Towards Enriching Robot's Actions with Affective Movements

Julian M. Angel-Fernandez
Automation and Control Institute, Vienna
University of Technology
Karlspltz 13, 1040
Vienna, Austria
julian.angel.fernandez@tuwien.ac.at

Andrea Bonarini
Dipartimento di Elettronica, Informazione e
Bioingegneria, Politecnico di Milano
Piazza Leonardo da Vinci 31, 20133
Milan, Italy
andrea.bonarini@polimi.it

ABSTRACT

Emotions are considered by many researchers as beneficial in social robotics, since they can enrich human-robot interaction with non-verbal clues. Although there have been works that have studied emotion expression in robotics, the mechanisms created to project emotion are usually highly integrated in each solution. This limits the possibility to develop a general approach. This paper presents a system that has been initially created for a theatrical robot to enrich its actions with emotions, but it has been designed to be adaptable to other fields. The emotional enrichment system has been envisioned to be used with any action decision system.

Keywords

Human-Robot Interaction; Emotions Enrichment System; Emotion Projection

1. INTRODUCTION

Bringing robots in a social context raise the challenge to increase their acceptance. It is expected that people would treat robots as humans, like they do with computers. [1], which makes necessary to devise robots that match people's expectations. Some researchers have suggested that embedding emotion expression capabilities to robots could improve their acceptance in social environments [2]. As a consequence, they added specific emotional poses and expressions to their robots. Others have studied how to convey emotions with specific platforms [3, 4]. Nevertheless, these research activities have developed modules to show emotions that are strongly integrated with their specific setting, which eliminate the possibility to re-use or adapt their solutions for other projects.

This paper presents an Emotional Enrichment System, which modifies actions' parameters and adds additional actions to create the illusion of emotion expression in a robot. Although the system was originally conceived to be used

in an autonomous performance robot to enrich actions with emotions [5], its design was devised to make it extensible to other platforms and adaptable to new tasks. To achieve this goal, the system relies on an Emotional Execution Tree, which is based on simple actions or primitive, sequential, and parallel nodes. Additionally, the concept of compound actions is adopted to group a bunch of nodes, which reduces the tree dimension and allows the reuse of recurrent actions generated by specific combinations of simple actions and other nodes.

2. EMOTIONAL ENRICHMENT SYSTEM

The main idea of Emotional Enrichment System is to blend a specific emotion with a desired action to generate "emotional actions". Differently from previous approaches, the system has been envisioned to allow: (i) interoperability among different platforms, (ii) introduction of new parameters and emotions, and (iii) interface with diverse action decision systems.

Therefore, the system could be thought as a black-box that receives a desired action and emotion, and produces an emotionally enriched action through modification of actions' parameters and addition of other actions to the original one. This modification is done following the description given in two configuration files. One defines how actions' parameters should be modified and what actions should be added to project the specific emotion. The second file describes "personal" traits to specific emotions. For example, one person could intensify his happiness, but conceal his sadness.

2.1 Basic Concepts

The system is based on six main concepts: *simple actions*, *compound actions*, *action message*, *emotional descriptors*, *character description* and *emotional execution tree*. *Simple actions* are actions that are considered as primitives: they are used as building blocks. Therefore, these actions are described in the system and are the ones in which the emotional enrichment takes place. Their description specifies mandatory and optional parameters that are required to execute an action. *Compound actions* are actions that are created from simple actions. These actions are not implemented in the system, but, if it is needed, they can be described in it (e.g. compound actions that are used often). *Action message* establishes the structure of the message to describe any kind of action (i.e., simple and compound). This message also specifies how the actions are executed (i.e., in parallel or in sequence) and which action is predominant

(i.e., primary or secondary). *Emotional parameters* describe how the emotional enrichment should be done to convey a specific emotion in a specific simple action. This description could also include addition of other simple actions. *Character description* enables the possibility to establish how to modify emotional expressions to generate diverse treats. Finally, *Emotional Execution Tree* is a computational representation of desired actions that should be executed. This tree is first created from the action message description and then modified using the emotional parameters and character description.

2.2 Emotional Execution Tree

Emotional Execution Tree is a connected acyclic graph $G(V, E)$ with $|V|$ vertexes and $|E|$ edges. The root and non-leaf nodes could be of either *parallel* or *sequential* type. The parallel node could be one out of four different sub-types: (i) action and emotion synchronous, (ii) action synchronous and emotion asynchronous, (iii) action asynchronous and emotion synchronous, or (iv) action and emotion asynchronous. Sequential nodes could just be one of two sub-types: emotion synchronous or asynchronous. Action synchronous means that each time that a parallel node receives a "finish" notification (i.e., success or failure), it will broadcast the message to all nodes that derived it and to its predecessor. If a sequence node receives a finish message, it will execute the next branch. When all branches have been executed, it communicates the end of the action. On the other hand, emotion synchronous means that each time that a node (either sequence or parallel) receives an emotion synchronization message, it will propagate the message to all branches to move to the consecutive emotional expression. If a node is principal and it has finished to execute all actions, it will notify its predecessor.

This distinction creates the possibility to synchronize emotional changes without affecting the normal execution of an action and it also enables synchronization among parallel actions. Finally, leaf nodes could only be simple action nodes that have been implemented in the system. Any node can belong to one of two levels: principal or secondary. If a node is principal, it will notify its predecessor about the messages that it has received, while the secondary node cannot propagate any message to its predecessor.Compound actions are implemented combining all type of nodes.

2.3 Process

When the system is launched, it starts reading emotion descriptors, and it creates an internal representation. Every time the system receives a new action message the verification process starts. This message includes a description of actions that should be executed, with their respective values. If the action is compound, it is first decomposed in simple actions, which will be labeled as primary actions, and checked if all actions exists in the system and the parameters correspond to the specification. If both conditions are fulfilled, the system generates an Emotion Execution Tree that represents the specified action. A similar procedure is done for simple action, but, instead of decomposing the action, the system directly verifies the parameters and then generates the execution tree. Once the Emotion Execution Tree has been created, the system proceeds to add actions that are required to express the desired emotion. Then, it proceeds to modify the parameters to convey the emotion,

considering emotion and character descriptors. When the process is completed, the system feeds information to the correct "drivers", which have been implemented to execute the desired action on a specific platform. Every time that a driver detects the end of the action, it informs the Emotional Enrichment System, which will take the proper actions.

2.4 Implementation

The system has been implemented in C++ and interfaced with ROS . To fulfill all the requirements, the system was implemented in two different ROS nodes : one containing the enrichment system, and the second an interface to connect the system with diverse platforms. The system has been tested in two physical platforms, with different degrees of freedom: Triskarino [5] and Keepon[1].

The simple actions implemented in the system were: *do nothing* waits for a time t before it communicates a finish message; *move body* moves the platform from its current position a to a desired position b; *oscillate body* generates an oscillation in the whole platform by an angle θ; *move shoulder* moves the shoulders to a desired angle θ; *oscillate shoulder* oscillates the shoulders by a given angle θ; *move torso* moves the torso to a desired angle in *yaw*, *pitch* and *roll*; and *oscillate torso* oscillates the shoulders by a given angle θ. The drivers for these actions were implemented for each platform. But, they were grouped by an interface, which hides the existence of them to the rest of the system.

3. CONCLUSIONS AND FURTHER WORK

An Emotional Enrichment System has been designed and implemented to enrich robots' movements with emotions. This has been tested in two different platforms, with different degrees of freedom, in which emotional actions were conveyed using the same enrichment system but different drivers.

Further work is envisioned to add additional parameters and new actions into the system. These new actions will correspond to other modalities, such as facial expression and voice/sound.

4. REFERENCES

[1] Reeves, B., Nass, C.: The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places. Cambridge University Press, New York, NY, USA (1996)

[2] Paiva, A., Leite, I., Ribeiro", T.: "emotion modeling for social robots"

[3] Li, J., Chignell, M.H.: Communication of emotion in social robots through simple head and arm movements. I. J. Social Robotics **3**(2) (2011) 125–142

[4] Brown, L., Howard, A.M.: Gestural behavioral implementation on a humanoid robotic platform for effective social interaction. In: Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on Robot and Human Interactive Communication. (Aug 2014) 471–476

[5] Angel F., J.M., Bonarini, A.: Towards an autonomous theatrical robot. In: ACII'13. (2013) 689–694

---

[1]The code is available in the repository https://github.com/julianangel/EmotionBot/tree/master/ROS/theatrebot_action_modulation