Data collection and processing for a multimodal Learning Analytic System

Emanuele Ruffaldi, Giacomo Dabisias, Lorenzo Landolfi (SSSA)
Daniel Spikol (MAH)
Practice-based Experimental Learning Analytics Research and Support

A EUROPEAN PROJECT LOOKING AT HOW TEACHER, LEARNERS AND TECHNOLOGIES CAN SUPPORT ONE ANOTHER IN HANDS-ON LEARNING OF SCIENCE, TECHNOLOGY, ENGINEERING AND MATH.
PELARS

LEARNING ACCREDITATION
Evaluation mechanisms via learning analytics

CLASSROOM TOOLKITS
Practice-based learning and feedback

TEACHING & CURRICULUM
Learning support via visualisation and learning analytics

EDUCATION POLICY
Improved STEM practice-based learning via understanding learners' knowledge
Introduction

The system is designed to:

- Acquire raw data from a set of sensors.
- Send data to a remote server.
- Process remotely data to produce learning traces.
- Produce visualizations for the different stakeholders (Teachers, Students, Researchers)
Overview

Collector Core

- Object Detection
- Face Tracker
- Marker Tracker

Local Storage

Server

- Events
- Arduino IDE
- Button

Kinect One

WebCam
Overview

• One central server
• Multiple clients
  – Single computing machine
  – Multiple sensors
  – Mobile app

Scalable architecture
Client

• Standalone C++ executable running under Linux.
• Distributed as opensource project on github.
• Runs inside a Docker virtual machine.
• Works online and offline.
Client

DEIGNED ENVIRONMENT
furniture, architecture and labware
w/ embedded sensors

REAL-TIME FEEDBACK
for learning support

SOFTWARE & HARDWARE KIT
w/ integrated analytics

LEARNING ANALYTICS
and reporting
Client

C920

Kinect2

Arduino kit
Client

Low level data acquisition:

- **Hand tracking** using Aruco markers.
- **Face** detection using openCV gpu detector.
- **Audio** power level recording.
- **Arduino ide** log.
- **Sentiment** **button interaction**.
- **Particle.io** events.
- **Video** recording from **kinect2** and **webcam**.
- **Object recognition**.
Server

- Coded mainly in **Java** and **javascript**.
- Two separate collector endpoints:
  - **Websocket** for data streams
  - **Servlets** for single requests
- **Mysql** database for persistence.
- **Hibernate** object mapper.
Server

Data collection

DB

Object Mapper

Websocket Collector

REST Collector

Collector

REST Jobs

REST Data Access

Web Interface

Visualizations
Server web interface

The server supports

- **REST operations** on the db data
  PUT/GET/DELETE based on User Access Control
- **Batch jobs**
  - Single valued and Data streams
- **Computation of learning analytics**
- **Produces dynamic visualizations**
Jobs

USER

PUT job

WEB INTERFACE

SCHEDULER

Thread

Thread

Thread

Result

GET ID status

JOB executing

GET ID status

JOB terminated

GET ID result
Mobile app

Students
- Images
- Text
- Videos

Mobile App

Researchers
- Work phases
- Rating
- Work description
Visualization

Support for different visualizations based on stakeholders

• Data timeline
• 3D data visualization
• Storyboard
• Piechart
• …
Visualization

- Timeline
- Piechart
- 3D viewer
- Storyboard
Results

- 33 trials
- 83 students
- 6 different locations
- 36 hours of recording
- 58 min average session length
- 15GB of collected data
Technical Challenges

• Distribution and installation
• Integrate all data sources
• Offline vs Online system state
• Quick bug fixes
• Computer science experts vs learning experts
• Data timing and integrity
Research Challenges

- Extract “Analytics” from unstructured sessions
- Identify patterns of behavior
- Identify groupwork
Future work

- Extract new learning analytics after trials
- Create and evaluate visualizations
- Debug system
Thank you!

Questions?