PERSONAL TEACHING AID

By Sara Aerssens, Caitlin Lagrand, Putri van der Linden, Lina Murady, Tirza Soute and Leila Talha

Personal Teaching Aid is a program that helps young students practice basic arithmetic problems. The software makes use of facial recognition and speech recognition, and the students' skills are constantly measured and updated in order to generate problem sets with the appropriate level of difficulty for each student.

INTRODUCTION

The Netherlands is currently battling with a shortage of primary school teachers, which has negative consequences for the achievements and development of students. Studies have shown that large classrooms can negatively affect the academic performance of students due to lack of personalized aid [Brühwiler and Blatchford, 2011]. A solution to this problem is the use of virtual learning environments. Other studies have shown that virtual learning environments can improve the performance of students [Shih-Wei Chou, 2005]. This leads to the goal of this project: to develop a program with human-like properties, so that it works as a basic simulation of a teacher (assistant). It recognizes students, interacts with the student (e.g. greeting) and guides it through questions. In the following sections, the implementation of the most important components of the program will be explained.

FACIAL RECOGNITION

When the program starts, ten pictures of the student are taken and they are marked as recognized if they are classified with an accuracy of at least 65%. If not, the student is classified as new and they are added to the database. In order to classify the photos, the OpenFace library is used. It uses a technique named Histogram of Oriented Gradients (HOG), which converts each image to its HOG image and uses training images to identify facial features. The face landmark estimation technique is used as well, to find the exact position of the eyes and mouth. This is used to align detected faces using affine transformations, in order to form a 128-dimensional representation of the face. These representations are used to classify faces in an image, which is done using a linear support vector machine.

PROBLEM GENERATOR

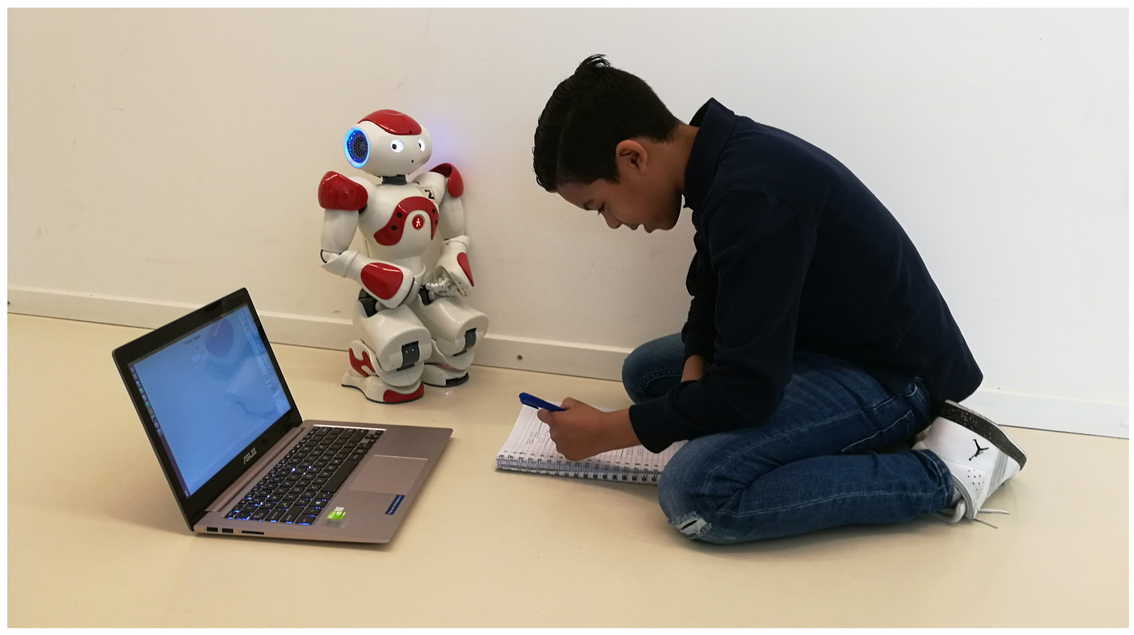
When a new student starts the program, the problem generator starts with very simple arithmetic problems. Their progress is saved and their skills are measured. According to their skill the difficulty of the problems is adjusted. If a student has trouble with multiplication, more multiplication problems are presented and the level of difficulty remains low until further improvement. The frequency of each operator is calculated by the student's ratio of correctly answered problems so far. The level of difficulty of a problem is influenced by the range of numbers, the amount of different operators, the amount of components and combinations of different operators. It all depends on the student's affinity with each operator.

SPEECH RECOGNITION

The student can respond to the given questions by saying the answer out loud. Analyzing the detected speech fragment is done with Google's Cloud Speech API. Using an appropriate threshold to ignore background noise, the tool returns the most probable sentence. The fragment is scanned for numerical values and it is checked whether these match the actual answer to the problem. The API does not give consistent output of numerical values, which is why we extended it with a number-text dictionary.

NAO ROBOT

The NAO robot is used to make the interaction between the student and program seem more natural and to motivate the student. The robot's camera is used for facial recognition and its text-to-speech is used. At the start of the program the student is greeted, and the problems are read out loud. As some operators were being mispronounced, these had to be corrected manually. It also tells the user whether an answer was correct, or not. All the sentences are predetermined, except the answers returned by the problem generator.



DISCUSSION

The facial recognition component currently just adds new users to the database, but does not train on new data yet. This could be improved by starting a new thread and train immediately, so that the new student can be recognized immediately next time. As for the speech recognition software that was used for the program, it would be better to use a more flexible alternative that can be trained to fit the need of this program. Ideally, the final goal of the program is for it to be able to communicate more naturally. It should be able to encourage the student when necessary and without sounding repetitive. The problem generator could be expanded as well by generating hints for each problem, as producing a hint generator was not feasible in the scope of this project.

REFERENCES

Christian Brühwiler and Peter Blatchford. Effects of class size and adaptive teaching competency on classroom processes and academic outcome. *Learning and Instruction*, 21(1):95–108, 2011.

Chien-Hung Liu and Shih-Wei Chou. Learning effectiveness in a web-based virtual learning environment: a learner control perspective. *Journal of Computer Assisted Learning*, 2005.