GTCards: A Video Game for Learning Geometric Transformations

A cards-based video game for learning geometric transformations in higher education

José Saúl, González-Campos*
Estudis d'Informàtica, Multimèdia i Telecomunicació, Universitat Oberta de Catalunya
jsaulg@uoc.edu

Joan, Arnedo-Moreno
Estudis d'Informàtica, Multimèdia i Telecomunicació, Universitat Oberta de Catalunya
jarnedo@uoc.edu

Jórdi, Sánchez-Navarro
Estudis d'Informàtica, Multimèdia i Telecomunicació, Universitat Oberta de Catalunya
jsancheznav@uoc.edu

ABSTRACT

Geometric transformations are a relevant subject taught at different educational levels, from elementary to higher education. This work focuses on designing an educational tool suitable to help students learn and practice transformations in the specific context of an undergraduate computer graphics course. Even when the subject of study is the same, the learning objectives are different in other educational contexts, such as elementary or middle education. We propose in this work a cards-based video game founded in a model that constrains the input variables of learning exercises without sacrificing the diversity of scenarios and difficulty levels. Incorporating this model into a video game is congruent because it privileges the visual over the numerical information that defines a specific "before and after" style of learning exercise. The game mechanics were designed around a set of requirements thought to meet the student's needs, such as learning by experimenting, instant feedback, and matrix representation, among other criteria. The video game was developed in the Unity engine, and it is ready to be tested in the classroom during the next phase of our research. We expect to gather qualitative and quantitative data to evaluate its potential as a learning tool for undergraduate students coursing computer graphics.

CCS CONCEPTS

• **Applied computing** → Education; Computer-assisted instruction; Computers in other domains; Personal computers and PC applications; Computer games; • **Mathematics of computing** → Mathematical analysis; Numerical analysis; Computation of transforms.

KEYWORDS

Serious games, geometric transformations, higher education

TEEM'21, October 26–29, 2021, Barcelona, Spain

ACM ISBN 978-1-4503-9066-8/21/10…$15.00
https://doi.org/10.1145/3486011.3486445

ACM Reference Format:
José Saúl, González-Campos*, Joan, Arnedo-Moreno, and Jórdi, Sánchez-Navarro. 2021. GTCards: A Video Game for Learning Geometric Transformations: A cards-based video game for learning geometric transformations in higher education. In *Ninth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'21) (TEEM'21), October 26–29, 2021, Barcelona, Spain.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3486011.3486445

1 INTRODUCTION

Geometric Transformations (GTs) are an important topic in education [1-3] and diverse professional fields. As a subject of study, GTs include translations, rotations, scales, and reflections as they are applied to real and virtual objects. This topic is taught at different levels, from elementary to higher education, with customized goals depending on the students' academic backgrounds. In the specific case of learning GTs as part of a computer graphics course, transformations are involved in the whole process of rendering a scene, like modeling, viewing, and projecting, among other structured steps in the rendering pipeline [4]. In this context, students must have a strong understanding of GTs to later utilize them in core processes in computer graphics. Unfortunately, not all students have previous experience or intuition to understand transformations easily and usually struggle to apply and dominate them; recognizing GTs implies developing cognitive abilities known as visual-spatial, which is not always a trivial task [5]. We detected an opportunity to address this issue, especially after verifying that most previous works are aimed at elementary or middle education, with different learning objectives than higher education, even with the same subject of study. Commonly, GTs are not represented with matrices at those lower levels of education and are almost always 2D oriented. Therefore, specialized tools are needed for computer graphics students that help them to understand both single and composite transformations (multiple GTs applied ordered, in a sequence). Furthermore, students need plenty of visual support to understand the meaning and application of GTs; most of the time, the matrix format and the analytical procedures alone are not enough. GTs are visual by nature because they correspond to our human perception of their effect on real-world objects. Hence, their learning implies abilities and strategies in both aspects, the visual and analytical [6]. Specific training may consist of exercises demanding to figure out a sequence of steps to transform an object, given an initial and a final state (this paper refers to this exercise style as "before and after"); this type of exercise was classified as the most challenging learning task involving GTs [7]. The rest of this paper is structured as follows: Section 2 summarizes the learning strategies proposed in previous works to teach GTs. Section 3 describes our proposed educational tool's requirements, provides a model to characterize the exercises' solutions, and justifies a series of constraints in the exercises definition that enhance its visual interpretation. Section 4 describes the game proposal in detail and, finally, Section 5 presents the conclusions and future work.

2 TEACHING GEOMETRIC TRANSFORMATIONS

Teachers have used various strategies and means to help students learn GTs according to students' educational level and background. For instance, in elementary and middle education, the emphasis is placed more on an intuitive concept of GTs, with a practical approach that relates this subject to the real world [8, 9] or the cultural context of the student [10, 11]. In this category are included numerous studies appearing in journals on mathematics education. Furthermore, in addition to the traditional teaching elements such as whiteboards, presentations, and multimedia materials, some authors have proposed additional tools, such as handheld calculators [12]. More recently, the so-called Dynamic Geometry Software (DGS), a computer program specially designed for interactive creation and manipulation of geometric constructions, has a ubiquitous presence in the classroom. Among others, software in this category includes Cabri, Geometer's Sketchpad, and Geogebra, and ranges from free of use to different licensing terms. Various studies have analyzed the impact of DGS from different perspectives [15-17]. Among several mathematical topics, this software has been used as a learning tool for teaching GTs [2, 13, 14], mainly in middle education. Learning GTs becomes more centered in the mathematical foundations and formal properties of translations, rotations, scales, and reflections in higher education. In the particular context of studying GTs as part of a computer graphics course, the standard way of study them is using their matrix representation. This treatment is compatible with the internal work of the graphics accelerator hardware and many low-level programming APIs that handle 3D graphics. We have detected few studies primarily focused on learning GTs in this context. Some studies, for instance, approached the computer graphics field in general, addressing GTs among many other subjects [18]. Other studies implemented didactical tools specifically for GTs. For example, one implemented an interactive tool that displayed an editable transformation matrix and simultaneously displayed a 3D model showing the visual effect [19]. Another study developed a gamified tool that used cards and aimed to help learn affine transformations by input values in the cards' fields and look at the results applied to a 3D model [20].

3 REQUIREMENTS AND EXERCISE MODEL

The following list shows a set of requirements that we consider relevant for the educational tool proposed in this work. They are aligned with the specific context of students learning GTs in a computer graphics course.

- **GTs are represented visually:** students should see the effect a transformation causes when applied to an object instead of only working with abstract mathematical entities such as vectors and matrices [21].

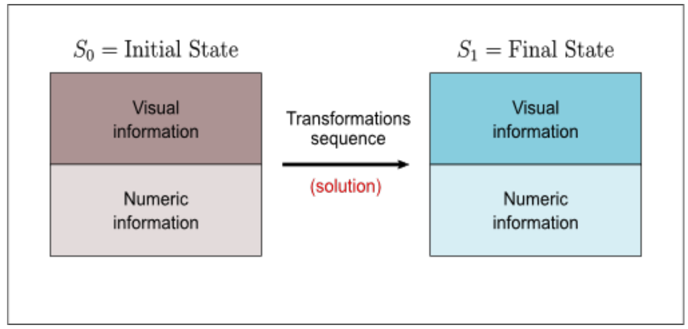

Figure 1: Exercise model.

- **Students dynamically work with the composition of transformations:** This requirement is about the free manipulation of a composition by adding, removing, or reordering the transformations and seeing the visual results [16, 21].
- **There is a connection between the transformations and their matrix representation:** Instant feedback about any single or composite transformation matrix values when solving an exercise [19].
- **Promote the students' awareness of the non-commutative propriety of composite transformations:** Students should see the effect of altering the order of transformations in a composition (sequence) [16, 21].
- **Provide a mechanism for adjusting the level of complexity of the learning exercises:** This is basic to keep students motivated to learn. Overwhelming them with early complex exercises may discourage them from learning [22].
- **Useful exercises can be derived from simple base cases:** The underlying concepts to understand and efficiently manipulate GTs can be exposed with well-selected exercises [23, 24].
- **The environment stimulates learning by doing:** The tool interface should encourage students to experiment instead of waiting for explanations, instructions, or a step-by-step solution [16].
- **The educational tool preferably has a game interface:** While this is only a choice, a ludic approach is preferable instead of a "lecturer" one. This is in alignment with a tool that favors the student's engagement in the learning process [25-27].

We developed a general model to represent the exercises generated by the game. This model is based on the "before and after" exercise's style mentioned earlier in this paper. In general, to solve this kind of exercise and avoid ambiguity (different interpretations for the same problem), data must be provided to the student that accurately describes the initial and the final states. This data can be split into two forms, visual and numerical. This condition is represented in Figure 1. Due to our human visual perception of the effect transformations have on real-world objects, it is unlikely that a student can solve a "before and after" exercise with only numerical data and with no graphical representation. On the other hand, it is not easy to define this kind of exercise exclusively with visual information [6].

Table 1: Selection of transformations

Transformations	Axes	Restrictions	Detailed options
Rotations	x, y, z	Multiples of 45 degrees	Rx(±45˚), Rx(±90˚), Rx(±135˚), Rx(180˚) Ry(±45˚), Ry(±90˚), Ry(±135˚), Ry(180˚) Rz(±45˚), Rz(±90˚), Rz(±135˚), Rz(180˚)
Reflections (Mirrors)	x, y, z	None	Mx, My, Mz
Scales	x, y, z	Double, Half	Sx(2.0), Sx(0.5), Sy(2.0), Sy(0.5) Sz(2.0), Sz(0.5)
Translations	x, y, z	Fixed distance	Tx(±1.0), Ty(±1.0), Tz(±1.0)

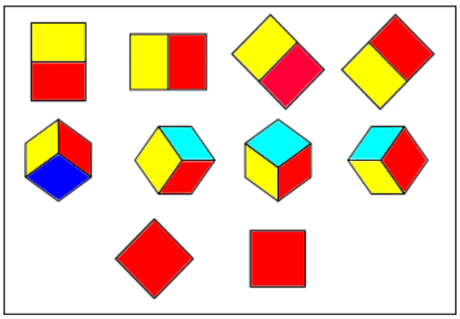

Figure 2: Selection of symmetric views

Therefore, our model establishes that most of the data is visual and restricts to a minimum the numerical one. To achieve this requirement, the model removes the continuous nature of transformations parameters and makes them discrete, leaving only a few options allowed in any exercise so that most of the numerical data can be inferred from the visual (graphical) data. Table 1 summarizes the total options considered in our model to generate exercises.

Even with the proposed restrictions in place, many views of an object can still be derived by combining the transformations in Table 1. Therefore, it is necessary to place additional constraints on the views to guarantee that only highly symmetrical ones are considered for the exercises. The student will infer the transformations involved in an exercise with symmetrical views easier than asymmetrical ones [28]. Figure 2 shows the selected views that are incorporated into the learning exercises defined by our model. Also, it is desirable to operate on a highly symmetrical object to take advantage of the constraints about symmetry; this is why the selected 3D object transformed in any exercise is a cube, as shown in Figure 2. Our model follows a uniform pattern to represent any solution; this allows us to quantify the total possible exercises that can be defined with all the constraints put in place. All possible combinations of the transformations in Table 1 (and restricted to generate the symmetric views in Figure 2) nearly total 350,000 exercises available (349,920). While not infinite, this number allows exercises to feel not repetitive and provides many degrees of complexity to be chosen.

4 GAME PROPOSAL

This section outlines a game prototype developed in Unity engine [29] that implements the features identified in section 3; it includes the description of the main interface, mechanics, adjustment of difficulty levels, and game elements design.

4.1 Main game interface

The main game screen consists of four zones: (1) Goal zone. In this zone, a cube is visualized with some applied transformations on it. The challenge for the player is to figure out what sequence of cards will correspond to this "goal state". (2) Player zone. Here, a cube in the initial conditions (with no applied transformations) is depicted. As soon as the player starts building a sequence of transformations, their cumulative effect will be seen on this cube. (3) Deck. This zone is reserved for the cards available to build a sequence (or steps) of transformations. (4) Sequence. Here, the selected cards are placed in the specific order chosen by the player, one by one.

4.2 Visual elements

Most of the information necessary to solve a goal is given as part of its visual representation. The following list summarizes the visual elements incorporated in the game interface that help the player interpret the goal transformations correctly.

- Scales are intuitively recognized by looking at the cube shape. It is not hard to differentiate only three scale possibilities: no scale, double, and half. Additionally, a colored "glow" at any edge in the cube will highlight a scale in that specific direction. The slight perspective projection (no orthographic) in the scene will help visualize this "glow feature" even in the depth Z direction.
- Scales are also highlighted by a geometric pattern in the cube faces because that pattern will look distorted in the presence of scales.
- Rotations are better recognized by giving a different color to any face of the cube.
- Rotations and reflections (and, to some extent, scales) are better recognized by including a visual representation of the system coordinates axes (additionally, each axis is colored differently).
- Translations are easily noted in the X-Y plane, but, unfortunately, it is not the case in the Z direction (depth). So, in this case, the perspective projection can help the player to be aware (to some extent) that the cube is translated in the Z direction by perceiving it farther or nearer. Additionally, a transparent colored panel in the X-Y plane at Z=0 highlights any cube's position at the negative Z-axis.

4.3 Game mechanics

First, the player is presented with a fixed number of cards in the "deck zone"; each card is labeled with a specific geometric transformation. The cards are not listed in any particular order. Then, the player can click and drag any card to the "sequence zone" until every slot is occupied (the sequence is complete). When the sequence is full, the game engine starts an animation that applies each transformation in the sequence, step by step, to the cube. At the same time, the answer is evaluated, and instant feedback is provided to the player about the sequence correctness. At this point, if the sequence was correct, the player's score is updated, and a new challenge (goal) is proposed. Otherwise (if the player fails), the same scene is loaded (with the same cards in the deck zone). To incentive the player to actively think for a solution, the highest score is given when the player solves the goal at the first attempt. After that, further attempts decrease the reward even when the answer is correct. This overall strategy intends to minimize the situation where only a random order of cards is chosen several times until they fit the goal state, which would have a minimal (or even no one) learning impact. Additionally, a pressure factor to keep the interest and dynamics of the player experience is the elapsed time; the player has a time limit to provide the right solution. However, this component is handled with caution because it is not desirable that the player selects a random choice just to avoid the time penalization. The game has various levels of increasing difficulty, which can be unlocked after completing lower levels in the complexity spectrum offered.

4.4 Difficulty levels

The proposed mechanics allows a great extent of flexibility in the degree of difficulty of the learning exercises. Difficulty can be adjusted with the following elements: (1) The ratio between the number of cards in the deck zone and the number of steps in the solution directly impacts difficulty. For example, a scenario with three slots in the sequence and three cards in the deck will have less complexity than three slots in the sequence and ten cards in the deck. A simple permutations comparison highlights the significant difference between these two cases as follows:

Permutations: ${}_{\mathrm{n}}P_{\mathrm{r}} = \frac{\mathrm{n}!}{(\mathrm{n}-\mathrm{r})!}$

Case 1: ${}_{3}P_{3} = \frac{3!}{(3-3)!} = 6$

Case 2: ${}_{10}P_{3} = \frac{10!}{(10-3)!} = 720$ (2) The types of transformations needed in a solution modify the overall difficulty. A controlled degree of complexity can be achieved by restricting the transformations applied to the "goal" cube. For example, a simple case requiring only translations is less complicated than other requiring translations, rotations, and scales. Also, subtle differences in difficulty are possible even within the same transformation. For example, variations in the degrees and axes of rotations can lead to different levels of complexity.

(3) The time available to solve an exercise also impacts the difficulty of solving it. However, this element is not considered much relevant because, to some extent, it is in opposition to the learning objective. Instead of acting based on time pressure, the player should carefully think, ponder, and analyze the given goal.

Figure 3: Game screen showing a learning exercise.

4.5 Transformations available and cards design

The game implements the selection of transformations defined in Section 3 (Table 1). Each of these options has a unique graphical representation in the cards. The design resembles the familiar icons and colors commonly found in design and animation software. It provides an intuitive representation of translations, rotations, scales, and reflections, as well as the involved axes (red, blue, and green, corresponding to X, Y, and Z). In addition to icons and colors, the only non-graphical information appearing in cards is a letter denoting the axis and a number that means different things depending on the card type (degrees for rotations, multiplicative factors for scales, distances for translations, and -1 for reflections). Figure 3 shows a portion of the game screen with the main elements in place: (1) A deck zone that allocates six cards, (2) a "sequence" zone with three steps, (3) the "goal" cube already in the desired position, and (4) the "player" cube in its original position. As soon as the sequence is complete, the game starts an animated sequence, step by step, showing the transformations and the final state achieved by the player. If there is a match with the goal, a score is given, and continue with the next challenge. If the submitted sequence is not correct, the player has the option to repeat de challenge. In this simple example, three cards will remain unused; this may discourage the player from random permutations guessing.

5 CONCLUSIONS AND FUTURE WORK

We consider that the prototype described in this work achieves the original requirements and can be extended to cover others. Furthermore, while previous works and various available tools aim at a similar objective, we consider that our proposal is more closely tailored to the specific context: higher education, computing, computer graphics, GTs, and matrices composition. Thus, different tools may be more effective in other contexts but may be less effective in this particular context, with this precise learning objective. Our proposal is still pending to be tested in the classroom, where we will gather data to clarify and quantify the game's potential as a learning tool. With this goal in mind, we also expect to incorporate game analytics that complements other feedback data, such as the students' performance in standardized tests. Future work includes developing additional studies to provide a solid foundation for categorizing difficulty levels according to the current variety of transformations. Also, complementary tests can be done by changing some aspects of the game design, such as adding objects, enhancing the game mechanics, providing alternative solutions, or allowing the manual definition of the cards deck, among other things.

REFERENCES

[1] Karen F. Hollebrands. 2003. High school students' understandings of geometric transformations in the context of a technological environment. The Journal of Mathematical Behavior. 22, 1 (2003/01/01/), 55-72. DOI= https://doi.org/10.1016/S0732-3123(03)00004-X.

[2] Guven Bulent. 2012. Using dynamic geometry software to improve eight grade students' understanding of transformation geometry. Australasian Journal of Educational Technology. 28, 2 (04/02). DOI= http://dx.doi.org/10.14742/ajet.878.

[3] Yanik H. Bahadır. 2014. Middle-school students' concept images of geometric translations. The Journal of Mathematical Behavior. 36(2014/12/01/), 33-50. DOI= https://doi.org/10.1016/j.jmathb.2014.08.001.

[4] Donald Hearn and Pauline Baker, 2004. Computer graphics with OpenGL. Pearson prentice hall Upper Saddle River, NJ:.

[5] James H. Mathewson. 1999. Visual-spatial thinking: An aspect of science overlooked by educators. Science Education. 83, 1, 33-54. DOI= https://doi.org/10.1002/(SICI)1098-237X(199901)83:1<33::AID-SCE2>3.0.CO;2-Z.

[6] Núria Gorgorió. 1998. Exploring the Functionality of Visual and Non-Visual Strategies in Solving Rotation Problems. Educational Studies in Mathematics. 35(03/01), 207-231. DOI= http://dx.doi.org/10.1023/A:1003132603649.

[7] Douglas R. Boulter and John R. Kirby. 1994. Identification of Strategies Used in Solving Transformational Geometry Problems. The Journal of Educational Research. 87, 5 (1994/05/01), 298-303. DOI= http://dx.doi.org/10.1080/00220671.1994.9941257.

[8] Alison Leonard, E. and Nicole Bannister, A. 2018. Dancing Our Way to Geometric Transformations. Mathematics Teaching in the Middle School. 23, 5, 258-267.

[9] Desha L. Williams. 2011. Math For Real: Hair Braiding. Mathematics Teaching in the Middle School MTMS. 16, 8 (01 Apr. 2011), 512. DOI= http://dx.doi.org/10.5951/mtms.16.8.0512.

[10] Febrian Febrian and S. Perdana, 2018. Triggering fourth graders informal knowledge of isometric transformation geometry through the exploration of Malay cloth motif.

[11] Crystal Kalinec-Craig,Priya V. Prasad and Carolyn Luna. 2019. Geometric transformations and Talavera tiles: a culturally responsive approach to teacher professional development and mathematics teaching. Journal of Mathematics and the Arts. 13, 1-2 (2019/04/03), 72-90. DOI= http://dx.doi.org/10.1080/17513472.2018.1504491.

[12] Michael T. Edwards. 2003. Visualizing Transformations: Matrices, Handheld Graphing Calculators, and Computer Algebra Systems. The Mathematics Teacher MT. 96, 1 (01 Jan. 2003), 48. DOI= http://dx.doi.org/10.5951/mt.96.1.0048.

[13] Shashidhar Belbase. 2013. Beliefs about Teaching Geometric Transformations with Geometers' Sketchpad: A Reflexive Abstraction. Journal of Education and Research. 3(09/02). DOI= http://dx.doi.org/10.3126/jer.v3i2.8396.

[14] Natalia V. Andraphanova. 2015. Geometrical Similarity Transformations in Dynamic Geometry Environment Geogebra. European Journal of Contemporary Education. 12, 2, 116-128.

[15] Daniela Ferrarello,Maria F. Mammana and Mario Pennisi. 2014. Teaching/learning geometric transformations in high-school with DGS. 21(01/01), 11-17.

[16] Karen F. Hollerbrands. 2007. The Role of a Dynamic Software Program for Geometry in the Strategies High School Mathematics Students Employ. Journal for Research in Mathematics Education JRME. 38, 2 (01 Mar. 2007), 164. DOI= http://dx.doi.org/10.2307/30034955.

[17] Tugba Uygun. 2020. An inquiry-based design research for teaching geometric transformations by developing mathematical practices in dynamic geometry environment. Mathematics Education Research Journal. 32, 3 (2020/09/01), 523-549. DOI= http://dx.doi.org/10.1007/s13394-020-00314-1.

[18] Romero Tori,João Luiz Bernardes Jr and Ricardo Nakamura, 2006. Teaching introductory computer graphics using java 3D, games and customized software: a Brazilian experience. In ACM SIGGRAPH 2006 Educators program ACM, 12.

[19] Petr Felkel,Alejandra J. Magana,Michal Folta,Alexa Gabrielle Sears and Bedrich Benes. 2018. I3T: using interactive computer graphics to teach geometric transformations. In Proceedings of the 39th Annual European Association for Computer Graphics Conference: Education Papers. Delft, The Netherlands, Eurographics Association, 1–8.

[20] Sebastian Oberdorfer and Marc Erich Latoschik. 2016. Interactive gamified 3D-training of affine transformations. In Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology. Munich, Germany, ACM. 2996314, 343-344. DOI= http://dx.doi.org/10.1145/2993369.2996314.

[21] Hilal Gulkilik, 2016. The Role of Virtual Manipulatives in High School Students' Understanding of Geometric Transformations. In International Perspectives on Teaching and Learning Mathematics with Virtual Manipulatives, P.S. Moyer-Packenham Ed. Springer International Publishing, Cham, 213-243. DOI= http://dx.doi.org/10.1007/978-3-319-32718-1_10.

[22] Karin A. Orvis,Daniel B. Horn and James Belanich. 2008. The roles of task difficulty and prior videogame experience on performance and motivation in instructional videogames. Computers in Human Behavior. 24, 5 (2008/09/01/), 2415-2433. DOI= https://doi.org/10.1016/j.chb.2008.02.016.

[23] Ann E. Fleury. 1993. Evaluating discrete mathematics exercises. In Proceedings of the twenty-fourth SIGCSE technical symposium on Computer science education. Indianapolis, Indiana, USA, Association for Computing Machinery, 73–77. DOI= http://dx.doi.org/10.1145/169070.169352.

[24] Anne Watson and John Mason. 2006. Seeing an Exercise as a Single Mathematical Object: Using Variation to Structure Sense-Making. Mathematical Thinking and Learning. 8, 2 (2006/04/01), 91-111. DOI= http://dx.doi.org/10.1207/s15327833mtl0802_1.

[25] Cheryl A. Bodnar,Daniel Anastasio,Joshua A. Enszer and Daniel D. Burkey. 2016. Engineers at Play: Games as Teaching Tools for Undergraduate Engineering Students. Journal of Engineering Education. 105, 1, 147-200. DOI= http://dx.doi.org/10.1002/jee.20106.

[26] Nicola Whitton, 2009. Learning with digital games: a practical guide to engaging students in higher education. Routledge, New York.

[27] Sara I. de Freitas. 2006. Using games and simulations for supporting learning. Learning, Media and Technology. 31, 4 (2006/12/01), 343-358. DOI= http://dx.doi.org/10.1080/17439880601021967.

[28] Bart Machilsen,Maarten Pauwels and Johan Wagemans. 2009. The role of vertical mirror symmetry in visual shape detection. Journal of Vision. 9, 12, 11-11. DOI= http://dx.doi.org/10.1167/9.12.11.

[29] Unity, 2021. Unity game engine. Unity Technologies, version 2020.2023.