Development of Robot-enhanced Therapy for Children with Autism Spectrum Disorders

Project No. 611391

DREAM
Development of Robot-enhanced Therapy for Children with Autism Spectrum Disorders

Grant Agreement Type: Collaborative Project
Grant Agreement Number: 611391

D5.2 Behaviour assessment model

Due date: **31/10/2016**
Submission Date: **31/01/2017**

Start date of project: **01/04/2014** Duration: **54 months**

Organisation name of lead contractor for this deliverable: **University of Skövde**

Responsible Person: **Serge Thill** Revision: **1**

Project co-funded by the European Commission within the Seventh Framework Programme		
Dissemination Level		
PU	Public	**PU**
PP	Restricted to other programme participants (including the Commission Service)	
RE	Restricted to a group specified by the consortium (including the Commission Service)	
CO	Confidential, only for members of the consortium (including the Commission Service)	

Contents

Executive Summary

This deliverable primarily documents efforts pertaining to task 5.2. It also covers changes to the overall approach in response to the last review meeting. This concerns in particular a very explicit decoupling of the behaviour assessment necessary for the proper functioning of the overall DREAM system – documented here – and longer term diagnostic methods as well methods to improve the current behaviour assessment – the subject of D5.3.

In specific terms, the present deliverable therefore documents

- Behaviour assessment driven by the needs of the intervention scripts. In other words, the system documented here specifically identifies behaviours that the intervention scripts expects and also evaluates the performance in terms of what the intervention scripts expect. Typically, the task is therefore to identify whether a given behaviour was carried out within a given timeframe.

- Engagement assessment as defined by the therapists. This effectively involves continuous monitoring for engagement indicators and computing the overall engagement based on those.

Together, the systems thus implemented allow the functioning of the overall DREAM system and are suitable for further improvements as part of T5.3, and for eventual use in diagnostic measures.

Principal Contributors

The main authors of this deliverable are as follows (in alphabetical order).

Haibin Cai, Portsmouth University
Cristina Costescu, Babeş-Bolyai University
Yingfeng Fang, Portsmouth University
Honghai Liu, Portsmouth University
Serge Thill, University of Skövde

Revision History

Version 3.0 (S.T. 20-01-2017)
Final version.

Version 2.0 (S.T., C.C. 14-01-2017)
Updated to include modifications agreed upon during the integration meeting Jan 11-13.

Version 1.0 (S.T. 18-11-2016)
First draft.

1 Introduction

1.1 Revised WP5 strategy

As a consequence of the Y2 review, a recommendation was made to revise the strategy for WP5 activities in light of a delayed delivery of some of the sensory analysis components and the resulting reduced demonstrations of functionality documented in deliverable D5.1. Following the Y2 review, this revised strategy was agreed upon at a developer's meeting between al code-producing partners of the project and essentially consists in clearly delineating two aspects of WP5 work:

- the development of behaviour assessment tools that are required for a functional DREAM system. This requires in particular the ability to detect child behaviours expected by the intervention scripts, and the ability to determine the performance level of the child when this is expected by the script. It also requires a first implementation of engagement assessment.
- the development of methods to improve these tools, including ensuring their suitability for secondary uses, such as the exploration of diagnostic tools built on them.

This document therefore focusses on the first part – the delivery of behaviour assessment functionality that fulfils the specific needs set forth by the intervention script and that has been integrated into the runtime system . Consequently, the submission of the present document had to be postponed until after the integration meeting (Jan 11 – Jan 13 2017) as the validation of this integration could only take place then.

1.2 Structure and purpose of this document

The following sections present the implemented systems for both performance and engagement evaluation. While the deliverable is therefore somewhat succinct, it is accompanied by the source code for the relevant DREAM system modules available in the repository, originally submitted for integration on November 18, 2016, and developed by both PORT and HIS.

2 Assessment of behaviour and performance

This section describes the functioning of the `assessChildPerformance` component as currently implemented in the DREAM system.

2.1 Expected behaviours according to the intervention scripts

The intervention scripts as specified after the Y2 review, in summary, define the following set of behaviours of interest (see the notes from the post-Y2 review developer's meeting):

0. The child displays no movement
1. The child performs a good sandtray move
2. The child touches a sandtray image
3. The child does not touch the sandtray
4. The child child touches the robot-owned sandtray image

5. The child looks left
6. The child looks right
7. The child points left
8. The child points right
9. The child speaks
10. The child waves with the hand
11. The child covers eyes with hands
12. The child places the hands on the head
13. The child illustrates a flying plane
14. The child illustrates a driving care
15. The child mimics drinking/smelling
16. Complex trajectory 1
17. Complex trajectory 2
18. Complex trajectory 3
19. Complex trajectory 4
20. The child imitates a knock on door

It can be observed that there are three different classes of events: some are related to actions on the sand tray (1 – 4), some can be determined directly from either the detection of a body pose or voice (5 – 9) and some require the analysis of changes in body pose over time (10 – 20, where "complex trajectory" merely refers to movements used in imitation tasks that do not have a more descriptive label).

2.2 Overall functionality of the component

The component is triggered by incoming information on the `GetInterventionStatus:i` port (typically sent from WP6, which maintains an overview of the intervention scripts). This defines amongst others if a behaviour is currently expected (and if so, which one of the list above), and within what time window this behaviour is expected to take place (including the possibility of no specific time window). Depending on the type of expected event, the component then reads information from the relevant sensory analysis ports for the specified duration and determines performance as detailed in the remainder of this deliverable.

The component typically returns – as defined and expected by the intervention scripts – a binary performance value: 0 indicates evidence for a bad performance was noted while 1 indicates that evidence for a good performance was found. The component does however also have the capacity to return intermediate values that may, for example, become relevant for behaviour where the analysis may carry a certain amount of uncertainty (such as 10 – 20), and indicate a confidence value that the expected behaviour was observed.

The component also has support for a return value of -1. This is an indication that, for some reason including possible bugs, the component was unable to make a performance assessment and, as such, is distinct from a confirmation of bad performance.

In all cases, the performance is then returned on the `GetChildPerformance:o` port and the value then determines further progress in the intervention script.

2.3 Implementation of sandtray events assessment

The performance of sandtray events is comparatively simple to classify since the sandtray reports the interactions the child has with it on the `getFromInteractionEvent:i` port. For these events, the present component is therefore reduced to a a simple message relaying service which monitors the aforementioned port for both behaviours that are in line with the expected behaviour (thus indicating good performance), and behaviours that are not (thus indicating bad performance). No sandtray behaviour at all is also evaluated as bad performance, unless that was in fact the expected behaviour.

2.4 Implementation of direct sensory events assessment

2.4.1 Pointing events

Pointing events are evaluated by tracking both wrists (whose coordinates are available on the `GetBodyPose:i` port) and looking for a movement to either the left or the right side consistent with the requested pointing. The resulting performance classification is binary.

2.4.2 No movement

Event 0, no movement, pertains to turn-taking in particular, in which the child is expected not to move over the Sandtray. Performance is therefore evaluated by tracking the skeleton for the duration of the time window and ensuring that no points move over the sandtray. This is a departure from the earlier interpretation, which was merely that the child should sit still. The component also contains the functionality to make this assessment, but at present, this has been disabled following the integration meeting where it was decided to focus on movements with respect to the sandtray instead. The resulting performance classification is binary.

2.4.3 Gaze events

Gaze events are evaluated by obtaining the gaze vector for the given window (available on the `GetEyeGaze:i` port) and verifying that a gaze in the correct direction was observed. The resulting performance classification is binary.

2.4.4 Voice event

The ID – an integer – of the owner of a detected voice event is available on the `IdentifyVoice:i` port. Since the system recognises a number of children but only one adult (the therapist), and given that only one child is expected in the therapy room at any point in time, the present component verifies 1) that a voice was detected, and 2) that it did *not* belong to the therapist. This is in part because it allows the system to accommodate any number of children without requiring changes in configuration (such as a list of known children) and in part because it is not trivial to guarantee on the sensory analysis side that a child voice is always correctly identified. The resulting performance classification is binary.

2.5 Implementation of complex sensory events assessment

2.5.1 Current implementation

The recognition of events 10 – 20 are developed and implemented as part of the *SensoryAnalaysis* component. For the present purposes, the `IdentifyTrajectory:i` port continuously receives a vector of length 12 (the 12th element being event 0 (no motion)), which is currently implemented as a one-hot vector identifying the current most likely match for the observed trajectory (if any).

It was further decided by UBB at the integration meeting in Portsmouth that it is sufficient for good performance that the child performs the expected behaviour at some point during the expected time window while performing other behaviours during the same time window is not considered a negative.

As such, performance assessment requires the monitoring of the `IdentifyTrajectory:i` port when a specific behaviour is expected, with a performance of 1 being reported if the expected trajectory is observed at some point during the time window; otherwise a performance of 0 is reported.

2.5.2 Alternative implementation

Alternative algorithms are also implemented. In particular, PORT has also provided experimental support vector machines that output the confidence that one of the complex trajectories is present. Specifically, one SVM is available per behaviour, and confidence values can be obtained for all of them. The present component contains functionality to interpret such return values as well: a good performance requires the confidence value to be highest for the behaviour in question, and the returned performance is then this confidence value.

While the functionality is present in the component, it is currently not actively used due to the computational costs of running the SVMs, the fact that the quality of their output depends on good training data, and the fact that the nuanced output which this approach gives on top of the currently active one is not needed given the actual needs of the intervention scripts. Nonetheless, future versions may re-activate this part should this become necessary.

3 Assessment of engagement

This section describes the functioning of the `assessChildEngagement` component as currently implemented in the DREAM system. Specifically, it was stipulated by UBB that a pragmatic and useful approach to quantifying engagement consists of tracking three predictors:

1. the position of the child with respect to the robot, in particular positioning in front of the robot
2. the facial expression of the child, in particular the presence of a smile
3. the presence of mutual gaze

It was further stipulated that based on these indicators, engagement is then quantified as a numerical value which counts the number of predictors currently present. For example, the engagement of a smiling child in front of the robot but not engaged in mutual gaze would be scored 2, whereas a child only demonstrating mutual gaze, but no smile or adequate position would be scored 1.

To assess the performance, the implemented system therefore tracks these three predictors. The information is available as follows on the following ports that receive their information from the corresponding ports in the `SensoryAnalysis` component:

- `GetBody:o` - this port returns the coordinates of the center of the body of the child. Since the location of the robot is known, the present component checks whether the child's position on the axis pointing from the robot towards the room is in line with the requirement that the child is in front of the robot.
- `IdentifyFaceExpression:o` - different facial expressions are indexed by integers and for the present purpose, the presence of the identifier for a smile is tracked.
- `CheckMutualGaze:o` - this port simply indicates the presence of mutual case with a binary value, thus removing the need for any further processing.

`assessChildEngagement` continuously monitors the information arriving at these ports, verifies which of the necessary predictors are present, computes the resulting engagement level and broadcasts it on the `GetChildEngagement:o` port.

In sum, the approach for assessing engagement is therefore simple and robust instantiation of the therapist requirements stated at the outset.

4 Conclusions and future work

Overall, the methods for behaviour assessment described here fulfil the revised strategy, have been successfully integrated into the runtime DREAM system, and provide the functionalities required for the run-time system. Future work will now focus on improvements to these methods whenever needed, and on the exploitation of these methods for use in diagnosis automation.