Scratch PoseNet Exergame Prototyping for Learning Process Support in Physical Education

Oleksandr Blazhko, Viktoriia Podhorna, Anastasiia Kokotieieva and Oleksii Ivanov
Odesa Polytechnic National University, Shevchenko Avenue 1, 65044 Odesa, Ukraine
{blazhko, podgorna.v.v, kokotieieva.a.s, o.v.ivanov}@op.edu.ua

Keywords: Exergame, PoseNet, Scratch.

Abstract: This paper suggests the Exergaming approach of computer game control for a human motion in sport computer games in order to increase the level of physical activity of students in online classes, which takes into account the initial knowledge of the specialised courses of the IT curriculums. The approach is based on open access repository of computer programs in the Scratch-based block language (47 games for 28 summer sports, 16 games for 10 winter sports). Scratch programs for human motion recognition are based on the PoseNet neural network, which allows changing the principle of controlling game characters through the keyboard and mouse. In October 2023, ninety two first-year students majoring in Computer Science participated in a five-week online Exergaming as second part of the Physical Education course, among them 70 students (76%) successfully completed all stages. 57% of students felt motivated to continue using the Exergame approach and demonstrated the desire to achieve decent results, that indicates skill "Positive Thinking" in the discipline "Physical Education", seven Hard Skills and four Soft Skills of Computing Curricula 2020 were indicated to compensate for the lack of classroom learning.

1 INTRODUCTION

The report "Computing Curricula 2020 (CC2020): Paradigms for Global Computing Education. Association for Computing Machinery" defines competences in the Knowledge-Skill-Disposition framework, where element "Skill" includes not only Computing Knowledge (Hard Skills), but also Foundational and Professional Knowledge (Soft Skills), for example: Analytical Thinking, Collaboration and Teamwork, Mathematics, Problem Solving and Trouble Shooting, Quality Assurance/Control, Self-Starter/Learner, Time Management and others [1]. The basis for quickly acquiring and maintaining these skills is physical activity with benefits for individual health and well-being, improved mental health, sleep and cognitive function. Therefore, an academic discipline "Physical Education" can be a useful tool to promote life/soft skills and to add skill "Positive Thinking" into previous Soft Skills [2]. Daily exercise and participation in sports competitions build competencies "Self-Starter/Learner" and "Collaboration and Teamwork" for students.

Unfortunately, in many universities due to COVID restrictions in 2020-2021 and in the Ukrainian universities due to war activities since 2022, a lot of students have not access to sports facilities for Physical Education classes. Therefore, the "Physical education and sports" department of Odesa Polytechnic National University decided to adapt the curriculum of this course for students of IT specialties only through the use of modern software technologies in cooperation with the "Information Systems" department. Today, there are tools for human-computer interaction through human motion recognition that allow you to control events in entertaining or educational computer games using Exergaming technologies ("exercise" + "game") [3].

For many years, non-contact motion control sensors are used to recognize human movements, for example, the MS Xbox 360/One game console, which contains an MS Kinect sensor with an infrared camera [4] to support 3D computer sports games Kinect Sport: Tennis, Table Tennis, Golf, Skiing, Baseball, Soccer, Basketball, Boxing, Track and Field (Sprint, Javelin, Discus, Long Jump and Hurdles). However, the use of Exergaming technologies in the curriculum for first-semester

students has limitations when conducting online classes:

1) lack of access to infrared cameras;
2) lack of access to commercial computer games;
3) lack of knowledge to create computer games.

The first limitation can be partially removed by using webcam image processing programs based on neural network, for example, PoseNet [5]. The second limitation can be partially removed by using the Scratch programming environment [6], a tool for teaching computational thinking skills in schools. Scratch allows students quickly create prototypes of 2D-computer programs and provides to modify computer games created by students around the world and posted in open file repositories [7]. Unfortunately, there are no examples of sports games with motion control in the Scratch repository. However, work [8] presents special programs for displaying human movement by using the MS Kinect v.1 cameras in Scratch v.1.4. These programs use different scenarios of games [9]: "Alien attack" controls the movements of body to the left/right and raising the arms above the head, "Hungry shark" controls the movements of the head to up/down and joining the hands together, "Hungry ant" controls the movements of the right hand. The work [10] presents additional games for the rehabilitation of stroke patients: "Whack-a-mole", "Harvest carrots", "Picking apples", "Bowling" and "Boxing". These games can be easily adapted for simple Webcams instead of MS Kinect cameras by using free software libraries for connecting neural networks with the Scratch environment, for example, PoseNet [11].

The analysis of the previous works determined the need to modify the Physical Education curriculum for students in IT specialties in learning process of first semester. These modifications should partially compensate for the lack of classes with students in the gym and take into account the list of discipline competencies of the current and future semesters.

This work presents the Scratch-Based Exergame Educational Approach (Exergame approach) for computer game prototyping on the topics of Olympic sports. This Exergame approach should take into account the simple technical requirements for students during online classes and low requirements for knowledge of programming languages:

- access to simple webcams and computers;
- knowledge of the basics of programming in the Scratch environment;
- access to the Scratch software library for connection to the PoseNet neural network;
- access to the open Scratch repository of Olympic sports computer games.

The purpose of this work is to establish a connection between Hard/Soft Skills of CC2020 and skills of Exergaming approach during online classes (staying students at home) of the discipline "Physical Education" for students in IT specialties.

2 EXERGAMING IN PHYSICAL EDUCATION

2.1 Education Concept Based on Exergaming Approach

The proposed Exergaming approach is based on (Figure 1):

1) the Open Scratch repository of 2D-computer game programs;
2) the open Scratch Web-server library for connection to the PoseNet neural network;
3) Scratch program code templates for human motion recognition based on Webcam and the PoseNet neural network.

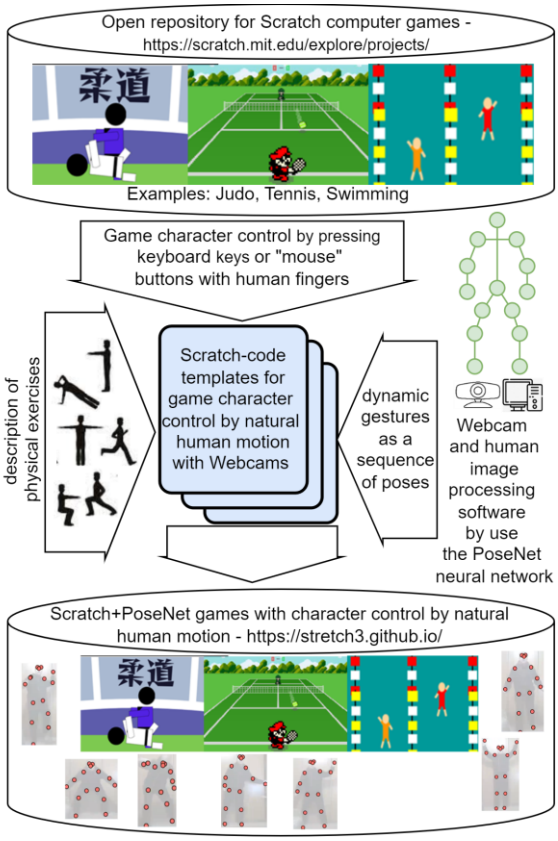


Figure 1: Student learning concept based on the Exergame approach.

The proposed approach gives recommendations for choosing a method for controlling game characters by using Webcams and descriptions of the characteristics of human dynamic gestures as a sequence of transitions between poses (static gestures).

The learning process includes the following stages:

1) students choose an example of a physical exercise and an example of a computer game in the Scratch open repository;
2) students analyze physical exercises and choose Scratch program code templates based on recommendations;
3) students use templates and make changes to the Scratch program code of computer games to control the game characters with gestures.

To determine the list of computer games that students will explore and modify during online classes in the Scratch-repository, we searched and selected 63 computer games: 47 games for 28 summer Olympic sports; 16 games for 10 winter Olympic sports. The Scratch version 3 uses an open source code and can be hosted on any Web server with additional software library PoseNet2Scratch to recognize human motion based on the PoseNet neural network [11]. The PoseNet allows present the human body in the form of 17 reference points (Figure 2).

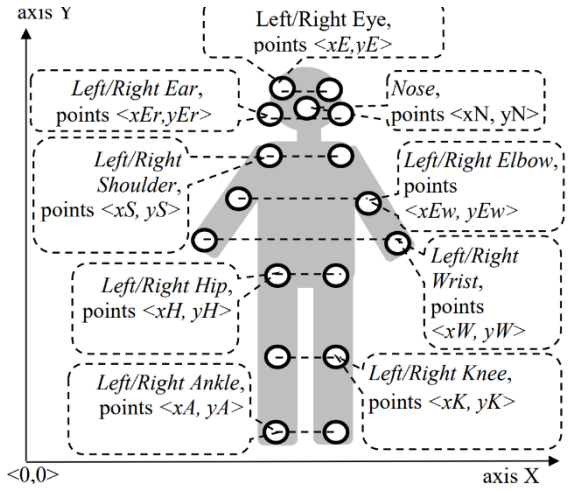


Figure 2: Reference points in PoseNet neural network.

Each description of a movement can be considered as a sequence of poses – positions taken by the human body, head and limbs in relation to each other. These poses can be considered as gestures: actions or movements of a human body. Exergaming approach uses the following terms:

- static gestures (sg) – unchanging position of the human body with fixed relationships of its various parts to each other;
- dynamic gesture (dg) – sequential transition in time between two static gestures.

For static gestures their coordinate description is proposed, which takes into account the following:

- for each pair of static gestures, reference points with significant difference between static gestures are searched, and a logical expression comparing the values of the coordinates of these reference points is described;
- these gestures must use a minimum number of comparisons between reference points, but guaranteeing a significant difference in their description between different static gestures.

Table 1: Examples of coordinate description of gestures.

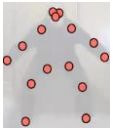
A kind of sport	Screen-shot of 1th static gesture	Screen-shot of 2nd static gesture	Coordinate description of static gestures
Judo	 *sg1* – protection	 *sg2* – attack	*sg1:* *(xK < xH)* *and (xW < xA)* *and ((ySL - yHL) = (xHR - xHL))* *sg2:* *(yWL = yN)* *and (yWR < yEwR)* *and (xN = xH)*
Tennis	 *sg1* – racket ready to hit	 *sg2* – hit with a racket	*sg1:* *(yWL > yEwR)* *and* *(xWL > xEwL)* *sg2:* *(yWL > yEwR)* *and (xEwL = xEwL)*
Swimming	 *sg1* – swing by right hand	 *sg2* – swing by left hand	*sg1:* *(yWR > yN)* *and* *(yWL < yEwR)* *sg2:* *(yWL > yN)* *and* *(yWR < yEwL)*

Table 1 presents three computer games examples of Web camera screenshots in the form of reference points of a human body with static gestures in the

Scratch program with a connected PoseNet neural network and a coordinate description of gestures.

2.2 Description of Scratch PoseNet Program Library Templates

To control anchor points, the PoseNet2Scratch software library adds a section of green blocks to Scratch. Figure 3 shows fragments of the sprite placement rendering Scratch-blocks associated with the anchor points of a right wrist. In addition, to program the recognition conditions of the specified gestures, it is recommended to create variables, for example, *nose_y*, *left_wrist_y*, *right_wrist_y*, the values of which are set in the software modules for visualization of reference points.

Figure 3: Fragments of Scratch-blocks for visualization of anchor points for a right wrist.

It is proposed to use an algorithmic template (example in C-language style) to recognize dynamic gestures by using system states of static gestures:

```
old_static_gesture := 0;
new_static_gesture := 0;
while (true) {
if static_gesture_1_definition() {
 // change system state
 old_static_gesture=new_static_gesture;
 new_static_gesture = 1;
}
   …
if static_gesture_n_definition() {
 // change system state
 old_static_gesture=new_static_gesture;
 new_static_gesture = n;
   }
}
if dynamic_gesture_1_definition(
 new_static_gesture,old_static_gesture)
 dynamic_gesture = 1;
…
if dynamic_gesture_n_definition(
 new_static_gesture,old_static_gesture)
 dynamic_gesture = n;
}
```

The training task includes programming algorithms for recognizing two dynamic gestures.

Students translate this algorithmic template into blocks of the Scratch language. For example, to recognize two static gestures “Both hands below nose” and “Both hands above nose” program code can include the following steps:

1) checking the condition of recognition of the first static gesture, which is defined by the logical expression "*sg1: (yN > yWL) AND (yN > yWR)*";
2) if the static gesture returns true, then set variable *new_static_gesture = 1* (state *sg1*);
3) checking the condition of recognition of the second static gesture, which is defined by the logical expression "*sg2: (yWL > yN) AND (yWR > yN)*";
4) if the static gesture returns true, then set variable *new_static_gesture = 2* (state *sg2*).

Dynamic gesture recognition blocks take into account the transition between a previously recognized static gesture and a new recognized static gesture. For the above example, two dynamic gestures may be recognized:

- *dg1* – “hands raised above the nose” through previously programmed recognition processes of two static gestures by using order $sg_1 \rightarrow sg_2$;
- *dg2* – “hands lowered below the nose” through previously programmed recognition processes of two static gestures by using order $sg_2 \rightarrow sg_1$.

To establish the fact of recognition of dynamic gestures, the change *dynamic_gesture* is used, which can later be used in the program code of the analysis of the conditions of controlling the game character to replace the classic version of control in the game through a computer keyboard or a computer mouse.

The exergaming approach offers students only to make changes to an existing computer game. However, to understand the basics of game design, students consider classic model “Mechanics-Dynamics-Aesthetics” [12]. Particular attention is paid to the Dynamics-component, which takes into account the actions of the players, depending on their psychophysiological abilities, for example, the speed of the brain's reaction, the speed of the fingers movement when pressing the keys on a keyboard. Tuning the Dynamics-component should balance the game. The computer games are already balanced. But after replacing the key control with gesture control, the computer game becomes unbalanced. Therefore, students should find Scratch-blocks for balancing the game and conduct experiments to adjust the new characteristics of game dynamics.

During the analysis of the steps of using the Exergame approach, connections between Hard/Soft skills of CC2020 and skills of the Exergame approach were established. Besides, two important steps determined connections with Hard Skills in the Physical Education Curricula (Figure 4).

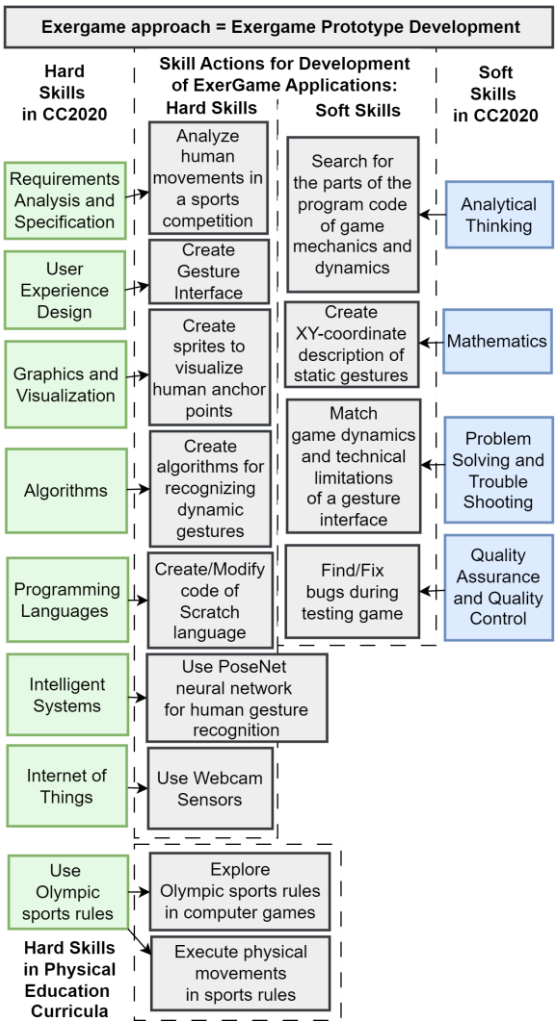


Figure 4: Examples of connections between Hard/Soft Skills of CC2020 and skills of the Exergame approach.

3 CONCLUSIONS

In October 2023, ninety two students participated in five-week online classes of the "Physical Education" discipline. Seventy students (76%) successfully completed all steps. The analysis of the results confirmed that students starting to study "Physical Education" in the first semester and lacking sufficient skills in professional programming, were already able to create computer games with 2D graphics and could assess/test their motor skills. An effect of implemented the Exergame approach consists of:

- 57% of students felt motivated to continue using the Exergame approach and demonstrated the desire to achieve decent results, that indicates skill "Positive Thinking" in "Physical Education";
- seven Hard Skills and four Soft Skills of CC2020 were indicated to compensate for the lack of classroom learning.

REFERENCES

[1] "CC2020 Task Force. Computing Curricula 2020: Paradigms for Global Computing Education". Association for Computing Machinery, New York, USA, 2020.

[2] S. Aliberti and G. Manzo, "Physical and sport education for life and soft skills development". Formazione & Insegnamento, vol. 21(1S), pp. 79-85.

[3] J.M. Krause and S.E. Jenny, "Physical educators' exergaming integration experiences, attitudes, and self-efficacy beliefs". The Physical Educator, vol. 80, 2023, pp. 91-107.

[4] M. Kourakli, I. Altanis, S. Retalis, M. Boloudakis, D. Zbainos, and K. Antonopoulou, "Towards the improvement of the cognitive, motoric and academic skills of students with special educational needs using Kinect learning games", International Journal of Child-Computer Interaction, vol. 11, 2017, pp. 28-39.

[5] J.-L. Chung, L.-Y. Ong, and M.-C. Leow, "Comparative Analysis of Skeleton-Based Human Pose Estimation", Future Internet", vol. 14(12), 2022, doi: 10.3390/fi14120380.

[6] S. Grover, "Teaching and Assessing for Transfer from Block-to-Text Programming in Middle School Computer Science", Research in Mathematics Education, Springer, Cham, 2021.

[7] "Scratch project Explore", [Online], Available: https://scratch.mit.edu/explore/projects/games/.

[8] S. Howell, "Install Scratch and Kinect2Scratch", In J. St. Jean (Ed.), Kinect Hacks: Tips and Tools for Motion and Pattern Detection, Sebastopol, CA: O'Reilly, 2012, pp. 184-206.

[9] S. Howell, "Kinect2Scratch (Version 1.5) [Computer Software]", [Online], Available: https://stephenhowell.github.io/kinect2scratch/.

[10] J.-W. Hung, et al., "Comparison of Kinect2Scratch game-based training and therapist-based training for the improvement of upper extremity functions of patients with chronic stroke: A randomized controlled single-blinded trial", European Journal of Physical and Rehabilitation Medicine, vol. 55, 2019, pp.542-550.

[11] J. Ishihara, "PoseNet2Scratch", [Online], Available: https://github.com/champierre/posenet2scratch.

[12] L.C. Duarte and A.L. Battaiola, "Distinctive features and game design", Entertainment Computing, vol. 21, 2017, pp. 83-93.