RoboCanes-VISAGE 2019
Team Description Paper

P. Peña[1], J. Masterjohn[1], L. Beaufils[1], M. Davis[1], U. Visser[1], and C. Lisetti[2]

[1] University of Miami, Coral Gables FL 33146, USA
robocanes@cs.miami.edu
[2] Florida International University, Miami FL 33199 USA
affect@cs.fiu.edu

Abstract. RoboCanes-VISAGE consists of faculty and PhD students from two Universities in Miami, FL: the University of Miami (UM), Department of Computer Science and Florida International University (FIU), School of Computing and Information Sciences. Together, our research covers artificial intelligence, robotics, human-robot interaction, and affective social computing. This paper describes our approach for the 2019 competition.

1 Introduction

RoboCanes was incepted in January 2010 at the University of Miami in the USA under the supervision of Dr. Ubbo Visser. The team has designed, developed, and implemented its autonomous agent framework and software from the ground up in both the 3D Soccer Simulation League and the Soccer Standard Platform League (NAO robots). This framework has evolved to a flexible research platform that led to over 40 publications and shared software (see section 4) over the years.

For RoboCup@Home, RoboCanes will team up with the VISAGE lab from FIU to form a new team: RoboCanes-VISAGE. The VISAGE lab is led by Dr. Christine Lisetti, a faculty from the School of Computing and Information Sciences at FIU who is one of the founders of the Affective Computing research field. Dr. Lisetti's group is known for it's research on social 3D Virtual Avatars.

We will use Toyota's HSR to leverage latest progress in affective social computing and socially intelligent agents as well as using latest technology in AI and robotics to address the RoboCup@Home challenges. It is an excellent platform to embody the integration of UM's RoboCanes agent with the FIU VIrtual Social AGEnt (VISAGE). Our RoboCanes framework is similar to the ROS architecture and although we will use ROS as the framework for HSR we will be able to use software components from our RoboCanes framework (e.g. HOG detector).

The RoboCanes agent will be mainly responsible for managing and controlling navigation, object manipulation, grasping, etc., while the VISAGE agent will handle face and facial expression recognition of the human interacting with the HSR, voice recognition and speech synthesis, and 3D-graphics facial and gesture synthesis of the VISAGE agent.

2 Research Focus

We propose to leverage and expand latest research on social robotics to enhance and personalize collaborative robots' (co-robots) natural verbal and non-verbal communication with humans. We will focus on co-robots' communication and collaboration in the home environment, where natural communication is of essence. Literature reveals that a vast majority of research is focused on low-degree of freedom anthropomorphic robots. We will develop our interactive agent on a high-degree of freedom hybrid anthropomorhic robot.

The RoboCanes agent will be mainly responsible for managing and controlling navigation, object manipulation, grasping, etc., while the VISAGE agent will handle face and facial expression recognition of the human interacting with the HSR, voice recognition and synthesis, and 3D-graphics facial and gesture synthesis of the VISAGE agent.

We will integrate and coordinate both agents toward a coherent and engaging multimodal model of communication with the human user.

3 Innovative technology

Our group has made a variety of scientific contributions in the areas of artificial intelligence, robotics, virtual agents/human-computer interaction, and with both groups (UM, FIU) together also a first human-robot interaction (section 3.6) and a human-robot interface (section 3.1).

3.1 eEVA as a real-time multimodal agent human-robot interface

We have developed a multimodal human-robot interface [1] for the Toyota's Human Support Robot (HSR, designed to help people in homes or offices) which integrates the RoboCanes agent and the Embodied Empathetic Virtual Agent (eEVA) developed by FIU's VISAGE lab. The RoboCanes agent is responsible for managing and controlling navigation, object manipulation, grasping, among other physical actions, while the VISAGE agent is responsible for recognizing and displaying social cues involving recognizing the user's facial expression and speech, synthesizing speech with lip-synchronization, and portraying appropriate facial expressions and gestures.

We created a greeting context for the pilot study of our first social human-HSR interactions with our RoboCanes-VISAGE interface by designing a small set of greeting gestures to personalize Toyota HSR with its users greeting preferences (and to establish some initial rapport in future more advanced studies): the Toyota HSR generates greeting gestures from four different cultures such as waving-hand (Western), fist-bump (informal Western), Shaka (Hawaii), and bowing (Japan) greeting gestures. The HSR's gesture greetings are performed based on the user's spoken selection of one of the four greetings and our pilot questionnaire aimed to assess the impact of combining the virtual agent interface on the user's experience (*e.g.*, feelings of enjoyment, boredom, annoyance, user's perception of the robot's friendliness or of competence).

3.2 Toolbox: Low footprint reinforcement learning library RLLib

One of our main accomplishments is the creation of the RLLib tool package that we have made available for other scientists. **RLLib has been downloaded by other peer researchers more than 5,000 times to date** [2], a recognition of the significance of our contribution to the scientific community. RLLib is a C++ template library to learn behaviors and represent learnable knowledge using on/off policy RL standard, and gradient temporal-difference learning algorithms in RL. It is an optimized library for robotic applications that operates under fast duty cycles (e.g., $\leq$ 30 ms). This is a significant difference to other available packages. RLLib has been tested and evaluated on RoboCup 3D soccer simulation agents, physical NAO V4 humanoid robots, and Tiva C series launchpad microcontrollers to predict, control, learn behaviors, and represent learnable knowledge. The implementation of the RLLib library is inspired by the RLPark API, which is a library of temporal-difference learning algorithms written in Java. RLLib garnered attention when presented at RoboCup Symposium and has been used by third parties as well, e.g. on malware detection by Bidoki et al. [3]. We will make use of the library also for the HSR. We will make use of the library also for the HSR.

3.3 Monitor/Debugger: Real-time 3D visualization

Another line of research is about high-quality and real-time 3D visualization for autonomous robots. We have developed **RoboViz**, a software program designed to assess and develop agent behaviors in the RoboCup Soccer Simulation 3D league multiagent system [4, 5]. Starting with the German Open in Magdeburg, Germany in April 2011, **RoboViz has been officially adopted as the default monitor (visualization tool) by the RoboCup Federation** in the 3D Soccer Simulation League. The tool facilitates real-time visualization of agents running concurrently on the RoboCup simulator to provide higher-level analysis of agent behaviors not currently possible with existing tools. Provided with appropriate hardware, the monitor and debugging tool can produce high-quality stereo vision images. RoboViz is an interactive monitor that renders both agent and world state information in a three-dimensional scene. In addition, RoboViz provides programmable remote drawing functionality to agents or other clients that can communicate over a network.

3.4 Dynamic motions: Walking and kicking engine on bipeds

We have developed a new **Walking Engine** for adaptive walking on bipedal robots. It is a LIPM-based closed-loop walk, that adapts to differences in the physical behavior of the robot by optimizing parameters of the model directly on the NAO while walking and executing other tasks. A significant amount of errors in the model predictions can be reduced without using a more complex model, simply by adjusting the LIPM to fit the observed behavior. Our experiments show that the optimized model yields a more controlled, faster and even more energy-efficient walk on different NAO robots, and on various surfaces without additional manual parameter tuning [6, 7].

We also developed a new dynamic **Kicking Engine** for bipedal robots. It can generate a kick trajectory with an arbitrary direction without prior input or knowledge of the parameters of the kick. The trajectories are generated using cubic splines (two degree three polynomials with a via-point). The trajectories are executed while the robot is dynamically balancing on one foot. When the robot swings the leg for the kick motion, unprecedented forces might be applied on the robot. To compensate for these forces, we developed a Zero Moment Point (*ZMP*) based preview controller that minimizes the ZMP error. Parameters such as kick configuration, limit of the robot, or shape of the polynomial can be optimized. We showed an optimization framework based on the Webots simulator to optimize these parameters. Experiments of the physical NAO robot show positive results [8, 9].

3.5 Motivational interviewing with intelligent virtual agents (IVA)

We developed a virtual counseling system which can deliver brief health interventions via a **3D anthropomorphic speech-enabled interface** – a new field for spoken dialog interactions with intelligent virtual agents in the health domain. We developed our dialog system based on a MDP framework and optimized it by using RL algorithms with data we collected from real user interactions. The system begins to learn optimal dialog strategies for initiative selection and for the type of confirmations that it uses during the interaction. We compared the unoptimized system with the optimized system in terms of objective measures (e.g. task completion) and subjective measures (e.g. ease of use, future intention to use the system) and obtained positive results. The system is able to learn dialog strategies for initiative and confirmation selection. Our **contributions to the Spoken Dialog Systems domain** include the creation of a RL paradigm to the completely new domain of behavior change - where our dialog length is 4-5 times longer and where the nature of the dialog is less restricted than spoken dialog systems operated in tourist information domain. We **contributed to the healthcare domain** with the first system to use speech as an input medium with a RL-based approach. Our initial evaluation showed that the dialog managers that are optimized with RL have the potential to reach optimal behavior, given enough training data [10, 11].

3.6 Human-robot interaction with a humanoid robot

We combined a spoken dialog system that we developed to deliver brief health interventions (see above section 3.5) with the a NAO robot. The dialog system is based on a framework facilitating a MDP and is optimized using RL algorithms (we used our own RLLib [2], section 3.2). The spoken dialog system for the humanoid robot was a novelty at that time and exists as a proof of concept. We anticipate that the NAO robot will become a very likable and effective mode of delivery for brief interventions for target behaviors such as poor diet, overeating, or lack of exercise, among others. The appeal of the NAO to children makes it particularly suitable to become a child's favorite health coach, say, to discuss eating more fruits and vegetables on a daily basis.

3.7 Knowledge representation and reasoning

This line of research **combines** modern symbolic knowledge representation and reasoning techniques from the **Semantic Web** domain with modern **autonomous robots**. Knowledge should be represented in real-time (i.e., within ms) and deduction from knowledge should be inferred within the same time constraints. We proposed an extended assertional formalism for an expressive $\mathcal{SROIQ}(\mathcal{D})$ Description Logic to represent asserted entities in a lattice structure [12]. This structure can represent temporal-like information. Since the computational complexity of the classes of description logic increases with its expressivity, the problem demands either a restriction in the expressivity or an empirical upper bound on the maximum number of axioms in the knowledge base. We have conducted experiments in the RoboCup 3D Soccer Simulation League environment and provide justifications of the usefulness of the proposed assertional extension. We showed the feasibility of our new approach under real-time constraints and conclude that a modified FaCT++ reasoner empirically outperforms other reasoners within the given class of complexity. We intend to use our approach with incremental reasoning on a physical robot to model beliefs and interpret entities in uncertain environments in the near future.

4 List of externally available components

We have produced and shared a variety of our system components to support various research communities. Some of the components are part of the prior section (3) and will only be short-listed here.

HapFACS[3] is an open source software that enables (without prior knowledge of computer graphics) the animation of speaking 3D virtual human-like characters with physiologically realistic facial expressions that have been validated by experts in facial expressions [13]. Specifically, HapFACS provides the ability to manipulate the activation – in parallel or sequentially – of combinations of the smallest groups of (virtual) facial muscles capable of moving independently in the human face, for the creation of physiologically and socially believable speaking virtual agents.

RLLib[4] (cf. section 3.2) is a C++ template library to learn behaviors and represent learnable knowledge using on/off policy RL standard, and gradient temporal-difference learning algorithms in RL.

RoboViz[5] (see section 3.3) is a software program designed to assess and develop agent behaviors in the RoboCup 3D Soccer Simulation league.

5 Applicability to the Real World

We propose to model short- and long-term social interactions with HSR, based on a safe HRI social agent able to establish rapport and conduct useful tasks in

[3] http://ascl.cis.fiu.edu/hapfacs-open-source-softwareapi-download.html
[4] https://mloss.org/software/view/502/
[5] https://sites.google.com/site/umroboviz/

the primary home care environment with carefully synthesized, synchronized and safe motions. The home care context for human-robot interaction is predicted to become increasingly important, especially with the baby boomers aging generation in mind. This generation is tech-savvy, highly educated, and are anticipated be open for a robot assisting them in their home, to remain mobile and to live independently longer in their home. Our research can therefore be of high social impact. We will now present a technical overview of each subsystem developed by RoboCanes-VISAGE when solving such a task:

Computer Vision: Our object recognition system is based on a variety of methods and can handle objects on tables and shelves. We are using the Point Cloud Library 1.7 (PCL) for our point cloud processing [14] and random sample consensus (RANSAC) method on our point clouds to estimate model parameters for planar regions in the image in order to identify plane candidates. Those plane candidates are used to identify the surfaces the objects are lying on. We then facilitate YOLO [15] for real-time object detection. We are optimizing/training YOLO based on the objects given for the competition. The bounding boxes received from YOLO are then used to extract the depth information from the point cloud data to get the cartesian coordinates of the object. This information is passed to the object manipulation node described next.

Manipulation: Our manipulation is based on the software provided by ROS/TMC. We are running our path planning and inverse kinematics nodes using MoveIt! [16]. We developed software providing information for the motion planner, especially for collision detection and orientation of the gripper. MoveIt! plans the end effector to be at a certain distance of the object defined in our node as the pre-grasp state. In the post-grasp state, the robot then uses the arm and base controller to get close to the object. This state machine allows the robot to approach the object without altering it's state by tipping it over. Once MoveIt! finds a successful goal state from the start state, the robot executes the plan. Based on the perception from the Computer Vision node, the Manipulation node decides on which side to grab the object. We then use the gripper controller provided by TMC in order to provide a force for the grip on an object. We also developed the software that adds/removes objects from the kinematic tree of the robot and planning scene.

Natural Language Understanding: In order to understand the user, we developed a system that integrates multiple Natural Language Understanding, *NLU*, technologies to allow a slightly more robust dialogue between the agent and the user. Chatito [17] is a tool used to generate training data for Rasa [18] and Snips [19], which are open source NLU frameworks. The way we use them is as follows: we train both on the same intent-entity pairings so that they could recognize instruction and parse it properly. The key difference is that Snips was trained to be implicit, whereas Rasa was not. That is to say, Snips will only accept instructions that perfectly match the generated Chatito data, and if not, it will fail. Rasa, on the other hand, learns and generalizes slightly, which is helpful to understand instructions on which it was not explicitly trained, but can lead to errors due to overgeneralization and misclassification. By having two frameworks work in parallel, the agent can intend to undertand the intention of the user more reliably. The input into the NLU is Google Speech API.

References

1. Pedro Peña, Christine Lisetti, Mihai Polceanu, and Ubbo Visser. eEVA: Real-time Web-based Affective Agents for Human-Robot Interface. In Dirk Holz and Katie Genter and Maarouf Saad and Oskar von Stryk, editor, *Robot World Cup XXII*. Springer Berlin / Heidelberg to appear, 2019.
2. Saminda Abeyruwan and Ubbo Visser. Rllib: C++ library to predict, control, and represent learnable knowledge using on/off policy reinforcement learning. In *Robot Soccer World Cup*, pages 356–364. Springer, 2015.
3. Seyyed Mojtaba Bidoki, Saeed Jalili, and Asghar Tajoddin. Pbmmd: A novel policy based multi-process malware detection. *Engineering Applications of Artificial Intelligence*, 60:57–70, 2017.
4. Justin Stoecker and Ubbo Visser. RoboViz: Programmable Visualization for Simulated Soccer. In Thomas Röfer, Norbert Michael Mayer, Jesus Savage, and Uluç Saranli, editors, *RoboCup 2011: Robot Soccer World Cup XV*, 2012, to appear.
5. Justin Stoecker and Ubbo Visser. Visualizing and Debugging Complex Multi-Agent Soccer Scenes in Real Time. In in print, editor, *RoboCup 2013: Robot Soccer World Cup XVII*. Springer Berlin / Heidelberg, 2014.
6. *Andreas Seekircher and Ubbo Visser*. A Closed-Loop Gait for Humanoid Robots Combining LIPM with Parameter Optimization. In Sven Behnke, Daniel Lee, Sanem Sariel, and Raymond Sheh, editors, *RoboCup 2016: Robot Soccer World Cup XX*. Springer Berlin / Heidelberg, 2017. to appear.
7. *Andreas Seekircher* and Ubbo Visser. An Adaptive LIPM-based Dynamic Walk using Model Parameter Optimization on Humanoid Robots. *Künstliche Intelligenz (KI), Special Issue RoboCup*, 30(3):1–12, July 2016.
8. Pedro Pena, Joseph Masterjohn, and Ubbo Visser. Optimizing kick trajectory: A comparative study. In Christoph Benzm\"uller, Christine Lisetti, and Martin Theobald, editors, *GCAI 2017. 3rd Global Conference on Artificial Intelligence*, volume 50 of *EPiC Series in Computing*, pages 239–245. EasyChair, 2017.
9. Pedro Pena, Joseph Masterjohn, and Ubbo Visser. An Omni-directional Kick Engine for Humanoid Robots with Parameter Optimization. In Hidehisa Akiyama, Oliver Obst, Claude Sammut, and Flavio Tonidandel, editors, *RoboCup 2017: Robot Soccer World Cup XXI*. Springer Berlin / Heidelberg, 2018. to appear.
10. Ugan Yasavur and Christine. Lisetti. Let's talk! speaking virtual counselor offers you a brief intervention. *Int'l Journal of Multimodal User Interfaces*, 8:281–306, 2014.
11. Christine Lisetti, Reza Amini, and Ugan Yasavur. Now all together: Overview of virtual health assistants emulating face-to-face health interview experience. *KI-Künstliche Intelligenz*, pages 1–12, 2015.
12. Saminda Abeyruwan and Ubbo Visser. A New Real-Time Algorithm to Extend DL Assertional Formalism to Represent and Deduce Entities in Robotic Soccer. In H. Akin Reinaldo, A. C. Bianchi, Subramanian Ramamoorthy, and Komei Sugiura, editors, *RoboCup 2014: Robot Soccer World Cup XVIII*, LNAI, pages 270–282. Springer Berlin / Heidelberg, 2015.

13. Reza Amini and Christine Lisetti. Hapfacs: An open source api/software to generate facs-based expressions for ecas animation and for corpus generation. In *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*, pages 270–275. IEEE, 2013.
14. Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.
15. Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
16. Ioan A. Sucan and Sachin Chitta. Moveit! available: http://moveit.ros.org, last visited on 02/16/2018.
17. Rodrigo Pimentel. Chatito available: https://rodrigopivi.github.io/chatito/, last visited on 10/26/2018.
18. Tom Bocklisch, Joey Faulker, Nick Pawlowski, and Alan Nichol. Rasa: Open source language understanding and dialogue management. *arXiv preprint arXiv:1712.05181*, 2017.
19. Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*, 2018.

A

3rd Party Robot's Software

- YOLO
- Moveit!
- Snips NLU
- Rasa NLU
- TMC software
- PCL (Point Cloud Library)
- Google Speech API
- Robot Operating System (ROS)

B

External Computing Devices

- Alienware 17 Gaming Laptop
  - Processor: Intel Core i7-8750H CPU @ 2.20GHz x 12
  - Graphics: GeForce GTX 1070/PCIe/SSE2
  - OS Type: 64-bit
  - Disk: 1.2 TB

Fig. 1: HSR in our lab, facing Toyota's lego block challenge