An IGS Gesture Dictionary for Modeling on Mobile Devices

Seiji Isotani[1], Helena M. Reis[1], Danilo Alvares[3], Anarosa A. F. Brandão[1], Leônidas O. Brandão[1]

[1]University of Sao Paulo, Brazil

sisotani@icmc.usp.br, helenamcd@gmail.com, anarosa.brandao@poli.usp.br, leo@ime.usp.br

[2]Department of Statistics and Operations Research - University of Valencia, Spain

daldasil@alumni.uv.es

Abstract

Interactive Geometry Systems (IGS) are educational tools that help to teach and learn geometry in an dynamic environment. Traditionally, the interaction with IGS is based on keyboard and mouse events where the functionalities are accessed using menu of icons in the interface. Nevertheless, recent findings suggest that such a traditional model of interaction has a steep learning curve and is inadequate to develop IGS for devices with multi-touch screens (e.g. tablets). Thus, this work proposes a new interaction model for IGS based on a gesture dictionary which enables the construction and manipulation of geometric objects without the need of accessing a menu of icons. The proposed dictionary is divided in three types: (i) kernel gestures: which are the basis for defining gestures; (ii) navigation gestures: related to the manipulation and editing of geometric objects; and (iii) basic gestures of construction: to construct geometric objects. To validate our work, an IGS for mobile device, referred to as GeoTouch, has been developed using the proposed gesture dictionary. Usability tests were performed comparing GeoTouch with three other IGS for mobile devices available to date. The results indicate that the GeoTouch interface has fewer usability problems than the other three IGS software and it is easier to learn and interact with.

Keywords: dynamic geometry, gestural interfaces, educational technology, interface design, usability

This version has a few updates over its correspondent published version at: Journal Interactive Learning Environments - http://www.tandfonline.com/doi/abs/10.1080/10494820.2017.1325377

Original citation:

Isotani, S., Reis, H. M., Alvares, D., Brandão, A. A., & Brandão, L. O. (2017). *A DGS gesture dictionary for modelling on mobile devices*. Interactive Learning Environments, 1-17.

1. Introduction

Since the popularization of personal computers at the end of the 1980s, the emergence of a myriad of computer systems for teaching and learning could also be observed. Among these, **Interactive Geometry Systems** (IGS) appeared, which introduced a new paradigm for geometry learning (Isotani & Brandão, 2008). By using IGS on computers, learners can explore geometric constructions and, more importantly, interact with them (Roanes-Lozano *et al.*, 2003; Kortenkamp and Richter-Gebert, 2004). These constructions are made from geometric objects, which can typically be lines, points, circumferences, and connections between these objects. In IGS, these connections are essential so that the system can redesign the whole construction by moving some of its objects, but maintaining properties from the initial construction (Isotani and Brandão, 2008; Ng & Sinclair, 2015).

Since the first IGS, the interaction model implemented in most of these systems has been based on icons, in which objects and operations are represented on them (Jackiw, 1995; Kortenkamp; 1999; Reis *et al.*, 2016; Borges *et al.*, 2016). In the early 2000s, several of these systems had interfaces with a large number of construction options, which could hinder learning, especially of a novice user. This difficulty was found in some studies that analysed the number of icons at the interface of IGS, as in Schimpf and Spannagel (2011) and Reis *et al*. (2012) An additional complicating factor in interfaces with many icons, as highlighted by Borges *et al*. (2016), is the fact that there are similar icons, which can cause more doubts and construction errors, and consequently discourage and frustrate learners during his knowledge construction process.

As a possible response to this difficulty an new interaction model and interface was proposed by Santos *et al*. (2006), in which the use of icons was abolished in favour of a metaphor of geometric design instruments, such as the set-square, compass and pencil. However, such an interface was primordially intended for geometric design, where it does not use connections between the properties of objects as in IGS; and therefore students could not interact and freely manipulate geometry constructions. Another proposal to deal with the difficulty caused by the number of icons is the use of gestures (Hinrichs and Carpendale, 2011), especially in devices with touch screens (Kortenkamp and Dohrmann, 2010; Ehmann et al., 2013).

If on one hand the use of interfaces based on gestures (gestural interfaces) can contribute to reducing elements at the interface, making them visually more simple (Nacenta *et al.*, 2013), on the other hand, the recent popularity of touch screens and their miniaturization of mobile devices, indicates the relevance of this approach. In this context, there is a need to propose a new interaction model for IGS that (i) reduces the learning curve; (ii) does not require icons in the interface to interact with the user; and (iii) can be easily applied in touch screen interfaces. Thus, in this work we propose a dictionary of gestures that can be used to interact with IGS in order to create/edit/manipulate geometric objects as well as access the most common functionalities of a IGS without using icon-based interfaces. We also developed a IGS for mobile device referred to as GeoTouch that implements our proposed dictionary and compare it with other three IGS available to date. The results show a significant improvement in the reduction of errors and interaction problems.

2. Related work

According to the work of Reis *et al.* (2016), who performed a literature review covering 998 articles published in more than 10 years of research in the field of interactive geometry, there are still few studies that demonstrate the use of interaction through touch interfaces allowing for the execution of gestures to construct geometric objects. One of them is the study conducted by Blagojevic *et al.* (2012) that combine tablets' surface and traditional tools, such as a ruler and compass, to create geometric objects. Although this study is not exclusively about IGS, the authors emphasize that this tool may also be used for teaching geometry using interactive figures. Another interesting study is presented by Blanke and Schneider (2011) in which students may interact with geometric objects using a multi-touch interface develop for tabletops. However, this paper does not address the free creation of geometric constructions available in conventional IGS, nor does it propose gestures to manipulate these objects properly. Finally, a recent study conducted by Vitale *et al.* (2014) explored the relation between how students interact with geometric objects and how they gain knowledge related to geometric properties. The results showed that the way the interaction is defined may either positively or negatively affect the teaching-learning process. Despite the benefits of this study, it did not focus on developing, implementing, and evaluating actions to construct geometric objects.

With respect to the existing IGS that include some types of multi-touch interaction to build geometric objects, we identified three tools available to date:

- **Sketchometry**[1] was released in July 2012 and developed at Bayreuth University, Germany. The software was implemented using HTML5 technology, which allows it to be run in any browser that supports this technology. The researchers' goal was to provide IG software that may be run on desktop computers, as well as mobile devices such as tablets and smartphones.
- **Geogebra**[2] was started by Markus Hohenwarter in 2001 at the University of Salzburg and is also free. The software supports primary schools to universities, and is available in multiple languages and joins features of geometry, algebra, tables, graphs, probability, statistics and symbolic calculations. Its first version was developed in Java for desktop, however a beta version for mobile devices that have Android and iOS as operating systems was made available at the end of 2013.

1 http://sketchometry.org
2 http://www.geogebra.org/

- **GeometryPad** was developed by Byte Arithmetic LLC and is available only for mobile devices with Android and iOS operating systems. Some features, such as constructing tangent or geometric objects by algebraic equations may be used only after the user buys the software. Similarly to Geogebra, the interface of GeometryPad mimics the interface of IGS available on desktop computers.

The interface of these tools often mimics the available IGS for desktop computers. Thus, this work proposes a gesture based interface that reduces the amount of icons in the interface and allows for the association of gestures to mathematical concepts which may provide advantages to learning (Yerushalmy, 1999; Schwartz, 1995) and can be more suitable for multitouch screen available in mobile devices.

3. Development of Gestures to Interact with IGS

To design a new interaction model that connects gestures with geometric concepts, interviews with mathematical experts were conducted in order to investigate which activities are carried out within the learning environment to support geometry learning using IGS. At this stage, four experts in Math Education were interviewed individually for about an hour. The four interviewed professors are Math Educators with major in Mathematics and Ph.D. in Education. All of them have more than 20 years of professional experience as teachers in k-12 as well as positions in University at the Department of Math Education. Their main research focus on this topic is related to developing new pedagogical approaches with the advent and use of new technologies in classroom. In order to conduct the interviews a set of standard questions was defined. Furthermore, freedom was given to the experts to express their expectations and needs about the use of technology, particularly IGS on mobile devices in the school context.

Interview data were transcribed and tabulated to help define both family gestures to construct geometric objects and the requirements of IGS for mobile devices. In particular, the interviews showed that learning geometrical concepts may be hindered, if the gestures created do not have any relation with the geometric properties of objects, corroborating with initial previous findings (Vitale *et al.*, 2014). For instance, if a circle is drawn with a circle gesture, the student will not associate that the whole circumference has a radius r and that all points of this circle are at a distance r from its center (i.e., the points are equidistant from the center), causing conceptual problems that may affect learning. To address this problem, the significance of each gesture performed is important, that is, mathematical concepts underlying the construction must be used to define a gesture. In addition to the educational benefits that such an approach may offer, there are also advantages related to the development of an IG interface, because the student may apply prior knowledge of geometry to interact with the software and perform actions.

Besides identifying the most common features used during the activities cited by the interviewed professors (i.e. creating and editing circumferences, lines, semi-lines, line segment and points), the requirements according to the available versions of IGS were also listed. Using this information, case diagrams could be used and the main features for designing new ways to interact with IGS were defined, which will be presented in the following sections.

Based on the identified features and requirements, we created a gestures dictionary to construct geometric objects and interact with the system (Figure 1). The gestures dictionary consists of the textual description of the object, its relations with the geometric concepts, and a set of illustrative figures that show how to perform gestures on a multi-touch screen-based interface. Initially, we generated multiple gestures to address the same functionality or requirement and, after multiple prototypes and interactions with the users, three categories of gestures were defined to systematize the way of interacting with the system and allow a high degree of flexibility and extensibility. These categories are described below:

- **Core gestures:** main gestures of the software, which are the basis for defining gestures of other categories (top-left of Figure 1). In this category the following gestures are defined: brief touch, press, movement, drag, rotate and adjust;
- **Navigation gestures:** This category groups the gestures used to perform activities such as manipulating and editing objects (top-right of Figure 1);
- **Basic gestures of construction:** The gestures of the category "basic gestures" correspond to the gesture needed to create simple geometric objects. Thus, in this category gestures to create points, midpoints, circumferences, lines, line segments, and etc, are described. All movements are based on the core gesture category and were created in order to be directly related with the geometric concepts involved with the constructed object (bottom of Figure 1).

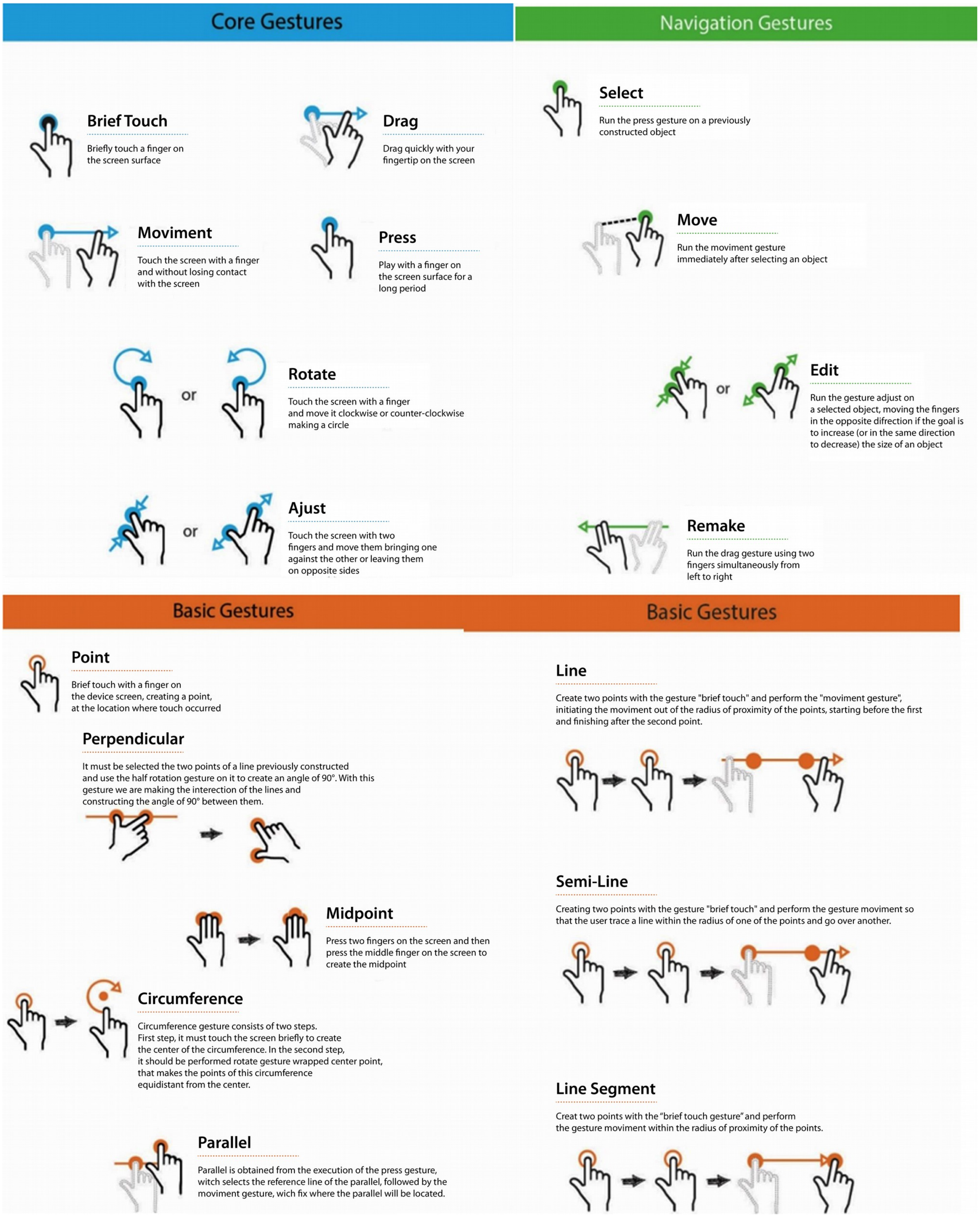

Figure 1. Gesture Dictionary to interact with IGS.

To have a better understand about the rationale behind the development of basic gestures of construction and compare it with previous approaches, we will present some classic examples of basic gestures to construct geometric objects. In the following paragraphs, we will indicate the set of basic gestures related to the act of construction using our gesture dictionary (Figure 1); then we will make the relation with the mathematical concept related to the geometric construction; next we will show the description and a figure to perform the gesture; and finally, we will present how to carry out the same geometric construction in three other IGS software available to date, namely Sketchometry, Geometry Pad, and Geogebra, explored in Section 2 of this paper. It is also important to emphasize that in Section 6 we will present the results of an heuristic evaluation using IHC techniques to compare these forms of interaction.

Gesture: Midpoint

Base gesture: Press, Brief touch

Conceptual definition: Given two points A and B, the midpoint is the point equidistant from A and B. The point makes up the midpoint, which is the center point equidistant from both end points of a segment of a line.

Description: The gesture consists of pressing two fingers on the screen and then pressing the middle finger on the screen to create the midpoint.

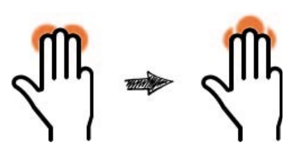

Figure 2. Midpoint gesture.

Comparison: In Geogebra software, the midpoint is constructed by pre-selecting it on the menu. Subsequently, the user must create the first and second points and the midpoint is automatically constructed. In the GeometryPad and Sketchometry software, this feature is not available.

Gesture: Line

Base gesture: Brief touch, Movement

Conceptual definition: The line is an infinite set of geometric points, having no curvature, beginning and end. Moreover, for any given two points there is only a single line.

Description: The gesture consists of creating two points with the gesture "brief touch" and performing the "movement gesture", initiating the movement out of the radius of proximity of the points, starting before the first and finishing after the second point. This distinguishes the gesture of the "line segment gesture", as can be seen below. If there are already two points, the user may use them without needing to create new points.

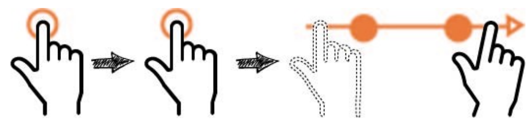

Figure 3. Line gesture[1].

Comparison: In the Geogebra and GeometryPad software, the user must first select the feature in the menu to construct the line and then briefly double touch on the screen in different areas to define the two points from the line. In the Sketchometry software, the user makes the line gesture on the device screen, and the created line has no points (i.e. you cannot move the line freely). In the Geogebra and GeometryPad software, you cannot construct a line without at least a point, which is different from the Sketchometry software. In the latter software, the user cannot identify that the line is an infinite set of geometric points.

Gesture: Semi-line

Base gesture: Brief touch, Movement

Conceptual definition: Semi-line is a subset of a line that has a limited end and the other goes to infinity.

Description: The gesture consists of creating two points with the gesture "brief touch" and performing the gesture movement so that the user traces a line within the radius of one of the points and goes over the other.

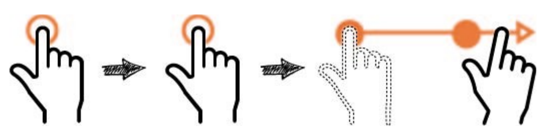

Figure 4. Semi-line gesture[2].

Comparison: Constructing a semi-line in the Geogebra and GeometryPad software is similar and the construction must be pre-selected in the menu. The user must make the first and the second points at different places on the screen, and constructing the second point determines the direction of the semi-line. In the Sketchometry software, the user should briefly touch the screen in different places to construct the points and then draw a line that goes over a point (to determine

1 http://www.icmc.usp.br/e/116ec
2 http://www.icmc.usp.br/e/86d95

the direction of the semi-line) and the boundary of the line ends at the other point. When the gesture is being done to construct the semi-line object in the Sketchometry software, it is clearly shown that the end of one side of the semi-line has a boundary and the other goes to infinity.

Gesture: Line segment

Base gesture: Brief touch, Movement

Conceptual definition: Line segment is the set of all points between two end points.

Description: The gesture consists of creating two points with the "brief touch gesture" and performing the gesture movement within the radius of proximity of the points.

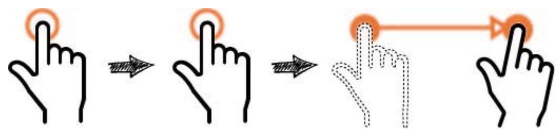

Figure 5. Line segment gesture.

Comparison: To create the line segment in the Geogebra and GeometryPad software, this feature must be selected in the menu. In both pieces of software, two points must be created on the screen in distinct places to construct the line segment. In the Sketchometry software, the user must create two points on the screen in different places and draw a line where the limits of this line have a short distance from the points (the line boundaries should go over the points). This gesture in Sketchometry makes it clear that the line segment is a set of all points between two end points, and in the Geogebra and GeometryPad software, the user may identify this concept only after constructing the line segment.

Gesture: Circumference

Base gesture: Brief touch, Rotate

Conceptual definition: A circumference is the set of all points that are equidistant from a given point. In other words, the circumference is the set of points on a circle (or plane) that are equidistant from its center.

Description: The circumference gesture consists of two steps. In the first step, the screen must be touched briefly to create the center point of the circumference. In the second step, the rotation gesture around the center point should be made, so that the points of this circumference are equidistant from the center.

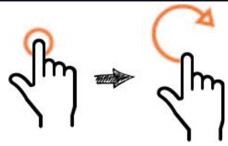

Figure 6. Circumference gesture[1].

Comparison: In the Geogebra and GeometryPad software, the construction of the circumference is similar, insofar as the feature must be selected before performing the construction. Thereafter, the user must briefly touch the screen twice; the first to define the center of the circumference and the second time to define the radius. In the Sketchometry software, the circumference is performed only with a rotation movement, without defining any point. Given the lack of a central point, the users may not identify that the circumference is the set of points that are equidistant from its center.

Gesture: Perpendicular

Base gesture: Press, Rotate

Conceptual definition: Given a line *r*, we say that line *s* is perpendicular to r if the angle between them is 90°.

Description: Two points of a line, previously constructed, must be selected and the half rotation gesture on it to create an angle of 90° must be used. By making this gesture, we mark the intersection of the lines and construct the angle of 90° between them.

Figure 7. Perpendicular gesture.

Comparison: In the GeometryPad software, the perpendicular line must first be selected in the menu. Thereafter, the user must briefly touch the line that he/she wishes to form the perpendicular. In the Geogebra software, the user must select the function on the menu to create the perpendicular and then briefly touch one of points on the line where the perpendicular must be created. The construction of this geometric object in Sketchometry is quite different from other software: the user must make the gesture in the shape of an "L" on the line where he/she wants to create the perpendicular. In the latter software, although this gesture is not intuitive, it is strongly related to the concept of 90° between two lines.

1 http://www.icmc.usp.br/e/3d2f5

Gesture: Parallel

Base gesture: Press, Movement

Conceptual definition: Given a line *r*, we say that line s is parallel to r if they are equidistant and hence never cross.

Description: The parallel is obtained from making the press gesture, which selects the reference line of the parallel, followed by the movement gesture, which fixes where the parallel will be located.

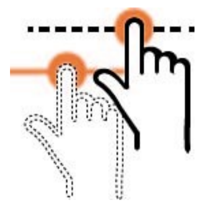

Figure 8. Parallel gesture.

Comparison: It was observed that the GeometryPad software does not have this feature. In the Geogebra software, the user must choose the construction of the parallel line previously on the menu and then touch the line where he/she wants to construct the parallel, and finally touch where the parallel must be located. In the Sketchometry software, the user must make the gesture in the shape of a "Z" horizontally on the line where he/she wants to create the parallel. In the latter software, the gesture is not intuitive and it is difficult to perform. Moreover, it is not related to the concepts of the parallel line.

4. GeoTouch: interactive geometry for mobile devices

Using the gestures dictionary developed in this work, we proposed the creation of an interactive geometry software for mobile devices with a multi-touch screen, called **GeoTouch** (Isotani et al., 2014; Reis et al., 2015). The initial version was developed using Android (v. 4.0) and it can be run on mobile devices such as 7 and 10 inch tablets or smartphones. Currently, GeoTouch is available *Google Play* and may be accessed at: https://play.google.com/store/apps/details?id=com.usp.icmc.geotouch.

GeoTouch has all the features that involve basic constructions of geometric objects to teach Euclidean geometry. Besides constructing these objects, it is also possible to manipulate (e.g. moving an object) and edit them (e.g. increase the size of a line or circumference). Other basic features have also been developed (e.g. New, Open, Save, View Cartesian Axis, Measure Distance). An illustrative video using the GeoTouch is available at: http://www.icmc.usp.br/e/185e3.

Figure 9 shows the initial screen of GeoTouch with and without the side menu. This menu (Figure 9 (a)) is accessed by an icon located in the top left hand corner and was developed so that the user may hide it when he/she wants to work in the drawing area (Figure 9 (b)) and shows it when he/she needs. It is important to note that the menu and icons are used for tasks *not related to constructing and editing geometric objects*. This enabled a significant reduction in the number of icons that is displayed, and proposed a natural way to interact with IG software.

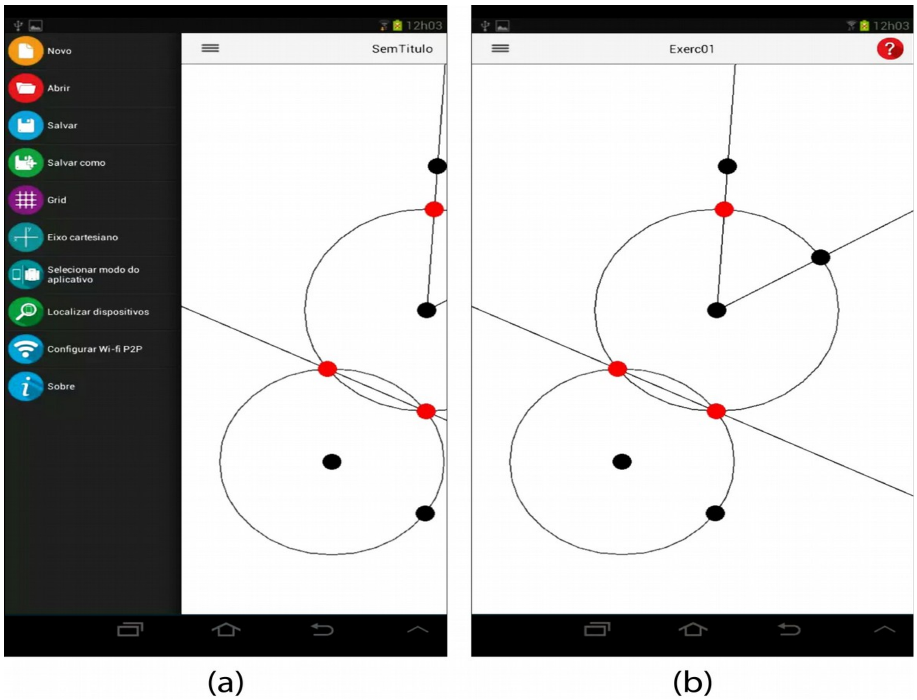

Figure 9. Initial screen of GeoTouch software.

Geometric objects are created from screen touches, and a set of these touches comprises the gesture as we presented in section 3. Students can check all gestures by touching the help button on the top right of the screen and a tutorial to create geometric objects will be displayed as shown in Figure 10.

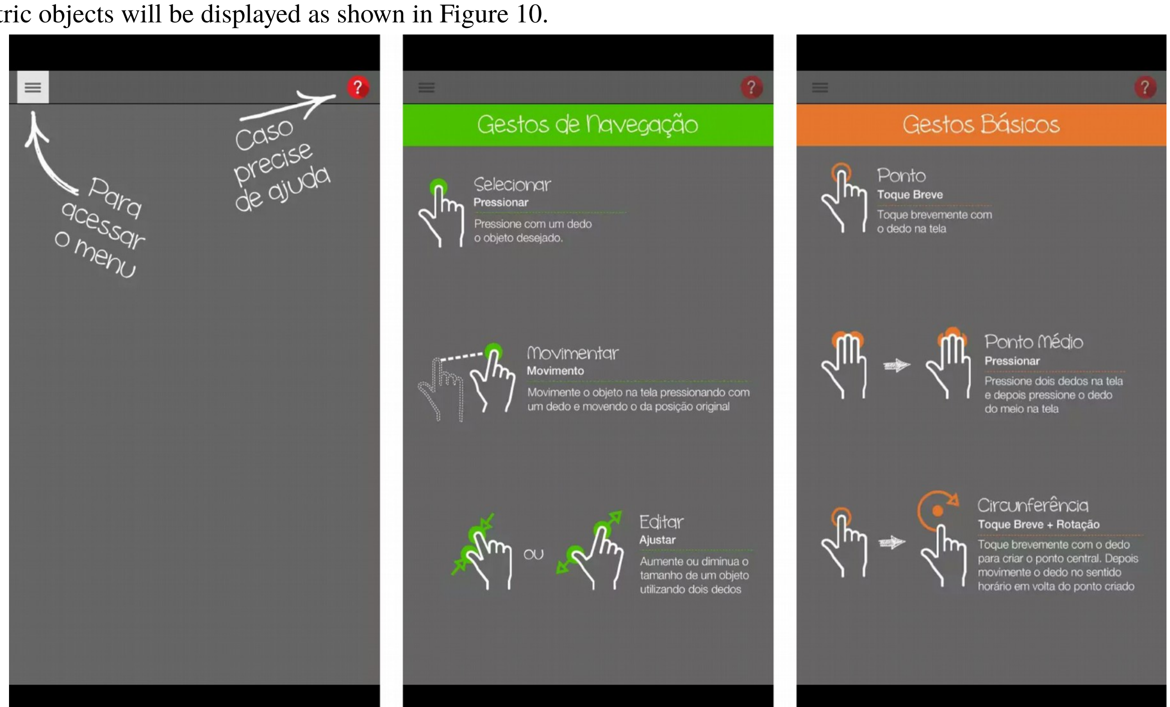

Figure 10. Tutorial to create geometric objects using gestures.

All screen touches are analysed by our gesture recognition algorithm responsible for identifying the type of touch (e.g. brief touch, movement or press) and decides which action will be executed (e.g. selection, movement or creation). A decision tree is adopted decide what actions should occur when there is an action in the interface. Part of this tree is shown in Figure 11. In the figure a touch is found in three types of actions: *Screen Touched*, *Moving fingers across the screen* and *Remove fingers from the screen*. When the user touch the screen, our algorithm checks if there is already a geometric object present in the same coordinate. If the answer is yes, then it will select the object or indicate the user wants to build another object that is connected with the current one. If there is no object, then the algorithm will prepare to create a new object (e.g. point) in the touched coordinate of the screen. If the user moves his/her finger(s), the algorithm will check if there is any object which has been selected to have this movement; otherwise, the algorithm will try to find a pattern based on our gesture dictionary to create a circumference, line, line segment, semi-line, parallel, midpoint or perpendicular. Finally, when the user remove his/her finger from the screen the algorithm will check if the pattern matching returns an object to be built, and thus, call a method to create a specific geometric objects. If there is no matching, nothing is done.

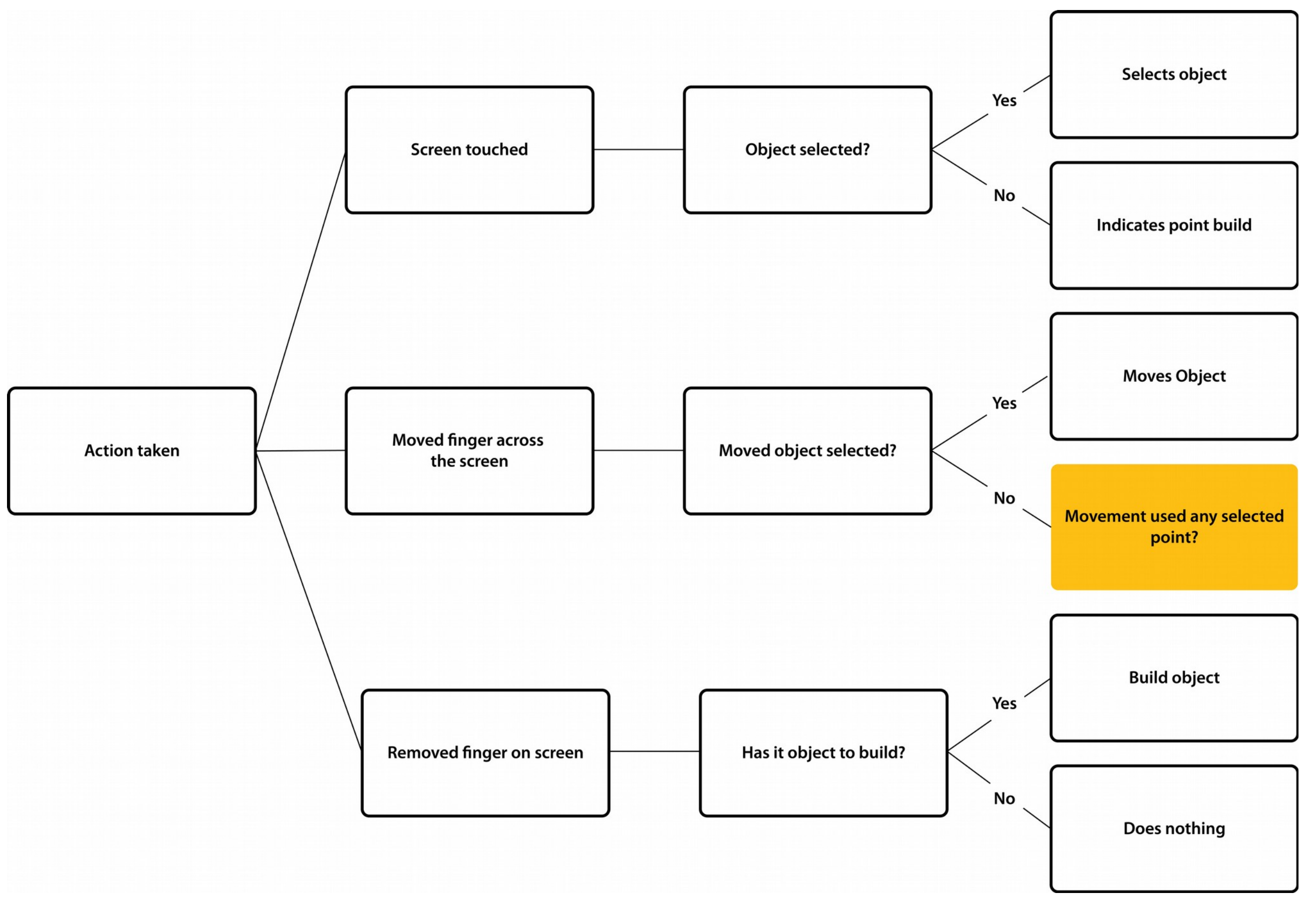

Figure 11. Part of the decision tree to decide what actions should be perfomed by GeoTouch.

5. Evaluation

An evaluation was carried out in order to verify the usability of our interaction model and interface using GeoTouch as well as to compare them with the standard format of interfaces and interactions utilized by others IGS. The technique selected for the evaluation was the usability analysis using the heuristic evaluation defined by Nielsen (Nielsen, 1993). Nielsen defines the usability criterion as a set of factors that analyses how well a person can interact with a computer system. The aim of this criteria is to find usability problems which help to evaluate the interaction of the system interface with the user and to find solutions to develop improved human-machine interfaces and interactions (Baker et al., 2001). The 10 heuristics proposed by Nielsen are: (H01) system status visibility; (H02) Correspondence between the system and the real world; (H03) User control and freedom; (H04) Consistency and standards; (H05) Error prevention; (H06) Recognition rather than memory; (H07) Flexibility and efficiency of use; (H08) Aesthetic and minimalist design; (H09) user to help identify, diagnose and correct errors; (H10) Help and Documentation.

According to Dringus (1995), the heuristic evaluation can be applied in educational software interfaces, so as to identify several issues that may affect learning. Heuristic evaluation is often used by researchers seeking to evaluate the usability of an interface in a short time and at a low cost. It is usually conducted with the participation of 3-5 evaluators, because according to many researchers, this number of evaluators is sufficient to identify around 80% of all usability problems. The process can also be well documented and has a high degree of replicability (Nielsen and Landauer, 1993; Hvannberg & Lárusdóttir, 2007).

GeoTouch was analysed together with the other three IGS available for mobile devices presented in Section 2. In order to conduct the heuristic evaluation, five senior researchers with several years of experience in software engineering, usability, and heuristic evaluation techniques were invited to participate. Each of these researchers evaluated each software for at least two hours, totalling forty hours of evaluation. All the evaluation activities were monitored by at least one of the authors of this paper and notes concerning the problems of interaction were taken. It is important to observe that no action from the observer was taken to not add any bias in the evaluators analysis. To perform the evaluation, one 7-inch tablet running the Android operating system was used. A computer was also used, so that the evaluators could answer a questionnaire to point out the usability problems. The test was performed in four phases as shown in Figure 12. A printed guidebook which contains the order and list of activities to be executed using each software was provided.

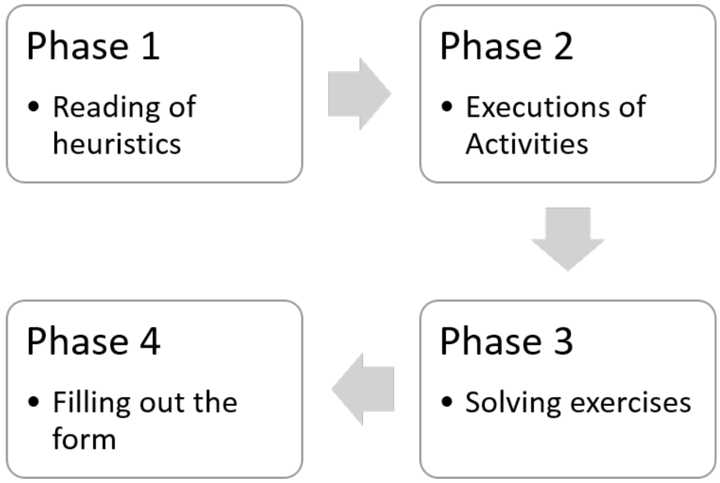

Figure 12. Heuristic evaluation phases of the software proposed by the evaluators.

The first phase consisted of the evaluators reading the ten heuristics and asking any question before starting the evaluation process. Then, the evaluators went on to the second phase, which was to carry out a set of 38 pre-defined activities and make them available as a structured guidebook. In this guidebook, each evaluator had to apply the ten usability heuristics to each item in the list of activities. These activities were grouped by type of functionality that the software provides (e.g. point, midpoint, line, line segment, semi-line, perpendicular, parallel, circumference, measured angle and intersection).

For each feature, activities involving the construction of basic geometric objects (e.g. creating points, midpoints or lines), handling these objects (e.g. moving the points), and editing (e.g. changing the point size) were requested to be carried out. In addition, standard tasks were proposed, such as undo a step of the construction, clear full screen, open a document, and save a document.

In the third phase, the evaluators were asked to solve a list of exercises related to the geometric construction of basic and intermediate levels. The purpose of this step was to simulate the use of software in geometry learning activities. Finally, in the fourth phase the evaluators completed the form on usability heuristics, according to the degree of severity of the problem as proposed by Nielsen (1993).

After the software was evaluated, a list was created for each software containing all the problems found and classified into heuristics. Note that due to the complexity of some activities, the errors and problems may have been classified into more than one heuristic, thus affecting the frequency of errors/problems reported.

6. Results and discussion

Initially, to provide an overview of the results, the Figure 13 highlights the frequency of problems found in each one of the evaluated heuristics and for each one of the evaluated software. On the x-axis the abbreviation of heuristics is shown and on the y-axis the amount of times that the problem was found is shown. In the figure, it is observed that GeoTouch was the

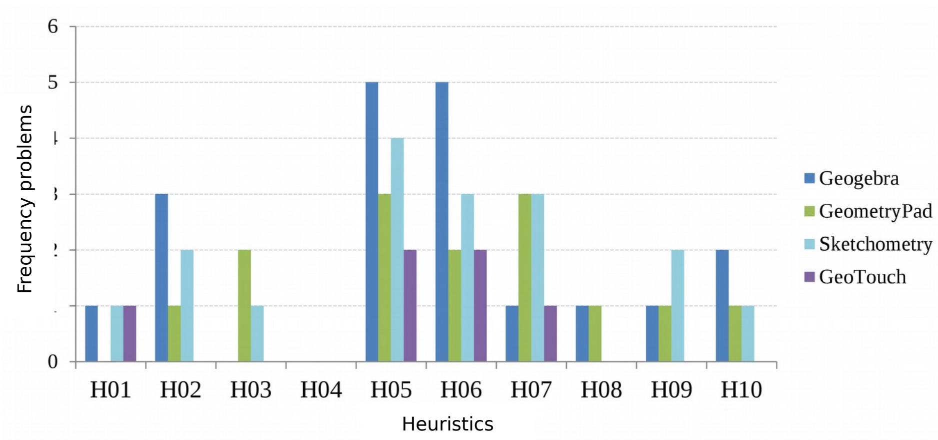

Figure 13. Frequency of problems found by heuristic and evaluated softwares.

system with fewer number of errors/problems in usability according to experts. GeoGebra was the system that had the highest number of errors/problems.

It also possible to observe that none of the evaluated software presented problems related to consistency and standards (H04), which shows a certain maturity of their interfaces since they are all adapted versions of interactive geometry software developed for desktop computers. Nevertheless, in all the evaluated software, a considerable number of problems in the following heuristics were found: H05: problems related to preventing errors; H06: recognition instead of recall; and H07: flexibility and efficiency of use.

The problem related to preventing errors (H05) covers the presentation of non-standard icons (i.e. icons commonly used), lack of description of the functions, error messages and step-by-step procedures to perform some activities. That is, the interface design and the way of exposing their elements did not prevent the evaluators from making mistakes when using the software.

The Geogebra and Geometry Pad did not present any shortcut key, affecting its flexibility and efficiency. However, the only software that presented problems related to aesthetics and minimalist design (H08) was the Geometry Pad, because it presented irrelevant information on the interface, such as the export feature located in the title bar. In addition, the Geometry Pad had a greater number of problems related to the user's control and freedom (H03), because it was not possible for the user to delete multiple objects at the same time. The Geogebra software had more problems in heuristics

H02, H06, and H10, compared with other software. These problems are related to (i) words or elements on the screen that are not familiar to users; (ii) difficulty remembering how a feature was used the last time; and (iii) the lack of software documentation.

Regarding GeoTouch, one of the identified problem was the lack of information (H01) about what object is going to be created while the gesture is being executed. For example, the gesture to construct a line may be similar to the gesture to construct a semi-line and line segment. In this case, if the information about pattern matching is provided in execution time, users could avoid mistakes. For evaluators, it is important that a message on the screen appears identifying which object are going to be constructed while the gestures are being performed. Another problem found in GeoTouch was the difficulty to remember the gestures of undo and remake (H06), leading one of the evaluators to consult the documentation several times. Finally, the last problem identified by the evaluators was the lack of shortcuts for some functions of the side menu, such as new, save, save as, grid and cartesian axis (H07).

To measure how much these problems may affect the usability of each software, the degree of severity of each problem was studied. Figure 14 shows the degrees of severity attributed to problems identified by specialists in each heuristic and for each software. The graphs are presented as polygons of 10 sides in the shape of a spider web, where the vertex represents the usability criteria (from H01 to H10). The degree of severity for each criterion is indicated filling in the polygon (from inside out). According to Nielsen, the degree of severity may be divided into four categories [15]: 0 - disagree that it is a usability problem (0% severity); 1 - **Cosmetic**: superficial problem that must be solved only when there is time and money (25% severity); 2 - **Mild**: problem of low usability and has a low priority to be solved (50% severity); 3 - **Serious**: problem of severe usability and has priority and should be corrected as soon as possible (75% severity) and; 4 - **Critical**: has top priority and needs immediate intervention to prevent users from inappropriately using the interface (100% severity). Thus, if the degree of severity of the H01 criterion is mild for example, it fills 25% of this vertex.

Concerning Geogebra, it can be observed in Figure 14 that the H02 (correspondence between the system and the real world) has a higher degree of severity (approximately 75%), compared to the other problems. It indicates that this problem is serious and needs immediate attention. Problems classified in heuristics H05 (prevention of errors) and H06 (recognition instead of recall) are classified with the second highest degree of severity (approximately 50%) and need to be analysed in future software releases. No problems were found related to the H03 heuristics (user control and freedom) and H04 (consistency and standards).

In the Geometry Pad, it is observed that the heuristic related to help identify, diagnose and correct errors (H09) has problems with greater severity, with approximately 75% of severity and therefore its solution is considered urgent. The heuristics related to the control and freedom of the user (H03), prevention of errors (H05), recognition instead of recall (H06), flexibility and efficiency of use (H07), and help and documentation (H10) presented approximately 50% of severity and they need attention. The H08 heuristic presented problems with the lowest degree of severity in the system, corresponding to around 25% of severity, where this is considered low priority. It should be mentioned that despite the frequency of errors in this software being less than the GeoGebra (Figure 13), the severity of these errors is higher (Figure 14) indicating that the evaluators had more difficulty in interacting to perform the proposed tasks.

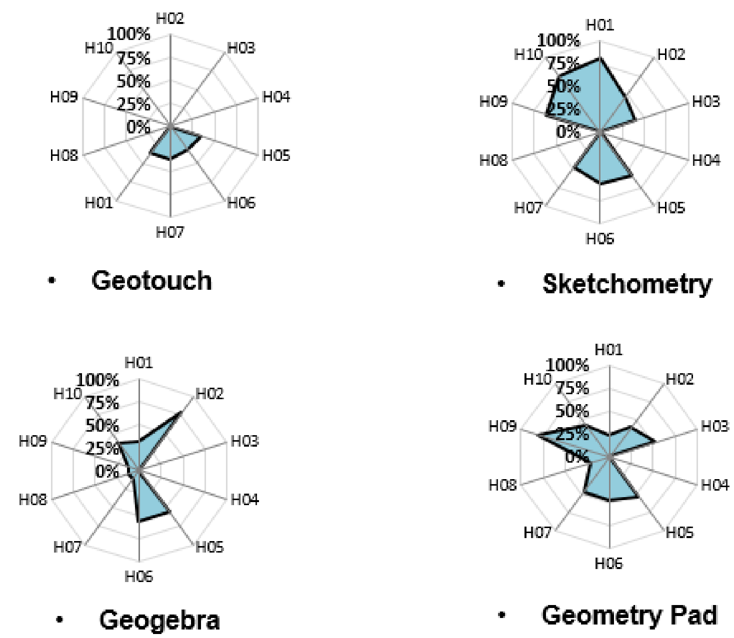

Figure 14. Degree of severity between heuristics.

In Sketchometry, the heuristic related to the system status visibility (H01) and help and documentation (H10) showed a high degree of severity, corresponding to around 75%. Therefore, the heuristics relating the correspondence between the system and the real world (H02), prevention of errors (H05), recognition instead of recall (H06), flexibility and efficiency of use (H07), and help to identify, diagnose, and correct errors (H09) presented severity of approximately 50%. Although this software provide a lower frequency of errors compared to Geogeobra and the Geometry Pad, most of them have a high degree of severity and may affect novice users in terms of using them or adopting them.

Finally, in GeoTouch, problems related to the system status visibility (H01), prevention of errors (H05), recognition instead of recall (H06), and flexibility and efficiency of use (H07) were found. Nevertheless, all of them were classified as cosmetic problems or mild severity (between 25% and 50% of severity). These results indicate that GeoTouch has the lowest error rate and the lowest severity of these errors among the four evaluated pieces of software (Figure 13 and 14).

7. Conclusions

The interaction model implemented by most IGS is based on menus and icons to represent the objects or operations on them. Besides the fact that this model is not suitable for mobile devices (with multi-touch screens), several studies indicate that IG interfaces with many icons may raise doubts, cause interaction errors, and consequently frustrate learners during the learning process (Schimpf and Spannagel, 2011; Reis *et al.*, 2012). Thus, this paper proposed the creation of a gestures dictionary as a means of interacting with the IGS without the need of menu of icons to make geometric constructions.

To define and develop these gestures, firstly, we conducted interviews with geometry teaching experts to identify the basic geometric concepts used for geometric constructions performed in teaching and learning situations. Then, the gestures that have a direct association with geometric concepts were defined. To the best of our knowledge, this is one of the first initiatives to define and standardize interaction gestures for IGS on multi-touch screens. This belief is reinforced by the results reported in a thorough systematic review of the literature, in which few studies related to the IGS on mobile devices were found, indicating a lack of research in the definition of gestural interfaces and interaction models for IGS (Laborde, 2007; Reis et al., 2016).

Based on the definition of the gestures dictionary, GeoTouch was developed, an IGS that implements all the basic features to construct geometric objects. To evaluate the GeoTouch interface, a usability test was performed. We also compared the results with other three IGS: Geogebra, Geometry Pad and Sketchometry. We applied a set of activities to evaluate, identify, and analyze the frequency of interaction errors and the severity of these errors. The results suggest that our proposal of new interaction model and interface, based on the gestures dictionary, is more appropriate when compared to the IGS for mobile devices already on the market.

We hope that GeoTouch as well our gesture dictionary become a reference to help developers to create better IGS interfaces for mobile devices as well as help students and teachers in the teaching and learning process of geometry using smartphones. For future work, we intend to conduct experiments in real classroom environments with students and teachers to find out the learning benefits of using GeoTouch comparing with traditional methods of using IGS in desktops.

Acknowledgements

We thank the support of CNPq, CAPES and FAPESP for sponsoring this research. We also wish to thank the evaluators who voluntarily spent their time and effort during the evaluation process.

References

Baker, K.; Greenberg, S.; Gutwin, C. (2001). Heuristic Evaluation of Groupware Based on the Mechanics of Collaboration. In: Proceedings of the IFIP International Conference on Engineering for Human-Computer Interaction, p. 123–140.

Blagojevic, R., Chen, X., Tan, R., Sheehan, R., & Plimmer, B. (2012). Using tangible drawing tools on a capacitive multi-touch display. Proceedings of the Annual BCS Interaction Specialist Group Conference on People and Computers, p. 315–320.

Blanke, D.; Schneider, G. (2011). TOM - A Multi-touch System for Learning Math. International Conference on Computer Supported Education, p. 199–206

Borges, S. S. ; Reis, H. M. ; Marques, L. B. ; Durelli, V. H. S. ; Bittencourt, I. I. ; Jaques, P. ; Isotani, S. (2016) Reduced GUI for an interactive geometry software: Does it affect students' performance?. Computers in Human Behavior, v. 54, p. 124-133.

Dringus, L. P. (1995). An iterative usability evaluation procedure for interactive online courses. Journal of Interactive Instruction Development, 7(4), p. 10–14.

Ehmann, M., Gerhauser, M., Miller, C., & Wassermann, A. (2013). Sketchometry and jsxgraph - dynamic geometry for mobile devices. South Bohemia Mathematical Letters, 21(1), 1-7.

Hinrichs, U.; Carpendale, S. (2011). Gestures in the wild: studying multi-touch gesture sequences on interactive tabletop exhibits. In Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, p. 3023-3032.

Hvannberg, E., Law, E., & Lárusdóttir, M. (2007) Heuristic Evaluation: Comparing Ways of Finding and Reporting Usability Problems. Interacting with Computers, 19 (2), 225-240.

Isotani, S.; Brandão, L. O. (2008). An algorithm for automatic checking of exercises in a dynamic geometry system: iGeom. Computers and Education 51(3), p. 1283-1303.

Isotani, S., Pedro, L. Z., Reis, H. M., Borges, S. S., Lopes, A. M., Souza, J. P., ... & Brandão, L. O. (2014). Interactive geometry goes mobile with GeoTouch. In Proceedings of the IEEE International Conference on Advanced Learning Technologies, 181-185.

Jackiw, N. (1995). The Geometer’s Sketchpad v3.0. Berkeley: Key Curriculum Press.

Kortenkamp, U.; Dohrmann, C. (2010). User interface design for dynamic geometry software. Acta Didactica Napocensia, 3(2). Available at http://dppd.ubbcluj.ro/adn/article_3_2_6.pdf

Kortenkamp, U.; Materlik, D. (2004). Geometry teaching in wireless classroom environments using Java and J2ME. Science of Computer Programming, 53(1), p. 71–85.

Kortenkamp, U.; and Richter-Gebert, J. (2004) Using automatic theorem proving to improve the usability of geometry software. Procedings of the Mathematical User-Interfaces Workshop. Available at: http://kortenkamps.net/papers/2004/ATP-UI-article.pdf

Laborde, C. (2007). The role and uses of technologies in mathematics classrooms: Between challenge and modus vivendi. Canadian Journal of Science, Mathematics and Technology Education, 7(1), p. 68–92.

Nacenta, M. A.; Kamber, Y.; Qiang, Y.; Kristensson, P. O. (2013). Memorability of predesigned and user-defined gesture sets. In: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI), p. 1099–1108.

Ng, O. L., & Sinclair, N. (2015). Young children reasoning about symmetry in a dynamic geometry environment. ZDM Mathematics Education, 47(3), 421-434.

Nielsen, J. (1993). Usability Engineering. San Francisco, CA.

Nielsen, J., and Landauer, T. K. (1993) A mathematical model of the finding of usability problems. Proceedings of ACM INTERCHI Conference, p. 206-213.

Reis, H. M.; Borges, S. S.; Durelli, V. H. S.; Moro, L. F. S.; Et Al. (2012). Towards Reducing Cognitive Load and Enhancing Usability through a Reduced Graphical User Interface for a Dynamic Geometry System: An Experimental Study. In: IEEE International Symposium on Multimedia, p. 445–450.

Reis, H. M.; Brandão L. O; Brandão, A. A. F.; Isotani, S. (2016). Interaction Interfaces in Interactive Geometry Software: Are we Exploring New Devices and Possibilities?. International Journal of Learning Technology, 11(4), (in press).

Reis, H. M., Isotani, S., Gasparini, I., & Mizoguchi, R. (2015). A Dictionary of Gestures for Multitouch-based Interactive Geometry Software. In proceedings of the IEEE International Conference on Advanced Learning Technologies, 102-104.

Reis, H., Isotani, S., Brandão, L., Cruz, W., Brandão, A., & Filho, R. (2015a). Concepção de uma Família de Gestos para Construção de Objetos Geométricos e sua Utilização em um Sistema de Geometria Interativa para Dispositivos Móveis: GeoTouch. Revista Brasileira de Informática na Educação, 23(02), 206-224.

Roanes-Lozano, E.; Roanes-Macías, E.; Villar-Mena, M. (2003). A Bridge Between Dynamic Geometry And Computer Algebra. Mathematical And Computer Modelling, 37(9-10), p. 1005–1028.

Santos, E. T., Lourenzoni, L., & de Oliveira, A. L. L. (2006). RISKO, an Educational Geometry Drawing Software With an Innovative Interface. Journal for Geometry and Graphics, 10(1), 115-124.

Schimpf, F.; Spannagel, C. (2011). Reducing The Graphical User Interface Of A Dynamic Geometry System. ZDM, 43(3), p. 389–397.

Schwartz, J. L. (1995). The right size byte: Reflections on educational software designer. In D. Persinks, J. Schwartz,M.West and S.Wiske (Eds), Software Goes to School (pp. 172–182). New York: Oxford University Press.

Ulrich Kortenkamp, (1999) Foundations of Dynamic Geometry, Ph.D. thesis, ETH Zurich, 11, http://kortenkamps.net/papers/diss.pdf.

Vitale, J. M.; Swart, M. I.; Black, J. B. (2014). Integrating Intuitive And Novel Grounded Concepts In A Dynamic Geometry Learning Environment. Computers and Education, 72(March), p. 231–248.

Yerushalmy (1999). Making exploration visible: On software design and school algebra curriculum. International Journal of Computers for Mathematical Learning 4(2-3): 169-189.