EAI Endorsed Transactions
on Creative Technologies

Research Article ICST.ORG

Towards the creation of a Gesture Library

Bruno Galveia[1], Tiago Cardoso[1], Vitor Santor[2] and Yves Rybarczyk[*,1]

[1] Faculdade de Ciências e Tecnologia - Universidade Nova de Lisboa, Caparica, Portugal
[2] Instituto Superior de Estatística e Gestão e Informação - Universidade Nova de Lisboa, Lisboa, Portugal

Abstract

The evolution of technology has risen new possibilities in the so called Natural User Interfaces research area. Among distinct initiatives, several researchers are working with the existing sensors towards improving the support to gesture languages. This article tackles the recognition of gestures, using the Kinect sensor, in order to create a gesture library and support the gesture recognition processes afterwards.

Keywords: Kinect Sensor, Gesture Recognition.

Received on 7 June 2014, accepted on 11 November 2014, published on 2 June 2015

doi: 10.4108/ct.2.3.e3

1. Introduction

The Natural User Interfaces have had a major improvement in last decade. The main aim of this interaction is to provide the possibility to control devices through "natural" communication means.

Gestures are one of the most intuitive forms of communication. The proof of this fact is that some gestures may be understood worldwide, whilst the spoken languages are associated to specific regions.

Devices like the Kinect Sensor and the Leap Motion can be easily found in the market. The focus of these devices is the recognition of skeletons, either totally or partially. Moreover, other than these skeleton recognition, these sensors may also map all the recognized points, giving the possibility to extract a movement notion. Nevertheless, despite the skeleton points mapping, these devices do not provide information along time – they only provide information of a single moment in time.

In what concerns gesture recognition, these devices only provide a small set of pre-defined gestures that might be recognized. They don't have any support for the definition of new gestures that one might want to recognize afterwards. The ability to create a gestures library is thus a challenge for the evolution, or next generation of devices like these. A structure like this could provide a programmer with the means to handle both the definition and usage / recognition of gestures.

Taking as a baseline the Kinect Sensor for the creation of such gesture library, there still exist some technical issues that constitute a problem, namely at the data acquisition level, as the position of the skeleton points are not always provided by this sensor.

The main objective of this research work is to improve the Software Development Kit of the Kinect Sensor, through the introduction of the time dimension. Introducing this element to the instantaneous information already provided by this SDK, it will then become possible to handle gestures.

The intended improvement consists of two major lines:

1. Gesture Library creation support – This first line will be responsible to capture gestures. Using the sensor abilities to get instantaneous data on the position of the distinct points / joins of the skeleton and adding the time dimension, the idea is to become able to store such gesture information so it can be used afterwards.
   In concrete terms, this gesture library will store the gestures that a programmer need in a specific system he or she is developing, like, for example, the Portuguese Sign Language (PSL) gestures.
2. Real-Time Gesture Recognition – The main objective of this second line is to provide the

[*]Corresponding author. Email: y.rybarczyk@fct.unl.pt

possibility to recognize any gesture previously stored in the library. This recognition should be made in real-time, meaning that as the user makes a gesture, the system should be able to make the recognition, without any knowledge about the starting or ending time of the gesture performed by the user.

The process for the creation of the library and the recognition that might take place afterwards was intended to be as simple as shown in Figure 1. The user first makes the gesture while the Kinect Sensor capture the needed information from such gesture and the gesture is added to the library. When the gesture recognition starts, all the stored gestures become active, meaning that if the user makes such gestures, the system will recognize them.

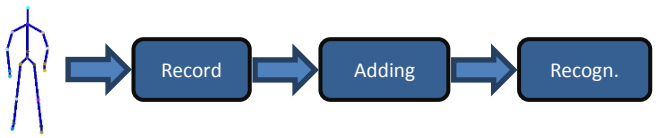

Figure 1. Sequence for capturing and recognizing of gestures.

2. Related Work

The communication with devices using Natural User Interfaces is an approach that has gained the attention of both the research community and the industry, especially in the last decade. From the user perspective, the preference goes to a more intuitive and natural way, rather than traditional means of interaction.

Several application areas have started to use and take advantage of these NUI approaches. One example area is the entertainment that started to introduce the control of systems through the user body movement. Devices like the Wii Motion, Move and the Kinect Sensor are widely used by the games' industry world.

On the other hand, the development of this kind of devices has also targeted other areas distinct from the gaming world. The Leap Motion is one example of sensor that supports the NUI approaches not in the gamming area.

These movement sensors work as good decoders of signal communication. Based on that, a project developed in China, with the Asian Microsoft Research support, used the Kinect Sensor to make gesture language recognition. The aim of this project was to facilitate the simple communication between people. One of the ways that this project works is through gesture recognition / translation – one person performs a gesture, the system recognizes it and shows the corresponding text. It is also possible to work in the reverse order, where a written word is presented by an avatar that performs the corresponding gesture [1].

This kind of systems always needs the previous creation of a set of gestures that might be recognized afterwards. The systems have their own mechanisms for the gestures definition / specification, along with a specific method for the recognition process, based on the way the gestures are defined.

Nowadays a few number of robust systems are able to create and recognize gestures. The project Full Body Interaction Framework is one of the open source examples in this area, as mentioned in Social Signal Interpretation Framework [2]. In this case, the gesture composition is built upon relative positions, angles, time and some other variables.

In what concerns the movement description, some of the proposals that were studied use a set of vectors. The project Gesture Recognition with a Wii Controller is one example of these cases [3]. In this case, the vectors represent the direction and acceleration of the movement that is being captured.

One other mechanism that is being used for the definition of a trajectory is the usage of little segments. These little segments have the information of velocity, acceleration weight, among others [4].

In what concerns the recognition methods, one example technique that is being widely used is the Pattern Matching, trying to find parts of the gesture that have similarities with previously saved gestures. [5].

3. Proposal

The objective of this work is to propose a new programming abstraction layer for the Natural User Interface devices.

Generally speaking, as shown in Figure 2, this abstraction layer surrounds the SDK from the devices, extending their functionality.

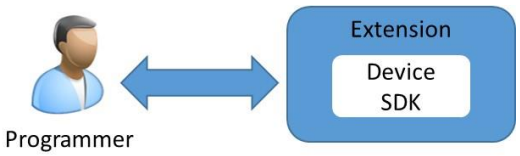

Figure 2. Interaction Diagram.

This new set of functionality aims at helping the programmer to create and detect / recognize gestures. In a first stage the task is the capture and modelling of the gestures. In a second stage, the task is recognising previously saved gestures in real-time.

In the gesture capturing and modelling phase, there is the sensor that is responsible for the external data acquisition. This data go through a workflow of transformations with the objective of representing the gesture as a sequence of equations. The modelling of gestures as equations and not as a simple storage of all the points for future comparison is one of the main aspects of this proposal and the main reason for this is the less needed computation power for the recognition phase.

In the real-time recognition phase, using the same external data acquisition mechanism, when the movements are captured, they are compared to the gestures that have been stored in the first phase. This comparison is made through the distance of the captured points to the equations representing the previously saved gestures.

Gesture Definition

In a first stage we have to define what a gesture is. In other words, what has been considered as a gesture in this work? A gesture is a movement of one or more parts of the human skeleton. In the perspective of the movement sensor, a set of Cartesian positions are selected and captured. This set is then used to model the form of an equation and then stored in the gestures library.
Figure 3 represents the movement of one articulation of the human body – the hand in this case, and the definition of the movement as a segment of an equation.

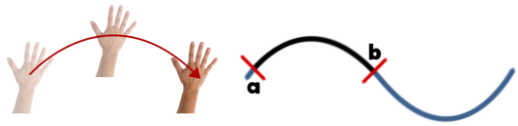

Figure 3. Definition of one movement as a segment of an equation.

Every movement may be defined as an equation. Even a straight line may be defined as a quadratic equation. The important element is the definition of a scale and the time frame. In the case of Figure 3, the right side of the equation (after b – in blue) does not seem to describe the gesture, but the left side of the equation seems very similar to the gesture of the hand (between a and b).
As the human movements occurs always in a 3D scenario, the gestures will always be defined with two equations, referring to Y and Z in terms of X, as the following 2 equations:

$$y = a \qquad [a\ b]$$
$$[a\ b]$$

There are several problems for the description of a movement as equations. One of the problems is when there is an inversion of the movement in the X axis, as an equation can only take one way. On the other hand, in some more complex movements, the trajectory may not fit directly on one single equation. Because of these two aspects, the current work proposes the definition of a gesture as a sequence of equations, as represented in Figure 4. The left side trajectory (in black) represents what we intend to model through the usage of equations. At the middle of the figure, the set of equations used to represent this trajectory are presented (in distinct colours). At the right-most side of the figure, the right segments of each equation are selected (in black), forming the final sequence that represents the left side gesture. Selecting the right equations at each time is a "trial and error" process and thus it is heavy in terms of processing speed needs.

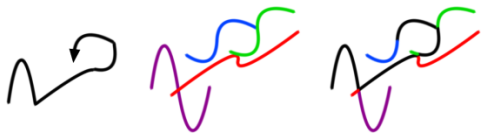

Figure 4. Defining a gesture through a sequence of equations.

Capture and Model

The first stage consists of capturing the movement of the articulations / joins of the human body during a period of time. In order to do so, the first task is the definition of the time frame and the joins that will take part in the movement being analysed.
After this first specifications, the data acquisition may start and the information of the position and movement of each of the selected articulation, or join, is gathered and stored, during the pre-defined time-frame. The experimentation of capturing this information reveals that in several cases, the information retrieved includes errors. These errors result in some points that seam out of the place they should be. This may happen because of the light in the room, for example. In order to deal with this constraint, a minimal and maximum threshold was defined in order to exclude points that are captured outside this framing.
One gesture can and should be captured more than one time. This repetition is then used to increase reliability. In other words, the final gesture model is the result of the "average" of each articulation / join of the human body that is included in such gesture. Moreover, after capturing one sample data for a gesture, it is also possible to define the start and the end of that gesture.
As several capturing samples are acquired, the "average" process is used to improve the trajectory definition. Although the distinct representations of a gesture may be similar, there are always slightly little differences. This normalization process is used exactly to decrease the effect of this deviation.
Figure 5 shows some sample capturing results for a single gesture. There, one can see that distinct capturing samples may have distinct dimensions and, as so, the normalization process tries to reduce this effect.

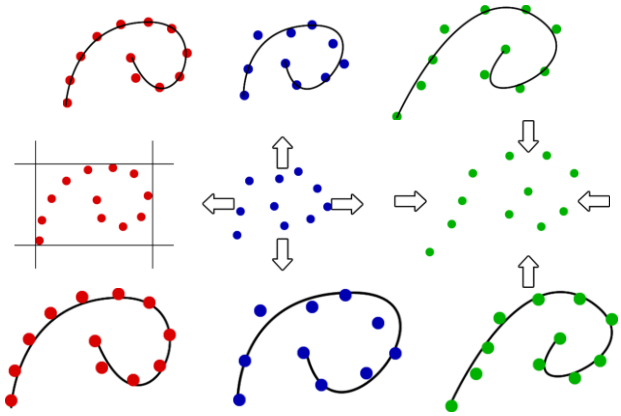

Figure 5. Redimensioning gesture repetition.

This normalization process is intended to smooth the distinct samples of the captured gesture. The algorithm used herein is the Simple Moving Average (SMA), which is an algorithm largely used in the economics area in order to deal with data analysis [6]. Figure 6 shows an example of the usage of this algorithm for distinct samples of a given trajectory.

Figure 6. Applying the SMA algorithm.

The main challenge faced in this research work was the mechanism to transform a set of points into an equation. The process of applying this SMA algorithm has largely improved this process.
The transformation of this set of points in an equation follows an approach of a polynomial equation of degree 3, using the minimum quadratic difference. This method is composed of a set of equations from witch the coefficients of the polynomial equation are extracted. Afterwards, a test is made in order to assess the quality of the obtained equation. This test is made using the points initially capture and normalized from the gesture and processing their distance to the equation that resulted from this process.
In the case where an equation does not completely define a gesture, the list of points is divided in shorter steps in order to repeat the process of finding an equation for each step and, as a result, the final representation of a gesture will consist of a sequence of equations.
In what concerns the selection of 3^{rd} degree equations and not from 2^{nd} or 4^{th}, for example, was made because of two reasons: 1 – 3^{rd} degree equations are already able to represent movements somehow complex; 2 – if one intend to use a sequence of equations, there is no need to go to more complex equations – the 4^{th} degree ones.

Recognition

The real-time recognition intends to monitor all the movements from all the articulations or joins from the human body and try to assess if some gesture is being made.
The data structures used in the real-time recognition work as a buffer that stores the points of each join during time. During this process, all the positions of each join are compared to the existing gestures towards reaching a matching between the movements the user is making and one specific gesture from the library.
The comparison of the points with the stored equations is made through a distance of each point to the equation. The movement of a join is valid for a given gesture if the distance of all the points follow the path or trajectory of the equation and maintain a minimal distance to it. The movement is valid during a specific time frame. For the gesture to be recognized, there must exist a time frame where all the articulations / joins included in that gesture become validated. Figure 7 shows, in the left side case, the movement where 3 articulations are recognized, whilst in the right side case the recognition did not succeed.

Figure 7. Time dimension in the recognition of each join / articulation.

Sometimes, the existing points do not invalidate a specific gesture but are not enough to reach the end of the equation. In this case, what happens is that the system should continue storing points in the buffer.
Every time new points are added to the buffer, a new validation takes place for each articulation / join and three results may happen:

- Detection of movement for that articulation / join,
- Invalidation of the sequence of points and the corresponding articulation / join, or
- Insufficient points.

In the case of the invalidation of the sequence of points, the latest point and the corresponding articulation / join is removed from the buffer and a new try is made for the validation, using the newly defined first point.
Figure 8 represents this process. In the left side, the last point (in red) exceeds the minimal distance to the equation and no other point is close to the end of the equation. The process removes the latest point (green). The assessment process is repeated starting in the newly defined first point (yellow). In this case the validation succeeds and the gesture is recognized.

Figure 8. Invalidating a points' set.

This recognition is continuously made until it becomes deactivated. The idea is to provide a framework to the programmer that, somehow is working all the time towards gathering the information from the sensor in what concerns the movements of the human body and, moreover, the gestures he or she might be performing.

4. Validation

The validation of this research work was made through the development and test of a prototype system. The selected sensor for this purpose was the Kinect Sensor, from Microsoft. This movement sensor uses an RGB camera and a depth sensor towards recognizing, at the firmware level, the articulations / joins of the human skeleton, resulting on the tracking of such joins [7]. The creation of this prototype was based on the information provided by the Kinect SDK in what concerns each articulation / join.

For testing purposes and in what concerns the physical environment, the sensor was placed at 2 meters far from the human actor and 1.4 meters from the floor, towards being able to capture the entire skeleton. The tests were carried out in a closed environment in order to avoid the infra-red disturbances.

The process of validation is shown as follows in two phases. The first phase is intended to test the proposed method for the storage of gestures in the library. The second phase intended to show the benefits of the proposed method for the real-time recognition of the gestures previously stored in the library.

Gesture Storage

The gesture storage starts by the capturing of the human movement.

For testing purposes, distinct gestures from the right hand have been captured. Each gesture capturing was made four times. Figure 9 represents the set of points stored in each capturing sample, using distinct colours.

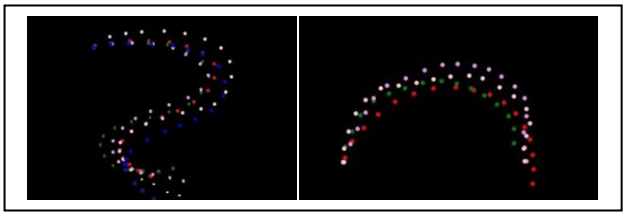

Figure 9. Set of points stored for each gesture (one colour for each sample).

Initially, a sequence of equations was retrieved using only one sample from the gesture capturing. Some similarities can be found in Figure 10 and the points from Figure 9. Figure 10 also shows that one of the gestures was modelled using two equations whilst the other gesture was modelled using four equations.

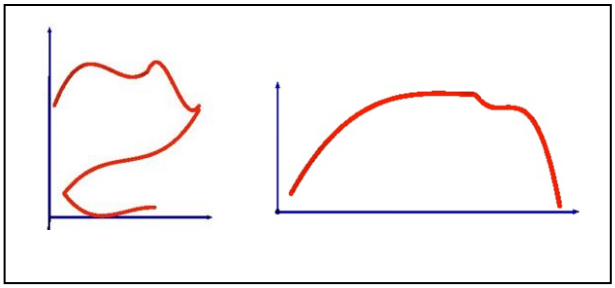

Figure 10. Equations for the definition of two gestures.

For the comparison purposes, the four sets of points were stored for each gesture, after passing through the normalization procedures. After these processes, using the same mechanism of finding the distances to the polynomial equation of degree 3, new sets of equations were found.

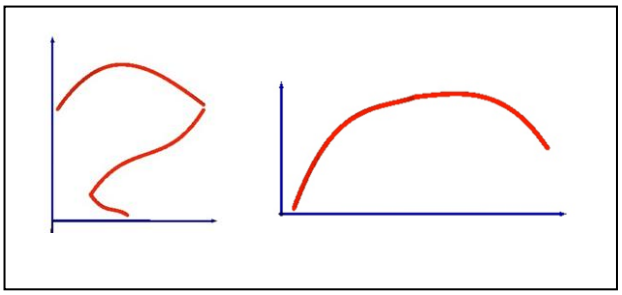

Figure 11. Equations for the definition of two gestures after several sample captures.

Comparing Figure 10 and Figure 11, one can notice that a reduction of the number of equation and a better definition of the trajectory took place, decreasing the error margin for the future recognition processes.

Recognition

This test was carried out towards showing how reliable the proposed methods is.
At a first stage, some gestures were stored in the gestures library. Each gesture was shown to a set of 10 persons that should reproduce such gesture in front of the Kinect sensor. The process was carried out resulting in a total of 40 samples stored for each gesture.
The gestures that were stored in the gestures library, as shown in Figure 12, were selected in order to have some mathematical complexity. On the other hand side, this selection also took into account how common these gestures are in the Natural User Interfaces area.
For these gestures, the only parts of the human body that are used are the hands. Each colour represents a single hand movement. It is also worth to mention that gesture D represents hand clapping. In order to validate this gesture, it has to be repeated three times. In other words, three hand claps.

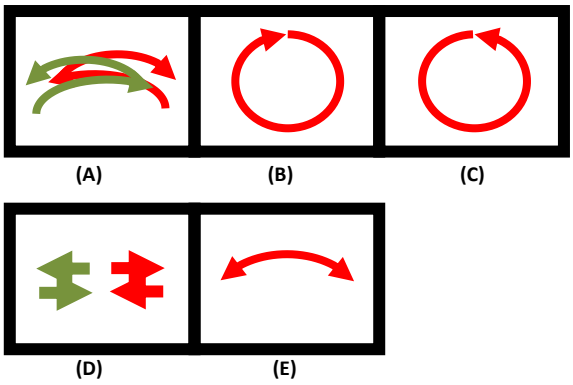

Figure 12. Gestures used for validation (swipe, clockwise circle, counter clockwise circle, hand clapping and no gestures).

Two kinds of tests were carried out, using this gestures library. First with a singular gesture in the library, and then with several gestures from the library.
In the first test, each gesture was validated individually. Table 1 shows the success rates for the recognition concerning this first test.
In what concerns the gestures that are composed of more than one gesture from the gestures library, using both hands, the success rates achieved slightly decreased because of some wrong information gathered whenever there is an overlapping between the hands.

Table 1. Success rate for individual gesture recognition tests.

	Success Rate
Gesture A	72,5%
Gesture B	82,5%
Gesture C	75,0%
Gesture D	75,0%
Gesture E	75,0%

In the second part of the test, the gestures library already had all the above mentioned gestures. There, the objective was to assess the correct decision in the recognition process of a given gesture. In other words, at the moment when the system recognises one gesture, it was validated if the recognised gesture really was the gesture made by the human actor or if it was a wrong recognition pointing out to another gesture. The results of this second test are shown in Table 2.

Table 2. Success rate for the recognition of distinct gestures performed by both hands simultaneously.

	Correct Recognition	Other Gesture
Gesture C	100%	0,0%
Gesture D	85,0%	15,0%
Gesture E	72,5%	27,5%

One of the possible explanations for the distinct success rate results may be based on the quality of the selected equations for each gesture. In order to check this possibility, the same procedure was carried out, but this time with the gesture equations directly selected by the human user, instead of the selection made by the system.
Table 3 and Table 4 show the results of these tests. The comparison of the two sets of tables one may verify that the decision on distinct gestures from a library may clearly be improved.

Table 3. Success rate for the recognition of singular gestures with a manually selected equation.

	Success Rate
Gesture B	77,5%
Gesture D	90,5%
Gesture E	87,5%

Table 4. Success rate for the recognition of several gestures with a manually selected equation.

	Correct	Other
Gesture B	97,5%	2,5%
Gesture D	100%	0,0%
Gesture E	95,0%	5,0%

5. Conclusions and Future Work

Natural User Interfaces are nowadays gaining more and more relevance as a communication means. Looking ahead, many systems are borrowing the interaction mechanisms of these kind of interfaces. Following this trend, it becomes mandatory to create new frameworks and tools for a smoother adoption of this paradigm to take place.
This work tried to contribute in this direction, through the creation of a working tool for the programmer, providing him or her:

- Mechanisms for the creation of new gestures during the production phase of their systems.
- Mechanisms for the recognition of the gesture stored in a gestures library created in the previous phase.

As the data acquisition device used for the proof of concept was the Kinect sensor. This device already provides the skeleton information at the firmware level. A new structure was created upon this skeleton information towards being able to model gestures as sequences of equations. Some algorithms were also developed in order to transform the points captured by the sensor in such equations.
Finally, a prototype system was developed in order to test the proposed methods.
During the gestures capturing phase, some difficulties happened sometimes in the Kinect side that retrieved wrong positions from the joins. Despite these errors, the presented prototype has shown some good success rates in the recognition processes, showing a new mechanism of how gestures can be created and recognized.
This validation has also shown that for the correct performance of the proposed method, the runtime environment has to be close to optimal, namely in what concerns light, because of the Kinect depth camera, on one hand side. On the other hand side, the machine must have good processing speed in order to be able to make the recognition at runtime. In other words, problems will surely arise for large gestures libraries and low processing speed.
In what concerns the structure selected for the gesture modelling as equations, it turned out to be fruitful, namely in what concerns the needed processing speed. Moreover, at the recognition level, a good selection of the equations to represent the gestures also greatly improves the success rates.

Future Work

As future work, naturally some aspects may be improved. In what concerns the capturing part, some difficulties have been experienced mainly due to the precision of the position of the joins of the human skeleton.
Another aspect that is interesting to study is modelling of gestures using polynomial equation of degree 2, instead of polynomial equation of degree 3.
In what concerns the real-time recognition, the evolution should follow the optimization of the needed processing speed, as a significant number of gestures demand a high values of this processing speed, resulting on undesired delays.
This new proposed tool may be integrated in any application that intends to use gesture recognition. Initially, it was created thinking about a Kinect SDK extension, but at a higher abstraction level, the proposed methods application to other SDKs and the corresponding devices seems reasonable as well.
The usage of this framework already took place in the creation of the Kinect-Sign [8], which is a game for the learning of sign language. This game is a good example for the usage of the gestures library, as it first creates the set of the gestures that compose a particular sign language (for example the Portuguese Sign Language), and then it passes to the recognition phase, where the persons perform the gesture with their hands, the system recognizes such gestures and the system behaves accordingly.
Other example systems may also be developed upon the gesture framework, in particular to support the transfer of motor skills from virtual to real environments [9].

References

[1] X. Chai, L. Guang, L. Yushun, X. Zhihao, Y. Tang, X. Chen and M. Zhou, “Sign Language Recognition and Translation with Kinect”, *AFGR 2013 : 10 th IEEE International Conference on Automatic Face and Gesture Recognition*, Shanghai, China, 2013.

[2] J. L. F. B. T. D. L. K. F. and A. E. Wagner, “The Social Signal Interpretation (SSI) Framework”, *ACM International Conference on Multimedia*, Barcelona, Spain, 2013.

[3] T. B. P. H. N. and S. B. Schlomer, “Gesture Recognition with a Wii Controller”, *TEI 2008 - Proceedings of the 2nd International Conference on Tangible and Embedded Interaction*, Bonn, Germany, 2008.

[4] W. Chen and S.-F. Chang, “Motion Trajectory Matching of Video Objects”, *SPIE/IS&T Storage and Retrieval for Media Databases*, San Jose, CA, 2000.

[5] G. Navarro, “Pattern Matching”, *Journal of Applied Statistics*, 2004.

[6] Y. L. S. P. J. and N. L. Ding, “HUNTS: A Trajectory

Recommendation System for Effective and Efficient Hunting of Taxi Passengers", *Mobile Data Management*, Milan, Italy, 2013.

[7] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of Kinect depth data for indoor mapping applications", *Sensors : journal on the science and technology of sensors and biosensors*, 2012.

[8] J. Gameiro, T. Cardoso and Y. Rybarczyk, "Kinect-Sign, Teaching sign language to "listeners" through a game", *Conference on Electronics, Telecommunications and Computers*, Lisboa, Portugal, 2013.

[9] G. Carrasco, Y. Rybarczyk, T. Cardoso and I.P: Martins "A serious game for multimodal training of physician novices", *6th International Conference of Education, Research and Innovation*, Seville, Spain, 2013.