

Augmented Reality Mobile apps Development with Unity and Vuforia SDK

S.I. Mustafina¹, A.R. Nurgalieva², Sh.R. Ibragimov², O.A. Medvedeva²

¹Bashkir State University, Zaki Validi street 18, Ufa, Russia, 450076

¹Kazan Federal University, Kremlevskaya street 18, Kazan, Russia, 420008

Abstract. This paper shows how to create an animation of a game character for a mobile application with augmented reality. The Unity platform, Vuforia SDK and Android OS were used for work. For this work, the Fighting Unity-Chan model was downloaded. This is an animation resource pack that specializes in fighting or action games. This code shows how a button is created (GUI.Button), it assigns an animation that will work when you click on it, the size of the button and its name. To create a smooth and correct transition between animations, Animator was created, in which all animations are shown taking into account the time of their action. For example, to switch from a state of rest (Idle) to some other state, for example, to a state of impact (Hicick), it is necessary to spend time in the amount of 1 second.

1. Introduction

In the past few years, augmented reality has become very popular. From expensive devices like the Hololens, to your own pocket devices with Apple's Arkit and Arcore for Android, augmented reality is fast becoming widely available.

Because augmented reality makes use of some hardware, we are going to need a few things. The first is an android phone running at least 7.0. It is possible to use an iPhone or an iPad, but, for this work we used an android phone because building to an android phone is a lot easier than building to iOS. Or, if you don't want to build to a phone you can use the webcam on your computer in most cases. Second, you will need to have a way to attach your phone to your computer (if you are using a phone). The last thing you need to do is import Vuforia. The only way to do this is to re-install Unity. Open up the download assistant (for Unity 2017.3 or any other version later than that) and check "Vuforia Augmented Reality Support" (also check "Android Build Support" since we will be targeting Android) in the package manager.

2. Main part

Vuforia is powered by Unity's cross-platform computer game development environment. Vuforia allows you to download free 3D models. For this work, the Fighting Unity-Chan model was downloaded. This is an animation resource pack that specializes in fighting or action games. This model can be seen in Figure 1. This model is displayed in a specific figure, i.e. if a certain pattern enters the field of view of the camera, then a model will be generated on it. This figure was chosen Figure 2.

To do this, it was necessary to create an ImageTarget, in which to further adopt our model, as shown in Figure 3. There were 10 animations in this package, all of which were displayed in the form of buttons, by clicking on which you could see a specific animation. The buttons were implemented as

C# code. The code can be seen in Figure 4 and Figure 5.



Figure 1. Fighting Unity-Chan.



Figure 2. A model generated.



Figure 3. ImageTarget.

```
C# IdleChanger.cs
1 using UnityEngine;
2 using System.Collections;
3
4 //
5 // Простой скрипт предварительного пространства анимации
6 //
7
8 // Require these components when using this script
9 [RequireComponent(typeof(Animator))]
10
11 #if UNITY_EDITOR
12 public class IdleChanger : MonoBehaviour
13 {
14     private Animator anim; // Ссылка на аниматора
15     private AnimatorStateInfo currentState; // Ссылка для сохранения текущего состояния
16     private AnimatorStateInfo previousState; // Ссылка для сохранения предыдущего состояния
17
18
19     // Use this for initialization
20     void Start ()
21     {
22         // Инициализация каждой ссылки
23         anim = GetComponent<Animator> ();
24         currentState = anim.GetCurrentAnimatorStateInfo ( layerIndex: 0);
25         previousState = currentState;
26
27     }
28 }
```

Figure 4. C# code.

```
C# IdleChanger.cs x
30
31 4 Event function
32 void OnGUI()
33 {
34     GUI.Box(new Rect(x: Screen.width - 400, y: 45, width: 220, height: 520), text: "");
35     if(GUI.Button(new Rect(x: Screen.width - 390, y: 60, width: 200, height: 40), text: "Job"))
36         anim.SetBool ( name: "Job", value: true);
37     if(GUI.Button(new Rect(x: Screen.width - 390, y: 110, width: 200, height: 40), text: "Hicick"))
38         anim.SetBool ( name: "Hicick", value: true);
39     if(GUI.Button(new Rect(x: Screen.width - 390, y: 160, width: 200, height: 40), text: "Spinkick"))
40         anim.SetBool ( name: "Spinkick", value: true);
41     if(GUI.Button(new Rect(x: Screen.width - 390, y: 210, width: 200, height: 40), text: "Rising_P"))
42         anim.SetBool ( name: "Rising_P", value: true);
43     if(GUI.Button(new Rect(x: Screen.width - 390, y: 260, width: 200, height: 40), text: "Headspring"))
44         anim.SetBool ( name: "Headspring", value: true);
45     if(GUI.Button(new Rect(x: Screen.width - 390, y: 310, width: 200, height: 40), text: "Land"))
46         anim.SetBool ( name: "Land", value: true);
47     if(GUI.Button(new Rect(x: Screen.width - 390, y: 360, width: 200, height: 40), text: "ScrewKick"))
48         anim.SetBool ( name: "Screw", value: true);
49     if(GUI.Button(new Rect(x: Screen.width - 390, y: 410, width: 200, height: 40), text: "DamageDown"))
50         anim.SetBool ( name: "DamageDown", value: true);
51     if(GUI.Button(new Rect(x: Screen.width - 390, y: 460, width: 200, height: 40), text: "Run"))
52         anim.SetBool ( name: "Run", value: true);
53     if(GUI.Button(new Rect(x: Screen.width - 390, y: 510, width: 200, height: 40), text: "Run_end"))
54         anim.SetBool ( name: "Run", value: false);
55 }
56 }
57 }
58 }
```

Figure 5. C# code.

This code shows how a button is created (GUI.Button), it assigns an animation that will work when you click on it, the size of the button and its name.

To create a smooth and correct transition between animations, Animator was created, in which all animations are shown taking into account the time of their action. For example, to switch from a state of rest (Idle) to some other state, for example, to a state of impact (Hicick), it is necessary to spend time in the amount of 1 second. This can be seen in Figure 6.

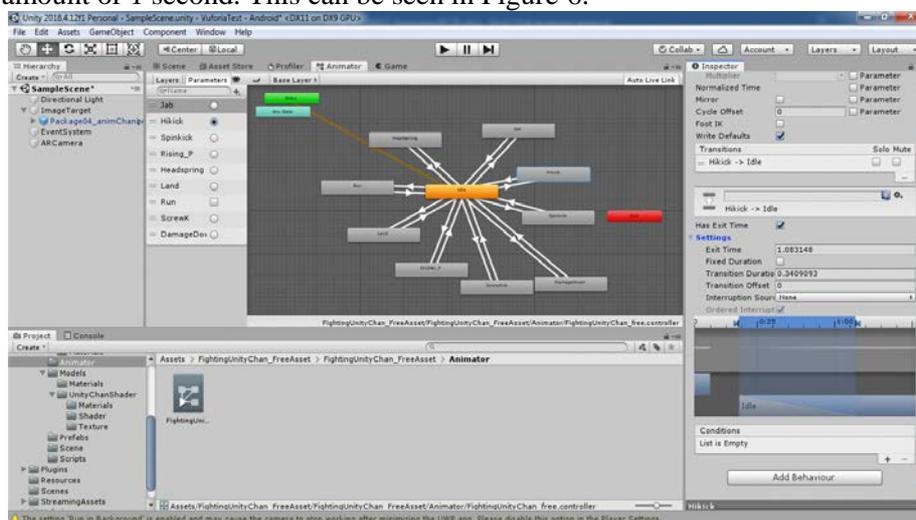


Figure 6. Animator.

In order for the script to start working, you must assign the script itself, called IdleChanger, to the model. We can observe this in Figure 7.

To run the model on the application, you need to download the apk package on your Android phone and have the picture shown in Figure 2. Next, the process of working with the model will be shown Figure 8 and Figure 9.

3. Conclusion

The primary value of augmented reality is the manner in which components of the digital world blend into a person's perception of the real world, not as a simple display of data, but through the integration of immersive sensations, which are perceived as natural parts of an environment. The gaming industry

embraced AR technology. A number of games were developed for prepared indoor environments, such as AR air hockey, Titans of Space, collaborative combat against virtual enemies, and AR-enhanced pool table games.



Figure 7. The script IdleChanger.

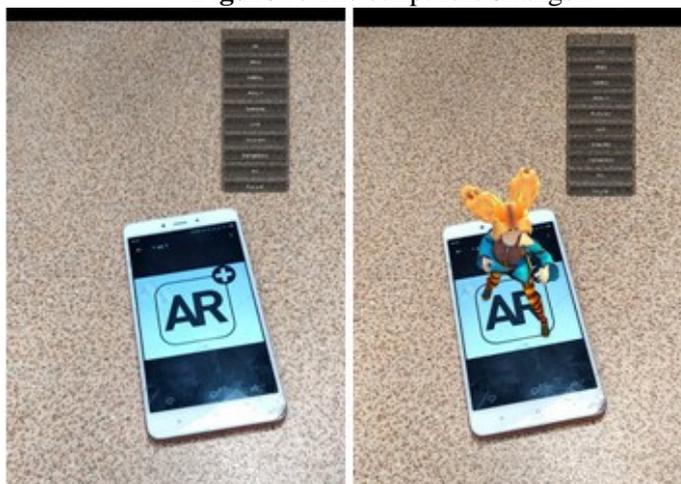


Figure 8. The process of working with the model.



Figure 9. The process of working with the model.

Augmented reality allowed video game players to experience digital game play in a real- world environment. Niantic released the augmented reality mobile game Pokémon Go. Disney has partnered with Lenovo to create the augmented reality game Star Wars: Jedi Challenges that works with a Lenovo Mirage AR headset, a tracking sensor and a Lightsaber controller. Augmented Reality Gaming (ARG) is also used to market film and television entertainment properties. In this work, an animated game character is created which can be controlled and played.

4. Acknowledgements

The authors express their gratitude and appreciation to the organizers of the conference of the Samara National Research University and Image Processing Systems Institute of RAS – Branch of the FSRC «Crystallography and Photonics» RAS for their help and support in preparing the article.

5. References

- [1] Bay, H. Surf: Speeded up robust features / H. Bay, T. Tuytelaars, L. Van Gool // Computer vision – Springer Berlin Heidelberg, 2006. – P. 404-417.
- [2] Bimber, O. Modern approaches to augmented reality / O. Bimber, R. Raskar // ACM SIGGRAPH Courses, 2006. – P. 101-120.
- [3] Bowman, D.A. New Directions in 3D User Interfaces // IJVR. – 2006. – Vol. 5(2). – P. 3-14.
- [4] Grigorev, R.A. Information system development using augmented reality tools / R.A. Grigorev, B.A. Valijanov, O.A. Medvedeva, S.A. Mustafina. – CEUR Workshop Proceedings. – 2019. – Vol. 2525.
- [5] Handheld Augmented Reality // Christian Doppler Laboratory on Handheld Augmented [Electronic resource]. – Access mode: <http://handheldar.icg.tugraz.at> (20.06.2019).
- [6] Kurz, D. Gravity-aware handheld augmented reality / D. Kurz, S. Benhimane // Mixed and Augmented Reality (ISMAR), 10th IEEE International Symposium, 2011. – P. 111-120.
- [7] Lee, K. Augmented reality in education and training // TechTrends. – 2012. – Vol. 56(2). – P. 13- 21.
- [8] Park, J. Natural feature tracking for extendible robust augmented realities / J. Park, S. You, U. Neumann // Proc. Int. Workshop on Augmented Reality, 1998.
- [9] Pasman, W. Augmented reality with large 3D models on a PDA: implementation, performance and use experiences // Proceedings of the ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry – ACM, 2004. – P. 344-351.
- [10] Pasman, W. Implementation of an Augmented Reality System on a PDA / W. Pasman, C. Woodward // Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality, 2003. – P. 276-289.
- [11] Phan, V.T. Interior Design in Augmented Reality Environment / V.T. Phan, S.Y. Choo // International Journal of Computer Applications. – 2010. – Vol. 5.
- [12] Vuforia Developer Portal [Electronic resource]. – Access mode: <http://developer.vuforia.com> (27.10.2019).
- [13] Wagner, D. First steps towards handheld augmented reality / D. Wagner, D. Schmalstieg // 16th International Symposium on Wearable Computers, 2003. – P. 127.
- [14] Yasumuro, Y. Consistent Presentation of Interactive Virtual Objects in Real Space with 3D Markers – Interactive Virtual Interior Design // Digital Image Computing: Techniques and Applications, Proceedings of the VIIth Biennial Australian Pattern Recognition Society Conference, 2003. – P. 653-662.
- [15] Yemez, Y. 3D reconstruction of real objects with high resolution shape and texture / Y. Yemez, F. Schmitt // Image and Vision computing. – 2004. – Vol. 22(13). – P. 1137-1153.
- [16] Zhou, F. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR / F. Zhou, H.B.L. Duh, M. Billinghurst // Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, 2008. – P. 193-202.