The Look, the Feel and the Action: Making Sets of ActDresses for Robotic Movement

Mattias Jacobsson, Ylva Fernaeus
Swedish Institute of Computer Science
Box 1263, 164 29 Kista, Sweden
[majac, ylva]@sics.se

Rob Tieben
Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, Netherlands
rob@robtieben.com

ABSTRACT
We present a series of design explorations for controlling autonomous robotic movement based on a metaphor of clothing and accessorising. From working with various sketches, scenarios and prototypes we identify a number of particular features of this form of interaction, as well potential challenges for designers of other systems based on this design concept. Finally we conclude with a few general implications, especially concerning the inert properties of visibility, physicality and modularity with respect to the particular case of interaction and robotic movement.

Author Keywords
Interaction design, Human robot interaction, Tangible interaction, Physical user interfaces

ACM Classification Keywords
I.2.9 Robotics: Operator Interfaces.
H.5.2 User Interfaced: User Centred Design.

INTRODUCTION
In the research areas of human-computer and human-robot interaction, as in interaction design at large, the physical forms of artefacts and how these shape the behaviours that people expect from them is becoming increasingly relevant to explore. Benford et al [1] have conceptualised this general theme of explorations in terms of what is sensed, desired and expected, and similarly Djajadiningrat et al [6] have discussed a related problem as striving to find a balance between appearance and action.

In our research a primary goal is to design for meaningful interaction, grounded in existing practices that can be or have been studied and observed among people. Interesting with respect to this is that it has been increasingly noted how people personalise their digital devices by different forms of physical means. Laptops are made personal by placing stickers on them, people buy or make their own customised cases, and they attach mascots and charms to their mobile phone handsets. Especially in the area of robotic consumer products, people have been reported to physically accessorise their technology, e.g., by sewing decorative covers for Roomba [17] and dressing up Pleo for different occasions [9].

Moreover, given the fundamental role that clothing has in human culture, and the interest in surface appearance in product design, our hypothesis is that this may be used much more concretely also as resources for controlling the software of physical devices. The concept of clothing may for instance provide a valuable link between physical hardware, expected behaviours and capabilities, and the varying contexts of use. For this purpose, we developed the concept of ActDresses [7] as an approach to designing for interaction where physical markings can be directly attached to a digital artefact to add or modify some digital property, action, or behaviour of that artefact.

In this paper we explore how such physical markings can be used for the specific case of controlling, programming, and predicting the movement of physical interactive artefacts such as robotic vacuum cleaners or toy vehicles (see Figure 1).

After a short overview of related work, we present a series of design explorations on this theme using scenarios, mock-

Figure 1. Four different design explorations, all based on the theme of controlling autonomously moving artefacts by attaching physical labels, signs or accessories.

DIS 2010, August 16–20, 2010, Aarhus Denmark

ups, and working robotic prototypes, conducted during the course of one year. We end by a discussion on particular reflections and design challenges that have emerged through this process, pointing at potential implications for future designs based on ActDresses as a design concept.

RELATED WORK – THEORETICAL EXPLORATIONS

This section presents related work in the areas of designing accessories for robotic products, the metaphor of clothing and dress codes in interaction design, and more broadly related research in the field of tangible programming and interaction.

Accessories for robotic products

Many existing robotic products include at least one accessory that symbolises a basis for physical interaction. Examples of such robots and accessories are Furby and the spoon, AIBO and the Pink Ball, and Pleo and the green Leaf. It is also possible to buy extra accessories to most of these kinds of products. The accessories play a double role: being a raft for intelligible agency and providing an added value from a marketing perspective. However, these accessories are more like external tools for interacting with the robot, rather than something that is meant to be attached to the robot surface.

The human-robot interaction research community has recently become increasingly interested in the physical appearance of robots e.g. studying the effects of different head-shapes [5] and costumes [16]. In particular this has become a core issue in the sense that capabilities that are suggested by its mere appearance appear to cause an elevation of peoples expectations that goes beyond its actual functionality [9]. The rare examples of accessories that provide more function also appear to be less directly attached, e.g. Roomba‟s infrared virtual wall beacons. It is interesting to note that much of human-robot interaction research so far has focused on the robotic artefact per se without paying much attention to interaction models involving accessories.

A recent study [18] explored the use of non-computational surface add-ons for the robot vacuum cleaner Roomba. Similarly the surface of the robot Nao can be slightly customised by interchangeable coloured plates. Although they do not add any additional computational behaviour, these kinds of accessories do provide the ability to personalise the robot, and to divide several robots into teams. Similar to children‟s play with inanimate toys [19], the representational manufacturer-given characteristics and format lend these kinds of artefacts to external representational structuring through accessorising (see Figure 2).

Figure 2. Customized covers for the Roomba vacuum cleaner robot, in the form of a frog (left) and tiger (right).

The metaphor of clothing and dress codes

Two notions that could be of particular interest to our designs are uniforms and costumes, which conceptually bind together function, practices and visually perceivable manifestations in socio-cultural settings. The difference between the two is subtle, but the uniform tends to be more strict while the costume would be more general in terms of encoding an outfit.

Rafaeli and Pratt [14] identify three dimensions for an analytic framework of uniforms – *attributes*, *homogeneity* and *conspicuousness*. Attributes refer to specific features such as colour, material and style, homogeneity is the extent to which there is a consistency between people and between occasions in how they dress, and conspicuousness is to what extent the dress code in unique compared to other groups in a society. Furthermore, dress codes not only account for deeper socio-cultural meanings e.g. social status and beliefs, but also more contextually oriented meanings e.g. allowance for passage or appropriateness to perform a certain action.

A dress code describes how individual pieces are recombined and put together, in our case into a complete ActDress. Each piece can be thought to have some of the following properties:

- Belonging to a set
- Visually representing a function
- Could be recombined with other pieces
- Could add a certain functionality

Similar ideas have been explored in terms of democratising hacking culture from a gender perspective by looking at DIY, tinkering and bricolage. Specifically Blackwell suggests that women‟s (stereotypically put) sense of assembling outfits - *séance d'essayage* – captures the sort of competence required for this kind of end user programming [2].

Within the American robotic toys industry the notion of „*skits*‟ carries the meaning of small acting repertoires or themes of motions that can be performed by a robotic device. For instance, using the robotic toy dinosaur Pleo it is possible to „run‟ skits by downloading script-files from the Internet onto a SD-card that then can be inserted into a SD-slot in Pleo‟s belly. Available skits are e.g. the Halloween, Christmas, Valentine and Watchdog themes. When combined, dress codes could be defined to encode for such skits in a more structured way.

Physical and tangible programming

Historically programming has included very physical forms of interaction using punch cards and changing of electron tubes. However for many decades now, the specification and control of interactive systems have been based primarily around on-screen developing environments, even for the case of highly physical robotic systems.

Figure 3 illustrates a simplistic case of how a robot may be made to spin in a certain way by using conventional programming code downloaded from a separate computer. The figure also illustrates the fundamental concept from semiotic theory that a sign (e.g. for controlling interactive systems) is always built up of two parts – the manifestation of the sign itself (signifier) as well as what it refers to (the signified) [4]. Importantly, a sign referring to a particular computational action, e.g. spinning in a certain way, may take more or less any form, including words, images, sounds, gestures, acts, and physical objects. In settings where the interactive artefact does not have a screen display of its own, the „code‟ is more or less hidden, often at a completely separate hardware device. A state-of-the-art example is the Choregraphe environment for controlling and programming the Aldebaran NAO robot [12].

In the development of new approaches to controlling robotic action, the challenges thereby include both the designs of the actual signs, as well as their coupling to meaningful computational action. A number of research projects have been exploring producing and predicting behaviours by means of physical manipulation only. Examples include Topobo [15], Sketch-a-move [10] and Curlybot [8], and a range of programming by example implementations in industry.

A closely related stream of research is the also early experiments within the domain of tangible user interfaces (TUI‟s), where physical objects have been designed to represent and load digital files and actions. Examples include Ullmer et al.‟s MediaBlocks [20], and Ljungstrand et al.‟s WebStickers [11], both conducted during 1999-2000.

DESIGN EXPLORATIONS

Our focus will be on a scenario of using signs in the form of physical markings attached to a digital artefact, where the physical markings act as signifiers, and the actions or behaviours that they would make the artefact perform would be the signified. Thus, ActDresses are displayed in the immediate physical context of the objects that they control, thereby addressing the general desire for a directly visible, inspectable and modifiable program representation. A related property is that they are meant to represent and produce *perceivable* actions in the computerised system, in ways that end-users may easily observe and understand.

Figure 3. Left: Signifier (the manifestation of the sign). Right: Signified (what the sign refer to).

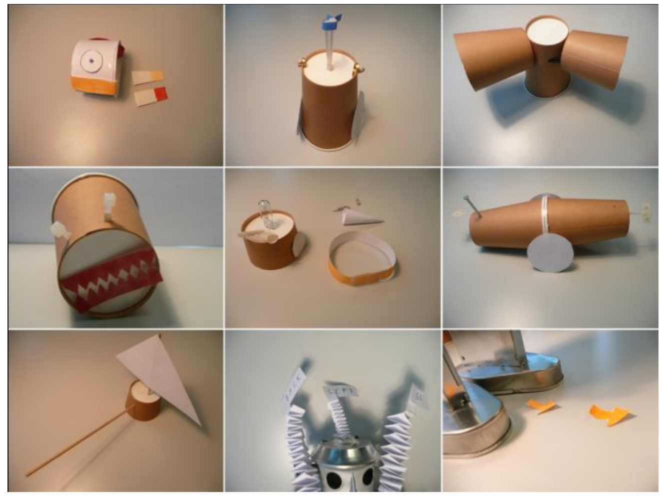

Figure 4. Paper mock-ups of behaviour visualisations using various physical shapes.

To give a sense of how the ActDresses concept could be realised concretely, a series of interaction scenarios and mock-ups have been created for the case of robotic movement. Figure 4 shows a series of paper cup mock ups, illustrating e.g. how the shape may make an object look like it is able to hear, see, move in a certain direction, or possess certain personality traits such as being aggressive, as manifested by its physical form factors. This kind of low-fidelity explorations helped gaining an understanding of the design space of physical shape in the representation of movement and physically enacted behaviours.

Below we present a series of semi-implemented explorations to illustrate how working with conceptual sketches in different materials may provoke assumptions and guide actual implementation further on. Our focus has been on simple autonomously moving robots, based on platforms such as the educational e-Puck and Roomba Create.

Hands-on manipulation of robotic movement

As a way of exploring the fundamental properties of direct manipulations with programming and control of robotic systems we made hardware experiments (Figure 4) to explore physical properties of manipulating robotic surfaces. The interaction principles we covered were (1) direct manipulation of sensors: how does a robot behave, and what does the user expect, if a user „blindfolds‟ a robot? (2) Direct manipulation of actuators: disabling a wheel changes the actual and expected behaviour of a

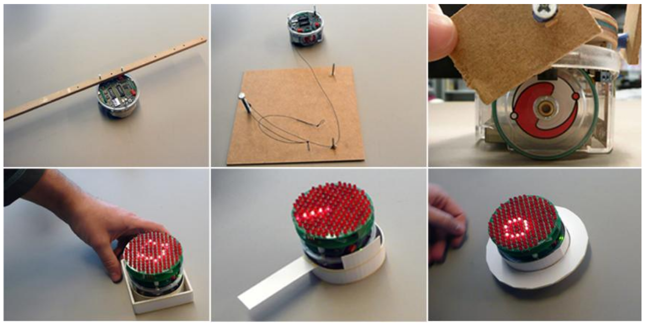

Figure 5. A series of explorations of hands-on manipulations of robotic movements by e.g. covering parts of its sensors, attaching it to strings and sticks, all affecting how the movement of the object could be controlled and predicted.

robot: it can only drive in circles (3) Direct manipulation of physical characteristics: by adding physical objects, the behaviour of a robot changes: a connected string literally constraints the movement possibilities (4) Tags and flags: a user can guide, direct, program or control a robot. How can tags and flags visualize this? (5) Direct manipulation through physical objects: adding a square shape makes the robot display a square shape, rotating it displays a rotating square, etc.

A second stream of explorations covered agent principles, how users through physical means could manipulate and in a sense program existing robots, and to an extent also how robots may influence each other. Conditions (If-then-behaviour), copying and emergent principles were explored using the GlowBots (based on the e-Puck platform, see Figure 6). While this allowed more complex behaviour of a group of robots, it appeared to remove the ActDresses principle for individual robots. For instance, when a robot had copied a behaviour from another robot, its visual appearance would no longer match its behaviour. One solution would be to allow the robot to adapt its own appearance, but since this would imply a whole new dimension to the design concept, we decided to take a step back and focus on interaction with a single robot at a time.

It became apparent that the *interpretation* of the appearance that was affecting the behaviour was crucial: e.g. if a user was not able to predict that blocking one wheel would cause the robot to drive in circles, then the correct appearance changing required some trial-and-error cycles. After „succeeding‟, and thus after observing both the appearance and the resulted behaviour, the mapping and interpretation became clear. Changing the appearance should thus be consistent with the resulting changes in behaviour, and the user should be able to learn this by trial-and-error iterations.

Interaction scenario: high-level behaviours

In the scenario explored in Figure 6 we use commercial vacuum cleaning robots as a prototype base for exploring

Figure 6. Interaction scenario for simplistic higher level behaviours. By attaching simplistic labels indicating a higher level state of activity, users would be able to put the device into different modes.

how the concept could enhance a task-oriented robot. Here we explore how comic book-style patches attached by magnetic tape can be used to extend the robot‟s basic behaviours with super positioned abilities. As inspiration, Braitenberg [3] introduced the concept of having very simplistic behaviours from hard-wired sensors-actuators. Such simple actions resulted in behaviours that people tended to interpret as psychologically driven, e.g. being curious, aggressive, nervous, etc. Reasons for experimenting specifically with signs borrowed from comics was to explore a type of notation that many people are familiar with and already know how to read.

In this scenario, a set of comic-like magnetic patches are attached to the metallic shield of the vacuum robot (prototyped by using an upside down cake pan in its place), putting it into different modes. In Figure 7, a „shy‟ sign has made the robot hide under the sofa, and is switched to another mode by the user, to make is spiral slowly and silently on the carpet. Apart from the standard sensors of most commercially available robot vacuum cleaners, this scenario would require an RFID-reader for reading of the signs, as well as sensors to allow for interaction patterns that respond to e.g. motion, sound, and light.

Interaction scenario: detailed instructions

As a complement to the very high-level behaviour explorations depicted in the above scenario, we sketched out a case that further emphasises actions and user interactions that can be sensed and performed on a lower level of abstraction. For the platform that we used for this scenario (the GlowBots), these actions could be narrowed down to very basic groups: (1) Navigation in space, (2) Patterns on its LED-display, (3) Generate sound, (4) Send and receive signals from other robots, and (5) Respond to user interactions, e.g. shaking and holding the robot in

Figure 7. Interaction scenario for how combinations of labels with more detailed instructions could influence robotic movement and interaction.

different ways. For each of these, a range of behaviours of different complexity could be developed.

The interaction scenario sequence in Figure 7 shows a user selecting from a collection of physical behaviour „amulets‟, attaching them to a GlowBot, and finally how a group of GlowBots move, glow, and interact according to items on their bracelets. The implementation strategy for this scenario is based on simple unit resistor pin-based decorations where combination is a main quality. Combining several such simple behaviours may then lead to more complex patterns.

IMPLEMENTATION EXPLORATIONS

From sketching in hardware and software we went onto implementing more complete demonstrators. The first implementation was used as a tool for reflecting about ActDresses from a visibility aspect and probing what other people thought about the concept in general. The second exploration focused on using wireless technology together with a scenario that had inherent modular properties. In the third exploration we started over again, went back to the roots and progressed more assertively by applying successful ideas and concepts from the previous experiments.

The Hat Trick

In the first implementation we did want to get a feeling for the concept by visualising simple behaviours using accessories like hats. In this case we use an e-Puck, a two-wheeled educational robotic platform the size of a coffee-cup. It has diverse technical capabilities e.g. infra-red sensors, accelerometer, Bluetooth, microphones, LED‟s and a speaker. The goal was to equip it with custom-made hats so that each hat would correspond to a certain behaviour that is activated as soon as the robot is activated with the hat (Figure 8).

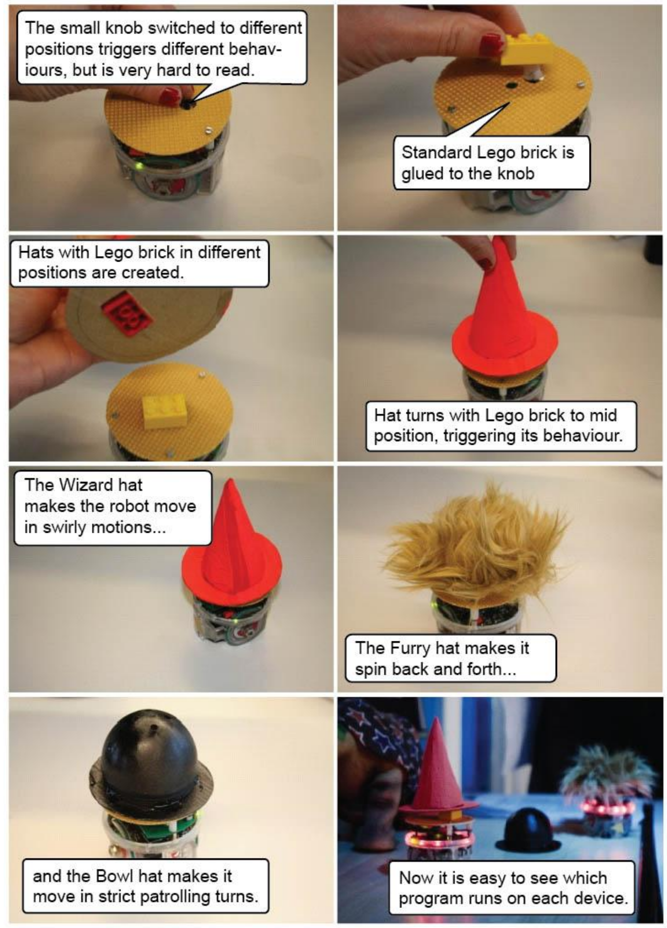

Figure 8. The Hat Trick. A demo illustrating how the changing of hats could simplify control and provide a clearer view of the position of a small knob that triggers different movement patterns of a device.

Three type of hats were designed, a Wizard hat, a Bowl hat and a Furry hat that would correspond to three different movement patterns – Swirly, Patrol and Spinning respectively. Each hat were designed to have a physical key that only fits in one way on the robot, thus forcing an electrical knob to be set in one out of four positions. The four positions each corresponds to four different software routines encoding the three movement patterns plus a default „no movement‟-state. Furthermore we experimented with using sound and blinking lights, although we quickly sensed that it made individual behaviours less articulated.

In order to discretely assess our prototype with external users we took the opportunity to demonstrate it at small public exhibitions. The reaction was that although it was a very easy concept to grasp, visitors generally sought more concrete and practical uses for it. Moreover we got suggestions that the same kind of principles might be applied to other electronic devices where accessories and personalisation would be important aspects, which shows that people quite easily could form their own ideas around the concept.

The Square Dancer

In order to look at different enabling technologies for ActDresses we invited students to come up with different scenarios that they thought of when learning about the project. Our second exploration looks at one of these resulting prototypes called *'Square Dancer'*. The scenario uses dance, or rather foot-patterns of a dance as inspiration for motions for a robot. The idea is to build more complex patterns of movements from basic trigonometric shapes. The Square Dancer is a half spherical shaped, omni-directional robot that can move in a fluent manner in combined patterns according to physical markings attached to its surface. This allows the robot to move in any direction or pattern without having to turn around as with ordinary wheels. Internally, the custom-made electronics are based on two circuit boards, one functioning as the master – reading and processing data from the RFID-slave circuit, driving the wheels and taking care of obstacle detection using a sweeping IR-sensor.

In line with the early explorations with the physical LOGO turtle in the 1970s [13], the students played with a simple set of geometrical figures as the basis for these explorations. Based on this, we created a set of physical tags depicting e.g. a square, a circle and a triangle resulting in corresponding movement patterns – if and while they were attached to the robot surface.

A specific problem that we wanted to address in this implementation concerned how different program instructions might be combined together, not only visually, but also in terms of how they would be put together computationally. This is related to aspects of program order and concurrency, and how the effects of several simultaneously connected tags could take effect at the moment of execution. This was approached by dividing each geometrical figure into smaller parts of their paths, and taking turns between these sections rather than do complete figures of movement. The behaviour combination functions are built in a way in which arrays will be split up in selected parts and then merged together in a dynamically allocated array. These parts can then be seen as patterns or steps in a dance.

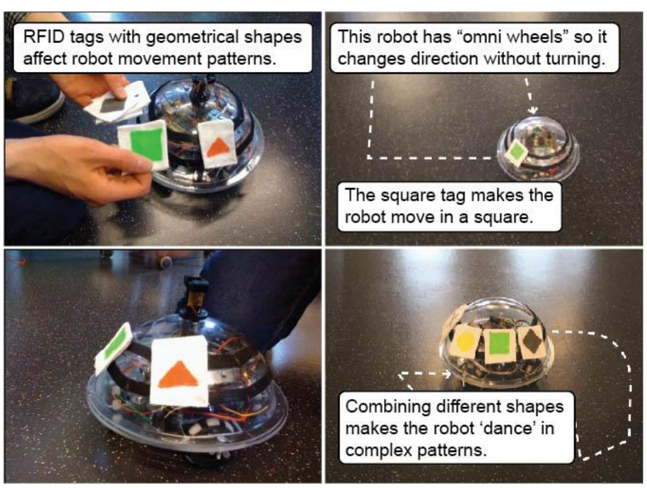

Figure 9. The square dancer. RFID tags with geometrical shapes attached to the robot to affect its movement patterns. Combining different shapes makes the robot 'dance' in complex patterns.

The interaction technology in focus was RFID and how it could be used to explore the practical aspects of using physical tags for controlling its behaviour. We quickly noticed that although RFID was used, the tags still needed to be physically attached. We first explored the possibility using magnets but eventually settled with Velcro. This allowed the tags to be attached more freely, but at the same time required additional attention from the user to see if the particular segments of the behaviour were properly executed. This issue was partially addressed by implementing a LED-light that indicated when a tag was positively identified by the RFID-reader.

Movement with Different Styles

The third and final implementation focused even more on the interaction and integration part, and explored this through broader iterations of exploration and prototyping, again attempting to address a realistic use case of controlling the movement of an autonomous vacuum cleaning robot.

In this step, we used the Roomba Creator platform, iButton technology[1] and a Viliv S5 MID device running Java. Different sorts of iButton-tags were created, each coupled to different behavioural aspects, e.g. „spiral driving‟, „sweeping mode‟, „follow wall‟, „random mode‟, „faster‟, and „slower‟. Attaching and combining these tags resulted in different behaviours of the Roomba.

A quick analysis first of all revealed the importance of the *surroundings* and *context*: wall-following in an empty room resulted in totally different behaviour than wall-following in a furnished room. In addition, the *building blocks* for programming behavioural aspects appeared skewed. With tags like „follow-wall‟, one would program a more full behaviour of the robot, yet this behaviour is not possible in a cluttered room. Instead, tags should imply movement characteristics, such as „stay close to objects‟ or „drive in circles‟, which would increase the mapping between user understanding and effective behaviour. Using a „stay close to objects‟ tag, the robot logically follows both walls and sofas.

In the final implementation, these concrete movement characteristics were implemented using iButton-equipped flags that could be attached to the Roomba via a magnetic connector (Figure 10). In addition we created three „dresses‟ that would signify more governing personalities of the Roomba. Learning through doing was a key to

[1] http://www.maxim-ic.com/products/ibutton/

Figure 10. ActDresses for Roomba. The user changes flags attaché to the robot to change the patterns or paths in which it moves. Covers in different styles are used to modify the ways in which this movement is performed.

understand this particular mapping issue. For instance, the „pink" personality in combination with the „stay close to objects" flag resulted in wild „object tracing" movements. Seeing this behaviour then allowed the user to predict that the same dress with the „drive in circles" flag would result in wild, circular behaviour. Besides allowing us to explore the ActDresses concept, this implementation also gave us practical insights into human-robot interaction through changing appearance.

DESIGN CHALLANGES

Following up on the implementations we here present the main design challenges that we found and relate them to three dimensions that form a design space for ActDresses. These will require further elaboration and exploration in creating a framework for the concept, but can be used as a starting point for compiling the results. The use qualities that we will follow up on here concern aspects related to *physicality, visibility*, and *modularity*. In particular we will focus on tensions between the initial scenarios and actual implementations.

Physicality

Naturally, physical objects cannot easily be copied, scaled, deleted. A direct mapping between what will happen to the object that an item is attached to means that the possibilities to scale up the system with larger amounts of objects and behaviours is restricted. Thus, turning „programming code" into physical form is not just another way of representing conventional computational instructions.

Physical constraints can be used as a resource for guiding manipulation, making it more intuitive (e.g. through the principle of tokens-and-constraints [22]). With the wireless approach using RFID we found that although the technology itself was wireless, tags still became "connected" by the fundamental implication of having them in the immediate context. Compared to wired solutions this offered more freedom in the sense that we were able to explore a variety of materials for attaching the tags. For instance we experimented with velcro, magnets and snap-buttons.

In the case of using a wired solution (e.g. iButtons), it implies that physical „sockets" restricts positioning of tags whereas the wireless solution can be designed to be both free and „socketed". On the other hand, a physical „contract" in the form of a physical wire would for instance give a subtle „enabled/disabled" tell whereas the more free wireless tag enters or leaves an invisible frictionless space. This fact appears to speak in favour of direct contact technologies. Furthermore, a final design challenge with respect to physicality is to keep in mind that robots often move around. For instance we noticed on several occasions that ActDresses tend to fall off or disconnect as the robot bumped into things.

One issue, which had already occurred in several prototypes but became even more apparent in the last case, was the notorious difficulty to change behaviour on-the-fly. While the robot was driving around the room, it was difficult to catch it and change the flags and cover. This insight made us implement a full stop when the robot was lifted from the floor and it indicates an important design consideration, that is directly drawn from the basic property of physicality. Other solutions for this challenge could for instance be to enable the robot to sense a certain presence or at least to have a way to pause the movement momentarily in order to allow physical manipulation.

Visibility

From our explorations we were reminded that meaning is always in the making and that the visual appearance must reflect an observed behaviour in order to be predictable. As illustrated in the four interaction scenarios, program actions could be visually manifested in a range of different ways, and the shapes and forms that the different representations could include a near-to-infinite amount of options and variations. The design space also ranges conceptually from visual representations of full persona costumes to detailed hardware instructions.

That objects are visible is thus important, but it also brings along a number of design challenges. For instance, when attaching several signs or labels to the same object, a consequence may be that they obscure the object, so that it is difficult to clearly see the object underneath. The items themselves could also overlap and obscure each other.

The role of appearance is perhaps one of the more obvious aspects of this, as the items may work simultaneously as controls on one hand and as a form of decoration on the other. In the second implementation we saw that the robot sometimes failed to recognise attached tags thus making it either unpredictable or incomprehensive depending on which stance we assume. Developing a stable solution so that what is seen or visible maps reliably with what is actually happening in the robot software is thus a fundamental aspect of this design challenge.

Modularity

Creating programs by putting together existing pieces of code into new arrangements relates to common popular practices of software development, e.g., the use of class libraries, interface widgets, and open source methodologies. Especially in educational settings and for novice programmers, such higher-level modes of program construction have been found particularly useful. This design concept is similarly based on a form of „high level" programming, in which detailed algorithmic detail is hidden from the user.

However, this also entails several challenges specific to those who intend to design such collections of programming items. As with any form of programming based on pieces of codes that should be possible to combine in a variety of ways, challenges come related to modularity. One important aspect is to make effective use of the compositional properties offered by the physical manifestation of an ActDress. This dimension is closely related to system integration and how the physical manipulation can be mirrored software-wise. Any individual item could belong to a set (complementary) or be combined with other items (linked). An item can be a container for other items or fit into another container (e.g. a pocket). An item can have sequential affordances (e.g. links, chains, snap-on, loops) or concurrent (e.g. badges). That is to say, any relationship between items must map to a relationship between aspects of behaviours.

In our second and third implementation we saw that the robots quite successfully managed to combine the different instructions into comprehensible behaviours. More specifically, this concerned aspects of program order and concurrency, and whether certain signs should be given higher priority than others at the moment of execution. Furthermore we saw that different types of signs appeared to be different classes of behaviour e.g. personalities and modes as described in the third implementation.

DISCUSSION

In our initial explorations of ActDresses as a design concept, we anticipated exploring user-friendly ways of controlling interactive devices by physical means. Looking at the specific explorations conducted here, including the three implementations, we conclude that a key issue in the interaction is the movement of the robot while trying to attach or change physical tags.

Another difficulty is related to how interaction and context unfold over time. A part of this problem can be related to Lucy Suchman"s dilemma of looking at „plans and situated actions": a robot is indeed situated and the appearance as well as the behaviour must therefore account for this [18]. Realising the implications for this fundamental principle make it possible to quickly assert any tendency of violation. In particular the first two design scenarios and the first two implementations show such tendencies. The approach we used was to account for context e.g. by using Braitenberg"s principles [3] as personality (e.g. hesitant or aggressive) and combine these together with more concrete movement characteristics (e.g. moving swirly or wall-following).

Furthermore, the qualities offered by physically signified means of interacting might still outweigh e.g. vocal commands, because of its complementary functions. Many cultural and social aspects that the ActDresses concept brings to the table regard subtle but interesting qualities such as aging, gender, social status, interests, etc.

CONCLUSIONS AND FUTURE WORK

We have presented four design scenarios and three implementations based on the concept of using physical clothing, labels, and accessories for interacting with and controlling robotic devices. In this set of explorations focus was on the simplistic act of specifying the movement patterns of autonomously moving robotic artefacts. The work is motivated by existing practices of physical customisation of electronic devices, the current trend towards commercial products with increasingly advanced control mechanisms, and experiences from the domain of end-user programming. Accessorizing is one such example, and the concept of ActDresses could be thought of as the corresponding interface that captures that practice in a bridge between physical and digital.

In developing new ways of controlling, programming and predicting the behaviour of physical consumer products, a deeper understanding of the fundamental theories explored in these and other fields may hold further benefits. Projects like these suggest further potential in using loose physical items, such as garments, jewellery and visual signs as resources for controlling, reading and predicting the behaviour of physical computing systems. However, all explorations presented in this paper were fairly technical in nature, and much more focus could be given towards how the concept could work in a certain practice, to offer more support for creativity, for play, and for personal expression in a social context.

This work also illustrates the necessity to consider physical shape, behaviour capabilities, and interaction modalities together, grounded both in existing use patterns, shape and capabilities of the respective platform. That being said, the truth is that the concept is still very much open in the sense that it is impossible to predict how it ultimately unfolds.

In a recently started parallel project we have begun to explore this concept in yet another domain, namely mobile phones. The mobile version builds on the same realisation – that people already do accessorise their personal handhelds. The general idea is to let this practice affect the software by enabling e.g. visual themes, media and games as shells and jewelry get attached to the devices.

ActDresses is an enabling concept and it generates far more ideas, scenarios and implementations than we can possibly explore alone. We hope to see more examples of this type of interaction in the future and together work towards a proper framework and reclaim control over not only appearance but also over action and in particular perceived action.

ACKNOWLEDGMENTS

This work was conducted as part of the EU funded project LIREC (Living with Robots and Interactive Companions). Important parts of this work was conducted by bachelor students Victor Luque and Johan Sundin who worked with developing the Squaredancer.

REFERENCES

1. Benford, S., Schnadelbach, H., Koleva, B., Anastasi, R., Greenhalgh, C., Rodden, T., Green, J., Ghali, A., Pridmore, T., Gaver, B., Boucher, A., Walker, B., Pennington, S., Schmidt, A., Gellersen, H., and Steed, A. (2005). *Expected, sensed, and desired: A framework for designing sensing-based interaction.* ACM Trans. Comput.-Hum. Interact., 2005. 12(1): p. 3-30.
2. Blackwell, A.F. (2006). *Gender in Domestic Programming: from Bricolage to Séances d'Essayage*, in *CHI'2006 workshop on End User Software Engineering*.
3. Braitenberg, V. (1984). *Vehicles: Experiments in synthetic psycology*. 1984, Cambridge, Massachusetts: MIT Press.
4. Chandler, D. (2002). *Semiotics: The Basics*. 2002, London: Routledge.
5. DiSalvo, C.F., Gemperle, F., Forlizzi, J., and Kiesler, S. (2002). *All Robots Are Not Created Equal: The Design and Perception of Humanoid Robot Heads*. Proc. of *DIS2002*: ACM Press
6. Djajadiningrat, T., Wensveen, S., Frens, J., and Overbeeke, K. (2004). *Tangible products: redressing the balance between appearance and action.* Personal Ubiquitous Comput., 2004. 8(5): p. 294-309.
7. Fernaeus, Y. and Jacobsson, M. (2009). *Comics, Robots, Fashion and Programming: outlining the concept of actDresses*. Proc. of *TEI'09*: ACM
8. Frei, P., Su, V., Mikhak, B., and Ishii, H. (2000). *Curlybot: designing a new class of computational toys*. Proc. of *CHI'00*: ACM Press.p. 129-136.
9. Jacobsson, M. (2009). *Play, Belief and Stories about Robots: A Case Study of a Pleo Blogging Community* Proc. of *Ro-Man*: IEEE
10. Jain, A., Klinker, L., Kranz, M., Stoeger, C., Blank, D., and Moesenlechner, L. (2006). *Sketch-A-Move - Design Inspired Technology for Children*. Proc. of *UbiComp (video)*
11. Ljungstrand, P., Redström, J., and Holmquist, L.E. (2000). *WebStickers: using physical tokens to access, manage and share bookmarks to the Web*. Proc. of *DARE*: ACM p. 23-31.
12. Monceaux, J., Becker, J., Boudier, C., and Mazel, A. (2009). *Demonstration: first steps in emotional expression of the humanoid robot Nao*. Proc. of *International conference on Multimodal interfaces*: ACM.p. 235-236.
13. Papert, S. (1980). *Mindstorms: Computers, Children, and Powerful Ideas*. 1980, New York: Basic Books.
14. Rafaeli, A. and Pratt, M.G. (1993). *Tailored Meanings: On the Meaning and Impact of Organizational Dress.* . The Academy of Management Review 1993. 18(1): p. 32-55.
15. Raffle, H.S., Parkes, A.J., and Ishii, H. (2004). *Topobo: a constructive assembly system with kinetic memory*. Proc. of *CHI'04*: ACM Press.p. 647 - 654.
16. Robins, B., Dautenhahn, K., Boekhorst, R.e.t., and Billard, A. (2004). *Robots as Assistive Technology - Does Appearance Matter?* Proc. of *RO-MAN 2004* IEEE Press.p. 277-282.
17. Sung, J.-Y., Guo, L., Grinter, R.E., and Christensen, H.I. (2007). *"My Roomba is Rambo": Intimate Home Appliances*. Proc. of *UbiComp 07*
18. Sung, J., Grinter, R.E., and Christensen, H.I. (2009). *"Pimp My Roomba": designing for personalization*, in *CHI'09*. ACM: Boston, MA, USA. p. 193-196.
19. Turkle, S., Taggart, W., Kidd, C.D., and Dast, O. (2006). *Relational artifacts with children and elders: the complexities of cybercompanionship.* Connection Science, 2006. 18(4): p. 347 - 361.
20. Ullmer, B., Ishii, H., and Glas, D. (1998). *mediaBlocks: physical containers, transports, and controls for online media*. Proc. of *Computer graphics and interactive techniques*: ACM p. 379-386.