Bringing Robots To Life: Applying Principles Of Animation To Robots

A.J.N. van Breemen
Philips Research
Prof. Holstlaan 4 (WDC 1.34)
5656 AA Eindhoven, The Netherlands
albert.van.breemen@philips.com
Tel: +31 40 27 43440

ABSTRACT
The acceptance of user-interface robots as "social friend" depends among others on the ability of the user to understand the robot's behavior − to understand what it is doing and thinking. Body gestures are a natural channel to communicate this. Traditionally, the control of robotic body parts is carried out by feedback control loops. This results, however, in rather machine-like behavior that does not reveal much about what the robot is doing or thinking. We argue that in order to bring robots to life − such that they show behavior that can be naturally understood and anticipated − principles known from the field of character animation should be applied. In this article we will discuss this idea and present results obtained from experiments with our user-interface robot "iCat".

Author Keywords
Robot, user-interface robot, character animation, Ambient Intelligence.

ACM Classification Keywords
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION
In the consumer electronics market a new genre of robots is introduced, known by names such as: **Personal Robots**, **Service Robots** or **User-Interface Robots** [7]. The application domain of these devices varies between messaging, multimedia applications (e.g. taking photos, playing melodies, telling stories), web-services and gaming. Some examples of these robots are AIBO ERS-7 [10], PaPeRo [8], ER1 Robot [6], and RoboScout [11].

Users create a special relationship with their user-interface robot and consider them as being a "social friend". Toys such as the "Tamagotchi" and "Furby" already introduced this aspect a few years ago. Owners of these toys are responsible for taking care of their toys − feeding them, playing with them and talking to them. From the robot's perspective, a good social relationship opens possibilities to build rich user profiles that can be applied to personalize the robot's tasks. From the user's perspective, personalization adds value to the robot application. However, in order to preserve the social relation it is important that the user understands the robot's behavior and is able to anticipate it [2].

Animated User Interface Characters have been applied for similar purposes as user interface robots and also here the principles of animation have proven their value [5].

Robot behavior manifests itself, besides sound and lights, through moving body parts. The movements of these parts reveal a lot about what the robot is doing and thinking. Traditionally, feedback loops control the movement of the body parts. For instance, an object tracking behavior is created by a feedback loop between the estimated object position from a camera and the servos of the robot's head [4]. This results in *machine*-like behavior that − in contrast to *life*-like behavior − can not be naturally interpreted.

Early day's animators dealt with a similar problem: creating the illusion of life of characters on paper, such that the audience understands what the character is doing and thinking [12]. Principles of animation were discovered that are the basis for creating the illusion of life. We consider user-interface robots to have the same problem of early day's animations: they miss the illusion of life, which is so important to be understandable for users. In this article we propose to apply the principles of animation to user-interface robots and show several results of this idea.

The remainder of this article is organized as follows. In the next section we present user-interface robots and discuss the problem of understanding and anticipating their behavior. Section 3 presents principles of animation and summarizes the twelve main principles. Section 4 explains how these animation principles can be applied to robots and present results obtained during experimentation with our user-interface robot "iCat". Section 5 summarizes our work.

USER-INTERFACE ROBOTS

Instead of being a universal house-hold robot like "Rosie" from "The Jetsons", user-interface robots currently entering the consumer market have specialized application domains. At Philips Research we are currently researching the usage of user-interface robots for home automation. The user-interface robot's main purpose is to provide an interface to devices in an Ambient Intelligence Home environment such as the "HomeLab" [1].

During previous research we build Lino (see figure 1), a mobile user-interface robot with a special head to create facial expression [3]. Lino is able to recognize spoken commands, to autonomously navigate in the home and to recognize objects by using vision. Our latest user-interface robot is the "interactive Cat", or just "iCat" (see figure 1). In contrast to Lino, iCat is smaller and lacks mobility, so that we can solely focus on the robot-human interaction. The iCat has 13 servos to move different parts of the head in order to create facial expressions, a stereo microphone to determine the direction of sound and a build-in webcam. Furthermore, iCat can be connected to a home network to control devices (e.g. VCR, TV) and to use the Internet.

Figure 1. Our user-interface robots Lino and iCat.

It's iCat's task to recognize users, build profiles of them and handle user requests. The profiles are used to personalize different kind of ambient functions performed by the robot. For instance, different light and sound conditions are used for one user asking iCat to create a 'relaxing atmosphere' and another user requesting the same. In order to learn rich user-profiles, a good social relationship between the iCat and the user is required, because both should understand each other and be willing to spend time in teaching each other things about themselves. As argued in the introduction, this relationship only lasts when the user is able to understand what iCat is doing and thinking.

From preliminary experimentation we observed that facial expressions can reveal what iCat is thinking. In one particular experiment a user had to tell iCat that "Madonna has released a new album". iCat reacted by showing a surprised face and said "I didn't know that". From this reaction, the user anticipated that iCat did like Madonna very much − something that is hard to infer when the sentence "I didn't know that" was spoken without any emotion. Experiments like this one let us hypothesize that the complete robot's reaction, including the movements of all its body parts, need to be properly choreographed in order to communicate the right message to the user.

PRINCIPLES OF ANIMATION

More is needed to create naturally understandable behavior then only having a robot react to stimuli from the environment. People need to be able to explain the behavior of the robot in terms of intentional, mental and emotional states [2]. Thomas and Johnson [12] mention in *Illusion of Life* that change of expression can reveal the thought process of a character: "It is the change of shape that shows the character is thinking. It is the thinking that gives the illusion of life. It is the life that gives the meaning to the expression." (p471).

Over the years animators have discovered several fundamental principles to animate characters. The next sections present twelve principles of animation mentioned by [12].

Squash and stretch

Squash and stretch is one of the most important principles of animation. Moving objects with a fixed shape appear to be very rigid (e.g. a chair or table). The movement emphasizes the rigidity. However, moving objects that change their shape appear to be flexible. Things made of living flesh always change their shape while moving (e.g. a moving arm with swelling biceps). Living objects are therefore always animated using the squash and stretch principle.

Anticipation

A viewer watching a character will not understand its actions unless these actions are preceded by a clearly planned sequence of other actions. This sequence of preceding actions is known as anticipation. For instance, before throwing a ball the character first swings its arm backwards.

Staging

Staging is the presentation of any idea so that it is completely and unmistakable clear. Actions, personality, expressions and moods can all be staged. If a character is sad you should not play happy music. If you want to stage an action, be sure only one action is seen. If too much happens the user might become confused.

Straight Ahead Action and Pose-to-Pose

There are two basic methods to create animations. The Straight Ahead Action method works from the first drawing to the next until the final drawing is reached. The Pose to Pose method starts with planning the key drawings and then starts to draw the in-betweens. Whereas the first method results often in more spontaneous animations, the second methods results in clearer animations with more strength.

Follow Through and Overlapping Action

Whenever the action of a character is suddenly stopped they appear to be stiff and unnatural. For instance, it is unnatural for a character that throws a ball to stop moving its arm when the ball is just released. Therefore, just like anticipation is used to precede an action, a sequence of actions is needed to end a major action.

A Follow Through is a sequence of actions that follows the major action. For instance, when throwing a ball, the character first moves backwards (anticipation), then throws and releases the ball (major action) and finally its arm overshoots and keeps moving along the direction of movement and finally returns to the end position (follow through).

An Overlapping Action is an action that is caused by the major action. For instance, a dog with long ears that moves and stops (major action) will have ears that keeps moving for a little while (Overlapping Action).

Slow In and Slow Out

The Slow In and Slow Out principle states that the movement of objects seems to become more natural whenever the in-betweens are close to the “extremes” or key drawings. Only one in-between should be drawn between two key drawings. Slow In and Slow Out causes objects to move nonlinear.

Arcs

Movements of living organisms always occur in arcs. Movements in straight lines are very mechanical and seldom performed by characters or parts of characters.

Secondary Action

A Secondary Action is a supplementary action that supports the main action. Secondary Actions make scenes richer and more natural. For instance, a sad person that wipes a tear as he turns away.

Timing

Timing is essential to animation. By changing the number of inbetweens the meaning of a movement can be changed. For instance, a fast eye blink makes the character alert and awake, whereas a slow eye blink makes the character tired.

Exaggeration

Exaggeration is used to accent actions, personalities and expressions in order to make them more convincing. It creates a bigger contact with the audience. So, if a person is sad, make him sadder. If an object is bouncing, make it really bounce.

Solid Drawing

The basics of solid, three-dimensional drawing are weight, depth and balance. Drawings that are very symmetrical (e.g. arms and legs having the same pose) appear to be very stiff and “wooden”. A character becomes more natural when each part of its body varies in some way from the corresponding opposite part.

Appeal

Appeal means anything that a person likes to see, a quality of charm, pleasing design, simplicity, communication and magnetism. Drawings lack appeal when they are too complex or hard to read.

ANIMATING ROBOTS

The traditional way to control a robot’s movements is to use feedback control loops that let the robot react to stimuli from the environment. However, this results in very machine-like behavior − constant velocities and moving in straight lines. More life-like behavior is created by designing pre-programmed motions based on the principles mentioned in the previous section. These motions could be stored in a library and played during the right situation.

In practice, a mixture between pre-programmed motions and feedback loops is required. The pre-programmed motions are designed to make the robot better understandable, whereas feedback loops let the robot react to stimuli from the environment. This mixture has implications for the software architecture of the robot. In the world of games specialized animation layers that combine pre-programmed motions and feedback loops are already known for virtual characters [9]. We applied these ideas to our iCat and developed a set of motions to animate iCat. The motions were carefully designed using the principle of animation mentioned in the previous section. Below, we will discuss two of them: turning iCat’s head to the left and letting iCat fall asleep.

Example 1: Turning to other side

There are many ways in which a character can turn its head, and there are also many reasons why a character would do it. For instance, a character might turn its head because it has been hit, or the character’s attention was caught suddenly, or the character wants to look severe at somebody in order for that person to be silent. In the first case the character will move the head very fast. In the second case the character will move its head quickly and will have its eyes and mouth wide open (probably with fast eye blinks). In the last case the character will move its head slowly, look angry and will have very few eye blinks.

Figure 2 (top) shows iCat turning to the left in a feedback loop-like manner: the robot moves its head with constant velocity. This case simulates for example, the robot tracking an object. The movement is unnatural, especially when we focus on the eyes of iCat. They just look into infinity, something we don't expect from living things. The iCat behaves zombie-like.

Figure 2 (bottom) shows an animated "turn to the left" movement. First, the eyes of iCat move to the left, as if iCat sees something at its left side. This action is used to prepare the user for the major action: turning to the left. An eye blink is added as secondary action to make the scene more natural. Also, all movements (head and eyelids) are performed using slow in and slow outs.

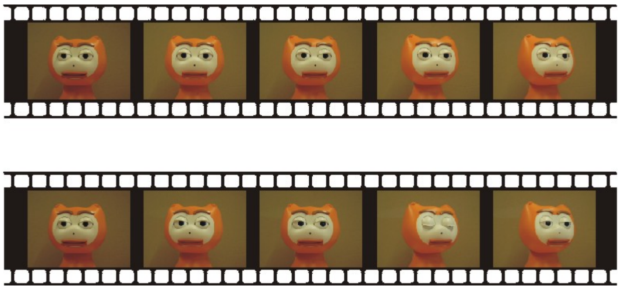

Figure 2. Robot animation example "turning to the left". Top: linear motion, Bottom: applying principles of animation.

Example 2: Falling asleep

Although there is no physiological reason why a user-interface robot would fall to sleep, every living being has a rhythm of being awake and being asleep. Suddenly falling asleep by closing the eyes and lowering the head is unnatural. Therefore, we use again some principles of animation in order to make the scene more natural. Figure 3 shows the movements of the iCat falling to sleep. The scene starts with an anticipating action, namely iCat first yawns. This should prepare the user for the major action: falling to sleep. Again, the falling to sleep movement is carried out by using slow in and slow outs.

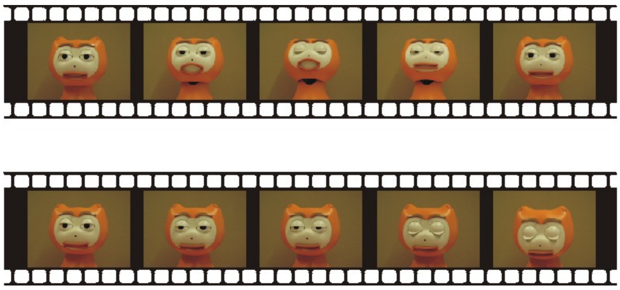

Figure 3. Robot animation example "falling to sleep".

As mentioned in the introduction, besides moving body parts, robot behavior manifests itself also by sound − e.g. music, noises and speech − and light. Robot movements alone are not enough for presenting a particular action. In addition to animated movements, spoken words, previous actions, background sounds and lights all contribute to the staging of an action.

SUMMARY

In this article we argued that the social relationship between a user and a user-interface robot depends on the degree a user is able to understand the robot's behavior and whether it can anticipate it. The way in which a robot moves reveals a lot about what the robot is doing and thinking. Therefore, robot movement needs to be carefully controlled.

Traditionally, feedback loops control the robot's movements. This results however in machine-like behavior that does not tell a user what the robot is doing and why it is doing it. We proposed to apply principles of traditional animation to make the robot's behavior better understandable. Examples from our user-interface robot iCat were given to illustrate this idea.

Using a preprogrammed set of carefully designed animation motions requires a change in the software architecture of the robot. A general architecture that merges pre-programmed motions and feedback control loops is needed.

REFERENCES

1. Aarts, E., Harwig, R., & Schuurmans, M. (2001). Ambient Intelligence. In P.Denning (Ed.), The Invisible Future, pp. 235-250, New York: McGraw Hill.
2. Breazeal, C. (1999), "Robot in Society: Friend or Appliance?". In Agents99 workshop on emotion-based agent architectures, Seattle, WA. 18-26.
3. Breemen, A.J.N. van, Crucq, K., Krose, B.J.A., Nuttin, M., Porta, J.M. and Demeester, E. (2003). "A User-Interface Robot for Ambient Intelligent Environments". In Proc. of ASER 2003, Bardolino, Italy, pp.176-182.
4. D'Azzo, J.J. and Houpis, C.H. (1995). Linear Control System Analysis and Design, McGraw-Hill, Singapore, 4th edition.
5. Diederiks, E. (2003). "Buddies in a Box: Animated Characters in Consumer Electronics" In Proceedings of IUI 03< Miami, USA, pp. 34-38.
6. Evolution Robotics (2004), ER1 Personal Robot System, http://www.evolution.com .
7. Krose, B.J.A., Porta, J.M., Breemen, A.J.N. van, Crucq, K., Nuttin, M. and Demeester, E. (2003). "Lino, the User-Interface Robot", First European Symposium on Ambient Intelligence (EUSAI 2003), November 3-4, Veldhoven, The Netherlands, pp. 264-274.
8. NEC (2004), Personal Robot PaPeRo, http://www.incx.nec.co.jp/robot/PaPeRo/english/p_index.html .
9. Rabin, S. Ed. (2002), AI Game Programming Wisdom, Charles River Media.
10. Sony (2004), AIBO ERS-7, http://www.aibo.com .
11. Sharper Image Design (2004), RoboScout, http://www.roboscout.com .
12. Thomas, F. and Johnson, O. (1981), The Illusion of Life – Disney Animation, Walt Disney Productions.