ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
Indian Journal of Science and Technology, *Vol 11(46), DOI: 10.17485/ijst/2018/v11i46/139719, December 2018*

Virtually Reactive Boxing Based Action Reflector Suits for Real Time Sensation

Muhammad Arsalan Jalees Abro[1]* and Irfan Ahmed Halepoto[2]

[1]Department of Mechatronics Engineering, Mehran UET, Jamshoro, Sindh, Pakistan; arsalan_jalees@hotmail.com
[2]Department of Electronic Engineering, Mehran UET, Jamshoro, Sindh, Pakistan; irfan.halepota@faculty.muet.edu.pk

Abstract

Objectives: Recently in the gaming industry, multiple games for different consoles and machines are being introduced and their sole focus is to bring the player closer to its avatar. **Methods/Statistical Analysis:** In the past, such avatar was achieved through the motion detection cameras such as Kinect by Microsoft and the introduction of virtual reality in games. Even with all these, the games are still not well connected with the players because the player do not feel the outcomes of the game in a way that if player's character gets hurt, simultaneously the player should also physically experience the action. **Findings**: This research is based upon the use of Kinect as an input device which will detect the player and record its movements in the game developed through Unity 3D. The game developed is based upon boxing. There are two characters controlled by two players and the concept is that when a player's character gets punched, the player also gets a feel through vibration motors attached to the player's suit. **Application/Improvements:** This virtual gaming suit will ensure that a gaming player must be the part of the game and will feel the action and sensation of the game in real time.

Keywords: Arduino; Boxing; Gaming Suit, Microsoft Kinect, Unity 3D

1. Introduction

There are multiple reasons for the gaming industry to gain the popularity midst the minds of people. Its purpose is to create and develop games which will enhance a person's perspective while providing entertainment. Through the years, games have been the source of free zone resting and being focused to just play and enjoy the graphics. As the Microsoft introduced Kinect, the idea of gaming has changed. The focus from only graphics point of view is diverted towards the interaction of games with their players. Kinect[1,2] not only provides fun but also gives a reason for the players to workout with their friends by staying at home. Players have become lazy and health issues have arrived as exertive games such as boxing are being neglected. So, such games will help players enjoy the game while working upon their health. To enhance the gamers experience and to make them more interested in such games, a suit was devised to help the gamer feel a blow of a punch as laid in a real boxing match. This will not only give the gamer an experience through vision, but also vibration motors will be put into action, making the gamer feel a hit on the point where punched. Gaming industries have been working on gaming simulators since ages. They have devised different strategies including a vibration motor in gaming controllers[3] just to depict an action such as a car crash in a racing game or movement in a wrestling game. Then there are some developers who

*Author for correspondence

are trying to devise suits which are to be introduced to the players but are either unable to generate enough funding for the project or are yet to launch the final product. In literature, some works has been on different simulators. Elsaco[4] has developed a car simulator called Motion-SIM, to depict the driving experience on a road by a player sitting on a seat in his/her room. This simulator has 4 moving legs, which are connected to the player's seat and move it according to the track and the movement of the car. The limitation of this simulator is that it is only for the environment of a car and while the driver crashes, he/she will not really feel the intensity of the crash. They also provide an aircraft simulator with the same features but differentcontrols.

The Haptika[5] has introduced a jacket named haptic jacket which is for the first person shooting game 'Counter Strike". The player will feel the gun shot but the limitation of this jacket is that it can only be used in Counter Strike and the players will have to use the keyboard as the controller. So the movement and agility of the player remains the same. The suit devised here, makes the player go through exertion by literally throwing punches standing in front of Kinect and when the player is hit, he/she will feel the blow as well. So, the overall state of this suit is much better as the player will use the entire body to fight.

2. Designing of Project

The project idea lies in developing a game using Unity 3D. The idea of the game is to have two characters which will box each other and they are to be controlled by two human players playing the game. The input device used for controlling is the Microsoft Kinect. The design is included the following steps:

1. A motion detection camera Kinect to read and detect the human skeletons of the players,
2. A two players' boxing game using Kinect,
3. A Unity 3D communication with Arduino, and
4. The output of the game serving as the inputs to the Arduino.

As gradually these steps were completed, the entire game came into existence. In the final stage, through User Datagram Protocol (UDP) communication, the punches thrown in the game were successfully detected by the suits worn by each player.

2.1 System Components

The following components have been used in this project: Unity, Kinect camera sensor, Arduino Mega 2560, Ethernet Shield, and Grove-Vibration Motors.

1. *Unity:* Unity[6] is a user friendly, open source game development platform. It is used to construct the virtual characters and scenarios. It is a powerful platform that can be used to develop 2D & 3D games and interactive applications for PC, mobiles, consoles and websites. Most major components required to develop games can be obtained in the free edition. Unity provides all necessary tools to create a complete game: an animation system, graphic solutions to create interactive user interfaces, audio controllers, lighting system, cameras, 2D/3D physics engines, and an Application Program Interface (API) library solution to write scripts and many advanced components to create a fully customized game. Asset Store, an online marketplace for Unity also embedded in the development environment, provides complete free and paid for packages such as terrains, artwork, complete games that can be imported into Unity projects. To maintain compatibility with multiple platforms, Unity uses "Mono" in the background; an open source development platform based on ".NET" framework of Microsoft, and includes among others a C# compiler, a runtime, and various libraries. The Unity 3D is basically a game engine which is a complete integrated development environment and as compared to other engines, Unity allows using several languages to

program games. For the development, Unity supports C# and JavaScript. All mentioned languages can be combined in one scene and even half of the scripts of an object can be written in one language and the rest in another language.

2. *Microsoft Kinect:* Kinect[7] is a line of gesture identifying input device made by Microsoft for the purpose and the game play of Xbox 360 and the latest console, Xbox One. Kinect lets the users control and interact with the console or a PC without using buttons or any other control. Just by the use of simple gestures and audio commands, the user can move through menus or play a game. With the help of Kinect, a person can remain seated and relaxed and there is no need of managing a controller or any other handheld input device. Kinect is equipped with a RGB camera along with two 3D depth sensors which provide a new experience to the users. When all three colors are combined it helps the consoles to view a human in a 3D view through which the gestures are made possible. These make the distance and placing of the humans be read easily. The software of Kinect has a skeleton recognition algorithm which functions by detecting the body in front of Kinect and giving a character with head, limbs, torso, so that the console can read it and can be placed upon the character in the games which support Kinect.

3. *Arduino Mega 2560:* Arduino Mega 2560 is a simple form of microcontroller board which is simple to use and is based upon the AT Mega 2560. This board is equipped with 54 digital input pins and output pins. Amongst these, 14 of the pins can be utilized as the PWM outputs, 16 of the pins as analog inputs, 4 as universal asynchronous receiver transmitter (UARTs), one single crystal oscillator of 16 MHz, USB connection, a power jack, an In-Circuit Serial Programming (ICSP) header, and a reset button. The only requirement is to connect it to a PC or a laptop by using a USB cable or it can be powered with an AC-to-DC adapter or even a battery.

4. *Ethernet Shield:* The work of an Ethernet Shield is to instantly enable internet connectivity for the projects of Arduino. This on board Wiz5100 Ethernet controller has the capability to handle up to 4 TCP and UDP connections by simply stacking it upon the Arduino to create networked devices.

5. *Grove-Vibration Motor:* The Grove-vibration motor is a small vibration motor which is suitable for a non-audible indicator. At a HIGH input, the motor will vibrate equivalent to the vibration of a cell phone.

2.2 Kinect Integration with Unity

Kinect is an input device which is our primary interactive device and manipulation functions can plug-in on Unity software. Unity is user friendly open source software. The purpose of this study is to create a game which will introduce the effect of real time feel while playing a game. Kinect here detects the depth of the human skeleton and applies the information of skeleton to the virtual characters scripted in Unity. There is just a need to plug the Kinect into a serial port. To select a character, a player has to stand in front of Kinect and the character will be activated. Kinect is invented by Microsoft so using it on Microsoft Windows is very easy. It is nearly plug-and-play just a few drivers are required, which can be downloaded from the official Microsoft website.

1. *Development with Unity 3D and Kinect:* Unity has been equipped with multiple libraries for the purpose of application development in Unity using Kinect. Mainly there are two libraries; one is the official Microsoft Kinect Software Development Kit (SDK). The SDK provided by Microsoft for the applications of this device is known as Kinect for Windows. The advantage in using the SDK is that it has all necessary drivers for the use of Kinect and

is loaded with sample projects. The other library is the open source version called OpenNI; The API is provided by OpenNI for those applications which use natural interaction. It is a multi-platform framework which aims at providing the features which are necessary to write applications which use NI.

2.4 Unity Integration with Arduino

The interface between Unity 3D, virtual environment, and Arduino Mega, is done through the .NET framework. The earlier versions of unity were limited as for the interfacing of custom hardware, extra plug-in were required but for that professional versions were needed. The .NET framework includes .NET and .NET.SOCKET libraries which open the ports for communication.

Unity and Arduino Integration Steps: Unity and Arduino can be communicated using the serial port or the UDP[8]. There are five steps to establish communication between unity and Arduino, as follows: Configuring, Opening, Writing, Reading and Communication. These all steps, for both types of communications, remain same only the codes are being altered. In serial, baud rate and the COM port is defined which is in use. Communicating using UDP requires the IP address of the client, the communication port and the IP address of the server. The basic difference between them is that in serial the data is sent through the USB port of the device and in UDP, the data is sent over a network which may or may not be a private network. The problem with serial communication is that sending a lot of data at once may overflow the serial port buffer which might either crash the system or create a severe lag. This is the reason the switch was made from serial communication to UDP communication. Though there is a chance of data loss, but that is a small cost as compared to a total lag or system crash.

3. Methodology

This undergoes how the game was developed by first explaining it into parts and then combining the entire concept together to shape the whole project into a single vision.

3.1 Project Methodology Flow Chart

The flow chart as shown in Figure 1 illustrates the step by step system methodology of the game. At the beginning,

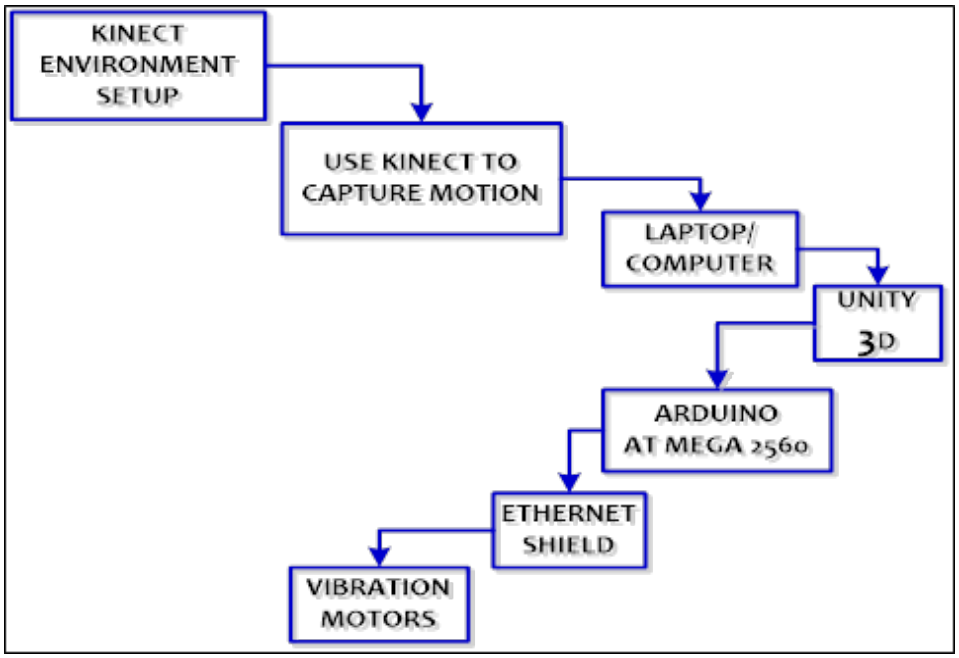

Figure 1. Project methodology flow chart.

the Kinect activates as commanded through the script placed in Unity 3D. After the initialization of Kinect is complete, it starts to capture the motion which the players make while standing in front of it. This recorded motion is sent to the machine upon which Unity 3D is installed. Unity 3D takes the input from the Kinect and forwards it to the Arduino AT Mega 2560 which communicates with the Ethernet shield. Upon the signals received from the shield, the vibration motors equipped upon the suits turn on and off simultaneously.

3.2 Game Methodology

This is a local two-player game starting with two players standing in front of Kinect. Kinect acts as an input device. It detects the movements of players and accordingly sends to the Unity environment. The author has developed a game through Unity 3D. Unity is the game interface containing two avatars "Dimple" and "Character" which are packed with colliders on their hands and different parts of the body. Each collider is attached to particular vibrator through programming. When collider of one avatar hits other avatar's collider, a signal will be sent to the Arduino which will activate the particular vibrator on which collider strikes. The game keeps running until a player gives up or runs out of stamina.

3.3 Complete Integrated System Flow Chart

The flowchart in Figure 2, gives an overview of the entire project's working. As seen in the left column which reads initiating game, first the user is identified after which skel-

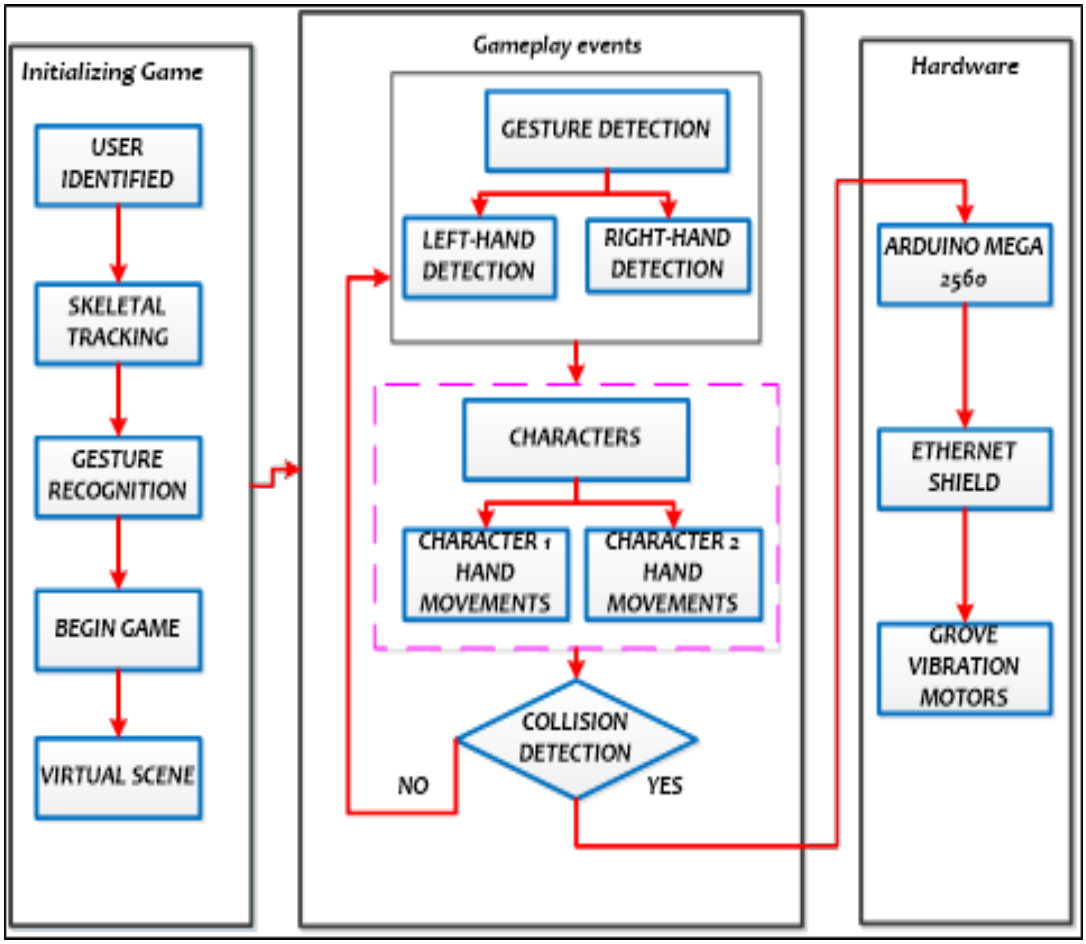

Figure 2. Complete integrated system flow chart.

etal tracking takes place. In order for the players to select their characters, they raise their hands that are right hand for player one and left for player two. Once the characters are selected, the game begins and the players are ready to box. The middle column shows how the game play takes place. As the gestures are detected, the game checks whether the collision took place or not. In the case of a collision, the signal goes into the Arduino which is shown in the third column titled as hardware. The communication between Unity and Arduino takes place through the Ethernet shield. As high signal is received, the vibrator upon either player's activates which is signaled through the Arduino and the player feels a punching sensation. The events in the second and the third columns keep repeating until and unless the game doesn't stop.

3.4 Game Setup

To play the game, the user must stand in front of Kinect, more than a meter away. The Kinect sensor will detect both the players through particular gesture recognition method. The avatars in the game will be controlled by two players. The avatars will follow the players' actions. When the players hit each other, the particular vibrators will be turned on for a certain time period. The vibration motors for vibrotactile feedback are sewn into a vest. This concept is illustrated in Figure 3.

4. Experimental Setup

This boxing game consists of two players who will box each other. For this purpose, their gestures are recorded through Kinect which interfaces with Unity 3D and then Unity 3D will interface with Arduino. This interfacing and implementation are already explained in the section III. The game consists of two characters each controlled by either player. As the players stand in front of the screen, their character is chosen for them and they are ready to play. Both the characters look different in size but their capabilities and strengths are equal as described in Figure 4.

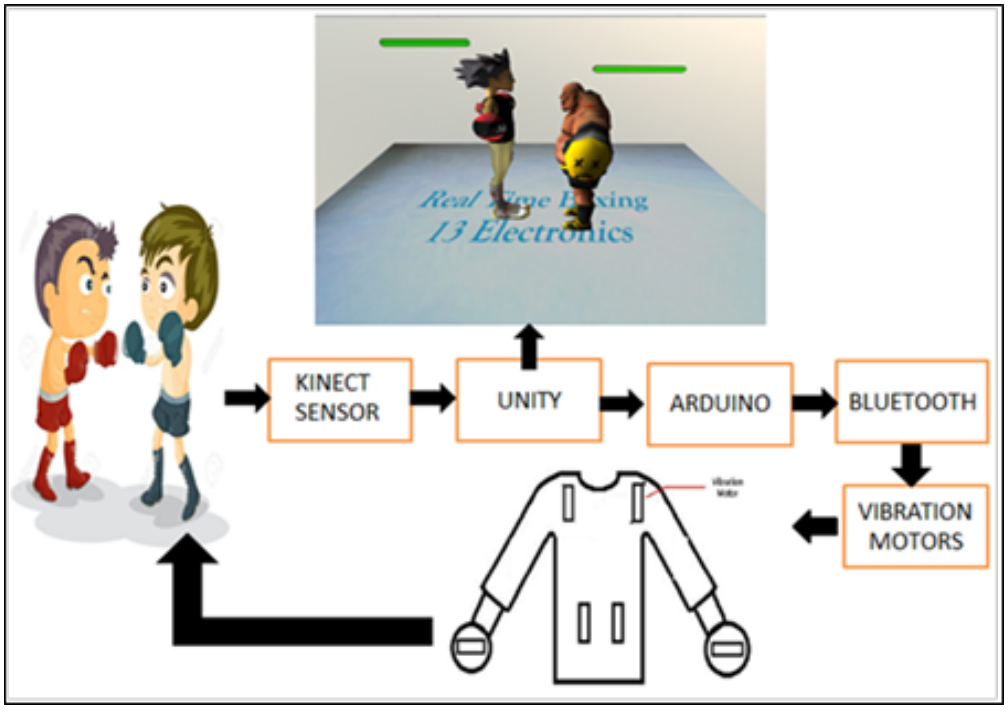

Figure 3. Complete game system with gaming suit.

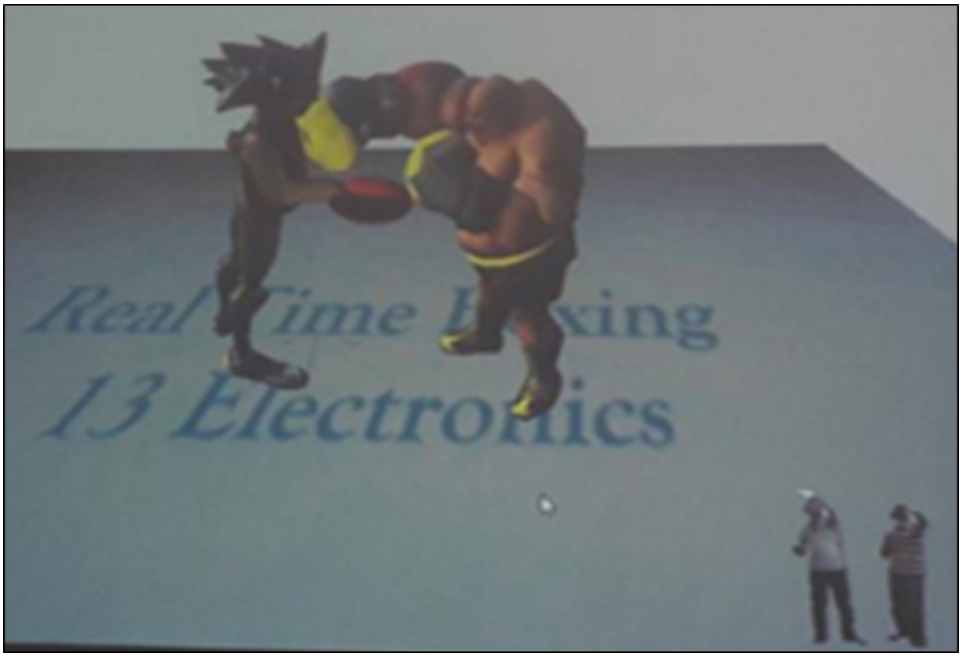

Figure 4. Game interactive environment.

4.1 Testing of First Character

The idea of the game relates a character's punch which intersects another character's body. When this happens, a message will be shown in the console window, stating both the target and the punch thrown. These messages will be shown through "debug.log" function. Table 1

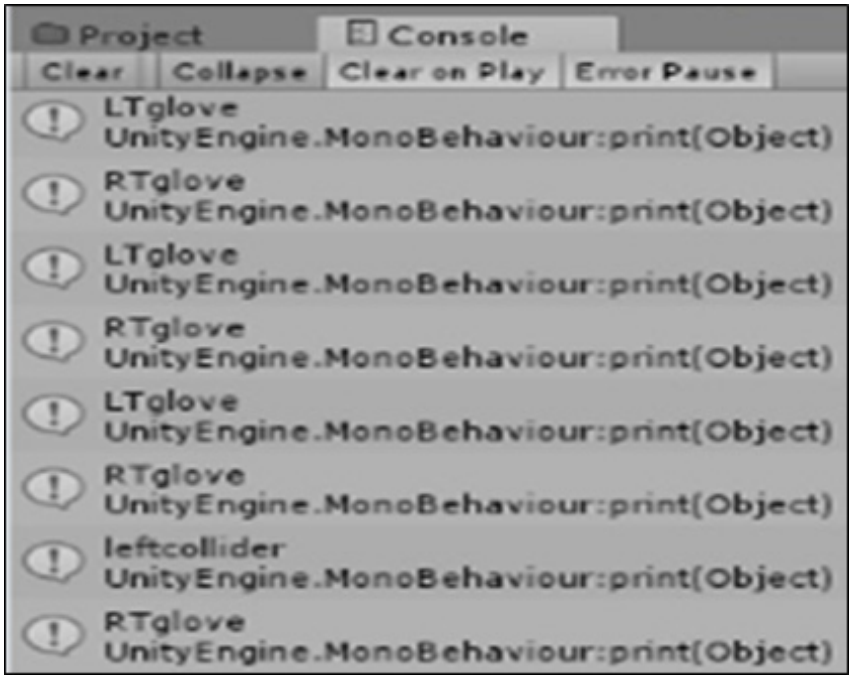

Figure 5. Console messages for first character.

Table 1. Test table for the first character

Test #	Punch Thrown	Hit Successful	LED #
1.	Left Hand	Yes	1
2.	None	No	-
3.	Left Hand	Yes	2
4.	Left Hand	Yes	1
5.	Left Hand	Yes	1
6.	Right Hand	Yes	2
7.	Right Hand	Yes	1
8.	Right Hand	Yes	2
9.	Right Hand	Yes	2
10.	Right Hand	Yes	1

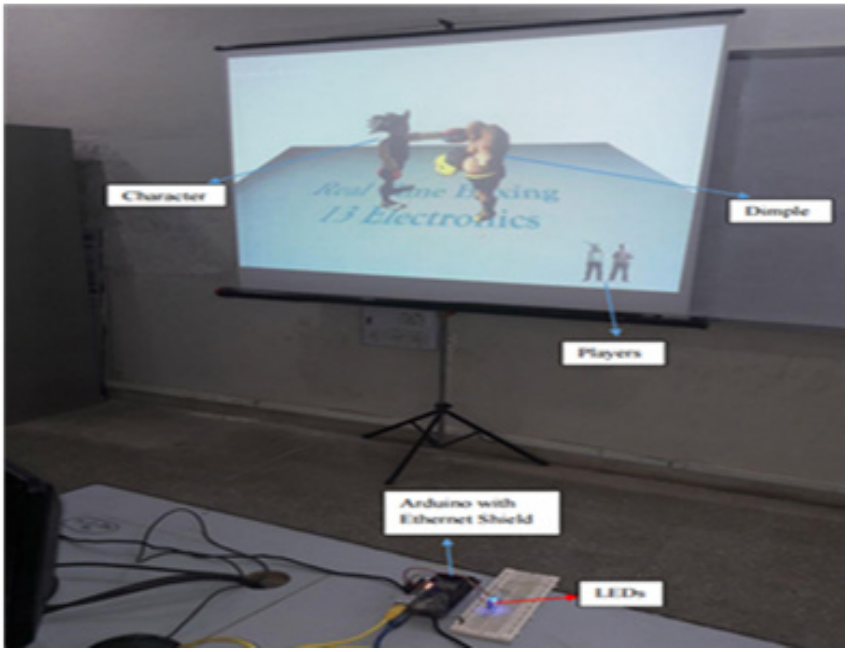

Figure 6. Collision detection through LEDs.

is the test table for the first character and successful hit is determined through both the console messages as shown in Figure 5, and as a result an indicator i.e., LED will glow. Here, LED1 denotes the hit taken at the right side of the second character and LED 2 denotes the hit taken at the left side of the second character as shown in Figure 6.

4.2 Testing of Second Character

The purpose and the working of the second character is same as that of the first character only the commands and controls lie in the hands of the second player. Table 2 is the test table for the second character. LED 1 denotes the hit taken at the right side of the first character and LED 2 denotes the hit taken at the left side of the first character. Figure 7 shows the console messages for this test. Figure 8 demonstrate the game views while the testing of both players took place.

4.3 Game Demonstration

As game, here are two shirts upon which total of 4 LEDs and 2 Vibrators were equipped. 2 LEDs and a vibrator for one shirt which was the first character's suit and 2 LEDs and a vibrator for the second shirt which was the demonstration for the second player's suit. When the players got hit, the LED or the vibrator placed at that point got powered showing that the proposed suits are working properly. The final demonstrated setup can be seen in Figure 9, where the entire setup is shown with the red

Table 2. Test table for the second character

Test #	Punch Thrown	Hit Successful	LED #
1.	Right Hand	Yes	2
2.	Left Hand	Yes	2
3.	Right Hand	Yes	1
4.	Left Hand	Yes	1
5.	Right Hand	Yes	2
6.	Left Hand	Yes	1
7.	Right Hand	No	-
8.	Right Hand	Yes	2
9.	Left Hand	Yes	2
10.	Right Hand	Yes	1

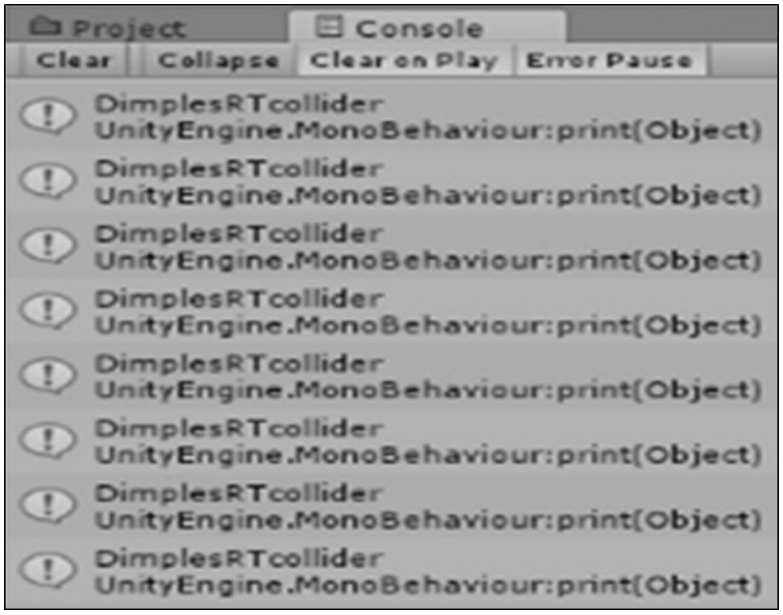

Figure 7. Console messages for second character.

Figure 8. Complete local two player game play.

shirt for player 2 and the white shirt for the player 1. The LEDs are connected with an Arduino board powered through the laptop upon which the game is being played. The Kinect in the corner of the table reads the gestures of the players and record it in the game.

Figure 9. The demonstrated version of the game.

5. Applications of the Proposed Work

From many applications the most identifiable is to ensure a gaming environment which will involve a player's sense to feel of being hit in real time. Till now, a player was just able to see the game in 3D view but with this game development, it will also be able to react to the game's actions. This will help in eliminating the laziness developed these days as playing a game will create a reaction, a player will be active and move with caution. This gaming suit can be used in multiple fields i.e., combat training, astronaut.

6. Conclusion

In this project, the game development environment is thoroughly studied enough to develop a game from scratch. Other than that, the communication of Arduino with Unity 3D was the difficult task as to get the output from the game that is the software and feed it as an input to the Arduino. This was done in order to turn on the vibrators in synchronization with the actions performed in the game. During this time, both serial communication and UDP communication were explored, giving a reasonable command over data sharing. Other than this, work was done on the Arduino libraries which allowed creating a sketch so that the signals to the vibrators be sent at proper instances and to the proper nodes. This project ensures a novel development in the gaming industry regarding the involvement of the user to a deeper essence of the character that the user deals with. It not only provides a platform for fun, but also a platform to release stress and to conduct a proper workout.

7. Acknowledgment

Authors are highly indebt to Dr. Muhammad Rafiq Abro, Dr. Bhawani Shankar Chowdhry, Engr. Khuhed Memon, and Engr. Azam Rafiq for their motivation and proficient guidance through the work. Authors are also thankful to Mehran University of Engineering and Technology, Jamshoro, Pakistan, for the support in providing the laboratories and research environment.

8. References

1. Zhang Zhengyou. Microsoft Kinect sensor and its effect, IEEE Multimedia. 2012; 19(2):4–10.
2. Yim Jeffrey, T. C. Graham. Using games to increase exercise motivation. Proceedings of the 2007 Conference on Future Play, ACM; 2007. p. 166–73. https://doi.org/10.1145/1328202.1328232.
3. Armstrong Brad A. 3D controller with vibration. U.S. Patent No. 6,906,700. 14 Jun. 2005.
4. Casas Sergio, Silvia Rueda. Lessons learned from the design and development of vehicle simulators: A case study with three different simulators, International Journal of Virtual and Augmented Reality (IJVAR) 2.1. 2018; 59–80. https://doi.org/10.4018/IJVAR.2018010105.
5. Ticknor Bobbie. Virtual Reality and the Criminal Justice System: Exploring the Possibilities for Correctional Rehabilitation. Lexington Books; 2018.
6. Calimeri Francesco, et.al. Integrating Rule-Based AI Tools into Mainstream Game Development." International Joint Conference on Rules and Reasoning. Springer, Cham; 2018. https://doi.org/10.1007/978-3-319-99906-7_23.
7. Zhang Mingshao, et.al. Recent developments in game-based virtual reality educational laboratories using the Microsoft Kinect, International Journal of Emerging Technologies in learning (iJET) 13.1. 2018; 138–59.
8. Silva Lucas, et.al. Development of a low cost data glove based on Arduino for virtual reality applications. IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA); 2013. p 55–59. https://doi.org/10.1109/CIVEMSA.2013.6617395.