postures; heads can be held at many different angles, and arms can adopt a huge variety of postures each with a different meaning: hands on hips or in pockets; arms crossed; scratching the head or neck, or fiddling with clothing. Computer animated characters often lack this variety of expression and can seem stiff and robotic; however, posture has been relatively little studied in the field of expressive virtual characters. It is a useful cue as it is very clearly visible and can be displayed well on even fairly graphically simple characters.

Research on posture generation has been limited relative to other modalities. Cassell *et al.*[7] have investigated shifts of postures and their relationship to speech, but not the meaning of the postures themselves. As such their work is

complimentary to ours. Coulson[14] uses an OCC model of emotion to generate postures. Bécheiraz and Thalmann[4] use a one-dimensional model of attitude, analogous to our affiliation, to animate the postures of characters. Their model differs from ours in that it involves choosing one of a set of discrete postures rather than continuously blending postures. This means that it is less able to display varying degrees of attitude or combinations of different attitudes.

The generation of gestures has been studied by a number of researchers. For example, Cassell *et al.*[6] have produced a character capable of extensive non-verbal behaviour including sophisticated gestures. Chi, Costa, Zhao and Badler[11] present a way of generating expressive movements, similar to gestures, using Laban notation. Gestures are closely related to speech and should be tightly synchronised with it. Cassell *et al.*[8] present a system that parses text and suggests appropriate gestures to accompany it. Gestures are less closely related to attitude than posture, though some connection can be made, for example head nodding while listening is a generally affiliative gesture.

As described in the previous section the attitude model generates a high level description of the behaviour of the character in terms of a value of each of a number of behaviour types. The behaviour modules themselves must translate this description into concrete behaviour. Each behaviour type can be expressed as a posture in a number of different ways, for example space filling can involve raising to full height or putting hands on hips while closeness can be expressed as leaning forward or making a more direct orientation (or some combination thereof). Actual postures are calculated as weighted sums over a set of basic postures each of which depends on a behaviour type.

The basic postures were designed based on the description in Argyle[1] and Mehrabian[23], combined with informal observations of people in social situations. The weights of each basic posture are the product of the value of its behaviour type and its own weight relative to the behaviour type. The weights of the basic postures are varied every so often so that the character changes its posture without changing its meaning, thus producing a realistic variation of posture over time. This is done using random group terms as shown in figure 3. Each basic posture is represented as an orientation for each joint of the character and final posture is calculated as a weighted sum of these orientations.

Gesture is generated using the same body animation system as postures, the main difference being that gestures are multi-frame animations and so weighted sums must be performed over a number of frames. They are also no longer merely static poses that can be held for a period of time; they must be repeated at appropriate intervals. More importantly gestures are more closely integrated with the flow of conversation and so must be synchronised with conversation. Of course as the conversation is textual the synchronisation does not have to be as exact as it would be with spoken language. We also do not attempt to parse text so gestures are not strongly connected to the meaning of the text as in Cassell *et al.*[8]. Our gesture model serves only to indicate when someone is talking and to express a degree of attitude. Figure 4 shows examples of postures and gestures.

4.2 Eye gaze

Natural eye gaze is critical to the realism and believability of an animated character. This is because eye gaze is fundamental in showing interest levels

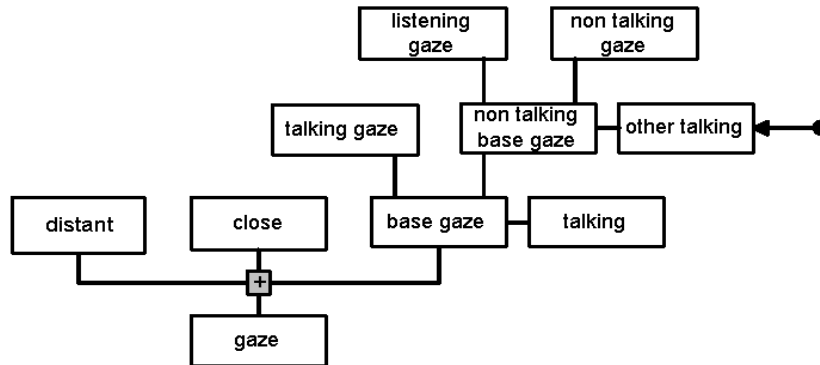


Figure 5: The section of behavioural controller dealing with eye gaze.

between characters and as means of anticipating events. Typically a person will look to another before exhibiting any behaviour, such as moving towards them or speaking to them. In conversation, a listener will typically spend a large proportion of their time looking at the speaker. A complete lack of gaze towards the speaker is a clear message of the lack of interest of the audience towards the speaker and will be picked up very quickly. Conversely, mutual gaze, in which two people are looking into each others' eyes is a powerful mechanism that induces arousal in the individuals, so typically mutual gaze is short (of the order of a second).

Argyle and Cook[2] have done extensive studies with pairs of individuals to understand levels of eye gaze, and mutual gaze, and has detailed results covering (among other things) conversations and the level to which individuals will look at the other while speaking (35%) and listening (75%) etc. We have used these results to influence our model of gaze and mutual gaze in-group settings. Eye gaze is also related to attitude. Higher affiliation results in higher levels of eye gaze. Argyle and Cook have demonstrated compensatory behaviour for eye gaze. People react to higher levels of eye gaze by reacting with more distant postures, and conversely people will look at each other less if they are placed close together.

Existing simulations of eye gaze fall into two broad categories. Gillies and Dodgson [17] and Chopra-Khullar and Badler[12] simulate the eye gaze of characters navigating and performing actions in an environment but do not handle social factors of gaze between people. Our work is closer to the other type of simulation that deals primarily with social gaze. Garau, Slater, Bee and Sasse[15] and Colburn, Cohen and Drucker[13] simulate the patterns of eye gaze between pairs of characters based on frequencies of mutual gaze. Vilhjálmsón and Cassell [31] use eye gaze to help regulate the flow of conversation by indicating when a speaker is about to finish talking, when someone wants to start or end a conversation and other similar information. Rickel and Johnson[27], in their character based virtual reality tutoring system, use gaze primarily as a method of indicating to the user an area of interest in the environment. Thórisson[29] simulates eye gaze in the context of more general work on multi-modal communicative behaviour during conversation.

Each character has a set of foci of interest, which are objects that it will

look at. The level of interest is specified as the proportion of time spent looking at that object. So for example if the character is in conversation with another character, while talking the level of gaze will be set to (say) 35%, and whilst listening to about 75% to approximate the natural gaze levels in conversation between two people.

The threshold and mean gaze values are generated by the Demeanour framework. As described above levels of gaze are different depending whether a person is talking, listening to some one else talking or neither talking nor listening. These three conditions are distinguished based on the “talking” and “other talking” parameters shown in figure 5. The “talking” parameter is set when the character is speaking and the “other talking” parameter is an import parameter that allows access to the “talking” parameter of other characters. The behavioural controller has a separate base gaze value for each of the talking conditions, which are parameters of the framework as shown in figure 5. One of these values is chosen based on switch nodes based on the “talking” and “other talking” parameters. Thus the “non talking base gaze” is either equal to “listening gaze” or “non talking gaze” depending on whether the “other talking” input is true. Using two switch nodes (“non talking base gaze” and “base gaze” in the figure) we choose one of the gaze proportions depending on the talking condition.

However, this base value is also affected by the affiliation attitude between the character that is looking and the one that is being looked at. A close attitude increases proportion of gaze (up to a maximum of 100%) and distant behaviour reduces it (to a minimum of 0%). This scaling is achieved by combining the base gaze values with the “close” and “distant” terms using a Sum of Products term as shown in figure 5.

In conversation between people a person will look at another then look away, usually by averting their gaze rather than moving their head, but they are not looking specifically at any other object, just averting their gaze. In our model we achieve this by having a number of ‘halo’ points around the head of a character that can be selected to look at if we need to look away, and have no other object that demands our attention.

5 Profiles

Demeanour provides a system of character profiles for off-line customization by end-users or world designers. A profile is a set of data that determines the unique behaviour of a character, i.e. how it differs from other characters. In Demeanour the behaviour language determines the structure of the behaviour network which controls the character’s actions. Customization is possible by altering the values of the parameter terms in the network e.g the weighting for how the closeness behaviour of other character affects a character’s affiliation. A profile can set this weighting to a positive value to achieve reciprocating behaviour, negative for compensation and a low or zero value for indifference to the other’s status.

Thus a profile consists of a number of values for parameters of the network (including weighting factors in sum of product terms). These values are stored in an XML-based format separate from the behaviour network definition. When a profile is loaded into a behavioural controller the values in the profile are used

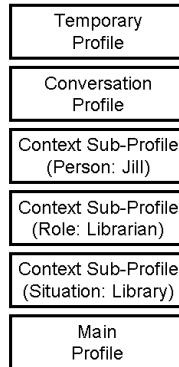


Figure 6: The profiles stack containing a number of loaded contextual profiles

to set the parameters of the controller (profile values are matched to parameters by name). Profiles are used as a means of customising a character; a means of providing contextual variation, and for regulating real time interaction. This means there will be a number of profiles loaded in a controller at any given time. They are stored in a stack as shown in figure 6. The base of the stack is always the main profile that contains the context independent customisations of a character. Above this, a number of context dependent profiles are loaded as described in section 5.2. At the top are two profiles that are used to store results of user interaction, the temporary and conversation profiles, as described in section 5.3. When a new context profile is loaded it is added above all the previously loaded profiles in the stack but below the temporary and conversation profiles. Profiles higher up the stack will override profiles lower in the stack, so recently loaded profiles override older ones and user input overrides other profiles. However, this process can be controlled by giving priorities to values within a profile. Values can have two priorities, *required* and *optional*. Required values always override values lower down the stack but optional values only override other optional values, and so are only loaded if no profile has a required value for that parameter.

5.1 Profiles for customisation

The primary function of character profiles is the customisation of characters. End users should be able to customise the behaviour of their character and designers of virtual worlds should be able to provide variety in the autonomous agents in their world. Each character has a main profile, at the base of the profile stack, containing values for the parameters of the behavioural controller that determines the unique behaviour of that character. This is the main customisation system for a given character.

To be an effective customisation method, easy to use tools must be provided for designing profiles. The most direct method is for the user to choose values to parameters whether by hand editing files or via a user interface. However, parameters are often closely linked to the internal workings of the behavioural controller and not necessarily intuitive to end-users, so this method should generally be confined to world designers and advanced users.

We propose the use of “adjectives”. These are names in natural language that describe a particular character trait or group of traits that is understandable to end-users. These adjectives are mapped onto actual settings of the internal parameters, each adjective affecting a number of parameters. For example, ‘extrovert’ might combine dominance with high affiliation while ‘easily intimidated’ might indicate compensation behaviour to dominance (i.e. responding submissively to dominant behaviour). Each adjective is a fixed set of parameter values and therefore is itself a self contained profile. They can be designed at the same time as the behaviour network, through direct profile authoring tools as above. An end-user designs their profile as a combination of the adjectives. They are presented with series of sliders each labelled with an adjective name, the values of the sliders represent the proportions of the various adjectives. The values contained in the adjectives are multiplied by the slider values and summed to obtain the final profile. This provides a customization tool that is easy to use, abstracts from the internal workings of the controller, and is itself easily customizable by world designers.

Section 5.3 describes another method of customisation that is integrated with real time control.

5.2 Profiles and context

As described in the introduction the variability of human behaviour is not solely between individuals but within individuals. People behave very differently in different contexts and it is important to also model this sort of variability. The importance of this type of adaptation is brought out in work by MacNamee, Dobbyn, Cunningham and O’Sullivan[20] and Maya, Lamolle and Pelachaud[22]. Goffman[19] provides a fascinating description of how people’s behaviour varies in different contexts.

In order to handle this sort of variability Demeanour uses a system of sub-profiles for specific context. A sub-profile is a small set of parameter values that are loaded in a given context to alter the behaviour of the main profile. These are loaded above the main profile in the stack as shown in figure 6, with more recently loaded sub-profiles overriding older ones.

The variation of a person’s behaviour in different contexts can depend on a number of different factors and so these contexts themselves can have different meanings, for example, a relationship with a colleague may define a context for interaction with that colleague but the context would also depend on whether they are at work or in a social context. We divide contextual sub-profiles into three types depending on when they are loaded and to some degree who designs them. The system could be augmented to add a number of other types.

Person sub-profiles define a relationship to a particular individual and thus represent the attitude to that person. They are loaded when an interaction starts with that individual. The parameter ‘Liking of other’ in figures 3 is an example of a parameter that represents an attitude to a particular person and is suited to inclusion in a person sub-profile. Person profiles are generally designed by end users.

Role sub-profiles represent the behaviour when the character is performing a particular role. These roles are often related to work, for example, a waiter

behaves very differently when actually serving customers than when not working or when interacting with other staff in the restaurant kitchen[19]. A practical example, might be eliminating all low affiliation behaviour when a waiter is at work to ensure politeness. Role sub-profiles can be loaded by the end-user or automatically loaded in a given situation. Role profiles can be created by world designers to give a user character specific behaviour in a role that the user might not have foreseen or they can be created by an end user.

Situation sub-profiles are specific to a particular environment or context in the world, for example, flirting behaviour might be disabled in an office environment but re-enabled in an office party context. They are loaded automatically when the character enters a situation and applied to all characters in that situation. Situation profiles are likely to be created by the designers of a virtual world for a given situation, though users might want to subsequently edit them.

5.3 Profiles and real-time control

Demeanour also contains a real-time control system where users can determine the affective state of their character through a number of commands as described in Gillies, Crabtree and Ballin[18]. When a user enters a command it not clear how long they intend a change of state to last, for example an increase in closeness might have a very short scope, just the length of the current utterance or it might indicate a permanent positive attitude to the person being talked to. Demeanour uses character profiles to allow users to choose between four different scopes for a change of state:

- A temporary change will last for a limited period, disappearing after a time out.
- Changes can be made to last for the whole length of the current conversations
- They can be permanent changes to the attitude toward the conversational partner
- Finally, changes can be a permanent change to the character's profile, allowing users to gradually update the behaviour of the character during use without extensive off-line customisation.

Initially, when a user enters a command that changes a parameter value, it is stored in temporary profile. This temporary profile is deleted after a short period of time and all the edits it contains are deleted. Thus the default scope for edits is that they are temporary. However, when a temporary profile is present (i.e. after the user sends a command) a button appears in the text chat interface allowing the user to save the profile. If the user clicks this button the temporary profile is saved into a conversation profile (as shown in figure 6), which has a longer scope lasting for the entire conversation. When temporary edits are saved into the conversation profile they are merged. This process is similar to the way values are overridden when profiles are loaded in the stack. Values from the new profile overwrite older values but *required* values take precedence over optional ones.

At the end of the conversation the user can merge the resulting edits into a permanent profile that is used in all future interactions. This can be to the character's main profile which controls its behaviour and is the chief method of customising a character. Users can also merge the conversation profile into a person profile for their conversational partner, thus developing the relationship between the characters during interaction.

Demeanour's real time control system allows users to adopt fleeting moods at an appropriate moment in a conversation. However, permanent profiles allow them to be used simultaneously as a method of long term customisation. Rather than explicitly defining a profile for the character a user can gradually correct and shape the behaviour during real time interaction. This means that the user avoids the long and possibly tedious task of defining a profile. It also means that the customisation is situated in a conversation. It is likely to be easier for a user to know what behaviour is appropriate when actually engaged in a conversation than to think about it abstractly during an off-line customisation step. The first factor is likely to be particularly important when creating contact profiles. Users are unlikely to want to spend time creating a new profile by hand for each new person they meet.

6 Conclusion

We have described the Demeanour framework's system of profiles which is applied in a number of ways, end user customisation, context dependence and real time control. This system shows promise in producing some of the variability and adaptability of real human behaviour. We are currently planning user trials to test the effectiveness to the methods described.

Acknowledgements

We would like to thank BT Exact for sponsoring this research. We would also like to thank the UCL Department of Computer Science Virtual Environments and Computer Graphics group for their help and support, and Amanda Oldroyd for the use of her character models.

References

- [1] M. Argyle, *Bodily communication*, Routledge, 1975.
- [2] M. Argyle and M. Cook, *Gaze and mutual gaze*, Cambridge University Press, 1976.
- [3] N. Badler, C. Philips, and B. Webber (eds.), *Simulating humans: Computer graphics, animation and control*, Oxford University Press, 1993.
- [4] P. Bécheiraz and D. Thalmann, *A model of nonverbal communication and interpersonal relationship between virtual actors*, Proceedings of the Computer Animation '96, IEEE Computer Society Press, June 1996, pp. 58–67.
- [5] B. Blumberg and T. Galyean, *Multi-level direction of autonomous creatures for real-time virtual environments*, ACM SIGGRAPH, 1995, pp. 47–54.

- [6] J. Cassell, T. Bickmore, L. Campbell, K. Chang, H. Vilhjálmsón, and H. Yan, *Embodiment in conversational interfaces: Rea*, ACM SIGCHI, ACM Press, 1999, pp. 520–527.
- [7] J. Cassell, Y. Nakano, T. Bickmore, C. Sidner, and C. Rich, *Non-verbal cues for discourse structure.*, 41st Annual Meeting of the Association of Computational Linguistics (Toulouse, France), 2001, pp. 106–115.
- [8] J. Cassell, H. H. Vilhjálmsón, and T. Bickmore, *BEAT: the behavior expression animation toolkit*, ACM SIGGRAPH, 2001, pp. 477–486.
- [9] L. Cheng, S. Farnham, and L. Stone, *Lessons learned: Building and deploying virtual environments*, The Social Life of Avatars, Presence and Interaction in Shared Virtual Worlds (Ralph Schroeder, ed.), Computer Supported Cooperative work, Springer, 2002.
- [10] L. Cheng, L. Stone, S. Farnham and A. N. Clark, and M. Zaner, *Hutchworld: Lessons learned a collaborative project: Fred hutchinson cancer research center and microsoft research*, Virtual Worlds (Jean-Claude Heudin, ed.), Lecture Notes in Artificial Intelligence, Springer, 2000.
- [11] D. Chi, M. Costa, L. Zhao, and N. Badler, *The emote model for effort and shape*, ACM SIGGRAPH, ACM Press/Addison-Wesley Publishing Co., 2000, pp. 173–182.
- [12] S. Chopra-Khullar and N. Badler, *Where to look? automating visual attending behaviors of virtual human characters*, Autonomous Agents Conference, 1999.
- [13] A. Colburn, M. Cohen, and S. Drucker, *The role of eye gaze in avatar mediated conversational interfaces*, Tech. report, Microsoft Research, 2000.
- [14] M. Coulson, *Expressing emotion through body movement: a component based approach*, AISB workshop on Animating Expressive Characters for Social Interactions (Imperial College London) (Ruth Aylett and Lola Cañamero, eds.), April 2002.
- [15] M. Garau, M. Slater, S. Bee, and M. A. Sasse, *The impact of eye gaze on communication using humanoid avatars*, ACM SIGCHI, 2001, pp. 309–316.
- [16] M. Gillies and D. Ballin, *A model of interpersonal attitude and posture generation*, Fourth Workshop on Intelligent Virtual Agents (Kloster Irsee, Germany) (Thomas Rist, Ruth Aylett, Daniel Ballin, and Jeff Rickel, eds.), September 2003.
- [17] M. Gillies and N. Dodgson, *Eye movements and attention for behavioural animation*, Journal of Visualization and Computer Animation **13** (2002), 287–300.
- [18] Marco Gillies, Barry Crabtree, and Daniel Ballin, *Expressive characters and a text chat interface*, AISB workshop on Language, Speech and Gesture for Expressive Characters (University of Leeds) (Patrick Olivier and Ruth Aylett, eds.), March 2004.

- [19] Erving Goffman, *The presentation of self in everyday life*, Pelican, 1972.
- [20] Brian MacNamee, Simon Dobbyn, Padraig Cunningham, and Carol O’Sullivan, *Men behaving appropriately: Integrating the role passing technique into the aloha system*, AISB workshop on Animating Expressive Characters for Social Interactions (Imperial College London) (Ruth Aylett and Lola Cañamero, eds.), April 2002.
- [21] S. C. Marsella, W. L. Johnson, and C. LaBore, *Interactive pedagogical drama*, the proceedings of the 4th international Conference on Autonomous Agents, 2000, pp. 301–308.
- [22] Vincent Maya, Myriam Lamolle, and Catherine Pelachaud, *Influences on embodied conversational agent’s expressivity*, AISB workshop on Language, Speech and Gesture for Expressive Characters (University of Leeds) (Patrick Olivier Ruth Aylett and Marc Cavazza, eds.), March 2004, pp. 75–85.
- [23] A. Mehrabian (ed.), *Nonverbal communication*, Aldine-Atherton, 1972.
- [24] A. Mortlock, D. Machin, S. McConnell, and P. Sheppard, *Virtual conferencing*, BT Technology Journal **14** (1997), no. 4, 120–129.
- [25] C. Pelachaud and I. Poggi, *Subtleties of facial expressions in embodied agents.*, Journal of Visualization and Computer Animation. **13** (2002), 287–300.
- [26] K. Perlin and A. Goldberg, *Improv: A system for scripting interactive actors in virtual worlds*, Proceedings of SIGGRAPH 96 (New Orleans, Louisiana), Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH / Addison Wesley, August 1996, pp. 205–216.
- [27] J. Rickel and W. L. Johnson, *Animated agents for procedural training in virtual reality: Perception, cognition, and motor control*, Applied Artificial Intelligence **13** (1999), 343–382.
- [28] S. J. Soltysiak and I. B. Crabtree, *Automatic learning of user profiles – towards personalisation of agent services*, BT Technology Journal **16** (1998), no. 3, 110–117.
- [29] K. Thórisson, *Real-time decision making in multimodal face-to-face communication*, second ACM international conference on autonomous agents, 1998, pp. 16–23.
- [30] X. Tu and D. Terzopoulos, *Artificial fishes: Physics, locomotion, perception, behavior*, ACM SIGGRAPH, 1994, pp. 43–49.
- [31] H. H. Vilhjálmsson and J. Cassell, *Bodychat: Autonomous communicative behaviors in avatars*, second ACM international conference on autonomous agents, 1998.