

Pianobot: an adaptive robotic piano tutor

Hannes Ritschel, Andreas Seiderer, Elisabeth André

Angaben zur Veröffentlichung / Publication details:

Ritschel, Hannes, Andreas Seiderer, and Elisabeth André. 2020. "Pianobot: an adaptive robotic piano tutor." In *HRI 2020 Workshop on Exploring Creative Content in Social Robotics*, March 23, 2020, Cambridge, UK. Augsburg: Universität Augsburg.
https://mypersonalrobots.org/s/HRI_2020_WS_Creative_SR_paper_10.pdf.

Nutzungsbedingungen / Terms of use:

licgercopyright

Dieses Dokument wird unter folgenden Bedingungen zur Verfügung gestellt: / This document is made available under these conditions:

Deutsches Urheberrecht

Weitere Informationen finden Sie unter: / For more information see:

<https://www.uni-augsburg.de/de/organisation/bibliothek/publizieren-zitieren-archivieren/publiz/>



Pianobot: An Adaptive Robotic Piano Tutor

Hannes Ritschel, Andreas Seiderer and Elisabeth André

Human-Centered Multimedia, Augsburg University

Augsburg, Germany

{ritschel,seiderer,andre}@hcm-lab.de

ABSTRACT

This paper presents a robotic piano tutor which aims to support and motivate students with gamification, hints and feedback. It uses a screen for displaying the musical score, a MIDI keyboard for monitoring the user's play and a social robot for providing feedback. Musical pieces are divided into four categories with different degrees of difficulty. An adaptation approach based on reinforcement learning is used to optimize the hints for the individual user.

1 INTRODUCTION

Social robots become increasingly popular, both for entertainment and education. This includes first commercial domestic companions in general but also research exploring robotic tutors in the context of music. For example, Han et al. [5] use a robot in elementary school music class as a teaching assistant in order to enhance the students' motivation and learning effects. Robots are also studied to act as music teachers in the context of autism patients [21] and as music therapists for people with cognitive impairments [23].

When it comes to playing instruments, recent work by Bagga et al. [1] presents the *instruMentor*, which acts as a robotic musical instrument tutor for teaching the recorder. 3D-printed hands show the correct finger positions on the robot's recorder while the user plays his own instrument. Microphones allow analyzing the user's performance during a lesson. Instructions and feedback in terms of smileys are shown on a touchscreen.

Playing the piano is subject of research by Jig and Lin [6], who present a robot for engaging children from low-income families in out-of-class practicing. It is built from every-day objects and does not provide a traditional piano keyboard but uses the robot's five pressure sensitive feet as input. The included game aims to improve both hands' dexterity and rhythmic precision.

There exist also many commercial apps without robots for piano practicing, either in combination with a MIDI keyboard or on the smartphone touchscreen. Typical features include displaying and following the score and feedback with regard to the user's performance. Open source implementations for PC include e.g. *PianoFromAbove*¹, which does not require the ability to read sheet music, and *PianoBooster*². The latter also features a visualization of the player's timing per note in the score, which makes it easy to recognize which notes were rhythmically inaccurate. There is also work on automatic key detection based on image processing from a camera [3] for displaying suggested fingering and highlighting individual keys in augmented reality style.

While embodied agents offer the opportunity to support creative processes and increase motivation, learning and practicing is an individual process for each student. For example, such an agent



Figure 1: Hardware setup including a robot, a screen and a small MIDI keyboard.

should be able to personalize its assistance to the user, e.g. based on the learning pace. In recent years, many experiments include some kind of adaptation process, e.g. with Reinforcement Learning (RL). This autonomous machine learning approach has in general become very popular for the adaptation of robots' behaviors, such as in the context of post-stroke rehabilitation therapy [24], second-language learning [4], playing games [7], intervention for children with autism spectrum disorder [8], exercising and sportive activities [18], to adapt a robot's linguistic style [9–11, 14–17] or humor [26], only to name a few. In the context of music, Tapus and Mataric [22] use a RL approach for a socially assistive robotic music therapist which aims to maintain attention of older adults with cognitive impairments.

Inspired by aforementioned research and experiments we present an interactive and adaptive robotic piano tutor which supports the user in practicing songs and piano pieces on a keyboard. It provides functionality typically found in piano teaching apps and gives feedback to the player. A RL approach, which is driven by the user's performance, selects hints on how to proceed with practicing.

2 AN ADAPTIVE ROBOTIC PIANO TUTOR

Figure 1 illustrates the scenario: the user sits in front of a screen and practices piano music with different level of difficulty on a keyboard. Sheet music is displayed on the screen, which is also used for interaction with the piano tutoring application. Besides notes, additional help is provided: a virtual keyboard and a cursor in the score highlights the next notes to play. Moreover, the graphical user interface includes a metronome, settings, score selection and

¹<https://github.com/brian-pantano/PianoFromAbove>

²<http://pianobooster.sourceforge.net/>

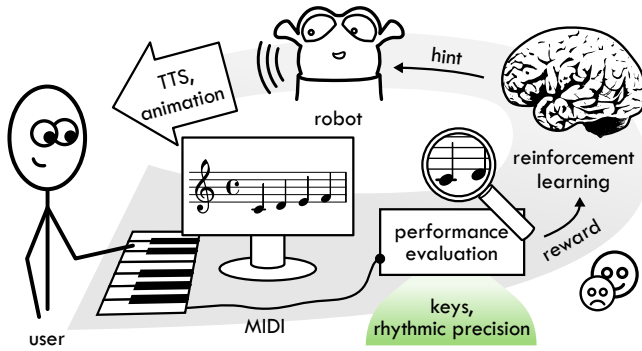


Figure 2: Overview of the adaptive robotic piano tutor.

playback. A social robot supports the human during practice: whenever the user finishes the performance it provides advice on how to proceed and which tools or strategies might be of advantage.

One core aspect of the robotic piano tutor is an adaptation process, which is illustrated in Figure 2. The user’s performance is monitored based on the data from the keyboard. The played keys and – optionally – rhythm are evaluated in order to detect errors. This information serves as input for the adaptation process. RL is used as a machine learning framework for exploring different hints which are presented by the robot. After each performance, the robot gives feedback to the user and – when necessary – suggests e.g. to use the metronome, to repeat the piece or to proceed with another one. The robot uses language to present the hints, underlined with facial expression. Hint selection is based on the learning agent’s former observations and experience. Initially, hints are selected based on trial & error, but the user’s performance is used to improve the hint selection over time.

The application requires prior knowledge from the user with regard to piano playing and music. This includes the ability to read sheet music, i.e. basic knowledge about notes and note values, rests, scales and accidentals. The robot cannot replace a piano teacher, but aims to provide support and increase motivation when practicing alone. It is not able to teach piano playing in general but to provide advice on how to practice the provided scores.

2.1 Gamification

Deterding et al. [2] define gamification as “the use of game design elements in non-game contexts”. Eight different forms of rewards are identified by Wang and Sun [25], including “unlocking mechanisms” or “access”. This type of reward is used by the robotic piano tutor as follows. Musical pieces are divided into four different levels of difficulty: *training*, *easy*, *medium* and *hard*. The first one consists of ascending one-handed and two-handed major scales for practicing black and white keys. Easy pieces include one-handed melodies for the left and right hand as well as two-handed songs (e.g. *Happy Birthday*, folk songs, etc.) with basic accompaniment (e.g. chords in the left hand, see Figure 4). Pieces with medium difficulty are longer, may use polyphony in one hand (two voices playing in parallel) or require larger leaps in the left hand (e.g. *Kalinka*). They also make more use of rests and dynamics. In the last category, pieces are more complex or train e.g. melody takeover between hands (e.g.

Für Elise). Easier pieces have a smaller tonal range, more difficult ones require a larger keyboard with several octaves.

Initially, the user is restricted to practicing pieces with the lowest difficulty and a lock icon is used for more difficult ones. As soon as all pieces in the current category are mastered, the next difficulty level is unlocked. This aims to ensure that the user is able to proceed with advanced challenges and to increase motivation. Moreover, a plot of the user’s performance error (see Section 2.3) is provided as additional feedback and control of the user’s practicing progress.

2.2 Hints

Siebenaler [19] analyzed piano lessons with expert piano pedagogues for children and adults. Different teacher behaviors were identified, such as clapping or singing, playing or talking, giving different directives, asking, showing different kinds of (dis)approval, music talk and off-topic talk. While their results show that students of relatively active piano teachers learn more effectively, the robot at hand is not able to react to errors in the user’s performance in real-time. Thus, after each performance the robot is able to suggest a hint depending on the overall errors during playing the score.

Inspired and adapted from Siebenaler, the robot can suggest to (1) use the metronome, which corresponds to the teacher’s clapping. A similar suggestion is to (2) listen to the song (playback) as a equivalent to the teacher’s playing. Hints with regard to the tempo include (3) playing slower. Moreover, it might also be beneficial to (4) play another piece. Finally, there are also suggestions specifically addressing the application’s virtual score and keyboard, which aims to replace a teacher’s ability to point at notes or keys directly with the hand. For example, one opportunity is to (5) focus on the score following cursor, which highlights the notes to play, to (6) focus on the virtual keyboard, which highlights the keys to play, and to (7) hide keys on the virtual keyboard which are not used in the piece.

The application does not enforce the suggestion automatically, the decision to apply the intervention is up to the user. The graphical user interface provides buttons and controls, e.g. to manipulate the tempo, to activate the metronome or to change the virtual keyboard’s visualization. The robot gives feedback after each of the user’s performances independently of the hints, e.g. “There are still some errors! Keep practicing.” or “That was awesome!”

Since the robot’s embodiment provides the opportunity to communicate with gaze [13], facial expression or sound [12], too, more options for feedback can be explored in the future. For example, its face could either reflect the player’s accuracy or encourage the user with friendly expression or funny grimaces in real-time. Robots with arms and hands could further directly point at wrong notes or keys to play.

2.3 Adaptation

The adaptation process aims to optimize the hints for the individual user based on the player’s performance. It is modeled as a k -armed bandit problem [20], which is a reduced form of RL. The agent’s goal is to find the best of k actions (here: set of hints) by estimating each action’s actual, unknown value q_* . This value is approximated iteratively based on a scalar feedback, the so-called reward signal R . In contrast to full RL, k -armed bandit problems do not have a notion of state.

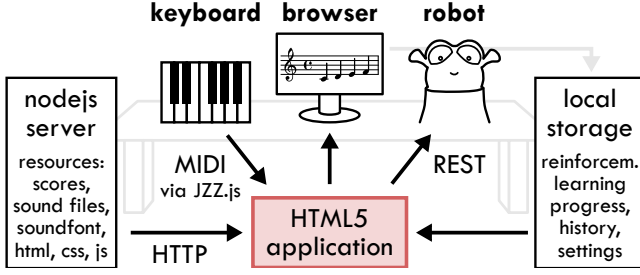


Figure 3: Overview of the technical implementation.

The learning loop is as follows. In each time step t the agent selects an action A_t (see below), executes it, receives a reward R_{t+1} and updates the action's new value Q_{t+1} to approximate q_* . Q_{t+1} is based on R_{t+1} , the old value Q_t and constant learning rate $\alpha \in [0, 1]$:

$$Q_{t+1}(A_t) = Q_t(A_t) + \alpha [R_{t+1} - Q_t(A_t)]$$

Action selection is an important aspect in RL: it must balance both *exploitation* and *exploration* for acting effectively. The former makes the agent *greedy* and uses the approximated values to pick the most promising actions for maximizing the expected reward. The latter addresses uncertainty with regard to the actual Q values and is essential to react to changes. This balance can be realized e.g. with the ϵ -greedy approach, which selects a random action with a small probability $\epsilon \in [0, 1]$ or the action with the highest estimated Q -value (the *greedy* action) with probability $1 - \epsilon$.

In the context of Human-Robot Interaction (HRI), k -armed bandit problems are often modeled as a stationary problem, such as in [7, 17]. The idea is that q_* is fixed and does not change over time. However, we expect that our problem at hand is non-stationary, which means that the actual q_* -values may change over time. This may occur due to fatigue when practicing the same over and over again. Thus, a learning algorithm (as presented above) must be used which is able to react to these changes dynamically.

Error Calculation & Reward. An error rate $e_t \geq 0$ is calculated based on the user's performance by comparing the notes in the score with those played by the user and analyzing their rhythmic precision. While the calculation is quite complex, the general idea is: the more deviations in notes and timing, the bigger the error rate. This value is used as a reward for the last action, depending on the note count $n_t > 0$ of the score:

$$R_{t+1} = -\tanh\left(\frac{e_t}{n_t}\right)$$

3 IMPLEMENTATION

Figure 3 outlines the most important components of the application. In terms of hardware, a keyboard, (touch)screen and the robot are required. Most of the program is written in JavaScript as a browser application.

3.1 Robot

*Reeti*³ is a robot with an expressive face. Its facial expression can be controlled with several motors for the eyes, eyelids, upper and

³<http://www.reeti.fr>

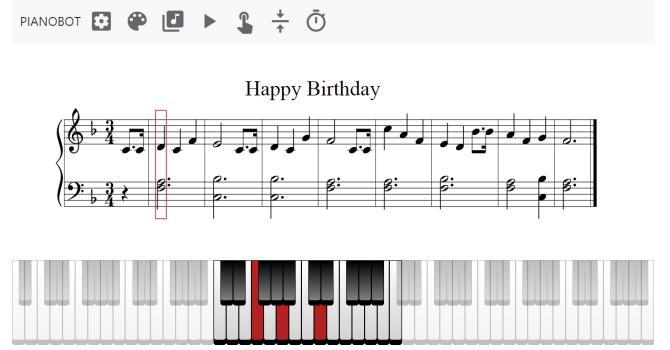


Figure 4: Screenshot of the browser application including menu, score and virtual keyboard.

lower lip as well as left and right cheek. Moreover, the head has three degrees of freedom and the ears can be rotated upwards and downwards, too. Each of Reeti's cheeks contains one RGB LED. Additionally, the robot has an inbuilt speaker and a Text-To-Speech (TTS) module. The manufacturer provides an URBI server and APIs for Java and C++ wrappers. We use a custom REST interface, which generates and sends URBI commands over the network to the robot.

3.2 Browser Application

3.2.1 Score Rendering, Following & Playback. All musical scores are provided as *abc* text files which are rendered by the *abc.js*⁴ JavaScript library. It allows rendering notes in the browser by generating a *Scalable Vector Graphic* (SVG) from textual input. In contrast to other JavaScript note rendering libraries, *abc.js* is performant enough to redraw the score after each keypress, which is required for score following. This cursor allows the user to see the current position in the score by highlighting the next note(s) with a bounding box (see Figure 4). As soon as the correct keys are pressed, the next note(s) are highlighted. This aims to avoid losing track when looking at the (virtual) keyboard. Scores with multiple lines automatically scroll as soon as the first note in the next line is reached. Thus, there is no need to put the hands away from the keyboard. In addition to the bounding box, the corresponding keys are highlighted on the virtual keyboard (see lower part of Figure 4) to find the keys more easily. Common performance instructions, such as repetition marks, are considered.

Moreover, *abc.js* allows generating and playing back the score's audio with a *SoundFont*⁵. The user can listen to the score by pressing the *play* button in the menu. During playback, the same score following functionality is employed as during training.

3.2.2 MIDI keyboard. MIDI support differs between browsers. The *JZZ.js*⁶ library is used for unified and asynchronous access to the keyboard. In general, any MIDI keyboard can be used for interfacing with the application. However, small keyboards with only two octaves, such as in Figure 1, only suffice for the easiest exercises. Four octaves or more are recommended.

⁴<https://www.abcjs.net/>

⁵<https://en.wikipedia.org/wiki/SoundFont>

⁶<https://jazz-soft.net/doc/JZZ/index.html>

4 CONCLUSION

This paper presented an interactive robotic piano tutor, which supports the user during practicing songs and piano pieces with different degrees of difficulty. The robot's feedback and gamification aim to support and motivate the player. The user's performance is monitored based on the input from the MIDI keyboard and used as a reward for an integrated Reinforcement Learning approach. Its goal is to adapt the robot's supportive hints to the individual user in order to optimize practicing.

Several aspects can be improved in the future, such as the error detection. Hints could be complemented with more targeted feedback, such as advice to practice certain parts of the score and more interventions identified by Siebenaler [19]. In order to provide maximum flexibility, users or teachers could import pieces on their own. Consistent fingering for all notes could be added and a keyboard with lit keys could be used instead of the virtual keyboard. Finally, more gamification aspects might further increase motivation, such as earning points according to the player's performance.

ACKNOWLEDGMENTS

The authors would like to thank the students Simone Mayer, Hannes Brugger and Philipp Langenmaier for the implementation of the prototype. This research was funded by the European Union PRESENT project, grant agreement No 856879.

REFERENCES

- [1] Shreyus Bagga, Benedikt Maurer, Tom Miller, Luke Quinlan, Lorenzo Silvestri, Dan Wells, Rebecka Winqvist, Mark Zolotas, and Yiannis Demiris. 2019. InStruMentor: An Interactive Robot for Musical Instrument Tutoring. In *Towards Autonomous Robotic Systems - 20th Annual Conference, TAROS 2019, London, UK, July 3-5, 2019, Proceedings, Part I (Lecture Notes in Computer Science)*, Vol. 11649. Springer, 303–315.
- [2] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart E. Nacke. 2011. From game design elements to gamefulness: defining "gamification". In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, MindTrek 2011, Tampere, Finland, September 28-30, 2011*. ACM, 9–15.
- [3] Adam Goodwin and Richard D. Green. 2013. Key detection for a virtual piano teacher. In *28th International Conference on Image and Vision Computing New Zealand, IVCNZ 2013, Wellington, New Zealand, November 27-29, 2013*. IEEE, 282–287.
- [4] Goren Gordon, Samuel Spaulding, Jacqueline Kory Westlund, Jin Joo Lee, Luke Plummer, Marayna Martinez, Madhurima Das, and Cynthia Breazeal. 2016. Affective Personalization of a Social Robot Tutor for Children's Second Language Skills. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*. 3951–3957.
- [5] Jeong-Hye Han, Dong-Ho Kim, and Jong-Won Kim. 2010. Physical Learning Activities with a Teaching Assistant Robot in Elementary School Music Class. *JCIT* 5, 5 (2010), 30–38.
- [6] Kyrie Jig and Chaolan Lin. 2018. Itchy, Scratchy: A Musical Robot for Low-Income Kid's Piano Practicing. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction, HRI 2018, Chicago, IL, USA, March 05-08, 2018*. ACM, 347–348.
- [7] Iolanda Leite, André Pereira, Ginevra Castellano, Samuel Mascarenhas, Carlos Martinho, and Ana Paiva. 2011. Modelling Empathy in Social Robotic Companions. In *Advances in User Modeling - UMAP 2011 Workshops, Girona, Spain, July 11-15, 2011, Revised Selected Papers*. 135–147.
- [8] Changchun Liu, Karla Conn, Nilanjan Sarkar, and Wendy Stone. 2008. Online Affect Detection and Robot Behavior Adaptation for Intervention of Children With Autism. *IEEE Trans. Robotics* 24, 4 (2008), 883–896.
- [9] Hannes Ritschel. 2018. Socially-Aware Reinforcement Learning for Personalized Human-Robot Interaction. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, 1775–1777.
- [10] Hannes Ritschel and Elisabeth André. 2017. Real-Time Robot Personality Adaptation based on Reinforcement Learning and Social Signals. In *Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, HRI 2017, Vienna, Austria, March 6-9, 2017*. ACM, 265–266.
- [11] Hannes Ritschel and Elisabeth André. 2018. Shaping a social robot's humor with Natural Language Generation and socially-aware reinforcement learning. In *Proceedings of the Workshop on NLG for Human-Robot Interaction*. 12–16.
- [12] Hannes Ritschel, İlhan Aslan, Silvan Mertes, Andreas Seiderer, and Elisabeth André. 2019. Personalized Synthesis of Intentional and Emotional Non-Verbal Sounds for Social Robots. In *8th International Conference on Affective Computing and Intelligent Interaction, ACII 2019, Cambridge, United Kingdom, September 3-6, 2019*. IEEE, 1–7.
- [13] Hannes Ritschel, İlhan Aslan, David Sedlbauer, and Elisabeth André. 2019. Irony Man: Augmenting a Social Robot with the Ability to Use Irony in Multimodal Communication with Humans. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS '19)*. International Foundation for Autonomous Agents and Multiagent Systems, 86–94.
- [14] Hannes Ritschel, Tobias Baur, and Elisabeth André. 2017. Adapting a Robot's Linguistic Style Based on Socially-Aware Reinforcement Learning. In *26th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2017, Lisbon, Portugal, August 28 - Sept. 1, 2017*. 378–384.
- [15] Hannes Ritschel, Kathrin Janowski, Andreas Seiderer, and Elisabeth André. 2019. Towards a Robotic Dietitian with Adaptive Linguistic Style. In *Joint Proceeding of the Poster and Workshop Sessions of Aml-2019, the 2019 European Conference on Ambient Intelligence, Rome, Italy, November 13-15, 2019 (CEUR Workshop Proceedings)*, Vol. 2492. CEUR-WS.org, 134–138.
- [16] Hannes Ritschel, Andreas Seiderer, Kathrin Janowski, İlhan Aslan, and Elisabeth André. 2018. Drink-O-Mender: An Adaptive Robotic Drink Adviser. In *Proceedings of the 3rd International Workshop on Multisensory Approaches to Human-Food Interaction (MHFI'18)*. ACM, Article 3, 8 pages.
- [17] Hannes Ritschel, Andreas Seiderer, Kathrin Janowski, Stefan Wagner, and Elisabeth André. 2019. Adaptive linguistic style for an assistive robotic health companion based on explicit human feedback. In *Proceedings of the 12th ACM International Conference on Pervasive Technologies Related to Assistive Environments, PETRA 2019, Island of Rhodes, Greece, June 5-7, 2019*. 247–255.
- [18] Sebastian Schneider and Franz Kummert. 2017. Exploring embodiment and dueling bandit learning for preference adaptation in human-robot interaction. In *26th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2017, Lisbon, Portugal, August 28 - Sept. 1, 2017*. IEEE, 1325–1331.
- [19] Dennis J Siebenaler. 1997. Analysis of teacher-student interactions in the piano lessons of adults and children. *Journal of Research in Music Education* 45, 1 (1997), 6–20.
- [20] Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement Learning - An Introduction (Second Edition)*. MIT Press.
- [21] Alireza Taheri, Ali Meghdari, Minoo Alemi, and Hamidreza Pouretmad. 2019. Teaching music to children with autism: a social robotics challenge. *Scientia Iranica* 26, Special Issue on: Socio-Cognitive Engineering (2019), 40–58.
- [22] Adriana Tapus and Maja J. Mataric. 2008. Socially Assistive Robotic Music Therapist for Maintaining Attention of Older Adults with Cognitive Impairments. In *AI in Eldercare: New Solutions to Old Problems, Papers from the 2008 AAAI Fall Symposium, Arlington, Virginia, USA, November 7-9, 2008 (AAAI Technical Report)*, Vol. FS-08-02. AAAI, 124–127.
- [23] Adriana Tapus, Cristian Tapus, and Maja Mataric. 2009. Music Therapist Robot: a solution for helping people with cognitive impairments. In *IJCAI Workshop on Intelligent Systems for Assisted Cognition, July, Pasadena, CA*.
- [24] Adriana Tapus, Cristian Tapus, and Maja J. Mataric. 2008. User - robot personality matching and assistive robot behavior adaptation for post-stroke rehabilitation therapy. *Intelligent Service Robotics* 1, 2 (2008), 169–183.
- [25] Hao Wang and Chuen-Tsai Sun. 2011. Game reward systems: Gaming experiences and social meanings. In *Proceedings of the 2011 DiGRA International Conference: Think, Design, Play, DiGRA 2011, Hilversum, The Netherlands, September 14-17, 2011*. Digital Games Research Association.
- [26] Klaus Weber, Hannes Ritschel, İlhan Aslan, Florian Lingenfelder, and Elisabeth André. 2018. How to Shape the Humor of a Robot - Social Behavior Adaptation Based on Reinforcement Learning. In *Proceedings of the 2018 on International Conference on Multimodal Interaction, ICMI 2018, Boulder, CO, USA, October 16-20, 2018*. 154–162.