Action Through Color Recognition for Sony's AIBO

M. Offerhaus, B. Postma, T. A. Unger

Abstract

Machines can behave more intelligently when able to analyze visual information. This information can be used to gain and remember knowledge about its environment, to inform future actions.

To demonstrate this, a specimen of Sony's AIBO was programmed to recognize certain colors, while ignoring others, and behave according to these observations.

Further research was initially considered, but ultimately not carried out due to time constraints.

The research that *was* carried out lead to the conclusion that current consumer-level technology is capable of analyzing visual information and can base its actions upon it.

1 Introduction

The sense of vision is a nearly universal trait for animals. By humans, cats and dogs, a significant amount of information is derived from this sense alone. Enabling machines to take advantage of visual information might therefore enhance their intelligence in interpreting their environment, as well as their ability to communicate that intelligence through visual communication.

The goal for the research described in this paper was to find out whether it is possible to program a specimen of Sony's AIBO (*Artificial Intelligence Robot*), model ERS-7, to:

- recognize and differentiate two different persons using the AIBO's on-board video camera,
- bring an object to either of these persons (using a random number generator), and

- learn that it would be consistently "rewarded" (e.g. touched on its head sensor) for bringing this object to one of these persons, but not the other, and let this learned knowledge create a bias of the random number generator towards choosing the course of action that leads to reward.

If the AIBO can be programmed to perform the actions stated above, it can be concluded that current consumer-level technology is capable of interaction with specific individuals and base its actions on knowledge about specific individuals.

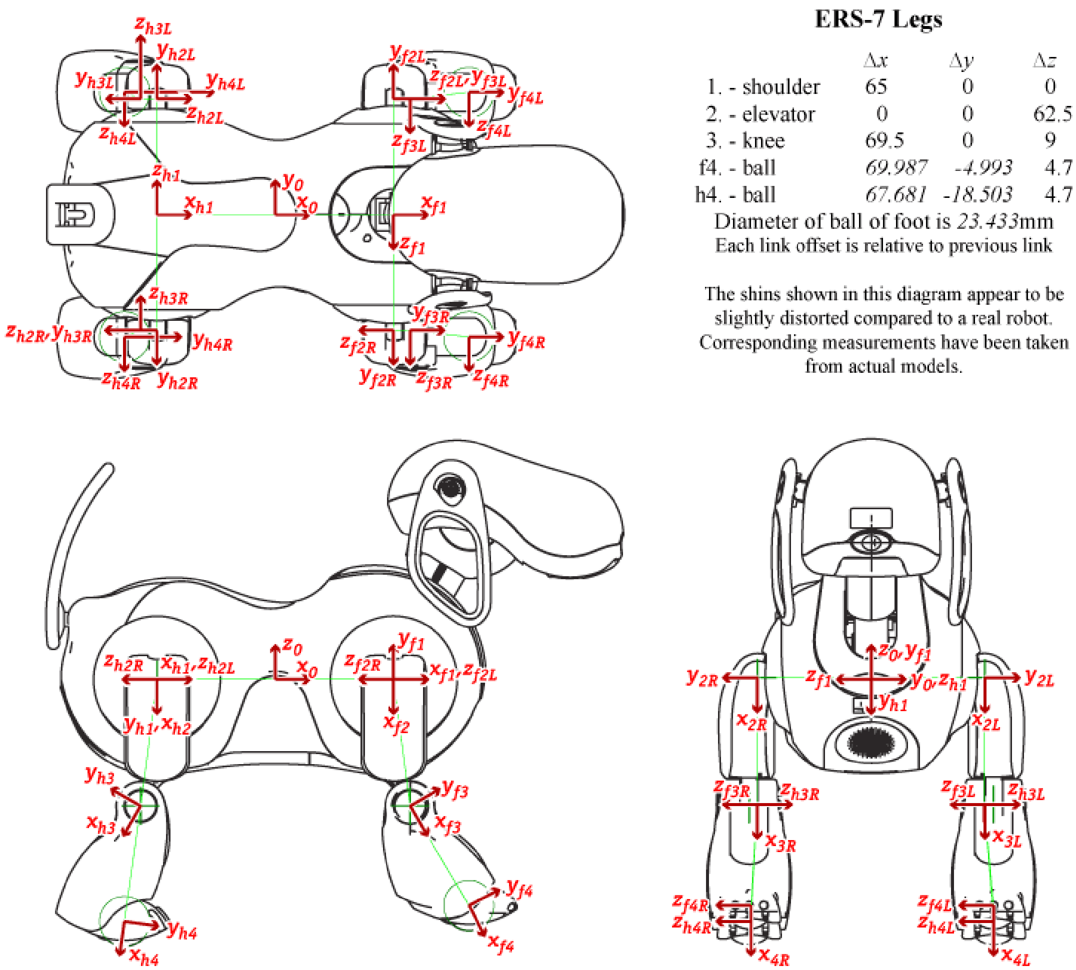

Figure 1: Schematic from Sony's documentation of the AIBO model used, containing the reference frames of the legs according to the Denavit–Hartenberg convention.

Earlier research has shown that the AIBO is indeed able to derive information from its on-board video camera, while using its on-board computer (Hasanuzzaman et al., 2004), or while being reliant on other computers (Geusebroek and Seinstra, 2005). To be clear, for the research in this paper only the AIBO's on-board computer was used to process visual information.

Other research has also confirmed the AIBO's ability to learn from interactions with its environment (Ullerstam and Mizukawa, 2004).

The AIBO runs a proprietary operating system, 'Aperios', developed by Sony, and accepts instructions written in the OPEN-R language (based on C++), also developed by Sony for use with the AIBO.

Given that prior experiments concerning visual recognition and learning using Sony's AIBO had been successful, it was hypothesised that the AIBO can be made capable of achieving the set goals by using the OPEN-R language.

2 Research Approach

Of primary importance was the ability to transfer newly written code to the AIBO. The AIBO has a slot for a memory stick on its underside. Code could be read from and written to files on this memory stick from a Linux, Windows or Mac OS X computer, of which the latter two were used. This also necessitated the use of a card reader which had been provided.

Although the material means to do research were available, little documentation had been provided on programming the AIBO. As a result, progress in this area was slow.

After a considerable amount of time investigating the files on the available memory sticks, the file `urbi.ini` was identified as likely being the file that is used for custom code. To confirm this suspicion, several commands in the file were replaced. Within `urbi.ini`, another file was referenced, named `motion.u`, as holding "walk functions". Referring to this file, which was available on the same memory stick, several commands could be found. One of these commands was used in place of the commands that had been removed earlier, in `urbi.ini`. The memory stick was then inserted into the AIBO's slot and, after starting up, the behavior changed as expected and according to the changes made.

This process also confirmed, incidentally, that the changed files did not need to be compiled before inserting the memory stick into the AIBO. While this was convenient in one respect, it was less convenient in tracking down syntax errors. To at least confirm or disconfirm whether code contained such an error, the execution of custom code would begin by turning on some of the LED lights in the AIBO's head. Would these not turn on, then it would be confirmed that the code contained a syntax error. This occurred several times throughout testing, but was ultimately inconsequential and shall not be discussed further.

By further inspecting the code already present on the memory stick, two types of statements were discovered. These were the 'at' and 'whenever' statements. These statements can be likened to 'if' statements commonly found in programming languages. In short, they allow certain code to execute, depending on whether or not something is true, e.g. whether a sensor is touched for a given amount of time, or whether something is visible. The latter was achievable using a "colormap" object, as was subsequently discovered.

The colormap object takes ranges of values in the YCbCr color space, consisting of a brightness channel (Y) and two color channels (Cb and Cr), which is different from the more commonly known RGB (red, green and blue) color space. The value of each of these three channels ranges from 0 to 255 (for a total of $2^8 = 256$ values, i.e. a byte). For the colormap object, a sequential subset of this range could be specified for each channel. When a captured image contains a pixel that falls within the specified range for each channel, the colormap object is recognized as "visible".

As a preliminary test, a colormap object was created that specified a range of 0–255 for each of the channels, meaning any value would be recognized. The AIBO was programmed to make a sound, then walk forward, when this object was visible.

Letting the AIBO execute this code, it did not behave as intended. The AIBO would make the sound, as expected, but move its legs only slightly and then repeat the sound and the slight movement for as long as the AIBO remained on.

It was hypothesised that this behavior was caused by the AIBO continuously recognizing the colormap object and continuously trying to start walking, effectively preventing itself from doing so. In an attempt to remedy this, a boolean variable was used, which would be set to “true” whenever the object was visible, prompting the AIBO to walk, and would be set back to “false” only when the AIBO had finished walking. After another test, it was found that this indeed remedied the AIBO’s unintended behavior.

Following these preliminary tests, the range of each channel was incrementally limited and tested, so that eventually only a small amount of values would be recognized. For each test, the AIBO was placed with its camera pointing to a black background. Several objects with a single, striking color were then held in front of the camera. After several iterations, the AIBO would only make a sound and walk when presented with a pink object, but not with a green, yellow or blue object.

At this point, most of the time allotted for research had unfortunately passed.

For the time remaining, a variation was made on the behavior produced thus far. The AIBO was instructed to start rotating for 10 seconds, or until the object specified would enter view. This proved to be rather unreliable, since the AIBO’s environment consisted of many colorful objects, as other research teams were also present in the same work area. However, the AIBO would stop, as intended, when a pink object was purposefully entered into the view of the camera.

3 Results and Discussion

Despite the goal initially set proving too ambitious, this research, albeit limited, shows that machines can display intelligent behavior through the use of visual information. If made identifiable by the color of their clothes, different individuals can be recognized as such by the AIBO.

Given that earlier, similar research has shown that the AIBO is capable of recognizing specific visual traits and is capable of learning, it does not seem any less likely than before the start of this research that the goals initially set are achievable, given a greater amount of time.

This research also shows that while visual recognition based on whether or not a certain color is visible can produce acceptable results, it is not adequate in real-world applications. More complex analysis of images, such as recognition of shape, size and location within the field of view, as well as temporal analysis and integration with other sensory information, would arguably produce more reliable behavior, although, in the case of the AIBO, its hardware may prove a limiting factor for some or all of these additional forms of analysis.

References

Geusebroek, J.-M. and Seinstra, F. (2005). Object recognition by a robot dog connected to awide-area grid system. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 1565–1566.

Hasanuzzaman, M., Ampornaramveth, V., Zhang, T., Bhuiyan, M., Shirai, Y., and Ueno, H. (2004). Real-time vision-based gesture recognition for human robot interaction. In *Robotics and Biomimetics, 2004. ROBIO 2004. IEEE International Conference on*, pages 413–418.

Ullerstam, M. and Mizukawa, M. (2004). Teaching robots behavior patterns by using reinforcement learning: how to raise pet robots with a remote control. In *SICE 2004 Annual Conference*, volume 1, pages 143–146 vol. 1.