

Real-time Simulation System of High Fidelity 3D Virtual Fitting Based on MR

Huixin Zhang*, Rui Jing

NorthChina Electric Power University, BaoDing 130600, China.

Abstract

In recent years, with the rapid development of economy, people have higher and higher requirements for wearing, as well as a growing number of materials and types of clothes are available. However, it's so unexpected that shopping malls appear crowded fitting rooms and other terrible phenomena. In order to reduce the occurrence of the similar phenomena, the real-time simulation system of high fidelity 3D virtual fitting based on MR came into being, which aims to simplify the fitting process by using virtual fitting to help people quickly find the right clothes for themselves.

Keywords

Hololens, Virtual Simulation, Unity3D, Human Modeling.

1. Introduction

High realistic 3D virtual fitting real-time simulation system has extraordinary significance for modern society. In recent years, with the development of non-physical clothing stores, remote shopping is no longer out of reach. You can just pick up your phone and shop easily. However, just as the two sides of the coin, the blossoming of the online economy has also brought some problems. The first is the virtual nature of online shopping, which has led to a spike in return rates due to failure to fit. The high frequency of returning goods brings harm to both merchants and consumers. For merchants, not only does the frequent return of goods by consumers damage the freight charges of merchants, consumes human and material resources, but also affects the reputation of merchants to some extent. For consumers, a long wait for the purchase of goods does not bring a happy result, but requires more time to return. This makes the DEVELOPMENT of 3D virtual fitting system more urgent.

The use of virtual try-on is very wide, which can be summarized as the following aspects: in the field of combining computer animation and games, virtual clothing can bring users a sense of experience closer to the real world; for fashion designers, the cycle of clothing design is reduced greatly, and the beauty and ugliness of the clothing and the suitability of the clothing can be judged in advance, which provides the possibility to design clothing on a large scale as well; for e-commerce platforms and clothing merchants, the realization of virtual try-on can reduce the number of returns and exchanges helpfully and increase revenue effectively; for users, it improves the accuracy of buying clothes, which helps consumers determine whether the size and style meet their own ideas and expectations. Therefore, the concept of virtual try-on can effectively solve these problems of the majority of e-commerce companies, at the same time, it brings the customer a special physical fitting experience. What's more, virtual try-on has high commercial value and practical significance. Making virtual fitting apply to the fact is already an inevitable reality.

2. The Way of Implementation

2.1 The Creation of Character Models

First of all, the character model is parameterized, and various data of the human body are obtained through 3D scanning technology or human body database. Then the human body is modeled

according to the parameters[1],the character model is divided into several parts, namely, height, chest, waist, hip circumference, weight, etc.the rest can be designed auxiliary parameters, such as arm length, thigh circumference, calf circumference and so on, and stretching bone makes changes to the character model. The relevant code is as follows.

```

1. public void setsize(string bodyparts)//bodyparts 航选参致
2. {
3. factor[0]=UIChangeProxy. instance. HumanParams. Height;
4. factor[1]=UIChangeProxy. instance. HumanParams. Bust;
5. factor[2]=UIChangeProxy. instance. HumanParams. Waist;
6. factor[3]=UIChangeProxy. instance. HumanParams. Hips;
7. factor[4]=UIChangeProxy. instance. HumanParams. Wight;
8. factor[5]=UIChangeProxy. instance. HumanParams. Arm;
9. factor[6]=UIChangeProxy. instance. HumanParams. Upthin;
10. factor[7]=UIChangeProxy. instance. HumanParams. Lowthin;
11. switch (bodyparts)
12. {
13. case
14. "CHANGETYPEes HEIGHT":TryClothManager. setClothPositon(cloth,factor[0]);
15. Bodymanager.ChangeSkeleton(new Vector3(100, 100, 100), Skeleton0,
16. UIChangeProxy.instance.HumanParams.Height); return;// bone deformity
17. case "CHANGETYPEes WAIST":Bodymanager.SetBlenshape(FromBody, 1,
18. UIChangeProxy.instance.HumanParams.Waist); break;
19. case "CHANGETYPE — RUSH":Bodymanager.SetBlenshape(FromBody, 0,
20. UIChangeProxy.instance.HumanParams.Bust); break;
21. case "CHANGETYPE — RM,,:Bodymanager.SetBlenshape(FromBody, 2,
22. UIChangeProxy. instance. HumanParams.Hips);break;
23. case "CHANGETYPEes UPTHIN":Bodymanager. SetBlenshape(FromBody,3,
24. UIChangeProxy. instance. HumanParams.Hips);break;
25. case "CHANGETYPE — OWTHIN":Bodymanager.SetBlenshape(FromBody, 4,
26. UIChangeProxy. instance. HumanParams.Hips);break;
27. case "CHANGETYPEes WEIGHT":Bodymanager.SetBlenshape(FromBody, 5,
28. UIChangeProxy. instance. HumanParams.Hips);break;
29. default: break;
30. }
31. Bodymanager.ChangeBodyCollider(colliderDateDeltaO, colliderDateDeltaI,
32. colliderDateDelta2, colliderDateDelta3,
33. colliderDateDelta4,colliderDateDeltaS,sphereChild, factor);// Impact sphere
    deformation
34. }

```

2.2 The Realization of Fitting Function

It is a relatively complicated process to change the costume of the character model, but a fitting demo is given by Unity official. From this demonstration, we can see that the reason for trying is that a common set of bones is shared. Then, when we pack, we separate the bones from the model and then pack the parts of the model separately. For example, the character packs the basic bones into a single package, and then packs the rest models of the body parts into other single packages separately. If the model consists of three parts: head, face and body, then there will be four resource packages after packing, namely basic skeleton, head, face and body, so the body parts of the model can be switched

separately. for this fitting system, I divided the character model into three parts, :upper body, lower body and shoes.

Click on the appropriate clothing, the system needs to provide the corresponding 2D picture of the clothing as the button to trigger the event. In this process, the following data needs to be replaced. Back to the theme of character model packaging, suppose a male character has 10 sets of clothes to change, but the common skeleton is a set, so at the time of packaging, the set of bones is still individually wrapped, and then all the other models are packaged. in a word, one set of bones corresponds to N models. These N models all share this set of bones. The practical realization of the dressing is to replace the corresponding body part. For example, if you want to change the top, merge the new top with a model other than the top top. In other words, replacing one part of the body is actually updating the entire character model. One thing to note here is that there is no need to update the base skeleton. This is also the implementation principle of this fitting system.

The relevant implementation code is as follows:

```
1. using System.Collections;
2. using System.Collections.Generic;
3. using UnityEngine;
4. // There are clothing categories such as pants, tops, shoes, body parts and so on
5. public class ClothesClass : MonoBehaviour
6. {
7.     public GameObject hideObj; // Wearing pants or a skirt usually hides the underwear
    because the overlap can be abnormal
8.     public List<GameObject> clothesList; // A list of all such clothes
9.     // Button click event
10.    public void ButtonClick(string aa)
11.    {
12.        // If you have clothes, you need to hide them all first
13.        foreach (var item in clothesList)
14.        {
15.            item.SetActive(false);
16.        }
17.        // Then the button corresponds to the garment
18.        clothesList[int.Parse(aa)].SetActive(true);
19.        // If you need to hide it, hide it like model underwear
20.        if (hideObj)
21.        {
22.            hideObj.SetActive(false);
23.        }
24.    }
25. }
```

2.3 Free Viewing and Model Scaling Implementations

As mentioned above, we need to zoom in and out after the characters have tried it on, and rotate the angle of view freely, so that we can observe every angle of the clothes, which is often referred to as the camera Angle. In the initial state, the camera's Angle of view remains directly facing the human body, and to do that requires a response. Generally speaking, in the Unity engine, there are two options to rotate the perspective freely. The first way is that when we move the mouse, the character doesn't move and the perspective moves. The second method is to move the mouse, the character with the mouse turns ,keeping the perspective unknown. This project adopts the first method of implementation.

The relevant implementation code is as follows:

```

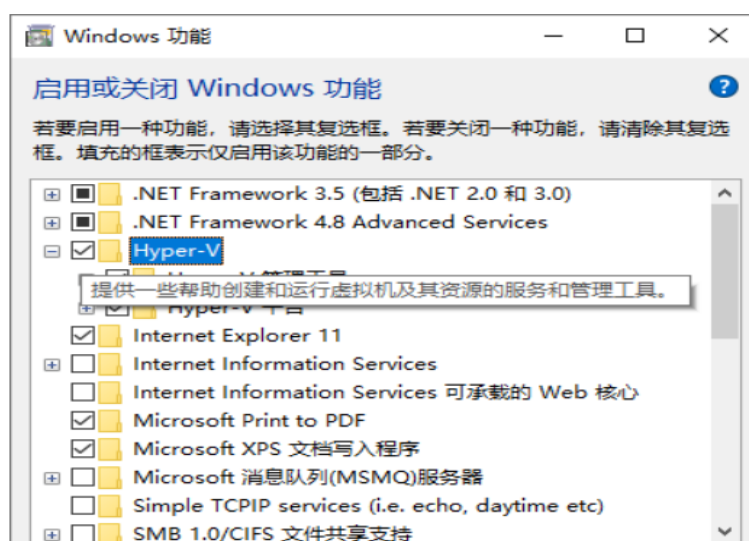
1. using System.Collections;
2. using System.Collections.Generic;
3. using UnityEngine;
4. public class ControlCamera : MonoBehaviour
5. {
6.     float speed = 20;
7.     public float distance_v;
8.     public float distance_h;
9.     public float rotation_H_speed = 1;
10.    public float rotation_V_speed = 1;
11.    public float max_up_angle = 80;
12.    public float max_down_angle = -60;
13.
14.    private float current_rotation_H; // Horizontal rotation result
15.    private float current_rotation_V; // Vertical rotation result
16.
17.    public Transform target; // Get the rotation target
18.    private void Update()
19.    {
20.        camerarotate();
21.        camerazoom();
22.    }
23.
24.    private void camerarotate() // The camera rotates around the target
25.    {
26.        // The camera rotates around the target
27.        transform.RotateAround(target.position, Vector3.up, speed * Time.deltaTime);
28.        var mouse_x = Input.GetAxis("Mouse X"); // Gets the X-axis mouse movement
29.        var mouse_y = -Input.GetAxis("Mouse Y"); // Gets the Y-axis mouse movement
30.        if (Input.GetKey(KeyCode.Mouse1)) // When pressing the right mouse button for moving the
            camera position
31.        {
32.            transform.Translate(Vector3.left * (mouse_x * 15f) * Time.deltaTime);
33.            transform.Translate(Vector3.up * (mouse_y * 15f) * Time.deltaTime);
34.        }
35.        if (Input.GetKey(KeyCode.Mouse0)) // When pressing the right mouse button for moving the
            camera position
36.        {
37.            transform.RotateAround(target.transform.position, Vector3.up, mouse_x * 5);
38.            transform.RotateAround(target.transform.position, transform.right, mouse_y * 5);
39.        }
40.    }
41.    private void camerazoom() // Camera wheel zooming
42.    {
43.        if (Input.GetAxis("Mouse ScrollWheel") > 0)
44.            transform.Translate(Vector3.forward * 0.5f);
45.        if (Input.GetAxis("Mouse ScrollWheel") < 0)
46.            transform.Translate(Vector3.forward * -0.5f);
47.    }
48.    }

```


2.4 Transferability

By combining the virtual fitting system with HoloLens, the so-called holographic virtual fitting system can be implemented. During the migration process, we needed to package the Unity project and build an XR-enabled UWP solution before we could generate a HoloLens simulation for Visual Studio to launch.

Before porting, we should first determine the required software and environment: Visual Studio 2019, Unity 2018.1.1, HoloLens 2 Emulator 10.0.18362.1082, Windows SDK (10.0.18362.0). If you are using Windows Home, you need to make some adjustments—download the Hyper-V for the first time. After downloading, you need to enable this feature in Windows Enabled and closed, as shown in Figure 4-2. Next, you need to enable virtual hardware. After entering BIOS, find Security, start Intel (R) Virtualization Technology to save and exit. If the startup fails in VS, you can try to find all project.lock.JSON configuration files in the published project and change the line "UAP,Version=v10.0.xxXXX" to "UAP,Version=v10.0" to confirm.



3. Test

3.1 Test Specification

The test of this project mainly lies in the following links:

- (1) The project expands the testing from two aspects: Functional test and Performance test.
- (2) Functional Test: Firstly, look at the requirements of the system, taking the requirements as a starting point, then use the white box testing scheme, and standing from the user's perspective, operate and use the functions of the system and then observe the results.
- (3) Performance Test: With the client of the system as the main body, through the start-up of the system, observe the loss of computer resources in the process of system running.

3.2 The Function and Performance Test of the System

(1) Functional Test:

- ① Start the system in this machine. When the system is turned on, the model and UI interface are shown in Figure 5-1.
- ② After that, we select the clothing model and try it on, and the result is shown in Figure 5-2.
- ③ Change the character model and try it on again. The result is shown in Figure 5-3.
- ④ By sliding and dragging the mouse wheel, the size of the character and the camera Angle can be changed to freely observe the fitting of the character from the ground. The result is shown in Figure 5-4.

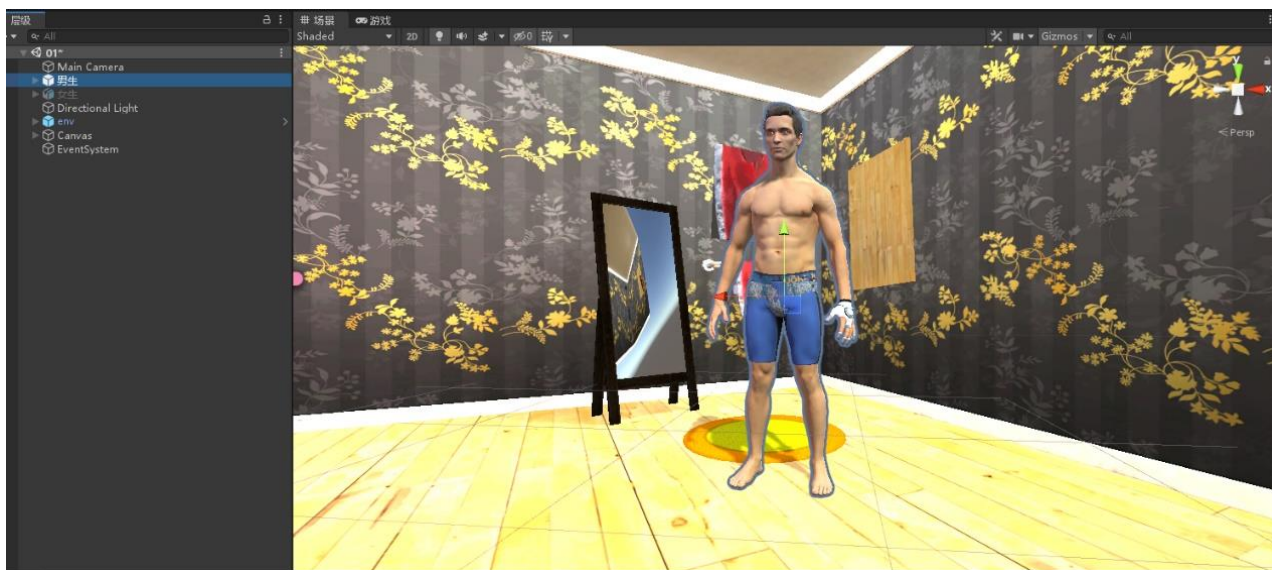


Figure 5-1 The startup page of the system is male model by default



Figure 5-2 The result of the first fitting



Figure 5-3 Changing the character model

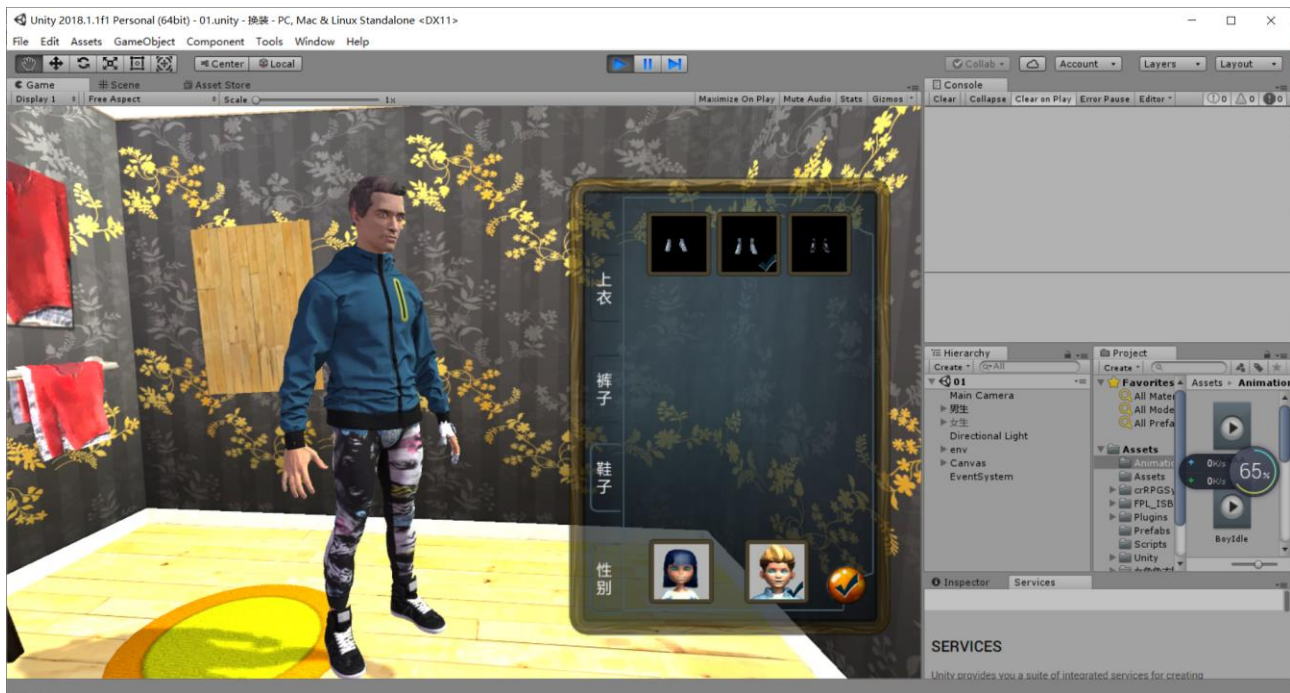


Figure 5-4 Demonstration effect from a free perspective

A summary of all the above fitting results is shown in the table below.

Table 5-1 System function test results

TestProject	Operation	Transformation	Result
enter the system	Run the System	Entered the homepage of the system	Successfully√
Virtual fitting of clothing	Click on the clothing model to change	Default model dress updated	Successfully√
Change the character model	Click on the character model to change	Character model replaced	Successfully√
Change clothes model	Secondary change of clothing	Clothes to replaced	Successfully√
Angle rotation	Drag the mouse/slide the pulley	The Angle of view changed	Successfully√

(2) Performance Test: In the performance test, we need to use the Profiler toolbox provided by Unity to observe the loss of the running line of this system. As shown in Figure 5-5.

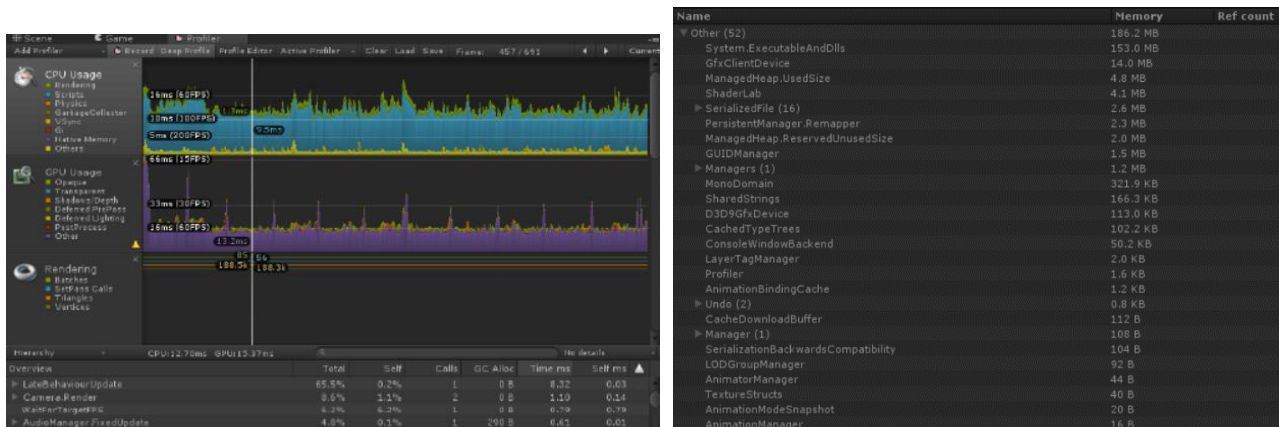


Figure 5-5 Profiler toolkit test results

3.3 Discussion of Test Results

First of all, the entire operation of the system has not been a serious error caused by the crash, in accordance with a system software for the reliability of the basic provisions. The results of the system test show that the virtual fitting system can meet the basic application requirements of customers, that is, customers can adjust the size of the body in a parameterized design. In addition, the same size of clothing can be tested by different bodies. Customers can also view the final results at will, having a good experience of human-computer interaction technology. The functional test results of the system show that the operation of the system is relatively smooth, and the loading of solid model resources and the operation of code take up a relatively small amount of CPU and GPU resources. However, the system can quickly 3D render fabric-based clothing models due to the large amount of running memory resources. Considering that there are many intermediate values in the simulation calculation of the fabric, it will occupy a lot of storage space, and system memory has more solid model and texture image resources, will also consume storage space. In addition, there is room for further improvement during the smooth time of the arrival simulation.

4. Prospects

In the context of the rapid development of the Internet economy, the virtual fitting real-time simulation system has a broad development space and brings people a better shopping experience because it can simplify the shopping process and minimize unnecessary shopping links.

References

- [1] Min Tang, Huamin Wang, Le Tang, Ruofeng Tong, Dinesh Manocha. CAMA: Contact-Aware Matrix Assembly with Unified Collision Handling for GPU-based Cloth Simulation[J]. EUROGRAPHICS, 2016.
- [2] Minsang Kim¹, Nak-Jun Sung, Sang-Joon Kim, Yoo-Joo Choi, Min Hong. Parallel cloth simulation with effective collision detection for interactive AR application [J], Multimed Tools Appl. 2018.
- [3] G. Song. 3D Virtual fitting Room based on unity3D Platform [D]. Wuhan Textile University. 2016.