Enhancing My Keepon robot: A simple and low-cost solution for robot platform in Human-Robot Interaction studies

Hoang-Long Cao[1], Greet Van de Perre[1], Ramona Simut[2],
Cristina Pop[3], Andreea Peca[3], Dirk Lefeber[1], Bram Vanderborght[1]

***Abstract*— Many robots capable of performing social behaviors have recently been developed for Human-Robot Interaction (HRI) studies. These social robots are applied in various domains such as education, entertainment, medicine, and collaboration. Besides the undisputed advantages, a major difficulty in HRI studies with social robots is that the robot platforms are typically expensive and/or not open-source. It burdens researchers to broaden experiments to a larger scale or apply study results in practice. This paper describes a method to modify My Keepon, a toy version of Keepon robot, to be a programmable platform for HRI studies, especially for robot-assisted therapies. With an Arduino microcontroller board and an open-source Microsoft Visual C# software, users are able to fully control the sounds and motions of My Keepon, and configure the robot to the needs of their research. Peripherals can be added for advanced studies (e.g., mouse, keyboard, buttons, PlayStation2 console, Emotiv neuroheadset, Kinect). Our psychological experiment results show that My Keepon modification is a useful and low-cost platform for several HRI studies.**

I. INTRODUCTION

Human-Robot Interaction (HRI) is an emerging field aimed at improving the interaction between human beings and robots in various activities. Researchers in this field are required to understand their research within a broader context due to the interdisciplinary nature of HRI [1]. According to [2], HRI differs from human-computer interaction and human-machine interaction because it concerns systems which have complex, dynamic control systems, which exhibit autonomy and cognition, and which operate in changing, real-world environments. Traditionally, robots are operated by experts and work as tools in industry. Many recent commercial robot platforms have been developed with the ability to exhibit social behaviors, and capability to work with non-expert users at home, school, hospital, museum, etc [3]. It is suggested that robots as partners can help us accomplish more meaningful work and achieve better results [2]. Hence, there is an observed niche market for social robots.

A big challenge to broaden the scope of research and apply the study results to daily life is the high cost of robot platforms. Successful robots such as Asimo [4], Nao [5], iCub [6], HRP-4C [7], Probo [8], Keepon Pro [9], are expensive due to their complexity with high degrees of freedoms. On the other hand, simpler robots like Furby [10], KASPAR [11], Pleo [12], are cheaper but not officially modifiable e.g. no SDK. Since many advanced studies need to connect the research platform to peripherals (e.g., mouse, keyboard, buttons, PlayStation2 console, Emotiv neuroheadset, Kinect, Xbox 360 controller), the cheap robots mentioned above prevent the possibility to adapt to the research requirements. Therefore, there is a need to have low-cost and expandable social robot platforms for HRI studies. Ono robot has recently been developed which can be reproducible at the cost of approximately €300 [13]. However, Ono is at the first steps of development and its electronics needs to be improved [13].

Modifying cheap commercial robots can be a solution to have low-cost programmable platforms for HRI experiments. In [14] and [15], Pleo robot is hacked by adding a bluetooth interface with a complete tutorial given by LIREC project. However, the instructions are complex and require a lot of materials. Hasbro's Furby toy (2012 version) can be hacked by using audio protocol with the official Furby applications for iOS and Android [16]. This method is simple but cannot enable Furby to connect with other devices. According to [17], hacking Furby's hardware is probable by rewriting its EEPROM but currently not practical due to the insufficient understanding of the data structure. Also, the form of Furby's casing with hidden screws makes the disassembling process difficult. In [18], My Keepon robot is modified to be a programmable robot platform. Its internal circuit board is replaced by an Arduino microcontroller board and motor drivers. Thus, the cost of modifying increases. Alternatively, there exists a method to hack My Keepon by sending I^2C commands to its microprocessors from Arduino. The source code is supported by the manufacturer Beatboxs [19], and from Nonpolynomial Labs [20]. The hacking cost is low since the internal electronics is retained. Moreover, the casing of My Keepon can easily be opened to access the hardware. To this end, My Keepon is a good candidate to be modified.

Although the source code to communicate with My Keepon is available, there is a lack of instructions to build the complementary hardware and user interface, and to adapt the source code with external devices. In this paper, after a short overview of Keepon Pro and My Keepon, we will present the technical details to connect My Keepon to Arduino with a user interface written in Microsoft Visual C#. Integrating the modified platform with PlayStation2 console and Emotiv neuroheadset will be introduced as a general example

[1]Vrije Universiteit Brussel, Robotics & Multibody Mechanics Research Group, Pleinlaan 2, 1050 Brussels, Belgium
[2]Vrije Universiteit Brussel, Department of Clinical and Life Span Psychology Group, Pleinlaan 2, 1050 Brussels, Belgium
[3]Babes-Bolyai University, Department of Clinical Psychology and Psychotherapy, Cluj-Napoca, Romania
Hoang.Long.Cao@vub.ac.be

of expanding the modification for advanced studies. Then Section IV briefly describes our psychological experiments to demonstrate the performance of the modified My Keepon in several HRI studies.

II. Keepon Pro and the Hackable My Keepon

Keepon Pro is a $30,000 research model developed by Hideki Kozima, and is sold commercially to Beatbots. It appears as a small yellow creature-like robot designed for simple, natural, nonverbal interaction with children to study social development of autistic children [9]. According to Beatboxs website [21], Keepon Pro structure has four degrees of freedoms (DOFs): turning ($\pm 180°$), nodding ($\pm 40°$), rocking side-to-side ($\pm 25°$), and bouncing with a 15 mm stroke. A PID controller generates a trapezoidal velocity profile for each DOF. Keepon Pro's playroom perceptions are transmitted to a therapist by two cameras in its eyes and a microphone in its nose. A rubber skin ensures safe and comfortable contact between the hands and the hidden buttons at each side of the robot. Keepon Pro is an interesting robot platform and has been widely used in HRI studies on social development and behaviors (e.g. eye contact, joint attention, touching, caregiving, and imitation) [9], [18], [22], [23], [24], [25].

My Keepon is a low-cost version of Keepon Pro at the price of $40. It is designed as an interactive robotic toy for kids with two modes: Touch and Dance. In the Touch mode, My Keepon responds to pokes, pats, and tickles detected by hidden buttons with emotional movements and sounds. In the Dance mode, it detects beat in music and automatically generates movements in synchronized rhythm [21]. As shown in Figure 1, the external structures of Keepon Pro and My Keepon are basically identical. However, My Keepon has limited capabilities because of its simple internal structure.

Nonpolynomial Labs initiated the idea of hacking My Keepon with Arduino by reverse engineering. Based on this idea, we improved the firmware and developed a software to control My Keepon at the Vrije Universiteit Brussel (Belgium). Beatboxs afterwards offered the official source code to make the robotic toy completely hackable. According to [20] and [26], My Keepon uses two microprocessors (PS232 and PS234) to control the movements and the sounds, which communicate with each other via I^2C protocol. The PS232 (Slave - address 0x52) deals with sounds and encoders. The PS234 (Master - address 0x55) handles driving the H-bridges, detecting button presses, and main processing. My Keepon can be controlled by sending commands to these two microprocessors over the I^2C bus.

III. My Keepon modification

In this section, we briefly describe the procedure to turn My Keepon to be a programmable robot platform. Since 2012 before the official source code released, we started to provide a tutorial of hacking My Keepon which has been used in our experiments. Going further from the Arduino code, we have also offered an open-source software written in Microsoft Visual C# with a user-friendly Graphical User Interface (GUI) to control My Keepon. For more details of the hacking process, please visit our "Hacking Keepon" website at *http://probo.vub.ac.be/HackingKeepon* (previously at *http://vikeepon.tk*).

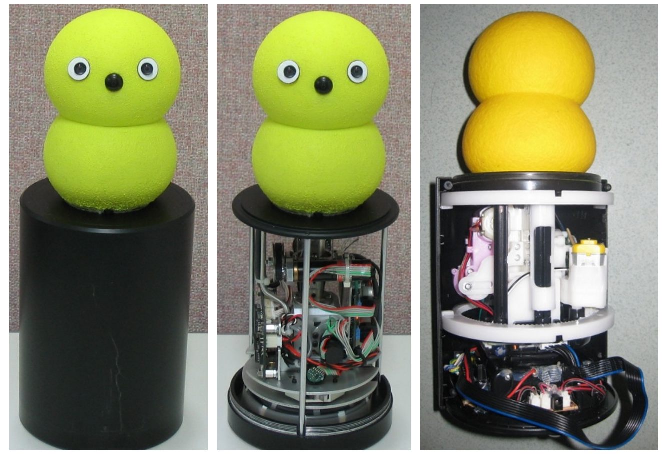
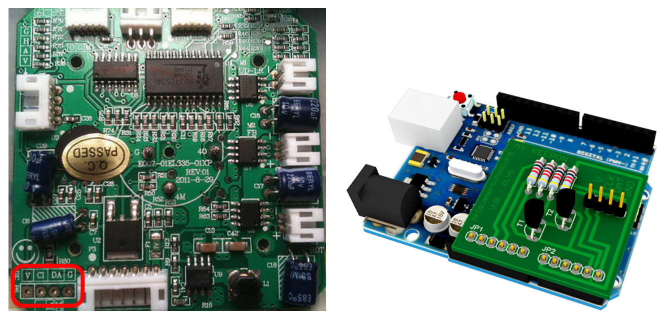


Fig. 1: Internal and external structures of Keepon Pro [9] and My Keepon.



(a) (b)

Fig. 2: My Keepon controller board (left) and Arduino shield for the modification (right).

A. Hacking the electronics

As previously mentioned, My Keepon can be hacked by sending I^2C commands to its microprocessors from an Arduino microcontroller board. In order to do this, we need to connect the I^2C pins of My Keepon's controller board to Arduino. This board lies under the mechanism after opening the casing (Figure 2a). The I^2C pads (V, CI, DA and G) are on the top right corner of the control board, indicated by a smiley face. These pads need to be connected to the corresponding pins of Arduino board: V-A0, CL-A5, DA-A4 and G-GND (Figure 4a). Since My Keepon's controller board works at 3.3V, a logic level converter such as the dedicated shield in Figure 2b is required if using 5V Arduino (e.g, Uno, Leonardo). Drill a hole above the power connector in order to let the four wires enter the casing, and then re-assemble the robot carefully. The last step is to upload the firmware (.ino or .pde extension) to Arduino board. After that, we can fully control My Keepon by sending string commands from

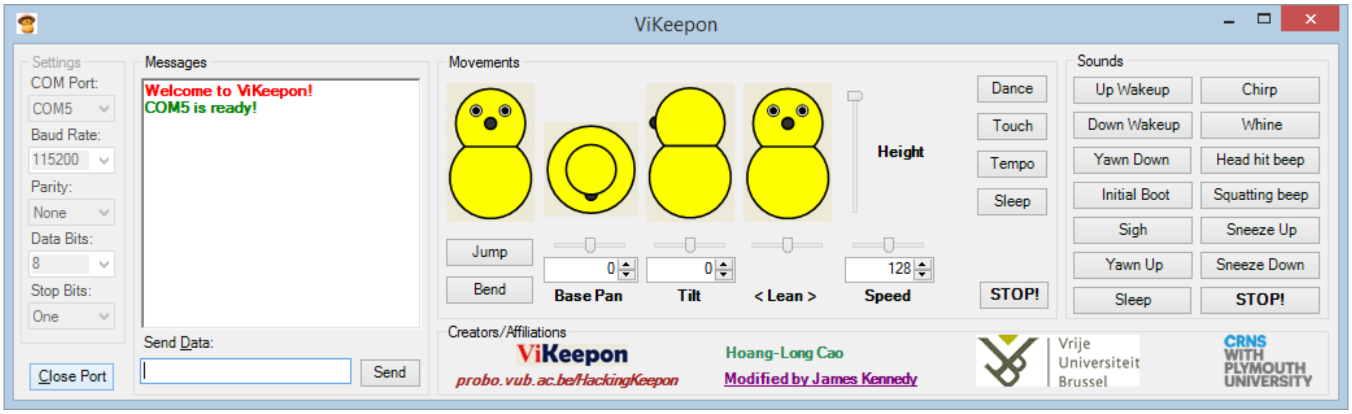


Fig. 3: Basic GUI to control the modified My Keepon. Adapted depending on specific experiments.

computer to Arduino via serial communication (UART). The list of commands with respect to Beatboxs firmware is given in Table I. Commands for Nonpolynomial Labs firmware are purely in hexadecimal values and not obviously understandable [20]. Arduino converts these string commands to the corresponding I^2C commands of My Keepon.

TABLE I: Commands and incoming data to control My Keepon [21].

Commands	Incoming data
SOUND (play, repeat, replay, stop)	BUTTON(built-in buttons)
SPEED (pan, tilt, ponside)	MOTOR (pan, tilt, pon, side)
MOVE (pan, tilt, pon, side, stop)	ENCODER (pan, tilt, pon, side)
MODE (dance, touch, tempo, sleep)	EMF (pan, tilt, ponside)
	POSITION (pan, tilt, ponside)
	AUDIO (tempo, mean, range,...)

Note that hacking My Keepon voids its warranty. Therefore, it is recommended to take care of the positions of the different parts to make sure My Keepon can be re-assembled. Apart from soldering wires to the I^2C pads, do not intervene other parts of the controller board. However, the hacking process is forthright and My Keepon can function at original condition if Arduino has no supply power.

B. Graphical user interface

Since most of non-engineering researchers (e.g. therapists, psychologists) are not familiar with I^2C commands, an easy-to-use software written in Visual C# provides a more convenient way to control My Keepon. This software is available to download from our website. Through a set of buttons of the GUI as shown in Figure 3, users are able to control all movements and built-in sounds of My Keepon. The software can also receive data from encoders and detect button presses. Functions of GUI components are as follows.

- **Settings:** configuration of serial port (e.g., port number, baud rate) and Close/Open button
- **Messages**: display outgoing (sound and motion commands) and incoming data (encoders, button presses)
- **Movements:** two buttons and four numeric track-bars with up-down boxes to change the movements
- **Mode buttons:** start Dance, Touch, Tempo, Sleep mode
- **Sounds:** buttons corresponding to different sounds generated by My Keepon's hardware

It is important to follow the following steps in order to make the modified My Keepon platform operate properly.

1) Connect the Arduino to My Keepon
2) Connect the Arduino to a computer using a USB cable
3) Launch the software (.exe file)
4) Power up My Keepon
5) Open serial communication (Baud rate: 115200, Parity: None, Data bits: 8, Stop bits: 1)

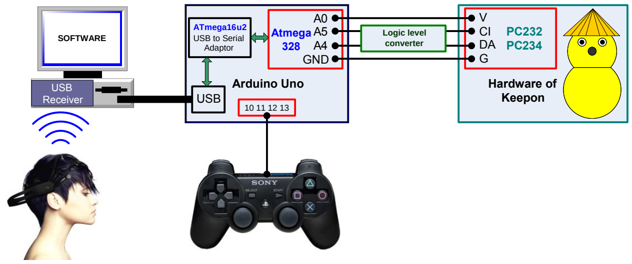
The software usability is intuitive. When users click on a button e.g. "Jump", the software converts it into a string command and then transmits to Arduino via a serial communication. Thanks to the firmware, My Keepon performs the corresponding movement or sound which is "bouncing" in this case. During this process, commands and incoming data from My Keepon are displayed in "Messages" groupbox for debugging purpose.

Basically, it is sufficient to set up experiments with manual or teleoperated mode by using the GUI components. With the Microsoft Visual Studio programming environment, advanced users can go further into the source code to create their desired functions such as generating complex movements or playing their own sounds. To do this, the basics of the communication protocol mentioned above must be understood. The software is compatible with Windows 8 (32-/64-bit), Windows 7 (32-/64-bit), and Windows XP.

C. Expanding the platform

In some HRI studies, it is necessary to integrate other devices to the modified My Keepon platform. This requires users to have knowledge of electronics and programming. The big resources from Arduino and Visual C# communities provide tremendous possibilities to expand the hardware and improve the software.

Devices are connected to Arduino or computer depending on their specifications. Simple devices (e.g., LEDs, buttons, keypad, joystick, PlayStation2 console) are typically connected to Arduino by virtue of standards and contributed libraries. On the other hand, advanced devices (e.g., Kinect,



(a) Hardware connections.

(b) User testing.

Fig. 4: An example of expanding the modified My Keepon platform with PlayStation2 console and Emotiv system.

Emotiv neuroheadset, Xbox 360 controller) need to be connected to the computer and communicate with the Visual C# software via SDKs.

We present here a general example to demonstrate the possibility to expand the modified My Keepon platform in terms of both hardware and software. Figure 4a illustrates the expanded platform in which a PlayStation2 console and Emotiv system are connected with Arduino and a computer, respectively. Therefore, Arduino firmware and Visual C# software have to be modified.

1) PlayStation2 console: PlayStation2 console is realized to be a handy tool to control My Keepon. Since it is widely used in DIY projects, tutorials of interfacing the console and Arduino are easily found on the Internet. The hardware connection is set up by connecting six wires of the console (clock, data, command, power, ground, and attention) to Arduino pins. The codes to handle button presses from the console need to be put inside the main loop of the Arduino firmware. The button press events are then converted to I^2C commands of My Keepon.

2) Emotiv system: Getting inspired from the recent studies on brain-machine interface (e.g., [27], [28], [29]), Emotiv system is integrated to control My Keepon by thought. The system consists of a neuroheadset with electrodes to measure brain signals, and a USB dongle to communicate with computer wirelessly. In order to read brain signals, the Visual C# software includes Emotiv API which is exposed as an ANSI C interface implemented in two Windows DLLs ($edk.dll$ and $edk\_utils.dll$) [30]. Thanks to this API, raw signals are translated into users' thought in form of EmoState structure such as forward, backward, rotate left, rotate right, etc. Emostate is then converted into string commands to control My Keepon.

Fig. 5: "Hacking Keepon" workshop at the 2013 International Summer School on Social HRI.

In this experiment setting, users are able to control My Keepon by a PlayStation2 console or Emotiv neuroheadset as can be seen in Figure 4b. The performance of user's thought detection significantly depends on the training process.

IV. Applications in HRI studies

The modified My Keepon platform to some extent can achieve similar performances as of Keepon Pro in HRI studies. The modification method is simple and does not require advanced knowledge of electronics and programming. We organized a workshop of hacking Keepon at The 2013 International Summer School on Social HRI for a multidisciplinary group of students such as engineers, computer scientists, psychologists, etc (see Figure 5). Even though many of them lack technical experience of soldering and programming, they were able to modify My Keepon in two hours. The workshop result proved that researchers can easily be familiar with the software, complementary Arduino shield and expand the platform. Hence, they can quickly set up the platform for their experiments.

Since 2012 at the Vrije Universiteit Brussel (Belgium) and Babes-Bolyai University (Romania), we have conducted three psychological experiments with the modified My Keepon with typically developing children and autistic children. In these experiments, we expand the hardware by adding simple devices such as buttons, LEDs, a buzzer to Arduino. The Visual C# GUI is extended so that the operator can easily control the actions of the robot depending on the requested scenario for the intervention. Specifically, buttons are created to make specific combination of movements (e.g, 45° left/right turns, nodding or bouncing within a period of time) by using string commands, and to play sound files. Another button is additionally made to turn My Keepon back to its neutral position.

A. Joint attention

A pilot-study was conducted to investigate the performances of typically developing infants in tasks focused on joint attention. Joint attention refers to a set of behaviors that serve to enable two partners to either vocally or non-vocally communicate about, or "jointly attend to" a third entity, object, or event [31]. Two types of joint attention behaviors were targeted in this pilot study, such as gaze-following and initiations of joint attention, during the infants interaction

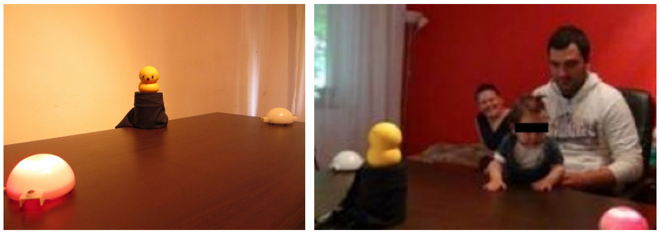


Fig. 6: Joint attention study with the modified My Keepon.

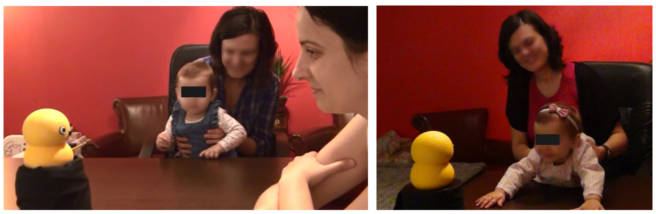


Fig. 7: Social agent study with the modified My Keepon.

with a robot (My Keepon) and a human partner. The extended platform for this experiment is shown in Figure 6. A pair of attractive objects was developed containing bright LEDs (red and blue) with an opaque white plastic cover. Every time when the target object was activated by the operator a pre-programmed light animation was started, together with a monotonic melody. The result of this study is beneficial for joint attention interventions for children who are impaired in this ability e.g. autistic children.

B. *My Keepon as a social agent*

The second experiment used My Keepon to investigate if infants perceive an unfamiliar agent as a social agent after observing an interaction between the robot and an adult. Twenty-three infants, aged 9-17 month, were exposed, in a first phase, to either a contingent or a noncontingent interaction between the two agents, followed by a second phase, in which the children were offered the opportunity to initiate a turn-taking interaction with My Keepon. The measured variables were: (1) mean looking time to each of the two interacting partners, (2) the number of anticipatory orientations of attention toward the agent that follows in the conversation, and (3) the number of verbal and motor initiations of the child toward the robot. A snapshot of the experimental setting is presented in Figure 7. In the contingent condition, My Keepon responds contingently, by a pre-programmed set of sounds and motions to the adult verbal initiations, while in the noncontingent condition, My Keepon remains still. In the testing phase, My Keepon responds by the same pre-programmed set of sounds and motions to any intentional babbling or motion of the child. At the moment, the data collected are still under analyses. The results will indicate if children, before their second birthday, attribute intentions to unknown agents, based on the dynamics of their interaction.

C. *Role of My Keepon in a cognitive flexibility task*

Another experiment was performed to investigate the role of My Keepon in a cognitive flexibility task performed by children with autism and typically developing children. The number of participants included in this study was 81 children: 40 typically developing children aged 4-7 years and 41 children with autism aged 4-13 years. Each participant had to go through two conditions: robot interaction and human interaction. In both conditions, they had a reversal learning task, meaning that they had learned to choose the correct stimulus location from a pair of locations to receive a positive feedback from My Keepon or from the human (acquisition). After making the correct choice over multiple trials, the rewarded stimulus location changed without warning (reversal). We have measured the number of errors from acquisition phase and from reversal phase, as primary outcomes, and shared attention and positive affect, as secondary outcomes. Even though data are still being analyzed, we expect that the children with autism will have better performance in the robot condition compared with human condition.

V. Conclusion

Taking the body of this paper as a whole, we present a method to modify My Keepon to be a programmable research platform for HRI studies. Nonpolynomial Labs initialized this idea by sending I^2C commands to the controller board of My Keepon from an Arduino microcontroller board. After that, Beatboxs supported the official firmware with a full set of commands. However, the use of this firmware is not intuitive for non-engineering researchers. Since 2012, we offered an open-source Microsoft Visual C# software to control My Keepon by a GUI. Our website gives a complete tutorial with instructions for hacking the electronics and guidelines for software usage. Users are welcomed to modify the source code or integrate devices to fulfill their research needs or educational purposes. Our psychological experiments are used as examples of using the modified My Keepon in HRI studies. This work is expected to solve the current problem in HRI studies, i.e., the lack of low-cost robot platforms to enlarge the experiment scale or popularize the research results in society.

Future work includes making the modified My Keepon platform compatible with the Robot Operating System (ROS) software framework. With the advantages of ROS, developing software for robot will be easier thanks to ROS tools and libraries, as well as code sharing among researchers in the community.

Acknowledgment

The authors of this paper would like to acknowledge James Kennedy (Plymouth University, UK) for his contribution to adapt our software to the official Arduino firmware of Beatboxs. This work is partially funded by the European Commission 7th Framework Program as a part of the project DREAM under grant no. 611391.

REFERENCES

[1] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: a survey," *Foundations and Trends in Human-Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2007.

[2] T. Fong, C. Thorpe, and C. Baur, "Collaboration, dialogue, human-robot interaction," in *Robotics Research*. Springer, 2003, pp. 255–266.

[3] P. Salvine, M. Nicolescu, and H. Ishiguro, "Benefits of human-robot interaction [TC Spotlight]," *Robotics & Automation Magazine, IEEE*, vol. 18, no. 4, pp. 98–99, 2011.

[4] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, "The intelligent ASIMO: System overview and integration," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3. IEEE, 2002, pp. 2478–2483.

[5] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, "Mechatronic design of NAO humanoid," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 769–774.

[6] N. G. Tsagarakis, G. Metta, G. Sandini, D. Vernon, R. Beira, F. Becchi, L. Righetti, J. Santos-Victor, A. J. Ijspeert, M. C. Carrozza, *et al.*, "iCub: the design and realization of an open humanoid platform for cognitive and neuroscience research," *Advanced Robotics*, vol. 21, no. 10, pp. 1151–1175, 2007.

[7] K. Kaneko, F. Kanehiro, M. Morisawa, K. Miura, S. Nakaoka, and S. Kajita, "Cybernetic human HRP-4C," in *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*, 2009, pp. 7–14.

[8] K. Goris, J. Saldien, B. Vanderborght, and D. Lefeber, "Mechanical design of the huggable robot Probo," *International Journal of Humanoid Robotics*, vol. 8, no. 03, pp. 481–511, 2011.

[9] H. Kozima, M. P. Michalowski, and C. Nakagawa, "Keepon," *International Journal of Social Robotics*, vol. 1, no. 1, pp. 3–18, 2009.

[10] Hasbro, "Furby," http://www.furby.com.

[11] K. Dautenhahn, C. L. Nehaniv, M. L. Walters, B. Robins, H. Kose-Bagci, N. A. Mirza, and M. Blow, "KASPAR–a minimally expressive humanoid robot for human–robot interaction research," *Applied Bionics and Biomechanics*, vol. 6, no. 3-4, pp. 369–397, 2009.

[12] E. S. Kim, D. Leyzberg, K. M. Tsui, and B. Scassellati, "How people talk when teaching a robot," in *Human-Robot Interaction (HRI), 2009 4th ACM/IEEE International Conference on*. IEEE, 2009, pp. 23–30.

[13] C. Vandevelde, J. Saldien, M.-C. Ciocci, and B. Vanderborght, "Systems overview of Ono," in *Social Robotics*. Springer, 2013, pp. 311–320.

[14] J. Gregory, A. Howard, and C. Boonthum-Denecke, "Wii nunchuk controlled dance pleo! dance! to assist children with cerebral palsy by play therapy." in *FLAIRS Conference*, 2012.

[15] J. Dimas, J. Leite, A. Pereira, P. Cuba, R. Prada, and A. Paiva, "Pervasive pleo: long-term attachment with artificial pets," in *Mobile HCI*, 2010.

[16] Iafan, "Hacksby," https://github.com/iafan/Hacksby.

[17] Michael Coppola's Blog, "Reverse Engineering a Furby," http://poppopret.org/2013/12/18/reverse-engineering-a-furby/.

[18] H. Admoni, B. Hayes, D. Feil-Seifer, D. Ullman, and B. Scassellati, "Are you looking at me?: perception of robot attention is mediated by gaze type and group size," in *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2013, pp. 389–396.

[19] Beatboxs, "MyKeepon," https://github.com/BeatBots/MyKeepon/.

[20] Nonpolynomial Labs, "Keepon hacking - proof of concept," http://www.nonpolynomial.com.

[21] Beatboxs, "My Keepon," http://beatbots.net.

[22] H. Kozima, C. Nakagawa, and H. Yano, "Attention coupling as a prerequisite for social interaction," in *Robot and Human Interactive Communication, 2003. Proceedings. ROMAN 2003. The 12th IEEE International Workshop on*. IEEE, 2003, pp. 109–114.

[23] H. Kozima, C. Nakagawa, and Y. Yasuda, "Children–robot interaction: a pilot study in autism therapy," *Progress in Brain Research*, vol. 164, pp. 385–400, 2007.

[24] E. Bernier and B. Scassellati, "The similarity-attraction effect in human-robot interaction," in *Development and Learning (ICDL), 2010 IEEE 9th International Conference on*, 2010, pp. 286–290.

[25] H. Admoni, C. Bank, J. Tan, M. Toneva, and B. Scassellati, "Robot gaze does not reflexively cue human attention," in *Proceedings of the 33rd Annual Conference of the Cognitive Science Society, Boston, MA, USA*, 2011, pp. 1983–1988.

[26] M. G. Marek Michalowski, Kyle Machulis and T. Hersan, "Hack "My Keepon" with an Arduino brain," http://makezine.com/projects/make-35/my-franken-keepon/.

[27] M. Rosen, "Mind to motion: Brain-computer interfaces promise new freedom for the paralyzed and immobile," *Science News*, vol. 184, no. 10, pp. 22–26, 2013.

[28] G. Ranky and S. Adamovich, "Analysis of a commercial EEG device for the control of a robot arm," in *Bioengineering Conference, Proceedings of the 2010 IEEE 36th Annual Northeast*. IEEE, 2010, pp. 1–2.

[29] H. Nisar, Q. Yeoh, H. Balasubramanium, W. Wei, and A. S. Malik, "Analysis of brain activity while performing cognitive actions to control a car," in *The 15th International Conference on Biomedical Engineering*. Springer, 2014, pp. 947–950.

[30] Emotiv, *Emotiv Software Development Kit - User Manual*.

[31] P. Mundy and A. Gomes, "Individual differences in joint attention skill development in the second year," *Infant behavior and development*, vol. 21, no. 3, pp. 469–482, 1998.