

详细设计文档

一、系统完整功能描述

利用 WebGL 创建手部(包含手臂)模型。设置两个板块，包括“手动调节”板块与“动画”板块。“手动调节”模块实现手臂、手掌、各手指关节的调节，以及主体缩放、特定手势显示等功能，并设置按钮恢复初始状态。“动画”板块分为“石头剪刀布”动画模块和“抓取”动画模块。“石头剪刀布”动画模块可随机生成三种手部形态，模拟“石头”、“剪刀”、“布”游戏过程。“抓取”动画模块，模拟手部抓取动作。并在这些功能基础上进行背景音乐等设置。

二、各模块流程

“手动调节”板块：

手臂调节旋转时，因为手臂是根部，其他部件都是在此基础上搭建的，所以所有其他部件会跟着该手臂进行同步旋转相同的角度。

手掌转动时，其他手指都是在此基础上搭建，而手臂不受其影响，手指和手掌进行同步旋转。

手指以食指为例，第一关节的转动不会影响手臂和手掌，只会带动第二关节和第三关节的转动。第二关节自会带动第

三关节的转动。第三关节不会带动任何关节的转动。

“动画”版块：

用户可自行选择“石头剪刀布”游戏动画或“抓取”动画。

“石头剪刀布”游戏动画状态时，按照现实游戏的操作，先是预备动作，手呈现石头状态晃动一定秒数，最后手呈现剪刀、石头、布的三种状态之一，状态选择依靠随机数的选取 $-\text{Math.floor}(\text{Math.random()} * 3)$ ，随机数在 $[0, 3)$ 之间随机区，再向下取整获得数字 0、1、2 分别对应剪刀、石头、布三种状态。

```
var flag = -1;
function shakehands() { //石头剪刀布动画
    var t = 1;
    timel = setInterval(function () {
        t = 1;
        stone();
        if (theta[UpperBase] >= 0) d = -10;
        else if (theta[UpperBase] <= -120) {
            d = 10;
            flag++;
            if (flag == 3) {
                clearInterval(timel);
                flag = 0;
                t = Math.floor(Math.random() * 3);
                t=0;
                if (t == 0) scissors();
                if (t == 1) stone();
                if (t == 2) cloth();
            }
        }
    }, 1000);
}
```

```

        theta[UpperBase] += d;
    }, 100);

    t1 = setTimeout(shakehands, 9000);
}

```

“抓取”动画状态时，模拟手部抓取动作，初始角度为 10° ，当角度超过 90° 时，进行手部“抓”动作的模拟，当角度为 0° 时，进行手部“放”动作的模拟。

```

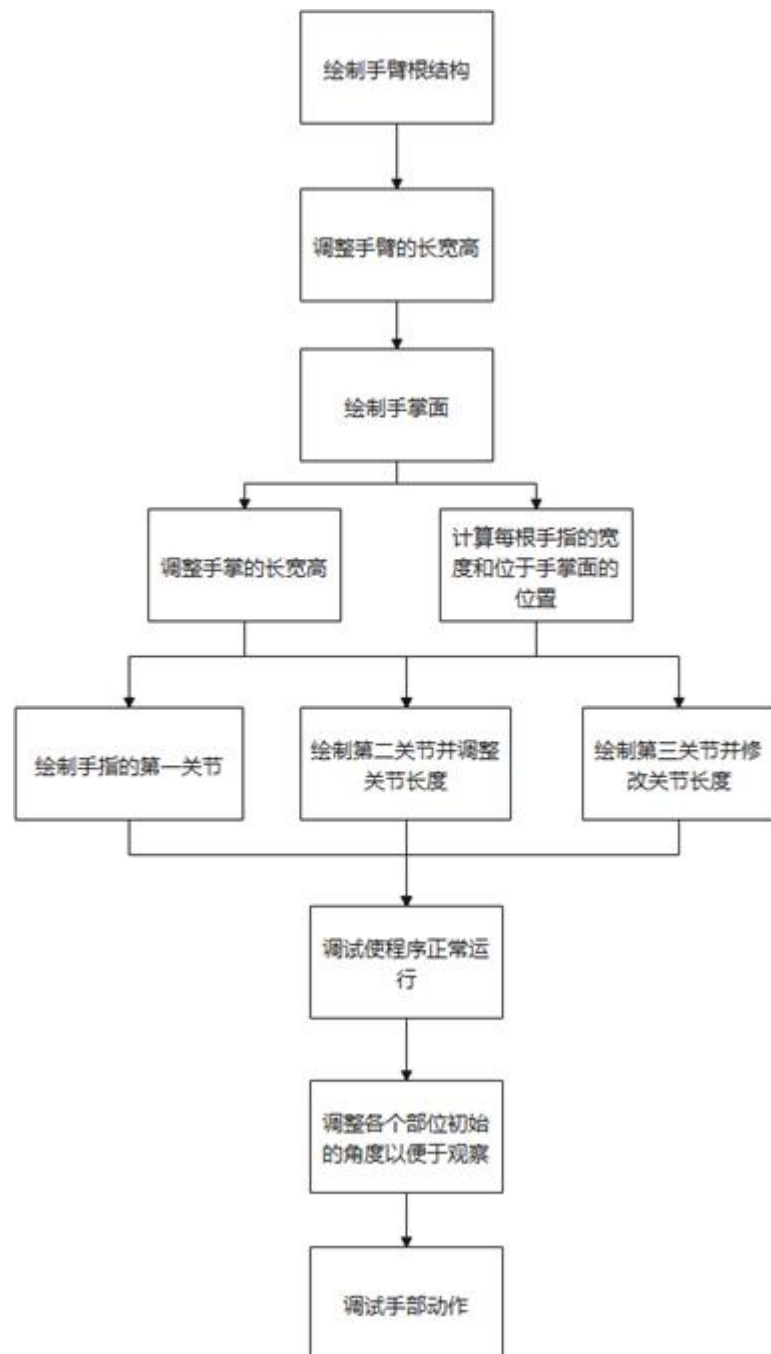
var ba = 10;
function trab(){//抓取动画
    time2 = setInterval(function(){
        for(var i = 2;i<=15;i++){
            if(theta[i]==90) ba = -10;
            if(theta[i]==0) ba = 10;
            theta[i]+=ba;
        }
    },600);
    t2 = setTimeout(trab,9000);
}

```

二、结构具体实现

具体实现逻辑为：首先绘制手臂根结构，调整手臂的长宽高再绘制手掌面，调整手臂的长宽高、计算每根手指和位于手掌面的位置后绘制手指的第一关节、绘制第二关节并调整关节长度、绘制第三关节并调节关节长度。随后，进行调试程序以正常运动，并调整各个部位的初始角度以便观察。

三、



四、关键函数

`setTimeout(shakehands, 9000);`

绘制手臂：`upperArmbase()`

绘制手掌：`lowArmbase()`

第一关节-大拇指绘制函数: lthumb()

第一关节-食指绘制函数: lindex_finger()

第一关节-中指绘制函数: lmiddle_finger()

第一关节-无名指绘制函数: lring_finger()

第一关节-小拇指绘制函数: llittle_finger()

第二关节-大拇指绘制函数: uthumb()

第二关节-食指绘制函数: uindex_finger()

第二关节-中指绘制函数: umiddle_finger()

第二关节-无名指绘制函数: uring_finger()

第二关节-小拇指绘制函数: ulittle_finger()

第三关节-食指绘制函数: third_index()

第三关节-中指绘制函数: third_middle()

第三关节-无名指绘制函数: third_ring()

第三关节-小拇指绘制函数: third_little()

手势设置-数字 1: numb1()

手势设置-数字 2: numb2()

手势设置-数字 3: numb3()

手势设置-数字 4: numb4()

手势设置-数字 5: numb5()

手势设置-石头状态: postStone()

动画选择: animation()

“抓取”动画设置: trab()

“石头剪刀布”动画设置: shakehands()

显示剪刀状态: scissors()

显示石头状态: stone()

显示布状态: cloth()

五、关键变量

手臂对应的角度位置: UpperBase

手掌对应的角度位置: LowerBase

以手臂为例, 高度、宽度、深度:

手臂宽度: UPPER_ARM_WIDTH

手臂高度: UPPER_ARM_HEIGHT

手臂长度: UPPER_ARM_LONG

弯曲的初始值: $\theta = [-120, 0, 30, 30, 30, 30, 30, 15, 15, 15, 15, 15, 15, 0, 0, 0, 0]$

第一关节-大拇指: l_thumb

第一关节-食指: l_index_finger

第一关节-中指: l_middle_finger

第一关节-无名指: l_ring_finger

第一关节-小拇指: l_little_finger

第二关节-大拇指: u_thumb

第二关节-食指: u_index_finger

第二关节-中指: `u_middle_finger`

第二关节-无名指: `u_ring_finger`

第二关节-小拇指: `u_little_finger`

第三关节-食指: `t_index_finger`

第三关节-中指: `t_middle_finger`

第三关节-无名指: `t_ring_finger`

第三关节-小拇指: `t_little_finger`

动画周期: `time1`

动画选取-为 1 时为“石头剪刀布”动画，为 2 时为“抓取”

动画: `choose`

六、实现工具、开发环境、工具库

开发工具:

Visual Studio Code: 一个运行于 Mac OS X、Windows 和 Linux 之上的, 针对于编写现代 Web 和云应用的跨平台源代码编辑器。

开发环境:

Google 浏览器

主要工具库:

WebGL, MV.js