

MiRo - a playful companion

| | |
|----------------------|--|
| Report Name | Project Outline |
| Author (User Id) | Aleksandra Madej (alm82) |
| Supervisor (User Id) | Patricia Shaw (phs) |
| Module | CS39440 |
| Degree Scheme | GG4R (Computer Science and Artificial Intelligence) |
| Date | February 7, 2019 |
| Revision | 1.0 |
| Status | Release |

1 Project description

“ MiRo - a playful companion ” project will develop a behaviour learning system for a pet-shaped robot MiRo. The robot will be able to detect user’s commands and learn new tricks based on provided feedback about its actions.

Most of existing pet robots are equipped with predefined set of possible user actions and corresponding robot behaviours. However, during interaction with real animals, it is often the training process that helps to develop the bond between the human and the animal. To simulate this kind of experience MiRo will be made fully trainable - given no initial knowledge of tricks or expected behaviours it will have to learn them by interaction with people. This will result in more realistic user experience but also allow people to personalize commands and resulting actions to their own preferences [1].

At the beginning MiRo will only be capable of performing a number of simple actions such as “go forward”, “turn left”, “bark”- each of them with equal chance of being chosen. These probabilities will later be affected by emotions associated with performing each action. Positive emotions, that could be triggered by positive user feedback e.g. stroking MiRo’s head, will increase the chance of action being chosen in the future while negative emotions will reduce it. No feedback after performing an action will be seen as negative reaction which will effectively allow MiRo to forget previously learned tricks if it is not rewarded for them anymore.

MiRo will also be able to detect and recognize commands given by user and develop separate sequences of actions for each of them [2]. When new command is detected it will perform random actions and wait for feedback to learn a new trick. If MiRo detects command it already knows it will perform set of actions associated with the best user’s reaction and proceed with random actions allowing to extend the existing sequence. If no action is associated with high enough reward it will finish the sequence and return to it’s idle state.

Learning rate will also be affected by MiRo’s overall mood [3] for increased realism - it will learn new tricks much faster if its happy and refuse to learn anything while annoyed or anxious. This will also mean that it will learn faster the longer the training session is, as the positive feedback required for learning will keep improving its mood.

After the play session is finished results will be saved into robots memory file so that it can remember previously learned behaviours at the beginning of next session.

The project will also involve developing a simple system for monitoring MiRo’s learning progress. This will display commands recognized by robot as well as sequences of actions associated with them.

2 Proposed tasks

- **Investigation of ROS** - the project involves using Robot Operating System to interact with MiRo. It will be necessary to explore its base concepts such as nodes, messages and topics as well as choose and become familiar with suitable testing framework. [4]
(weeks 2 - 4)
- **Setting up version control system** - a repository will have to be created to ensure possibility of recovering previous versions of the code and documents and control changes that have been made over time (week 2)

- **Project Meetings and Project Diary.** - two weekly meetings with supervisor will be organized. During the meetings progress on work, problems and new ideas will be discussed. A project diary will be created and updated with details to discuss on the meeting and any other useful information found for the future use in the project. (week 2)
- **Development**
 - **Base actions** - the decision will have to be made about types of basic actions robot will be capable of from the start. These actions will then have to be implemented. (week 3)
 - **Command detection** - different methods of command detection will have to be analyzed (e.g. sound pattern recognition, pose recognition [5]). After the choice is made implementation will amount to significant part of the programming work. (weeks 3 - 7)
 - **Sequence building** - creating actions sequence based on user feedback will be another large part of coding work. This will include building an ordered list of behaviours as well as adjusting probabilities of each action being performed as the robot is being trained. (weeks 5 - 9).
 - **Idle behaviour** - simple idle behaviour for the robot could be developed if there is enough time by the end of the project. As this is not strictly related to the main subject of the project (learning process) it is optional. This could involve randomized movements with obstacle avoidance. (week 10)
 - **Unit testing** - each part of code created will have to be properly tested for possible failures. This will be done continuously throughout the project alongside programming work. (weeks 3 - 10)
 - **Monitoring system** - this will be simple system for monitoring learned behaviours and presenting learning progress. Initial version would be text based (week 5) and improved GUI version could be created after other programming work is finished (week 9 and 10).
- **Preparation for demonstrations** - as a part of this project two demonstrations will have to be made. Preparation for them will involve creating overall plan of the demonstration, notes with important information to talk about and explanatory diagrams, as well as practising the demonstration process (week 6 and 15)
- **Creating final report** - final results and development process will be discussed in the report. (weeks 8-10)

3 Project deliverables

- **Documentation and instructions** - this will contain user instructions for teaching MiRo, using monitoring system and information about known issues (if there are any).
- **MiRo learning software** - files responsible for MiRos behaviour and learning process, monitoring system, MiRos memory files.
- **Test scripts** - set of automated unit tests, and manual tests description with results achieved by the end of the project.
- **Final Report** - discussion of created software, development process, techniques, third party libraries and tools used as well as scientific papers that have influenced work on the project.

- **Mid-Project and Final Demonstration Notes** - notes or slides with important aspects to cover during presentation, diagrams presenting learning process

References

- [1] <https://uhra.herts.ac.uk/bitstream/handle/2299/1898/902086.pdf?sequence=1>
Paper describing socializing and personalizing robots
- [2] http://socialrobotsinthewild.org/wp-content/uploads/2018/03/HRI-SRW_2018_paper_11.pdf
Paper describing developmental learning for social robots
- [3] https://consequential.bitbucket.io/Technical_Interfaces_Core_Interface.html
Information about MiRo's mood and emotions
- [4] <http://wiki.ros.org/ROS/Tutorials>
Tutorials for using ROS
- [5] https://github.com/JannisBush/ros_posenet
Library for estimating human poses