Journal of Universal Computer Science, vol. 24, no. 2 (2018), 149-170
submitted: 1/10/17, accepted: 6/11/17, appeared: 28/2/18 © J.UCS

Development of a Software that Supports Multimodal Learning Analytics: A Case Study on Oral Presentations

Roberto Munoz
(Universidad de Valparaíso, Valparaíso, Chile
roberto.munoz@uv.cl)

Rodolfo Villarroel
(Pontificia Universidad Católica de Valparaíso, Valparaíso, Chile
rodolfo.villarroel@pucv.cl)

Thiago S. Barcelos
(Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, São Paulo, Brazil
tsbarcelos@ifsp.edu.br)

Alexandra Souza
(Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, São Paulo, Brazil
alexandra.souza@ifsp.edu.br)

Erick Merino
(Universidad de Valparaíso, Valparaíso, Chile
erick.merinom@alumnos.uv.cl)

Rodolfo Guiñez
(Universidad de Valparaíso, Valparaíso, Chile
rodolfo.guineze@alumnos.uv.cl)

Leandro A. Silva
(Universidade Presbiteriana Mackenzie, São Paulo, Brazil
leandroaugusto.silva@mackenzie.br)

Abstract: Learning Analytics is the intelligent use of data generated from students with the objective of understanding and improving the teaching and learning process. Currently, there is a lack of tools to measure the development of complex skills in real classroom environments that are flexible enough to add and process data from different sensors and oriented towards a massive public. Based on this finding, we developed a free software system that permits to capture and to visualize a set of 10 body postures using the Microsoft Kinect sensor, along with the ability to track custom body postures and data from other sensors. The developed tool was validated by means of precision and usability tests. Furthermore, with the goal of demonstrating the potential of incorporating this type of software into the classroom, the software was used as a tool to give feedback to the teacher and to the students at the moment of giving and evaluating oral presentations. Also, a clustering analysis of data gathered from 45 student presentations indicate that presentations on similar topics with also similar complexity levels can be successfully discriminated.
Keywords: Multimodal Learning Analytics; Microsoft Kinect; Oral Presentations; Self-Organizing Maps
Categories: D.2.2, L.0.0, L.1.1, L.3.0, L.3.6

1 Introduction

The development and evaluation of professional skills (also known as soft skills), such as group work, efficient collaboration, and efficient communication are essential for good job performance [Lucas, Hanson and Claxton 2014]. The development of these skills has been standardized to priority level in Accredited Agencies of Engineering Education, such as in ABET (Accreditation Board for Engineering and Technology) and in INQAAHE (International Network for Quality Assurance Agencies in Higher Education). However, the development and evaluation of these skills is a challenge for higher education institutions. The reason for this is due, primarily, to two aspects. The first is associated with the educator who, at least in the first few years of teaching, is generally charged with numerous courses and heterogeneous students in terms of soft skills, which may reduce the time available to proper evaluation of skill improvement. The second is associated with the fact that the evaluation of these skills must be performed in the same learning environment where the skills are developed, and not through standardized evaluations, such as written exams or multiple-choice exams [Shuman, Besterfield-Sacre and McGourty 2005].

In addition to the mentioned challenges, feedback is a crucial aspect in the educational process as it can support the students' academic performance and promote their motivation and autoregulation skills; this may be a strategy to reduce the gap between the current and expected performance. However, properly obtaining and analyzing data for student feedback is a time-consuming task for educators. Hence, feedback is not a very frequent practice.

Given the aforementioned challenges, Learning Analytics (LA) arises as a methodology to measure, collect, analysis, and present data regarding students—the contexts and interactions that generate around them—with the goal of comprehending the learning process that is developing as well as optimizing the environments in which it is developed [van Harmelen and Workman 2012]. One modality of LA is based on the analysis of non-verbal language, where the evidence of learning is obtained through gestures, body postures, sound, etc. This modality is known as Multimodal Learning Analytics (MLA), which aims to maintain the richness and the highly contextual nature of traditional qualitative analysis mediated by people, but with the added benefits of quantifying this data in new forms, along with utilizing sensors to capture data that is not easily perceptible through human vision [Worsley et al. 2016].

Furthermore, the technological advances provided by Big Data and Machine Learning techniques in recent times have been of great utility for MLA [Worsley 2012]. In spite of the advantages provided by these advances, information provided by MLA techniques are still useful only when analyzed by experienced researchers without being transferred to the potential end user, that is, the educator.

Based on the above, this paper will present the design and development of a tool that permits the visualization and classification of body postures using a Microsoft Kinect sensor. Its architecture is designed in order to allow the visualization and classification of new types of multimodal data with relative ease. The proposed tool [1] (Lelikëlen – eyes open in Mapuche Language) currently allows the detection, storage

[1] https://github.com/leikelen-team/Leikelen

and display of body postures from recorded subjects, giving options of adding the detection of personalized postures, exporting and importing scenes, along with the display of identified postures in a time line. In order to validate the tool and demonstrate its usefulness we also present a case study, in which Lelikëlen is used to gather data for an evaluation of presentation skills of undergraduate students using learning analytics techniques.

A case study was conducted with 10 student groups that were required to present their class projects associated with 3 topics. These presentations were made in the same environment at the end of each unit. In each presentation, the body postures made by students were captured. Subsequently, data analysis was conducted using two different approaches. The first one attempted to identify the frequency of employed gestures at the time of presenting. The objective of this approach is to deliver feedback to the student in a formal classroom setting. The second one employs Self-Organizing Maps (SOM) type in order to identify patterns associated to the captured data. This is to say, if the positions and the measures captured were generating information that, in the future, could generate strategic knowledge about the students and the learning process.

The article is structured in the following manner. In the following section, related works are presented. Subsequently, the design of the solution and its implementation are shown. To continue, the technical validation of the developed tool is discussed by means of a real case study utilizing Lelikëlen. Finally, we present our conclusions and future works.

2 Related Work

Speaking and presenting in public are critical skills for academic and professional development. Based on this premise, several initiatives have emerged in which MLA techniques are applied to provide feedback to the presenter and facilitate his/her self-reflection and performance improvement.

The application of MLA techniques in this context can be justified, for instance, by [Tanveer, Zhao, Chen, Tiet and Hoque 2016], who indicates that one of the fundamental aspects for communicating ideas during a speech is the non-verbal language. The incorrect usage of this language modality or even the incongruence with the verbal language may directly affect the quality of the presentation. Bad idiom usage and bad synchronization between speech and gestures may distract the spectators. Hence, the authors present the development of a software tool using the Microsoft Kinect sensor aimed at informing the presentator about repetitive gestures. The tool was evaluated with 27 participants and the results indicate that they were able to correct their body postures during the presentations.

In [Nguyen, Chen and Rauterberg 2015], the authors argue that there are few software systems that implement mechanisms based on empirical investigation of non-verbal behaviors to deliver rich feedback information to presenters. Based on this premise, the authors use data from the Kinect sensor to train a Support Vector Machine to indicate the improvement of non-verbal behaviors on presentators. Validation was performed using data from 76 presentations and the recognition rate was 73.9%.

The proposal by [Schneider, Börner, van Rosmalen and Specht 2017] consider that most of the previously developed software tools are only able to provide an analysis after the oral presentation was performed. For this reason, a Presentation Trainer (PT) that provides real time feedback to students was developed. In this work the authors analyze, evaluate and extend the features of the software system so that students can improve their self-regulation skills on how they perform speeches. A preliminary evaluation was conducted with 12 students who did presentations about their projects.

In [Echeverría, Avendaño, Chiluiza, Vásquez and Ochoa 2014], a project is presented in which the individual oral presentations of 448 students are evaluated. Data from the presentations were captured using a Microsoft Kinect. The analysis criteria were based on visual contact and body postures, from which an effectiveness of 68% and 63% were obtained, respectively. The objective of this research was to identify non-verbal characteristics in students and their influences in order to provide a positive prediction on the development of oral presentation skills. Finally, the authors indicate that the results could be improved in order to later develop a tool whose objective would be to measure the performance of students, giving them rapid and concise feedback.

A relevant result is presented by [Leong, Chen, Feng, Lee and Mulholland 2015]. The authors provide an overview of related works on multimodal research tasks and educational activities. Furthermore, they conduct their work using depth sensors (using a Microsoft Kinect) to generate measurements of communication skills, in particular looking at public presentations. To finish, the authors also introduce an open-source package developed in Python that permits measuring body language. However, this package requires advanced knowledge in the use of programming languages.

Lastly, in [Chang, Nian, Chen, Chi and Tao 2014], the authors use the 3D sensor of the Microsoft Kinect with the objective of recognizing body postures to facilitate human-robot interaction. In order to facilitate recognition, the authors utilized postures based on the semaphore alphabet (letters and numbers signaled using flag movements). To develop this, they used the SDK (software development kit) of the Microsoft Kinect with a neural network, the Self-Organizing Map [Kohonen 1998] trained with a non-supervised algorithm. The proposed system, after recognizing a body posture and contrasting it with a crawl of information on skeletons, sends a command for a robot; however, the obtained results are still preliminary.

By comparing the aims and strategies described in the mentioned articles it is possible to identify some tendencies. Firstly, in all articles the Kinect sensor is used to obtain body postures; this is probably due to the wide availability of this sensor, and also its low cost. Moreover, two main limitations can be identified. The first of them is that, although all works are aimed at identifying and analyzing body postures, the proposed software systems do not allow extensions to incorporate the cross-analysis of other input modalities, in the spirit of MLA. The second limitation is that the proposed systems are not properly designed for end-user utilization. Also, aiming at both end-users and researchers, we consider that an MLA system based on body postures should be flexible enough to allow the identification of new patterns as desired. In the [Section 3] we describe the architecture and implementation of a visualization system aimed at overcoming the identified limitations.

3 Design and Implementation of an Extensible Body-Postures Visualizer

In this section, the architecture and interfaces of the developed system are presented together with the technologies utilized. As can be seen in Figure 1, the software architecture is designed for the incorporation of different types of sensors and devices.

Analyzing the related works, we demonstrate that a potential problem in the current solutions is that they are highly coupled to some specific sensor technology. Under this perspective, one of the design goals of Lelikëlen's architecture is a high scalability degree in terms of functionalities and integration of new devices for multimodal data capture. This decision is also related to the increased availability of low-cost devices that enrich data capture, which opens up a conducive environment for multimodal learning analytics.

In particular, for this research, we used a MS Kinect as a capture device, which is able to recognize up to six people simultaneously. However, in the current version of Lelikëlen, in order to present the scalability of the software, we also incorporated the possibility of natively using the EEG OpenBCI; furthermore, the expectation is to incorporate other types of low cost sensors, such as measured heart rate and directional microphones.

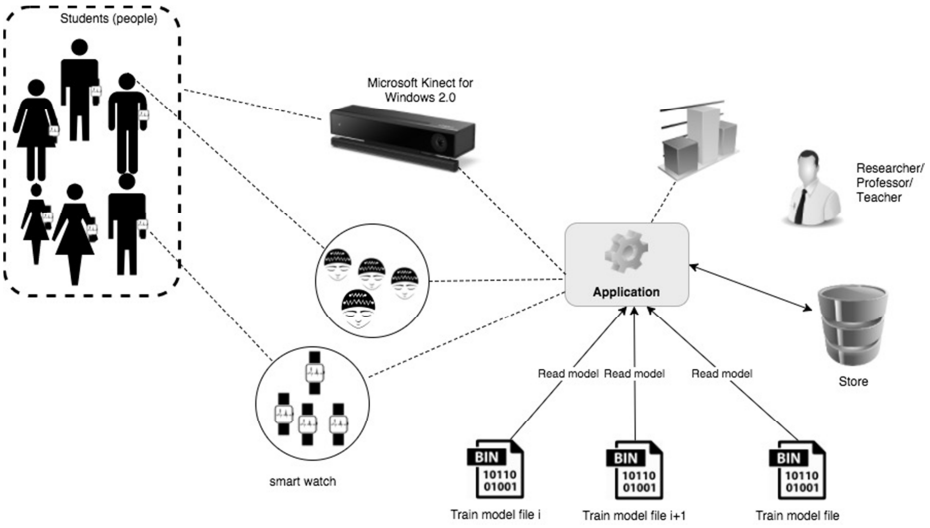

Figure 1: High-level functionality scheme

In the case of the MS Kinect, the sensor sends the frames captured through events to the application, which then, with its internal components, can detect the postures made by users in each second, as well as saving the obtained information in a storage layer represented by the *Store* component. This layer allows the incorporation of different training models. Furthermore, we also incorporated a set of body posture classifiers that are generally present at the time of oral presentations. The validation

of these classifiers is presented in [Section 3.4]. On the other hand, a validation was also performed using the tool, in a real environment, which will be presented in [Section 4].

Below, the main features of the developed application are presented.

3.1 Detailed Design at the interaction of the extensible Visualizer components

In order to ensure a scalable application, and thus to offer a tool that is not technology-dependent, a highly modular architecture was defined. This architecture is presented in Figure 2.

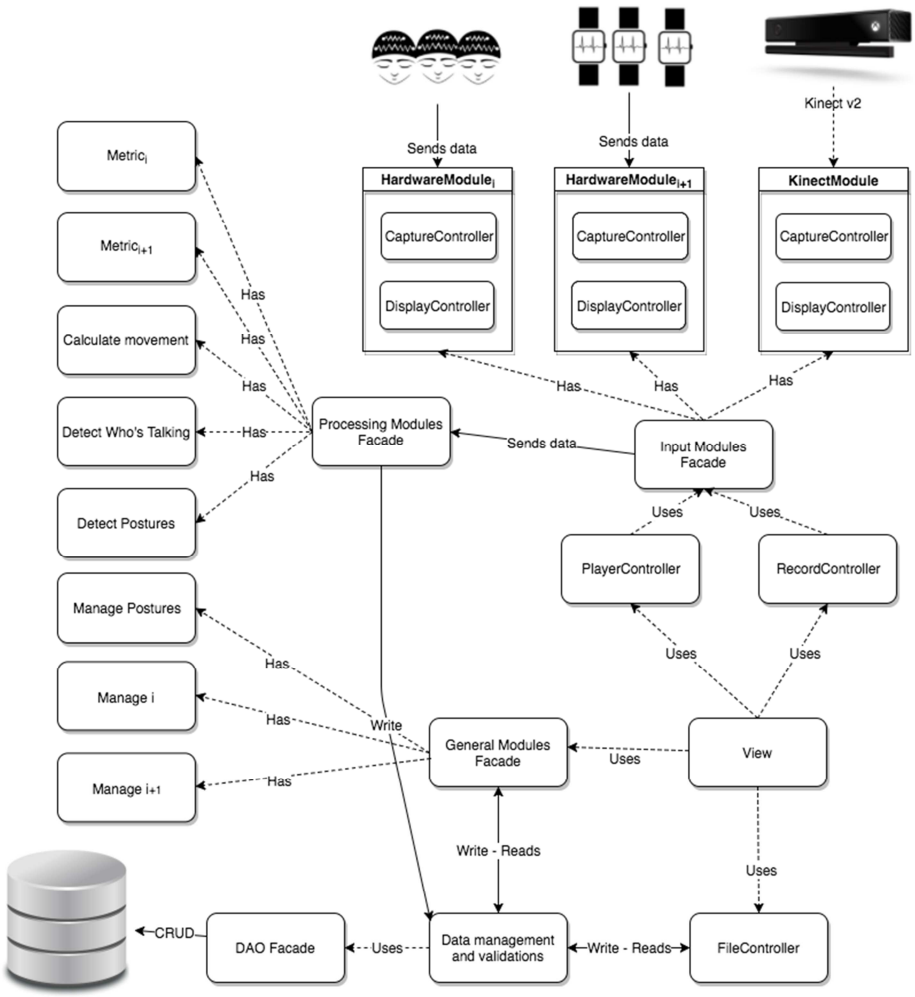

Figure 2: High-level functionality scheme

Upon starting the recording, the *RecordController* component commands the different input modules (via *Input Modules Façade*) to capture scene files (for later playback) within the corresponding folder. Furthermore, the component also sends the generated data to the processing modules that are compatible with that type of data. These modules are accessed through the *Processing Modules Façade*. They inherit from a common class and implement an interface for each compatible input module.

Each processing module completes some specific data processing and stores the results in main memory through data management and validations. Later, when recording is stopped, the processed results are stored in the database through a DAO (Data Access Object) component. We did this in order to optimize execution, considering the large amount of data generated in real time.

After the recording is stopped, the scene is stored in a common database and becomes available for playback mode. When playback is started through *PlayerController*, the scene data associated with each input module is obtained through *InputModuleFacade*; data is then sent to the view. The view supports a video and image player which is technology-independent. The player provides two overlayed layers to which each input module can add specific tabs to display their own data types. This strategy is known as view injection.

The view can also invoke the *FileController* to export and import external files. In this situation, the *FileController* communicates with the management and validation modules for obtaining data from the database or inserting new data to it.

Alternatively, a third type of module was defined as general modules. That kind of module can be active at any given time of execution, regardless of the current state of the application.

It is important to emphasize that input modules do not insert data into the database, and only read data concerning configuration directories to maintain a highly consistent structure in terms of roles and responsibilities. This means that the input modules do not interfere with the data model or the rest of the application, they only capture and provide data. Processing modules write model data (intervals, charts, and fixes), but do not delete or modify data. This was done to avoid data corruption because, as mentioned above, this data will be stored in main memory while it is being captured and will only be stored in the chosen database after the recording is complete. Furthermore, the general modules can do the four basic operations to the database (create, read, update, and delete). These modules serve to extend the application, and do not depend on the state of the application, but they can always be executed. They serve for tasks such as training classification models and other ancillary work. This way of accessing the data ensures that each class of the data validation and management module implements a certain interface, which are accessible to different modules in a differentiated way. To finish, it is important to emphasize that a public API has been generated and the application has been made available in a Git repository, so that it can be used and modified by the community in general.

3.2 Design of user interfaces

The design of the user interfaces is of great relevance for comprehension and approval of the application by its users [Snyder 2003]. This is due to the fact that the interface

design directly affects the interaction and associated results in manipulation that the user has over the application.

Digital prototypes were generated for the interface design which were then tested by 5 professionals in the educational field; this permitted determining communication breakdowns in use of the application [De Souza 2005]. Once the interfaces were validated, the next step was their final implantation. An example of the process to be passed from wireframe to the final interface is that presented in Figures 3 and 4, respectively.

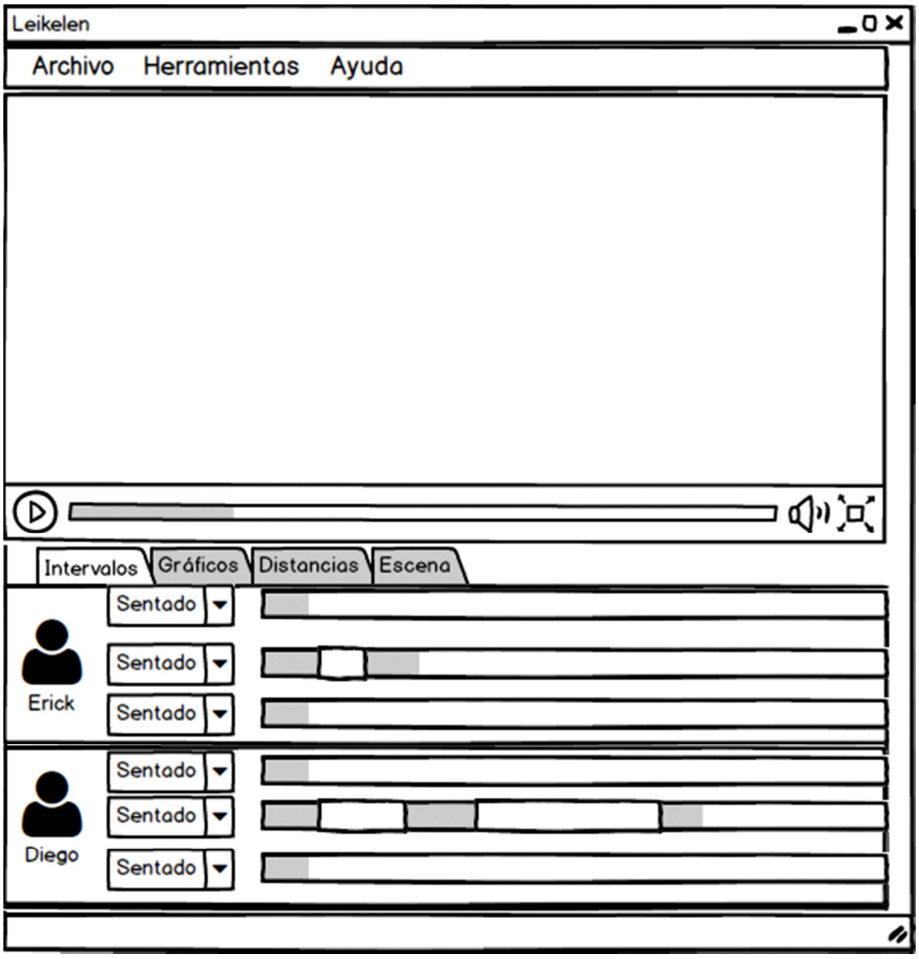

Figure 3: Wireframe of Multimodal Analyzer (Lelikëlen)

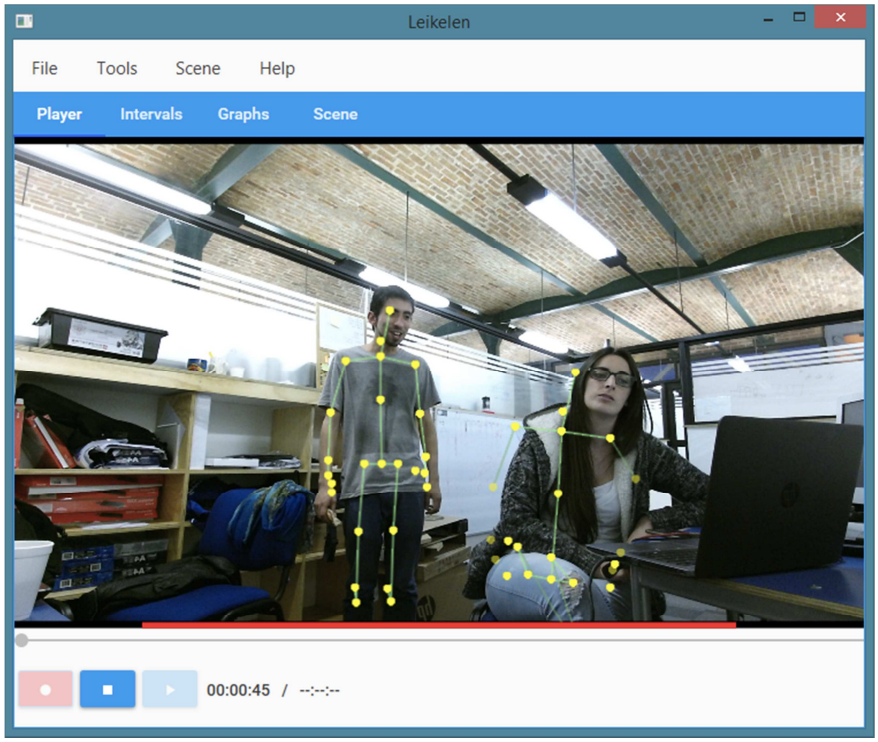

Figure 4: Main Interface of Multimodal Analyzer (Lelikëlen)

It is important to mention that the final interface varies slightly with respect to the prototype presented in Figure 3, since modifications were made after validating with potential users (among them the possibility of exporting to SQL). The functionalities of the developed Multimodal Analyzer are presented below.

3.3 Multimodal Analyzer Functionalities

Lelikëlen was developed with the objective of amplifying the number of potential users, focusing on the development of a usable interface with clearly defined functionalities.

As seen in Figure 4, the interface presents 7 principal functionalities that are listed below.

- Save Scene: Allows the storage of scenes in a centralized database specifically purposed for those scenes.
- Record Scene: Allows the storage of scenes for subsequent storage, reproduction, or reuse of data.
- Import / Export Scene: Allows the exportation of the generated scene and its data to an easily portable external file. This exported scene can be imported for analysis.

- Configuration Sensor: Allows the configuration of the input sensor, with configurations such as the input port, channels to use, filters, etc.
- Select Detection Algorithm: Allows the selection of the algorithm or method of detection and training to use with those implemented in the application.
- Manage People: Create, edit, delete, and list stored people.
- Manage Scene: Allows editing, selecting (for playback), and deleting a scene.
- Live View: Displays signals in real-time, and in the case of the algorithms or detection methods that permit them, to see detection results.
- Play Scene: Play a previously saved scene.
- View Timeline: The system shows a person's emotions in a timeline.

To conclude, as mentioned earlier in the system proposal, the use of EEG as an analysis of electrophysiological signals has also been integrated. Two specific features have been added to this.

- Train Detection (EEG): Train the classifier according to the detection algorithm; however, an already-incorporated model (independent of the person) is found associated with the detection of emotions by default.
- Detect (EEG): The system must detect what is defined in the model, in the case of not wanting to detect emotions of the person being analyzed.

3.4 Set of predefined postures

The software developed includes a set of 10 predefined postures (classifiers). These postures are based, principally, on those identified by [Chen, Leong, Feng, Lee and Somasundaran 2015, Echeverría, Avendaño, Chiluiza, Vásquez and Ochoa 2014]. The generated postures and trainings are: 1) Pointing; 2) Crossed arms; 3) Raised hand (e.g., asking a question); 4) Hand on the head; 5) Lowered hands; 6) Explaining with open hands; 7) Hands on the hip; 8) Hand on the face (e.g., hand on the chin); 9) Explaining with one hand; and 10) Seated. The seated and hand raised postures were added posteriorly, since they are of potential utility for researchers in education, although they are not necessarily common postures seen during presentations.

An example of each posture is presented in Figure 5.

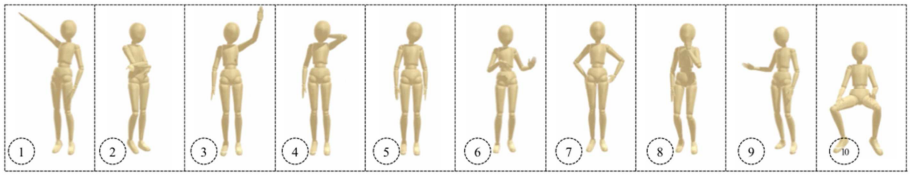

Figure 5: Example postures associated with classifiers

We trained and tagged each classifier using the Visual Gesture Builder. The VGB utilizes characteristics such as relative angles formed by articulations and movement of articulations to learn and generate the classifiers using the algorithm ADABoost [Freund and Schapire 1997]. Each classifier was trained using ADABoost with 1,000 weak classifiers. In 9 of the postures, the articulations below the waist were ignored

so that the tool would not discriminate if a person was seated or standing; articulations below the waist were only used at the moment of training the seated classifier. Table 1 presents the calculated statistics of our classifiers in terms of accuracy and precision [Badiru and Racz 2016]. To calculate, the analysis tool that the VGB provides was utilized.

ID	Posture	#Training frames	#Testing frames	Accuracy	Precision
P1	Pointing	13.155	5.370	82,16%	73,57%
P2	Crossed arms	27.574	5.488	91,63%	86,89%
P3	Raised hand	11.178	4.820	82,76%	82,57%
P4	Hand on the head	17.507	8.191	84,01%	89,19%
P5	Lowered hands	26.468	5.724	80,29%	77,42%
P6	Explaining with open hands	20.586	4.623	90,77%	85,14%
P7	Hands on the hip	16.150	6.396	82,96%	70,20%
P8	Hand on the face	13.847	5.289	72,40%	95,55%
P9	Explaining with one hand	14.210	5.318	77,96%	68,04%
P10	Seated	86.300	4.541	94,71%	90,24%

Table 1: Accuracy and Precision Classifiers Generated

One important aspect to mention is that the accuracy and precision of the generated binary classifiers was calculated with different subjects, those from the training. For example, with the "Crossed arms" classifier, an accuracy of 91.63% and a precision of 86.89% were obtained at the moment of classifying with test frames. On the other hand, after looking at Table 1, it can be evidenced that the classifier with the least accuracy and precision was that associated with "Explaining with one hand", with a 77.96% and 68.04% accuracy and precision, respectively. However, this continues to trend positively, given that the classification does not require a person to perform identification as it would with manual analysis.

3.5 Classification of Postures Evaluation

For the evaluation of classification of postures, a test was designed that contemplated an evaluation of 6 students in a real environment. The subjects were first-year students (males) in Informatics Engineering, that did not belong to the test or the training.

The camera was located in front of the presentation stage, which measured 4.0 meters wide and 4.1 meters long. In Figure 6, a scheme and image of the stage used are presented. One aspect of high importance to note is that the recordings were realized in an environment that was not controlled; this is to say with a backdrop without contrast; contrast facilitates the detection and illumination standard (as seen in Figure 6). This was performed with the objective of evidencing the potential use of these technologies in a real classroom.

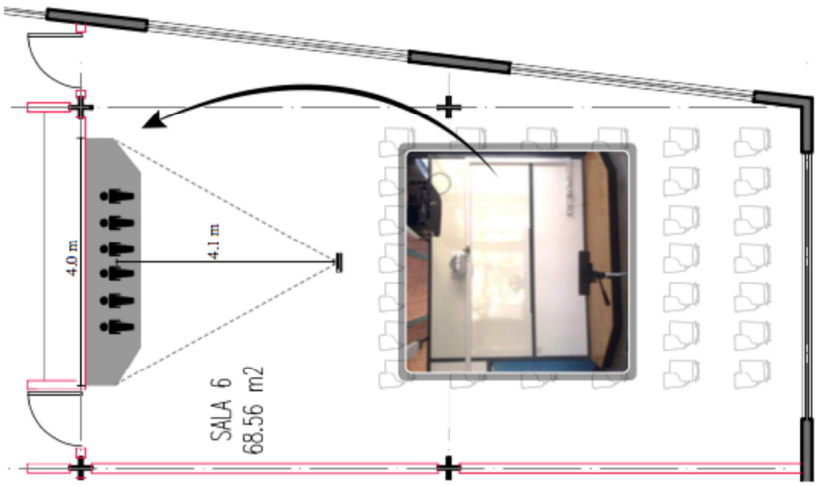

Figure 6: Test stage (real classroom)

Each student had to perform a sequence of 10 base postures on 10 occasions (presented in previous Section). This had to be realized in a maximum interval of 15 minutes; this is to say, each student had to perform 100 postures, automatically classifying a total of 600 postures (60 postures per classifier).

One important aspect to mention is that the subjects that realized the postures did not belong to the training set. Below, in Table 2, is presented the confusion matrix at the moment of automatically classifying the postures performed by the participants.

#	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**P10**
P1	50	0	1	0	0	0	0	2	1	0
P2	0	60	0	0	0	0	0	0	0	0
P3	3	0	47	1	0	0	0	0	5	0
P4	2	0	5	53	0	0	0	0	0	0
P5	0	0	0	0	59	0	0	0	0	0
P6	0	0	0	0	0	59	0	0	1	0
P7	0	0	0	0	4	0	55	0	1	0
P8	0	0	8	0	0	0	0	44	0	0
P9	5	0	0	0	0	0	0	1	49	0
P10	0	0	0	0	0	0	0	0	0	60

Table 2: Confusion Matrix Postures

The confusion matrix evidences that the classifiers incorporated in *Lelikëlen*, in agreement with the evaluation performed, classify in a satisfactory manner all of the postures. On one hand, in postures P8, P3, and P5, the smallest indications of

detection are present; however, these exceed 70%. For example, take posture P8 (Hand on the chin) into consideration: 73% was classified correctly, and 13% was classified incorrectly. In the case in which it was classified incorrectly, the cause was due to it being classified as posture P3 (Raised hand). On the other hand, only in 13% of the cases could this posture not be assigned to one of the predefined postures.

Finally, upon looking at the confusion matrix, it can be seen that in no case were the postures classified in an incorrect manner by the tool; rather, all of them were radically different, since each one followed a base pattern (e.g., Pointing when it should have been classified as Raised hand).

4 Case Study: Introduction to Informatics Oral Presentations

With the objective of evaluating the tool in a real environment, we requested to use Lelikëlen in the Introduction to Informatics course, offered in the first semester of 2017 as part of the Informatics Engineering degree at Universidad de Valparaíso. The course corresponds to one of the six subjects that students should complete in the first semester of the degree. The case study on this course was based on two aspects. The first one was an analysis of oral presentation of three group projectsfrom three distinct areas (Web Development, Microcontrollers, and Data Bases) that were developed by students. The second aspect examined one of the skills that should be developed, as declared in the course's plan of study: "efficiently communicating in an oral manner within distinct professional contexts".

The course has, approximately, 80 students. The historical dropout rate of the course is approximately 30%, due to various factors, one of them being the students' motivation. The subject matter of this year was dictated by 3 professors, one for each unit. Two of them (Units 1 and 3) belong to the Informatics Department, while the professor assigned to Unit 2, for unforeseen reasons, had to be contracted for the sole purpose of leading the unit.

Each group of students had to be made up of a minimum of 4 members and a maximum of 5. At the end of each unit (Web Development, Microcontrollers, Data Bases), the groups had to orally present the work developed by them in a maximum of 5 minutes, aided by a set of slides produced by each group. The first presentation was performed by 20 groups, while 12 groups took part in the second presentation and 13 groups took part in the third presentation. Before each presentation two students from each group were chosen in a random manner, via raffle, to present. Furthermore, the presentations were performed in the same environment as the one presented in Figure 6. One important aspect to mention is that every participant, before recording, signed an informed consent in which they were informed of the various aspects of the study and acknownledged that their information would be analyzed in a group manner.

After every presentation, each group received feedback based on the analysis performed by *Lelikëlen*. The feedback has the same format as the charts presented in Figures 7 to 9. After the presentation of results, all students took part in a 20-minute class on the importance of body language in oral presentations ans how it is related to some body postures.

The first presentation was performed in April 24th, the second in June 8th, and the third presentation in July 11th. An important point to emphasize is that the focus

of the presentations was to show how students had solved a problem involving the use of the contents of each thematic unit of the course.

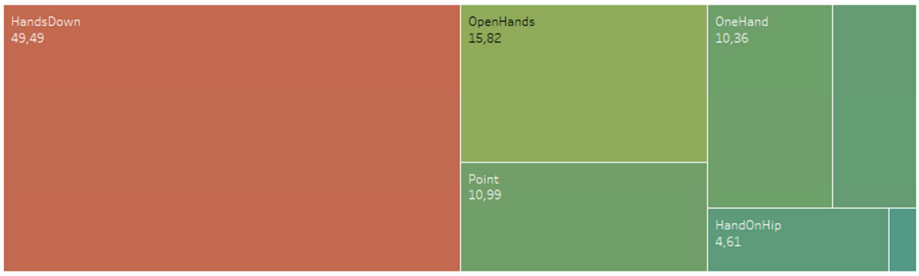

Figure 7: Tree map chart percentage of time per posture (Unit 1)

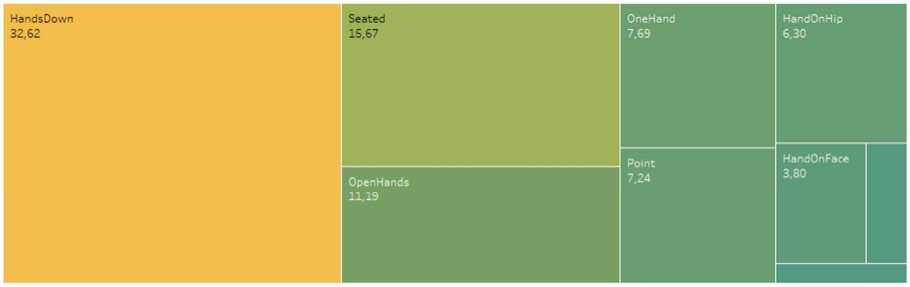

Figure 8: Tree map chart percentage of time per posture (Unit 2)

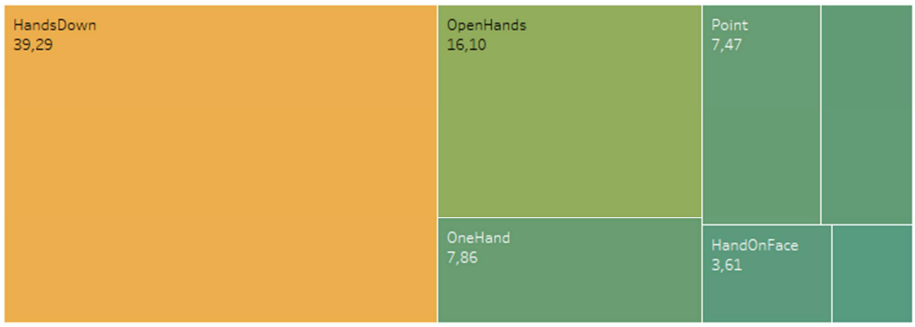

Figure 9: Tree map chart percentage of time per posture (Unit 3)

In general terms, during the first presentation, students were 50% of the time with hands lowered. One reason for this could be that the majority of participants, given that they are first-year students, were presenting for the first time in a university setting. A decrease of this posture can be seen during the second and third presentations. Although this change may be due to various factors, one of them can be attributed to the fact that after delivering the results of the first presentation, students

were given recommendations on characteristics associated with good oral presentations.

On the other hand, upon analyzing postures that are generally associated as ideal for oral presentations (i.e., explaining with one and two hands; pointing), it was noted that these remained relatively stable in the second and third presentation (more than 30% of the time). However, in the second presentation, this was slightly diminished. When looking at possible reasons for this, different factors can be identified. The first one could be due to the complexity of the Unit and the project that students had to present (integration of a microcontroller with software), given that the integration of hardware with software is not a common activity in the first-year courses of the Informatics Engineering career.

The second presentation was also the only one in which the *Seated* position was identified. Although this is not a position that is considered at the time of presenting, students often bended down while presenting to show their model (as shown in Figure 10).

Figure 10: Project Model - Unit 2

Furthermore, one aspect that may also have influenced the outcome of the presentation is that the teacher of this Unit (Unit 2) did not have previous experience in university teaching. A group analysis, using Self-Organizing Maps, is presented in the next section to illustrate the potential of drawing conclusions from data obtained from Lelikëlen.

4.1 Group Analysis Using Self-Organizing Maps

In general terms, the Self-Organizing Maps (SOM) are a type of interconnected and unsupervised artificial neural networks that maps a set of multidimensional data in a bi-dimensional plane. In this way, problems are solved by performing tasks related to data clustering, visualization, and abstraction. SOM can be used to perform studies of correlation between multiple variables (characteristics) present in the data set, without restriction in quantity, since that it has an ability to map data with elevated numbers of dimensions into reduced dimensions. For this, complex non-linear statistical

relationships are converted into geometrical relationships, all the while preserving the original topological relationship, given that the physical location of the data in the map shows the similarity between each in the original multidimensional space [Kohonen 2013].

The training of the SOM network consists of iteratively subjecting the randomly selected input data into a competitive learning model. The neurons, which are composed by a vector of weights, compete for the representativeness of each object in the training set. The weights are iteratively adjusted based on provided input data. The average distance between neurons is calculated as a measure of convergence of the clustering process. Then, each neuron represents a subset of objects used during the training, which are similar to each other, but which also possess similar characteristics to the closest neurons and distinct from the most distant ones.

4.1.1 Database

The analyzed database contained 2,497 records in a maximum of 5 minutes with information pertaining to 9 postures: *AskingHelp*, *HandOnFace*, *HandOnHead*, *HandOnHip*, *HandsDown*, *OneHand*, *OpenHands*, *Point*, and *Seated*. These postures were performed in the 3 sections whose number of presentations and team size are described in Table 7.

Theses records are combined in an analytic dataset with 91 rows and 9 columns (postures), containing the sum of percentage of time taken by the student in each posture per presentation.

Presentation	Number of Presentations	Team Size
1	20	2 students
2	12	2 students
3	13	2 students

Table 7: Summary of Analyzed Data

4.1.2 Results

The grouping analysis and the correlation-among-data analysis were performed using R language and its Kohonen package (Supervised and Unsupervised Self-Organizing Maps) [Wehrens and Buydens 2007]. The SOM parametrizations are shown in the Table 8.

Parameter	Configuration
Data	dataset with 91 records
Grid	Dimension 3x3 with lattice regular
Neighborhood	Retangular
Seasons	1000
Learning rate	Starting with 0.05 to 0.01 with linear reduction

Table 8: Parameterization of the SOM function utilized for the analysis

The data that was subjected to the training process of the SOM network is the duration of the postures that each student performed during oral presentations in proportion to the total presentation time. This was performed with the objective of verifying the influence of each posture on the composition of each node and the consequent topological distribution of the database elements within the map.

The experiment performed produced a set of maps generated by the SOM algorithm that can be analyzed in an isolated or combined manner.

Figure 11 shows the convergence graph of the mapping process, showing that, throughout the training, the variation of the mean distance between each element of the data set and its nearest node was reduced, which determines a greater similarity between its descriptive attributes; in this case, between 8 of the 9 gestures performed during the presentations.

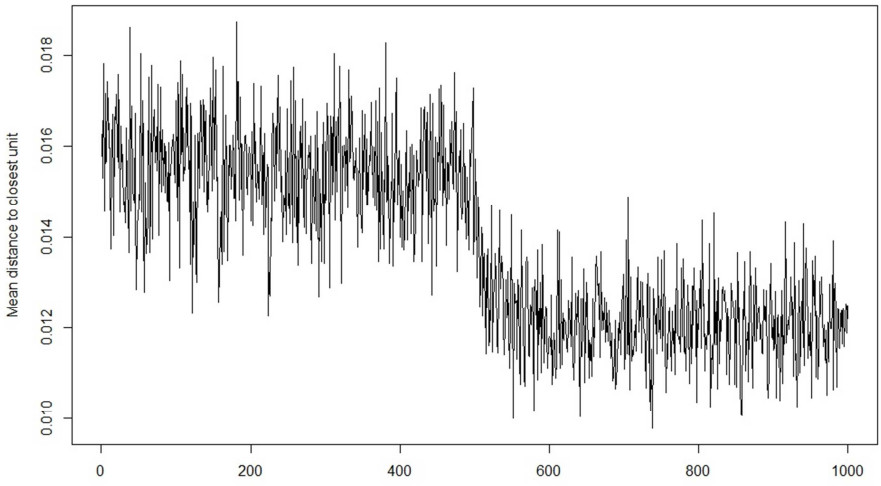

Figure 11: Convergence Graph of Mapping Process

Figure 12 shows the trained SOM map with 9 neurons (3 x 3 grid). This visualization allows to observe the weight of each posture in the composition of each node and the consequent topological distribution of the elements of the database used in the map. The size of colored circular elements is proportional to the values of the weight vector of each node that containing the relevance of each posture. In this way, is it possible to observe that, in greater or lesser intensity, the weight represented by the area of the circular element corresponding to the “Hand on Face” posture is present in all of the nodes. The postures associated with “Open Hands” and “Point” are only present in one node. However, “One Hand” is present in five of the 9 groupings generated. The posture “Asking Help” is present with greater intensity in the right side, both in the upper and lower parts. The posture "Hand on Hip" is present only in two nodes, both in the central column of the map -- the one with the higher weight in the lower line and the least weight in the center line. The gesture "Seat" has only two groups, also, and in this case, the greatest weight is in the neuron located in the lower line in the center of the map, and the least weight in the center line to the left. Furthermore, it is

possible to observe that the "Hand on Head" is only present in one neuron in the upper central part.

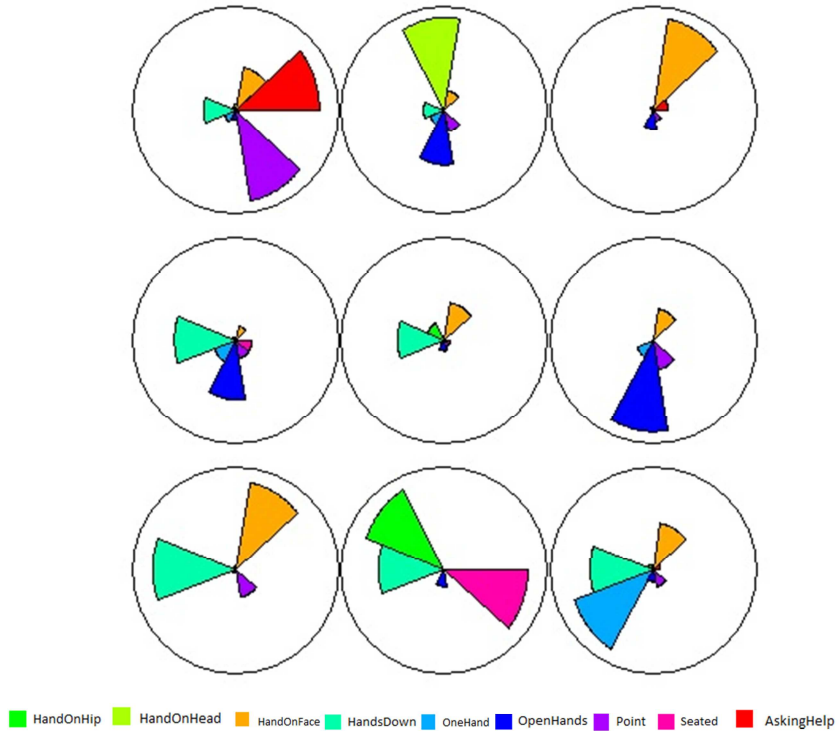

Figure 12: Map of the 9 postures performed during presentations, represented in the form of colored, circular elements

Figure 12, the SOM Map trained, is used for a projection of ID of differents teams. Here the ID team is the same for two students, i.e., the 2 students who participated in team 10, during Presentation 1, are represented with the same number on the map; in this case, they are located in the lower left-hand side.

By conducting a joint analysis of the maps of Figure 13 and Figure 14, it is possible to identify that in the map of Presentation 1, there is a higher concentration of students in the nodes present in the growing and decreasing diagonal and in the central line. Thus, it is possible to infer that there is a concentration associated with a difference of behaviors, this because the map has nodes that contain a greater number of students distant from each other, such as, for example, in the upper-left part and in the lower-right part.

The map of Presentation 2 indicates a concentration in the upper-left node and in the upper and lower right nodes. In a joint analysis with the map of Presentation 1, it is possible to note that the behavior present in Presentation 1 was not repeated in the entirety of Presentation 2, as shown in Figure 13. This is because, in this case, there

was a greater distribution of the students among the cluster nodes, which leads us to infer that difference in behavior exists; however, in a less concentrated manner.

The map of Presentation 3 shows a higher concentration in the nodes present in the growing diagonal and in the upper-left node, repeating, in part, the behavior of Presentation 1, as shown in Figure 14.

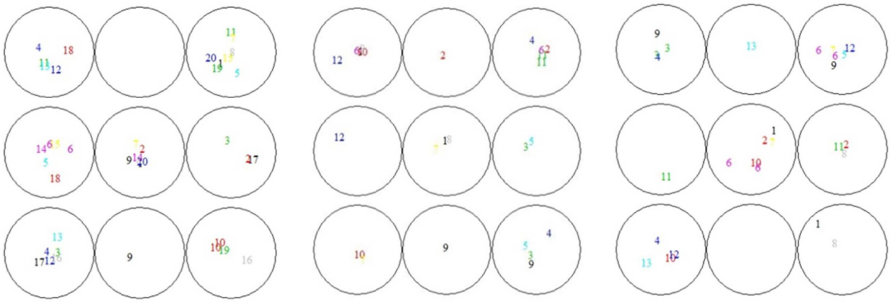

Figure 13: Correlation Map of the 9 postures performed during the presentations (with the attribute of identification of each student that composed a team during Presentations 1, 2, and 3)

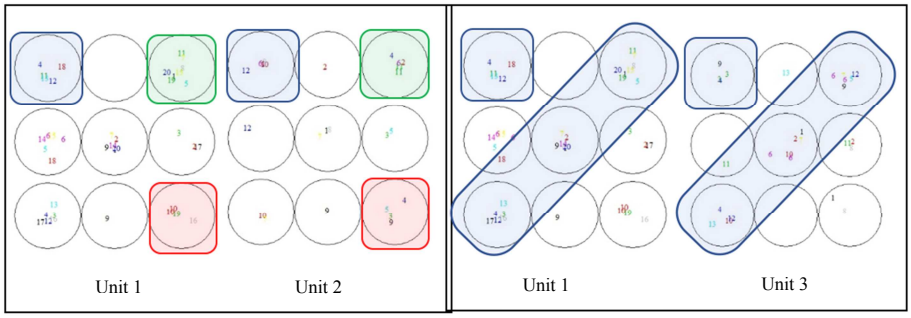

Figure 14: Correlation Map of the 9 postures performed during the presentations (with the attribute of identification of each student that composed a team during Presentations 1x2 and 1x3)

As mentioned, it can be seen that Presentation 1 and Presentation 3 have a similar distribution in the nodes regarding the students. However, upon contrasting them with the presentations associated with Unit 2, groupings are highly different. As mentioned in the beginning of this section, the presentation of Unit 2 had some previously identified particularities: the project was slightly more complex, involving an integration between software and hardware components. Furthermore, the teacher in charge of this unit had no previous experience in teaching. In Figure 14, it is possible to identify that the nodes that group most postures made by students in Unit 2, namely, nodes (0.0), (0.2) and (2.2), have higher weights for postures that indicate

demonstration of concepts, as *Point* in node (0.0) and *OneHand* in node (2.2). These postures were probably associated to the indication of parts of the physical prototype build by students. On the other hand, postures related to doubts, as HandOnFace in node (0.2) and AskingHelp in node (0.0) also had high weight values. The particular clustering pattern of data from Unit 2 already indicated that student behaviour was different; an association of this finding with the mentioned results also indicate the *nature* of these differences.

A large number of students were concentrated in the node from the center of the map (1.1), which is characterized as a node of non-recommended postures while presenting, during Presentation 1. However, during Presentation 2, this number was reduced and slightly increased in Presentation 3. Among the possible reasons for this, is that presentation 3 corresponds to the last unit of the course, and in many of these cases (about 40%), the students depended on the results of this evaluation to determine whether or not they passed the course.

On the other hand, when analyzing the SOM, it can seen that nodes in the lower left and upper right corners, in presentation 1 and 3, have a high concentration. These nodes have, as characteristics, the Hand on Face attribute, which is seen as their main attribute. Unfortunately, this posture indicates, mainly, confusion in the presenter. However, this type of posture would not have been possible to determine, had the use of this type of analysis to support the work of teachers and students not been incorporated. The similar clustering pattern of postures can also be correlated to the contents of Units 1 and 3. In Unit 1 students had to present a web site with content that was previously researched by them, and in Unit 3 they had to present the structure of a relational database. These contents are also covered in other courses in the same semester, in opposition to the hardware components that were studied in Unit 2. Again, the context of the presentations is reflected in the clustering pattern present in the SOM.

5 Conclusions and Future Work

Disciplines evolve when they can be assessed and measured. Under this premise, the incorporation of learning analytics in the context where the transformations occur is of vital importance for the constant improvement of the processes associated with education. Learning Analytics is of vital importance in the area of education, as it can be an aid when evaluating the performance of students during the process and not only at the end of the process. Informatics technologies are a vital support for multimodal learning analytics, given that they may support the measurement of complex performance, facilitating the storage of large volumes of data as well as performing an intelligent data analysis.

The absence of a learning analytics tool that considers the educator as the final user and that also allows future extensions in an easy way motivated the development of this application. It $_{\text{allow}}$ data capturing, analysis and visualization in real teaching-learning environments, presenting performed body postures in a comprehensible form and also allowing the exportation of those results in order to apply, for example, data mining techniques for associated postures, behavior, among other features, and relate those with performance aspects related to the task.

A case study was presented in which we analyzed on average 10 groups of 2 students during 3 presentations performed at the end of each unit of a university course. These presentations were associated with three major topics (web, micro-controllers, databases). In these oral presentations, we captured the body postures performed by students. The group was given feedback in order to promote the students' self-reflection. Furthermore, we used SOM in order to generate knowledge that could be relevant in future cases and also demonstrate the potential of analysis of data exported from the tool.

The SOM in the data analysis is shown as a tool for exploratory visual analysis of the results. It allowed the discovery of group types, the characteristics that differentiate them, and the correlation (at the level of map behavior) between groups. As the main objective of this work is the presentation of the developed tool, SOM was used to verify if the information collected brought about analytical details. With the use of SOM, this could be confirmed, but it is still necessary to conduct a larger study of how to incorporate these results within the framework.

Although the results show that it is not enough to merely declare the development of soft skills, it is important to note that the focus of this work was to demonstrate that it is possible to incorporate this type of tool in order to indicate skill development in a more accurate way. However, it is important to point out that such tools, although providing important feedback on the process of professional skills development, are meaningless if the information is not accompanied by action at the institutional level.

The designed architecture allows the integration of different low-cost sensors, such as GSR (Galvanic Skin Response), HRV (Heart Rate Variability), and EEG (Electroencephalogram), among others. The integration of these types of sensors may bring a high potential for new applications of multimodal learning analytics in learning-teaching enviroments. In future works the correlation between data gathered from these new sensors and body posture tracking data will be incorporated into the system, as well as new visualization strategies.

Acknowledgments

Roberto Munoz, is partially funded by INF-PUCV 2015 doctoral grant. Rodolfo Villarroel is funded by PUCV 2017 039.440/2017 grant. Finally, the authors would like to thank Travis Jones for his valuable contributions to the elaboration of this paper.

References

[Badiru and Racz 2016] Handbook of measurements: benchmarks for systems accuracy and precision: Boca Raton: CRC Press (2016).

[Chen, Leong, Feng, Lee and Somasundaran 2015] Chen, L., Leong, C. W., Feng, G., Lee, C. M., Somasundaran, S.: 'Utilizing multimodal cues to automatically evaluate public speaking performance'; In 2015 International Conference on Affective Computing and Intelligent Interaction (ACII) (2015), 394–400. https://doi.org/10.1109/ACII.2015.7344601

[De Souza 2005] De Souza, C. S.: 'The semiotic engineering of human-computer interaction'; Cambridge. Mass.: MIT Press (2005).

[Echeverría, Avendaño, Chiluiza, Vásquez and Ochoa 2014] Echeverría, V., Avendaño, A., Chiluiza, K., Vásquez, A., Ochoa, X.: 'Presentation Skills Estimation Based on Video and Kinect Data Analysis'; In Proceedings of the 2014 ACM Workshop on Multimodal Learning Analytics Workshop and Grand Challenge. New York, NY, USA: ACM (2014), 53–60. https://doi.org/10.1145/2666633.2666641

[Freund and Schapire 1997] Freund, Y., Schapire, R. E.: 'A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting'; Journal of Computer and System Sciences, 55, 1 (1997), 119–139. https://doi.org/10.1006/jcss.1997.1504

[Kohonen 2013] Kohonen, T.: 'Essentials of the self-organizing map'; Neural Networks, 37, (2013), 52–65. https://doi.org/10.1016/j.neunet.2012.09.018

[Lucas, Hanson and Claxton 2014] Lucas, B., Hanson, J., Claxton, G.: 'Thinking like an engineer: implications for the education system'; Royal Academy of Engineering (2014).

[Nguyen, Chen and Rauterberg 2015] Nguyen, A.-T., Chen, W., Rauterberg, M.: 'Intelligent Presentation Skills Trainer Analyses Body Movement'; Advances in Computational Intelligence, 9095 (2015), 320–332. https://doi.org/10.1007/978-3-319-19222-2_27

[Schneider, Börner, van Rosmalen and Specht 2017] Schneider, J., Börner, D., van Rosmalen, P., Specht, M.: 'Do You Know What Your Nonverbal Behavior Communicates? – Studying a Self-reflection Module for the Presentation Trainer'; Immersive Learning Research Network, 725 (2017), 93–106. https://doi.org/10.1007/978-3-319-60633-0_8

[Shuman, Besterfield-Sacre and McGourty 2005] Shuman, L. J., Besterfield-Sacre, M., McGourty, J.: 'The ABET ?Professional Skills? - Can They Be Taught? Can They Be Assessed?'; Journal of Engineering Education, 94, 1 (2005), 41–55. https://doi.org/10.1002/j.2168-9830.2005.tb00828.x

[Tanveer, Zhao, Chen, Tiet and Hoque 2016] Tanveer, M. I., Zhao, R., Chen, K., Tiet, Z., Hoque, M. E.: 'AutoManner: An Automated Interface for Making Public Speakers Aware of Their Mannerisms'; Proceedings of the 21st International Conference on Intelligent User Interfaces (2016), 385–396. https://doi.org/10.1145/2856767.2856785

[van Harmelen and Workman 2012] van Harmelen, M., Workman, D.: 'Analytics for learning and teaching'; CETIS Analytics Series, 1, 3 (2012), 1–40.

[Wehrens and Buydens 2007] Wehrens, R., Buydens, L. M.: 'Self-and super-organizing maps in R: the Kohonen package'; Journal of Statistical Software, 21, 5 (2007), 1–9.

[Worsley 2012] Worsley, M.: 'Multimodal learning analytics: enabling the future of learning through multimodal data analysis and interfaces'; Proceedings of the 14th ACM international conference on Multimodal interaction. (2012), 353-356. https://doi.org/10.1145/2388676.2388755

[Worsley, Abrahamson, Blikstein, Grover, Schneider and Tissenbaum 2016] Worsley, M., Abrahamson, D., Blikstein, P., Grover, S., Schneider, B., Tissenbaum, M.: 'Situating multimodal learning analytics'; Proceedings of International Conference of the Learning Sciences, ICLS, 2 (2016), 1346–1349.