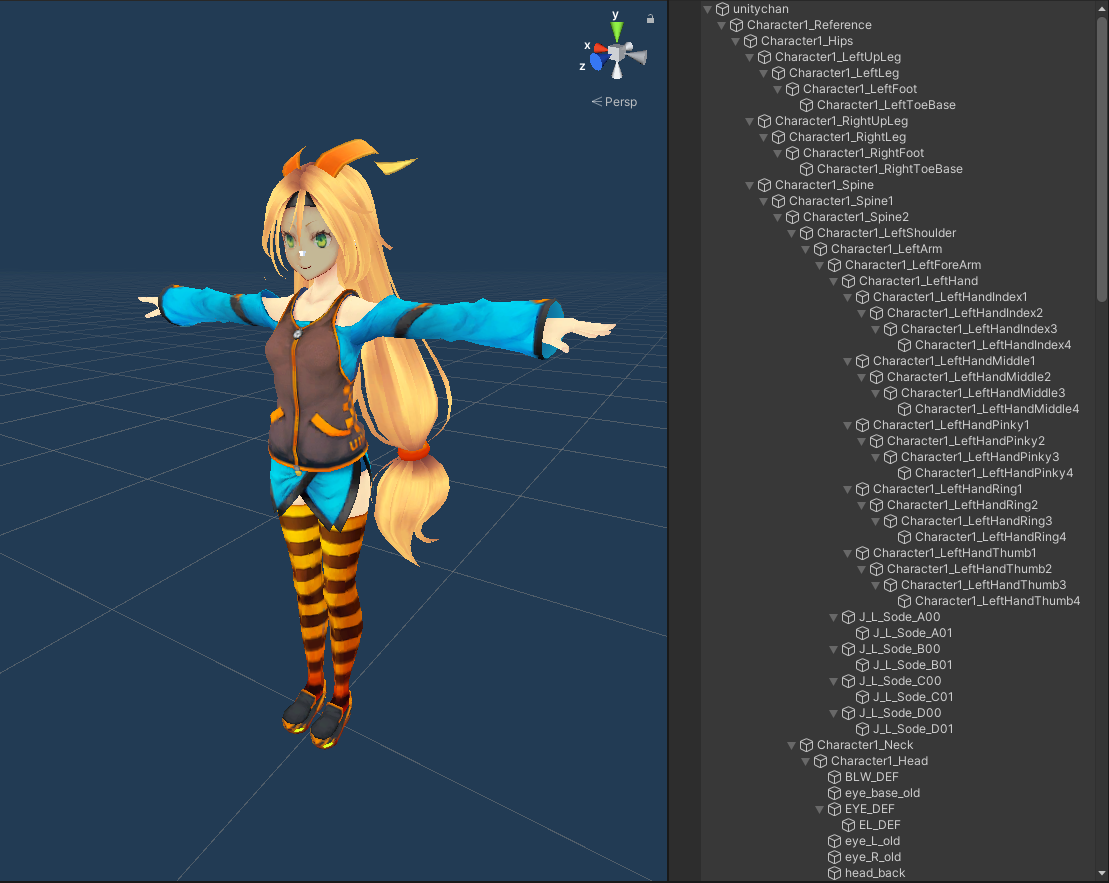
Technical Details in Game

1. Matching Output with Game Character

We chose “Unity Chan,” who was released officially by Unity for free, as our main character in game. She was built meticulously with 153 joint points, composed of organs and clothing (Figure 1-1). Her body structure was too complete for our model, which includes only 24 joint points, thus the coordinates of most of her body parts were calculated. Below are some of our estimations:

1. Hip: Locate hip position as the middle of the roots of left and right thighs. Adjust the z-positional offset with the bottom of her spine.
2. Neck: Locate the neck position with both shoulders.
3. Head: Locate the head parental position with the top of her head and the position of her neck.
4. Facial feature: The positions follow the head position which follows the neck position.
5. Spine: Locate the spine parental position with her upper abdomen.
6. Fingers and toes: The position of fingers and toes follow their parent components.
7. Clothing: Her upper clothing follows her shoulders and spine, when her lower clothing follows her spine, hip, and legs.

(Figure 1-1: Unity Chan and part of her components)

1. Shifting Offset for Gaming

Since a parkour game requires wide space to show players’ movement, our game was designed with a large scene. Nevertheless, webcam wasn’t able to capture a huge enough area. We had to scale up the offset when character shift horizontally. Based on the shifting offset of the hip position from the previous position, we adjust the multiplication until it looked suitable in gaming (Figure 1-2).

Furthermore, due to the skeleton mechanism in Unity, Unity Chan could traverse her whole body according to the shifting of her hip. In this case, her body wouldn’t be folded or unnaturally stretched.



(Figure 1-2: Shift Scaling)

1. Looking Rotation of Joint Points

Although the coordinate of each organ was set perfectly, the rotation should be appropriate to look natural. Here, we use the ‘LookRotation’ function to specify the forward and upwards direction. Below are some implementations:

1. Hip: Forward direction was the triangle normalization of hip position, left and right thigh bending position.
2. Gaze: Gazing direction was set as the vector from head position to nose position.
3. Hands: Forward direction was the vector from thumb to middle finger position. Upward direction was the triangle normalization hand position, middle finger position, thumb position.

With these looking rotation adjustment, Unity Chan could move naturally with no abnormal distortion.