邹天舒

指导老师：魏祥

实验报告

机器学习之位置识别

**目录**

[**1. 需求分析 2**](#_Toc38906659)

[**2. 代码 2**](#_Toc38906660)

[**1) 滑动条 2**](#_Toc38906661)

[**2) 获取两个训练样本 2**](#_Toc38906662)

[**3) 训练展示 2**](#_Toc38906663)

[**4) 目录结构 3**](#_Toc38906664)

[**3. 运行截图 4**](#_Toc38906665)

[**4. 实验总结 4**](#_Toc38906666)

# 需求分析

基本需求：

1.    具有两个bar分别位于canvas的水平方向和垂直方向作为拟合位置输出。

2.    训练模型，能够令标定方块在二维平面内移动，并且实现跟踪待识别物体位置的功能。

# 代码

## 滑动条

因为需要获取水平和垂直两个方向的训练样本，所以就需要创建两个滑动条代码如下：

*slider1* = createSlider(0, 1, 0.5, 0.01);  
*slider2* = createSlider(0, 1, 0.5, 0.01);

这两个滑动条分别获取水平垂直的方向。

## 获取两个训练样本

*feature1* = ml5.featureExtractor('MobileNet', *ModelReady*);  
*feature2* = ml5.featureExtractor('MobileNet', *ModelReady*);  
predictor1 = *feature1*.regression(*video*, *videoReady*);  
predictor2 = *feature2*.regression(*video*, *videoReady*);  
  
  
*slider1* = createSlider(0, 1, 0.5, 0.01);  
*slider2* = createSlider(0, 1, 0.5, 0.01);  
addButton = createButton('Add');  
addButton.*mousePressed*(function () {  
 predictor1.addImage(*slider1*.value());  
 predictor2.addImage(*slider2*.value());  
});

如上代码，将设置两个样本添加的类别，然后分别通过value()获取到滑块的值，添加到不同的分类上面。

## 训练展示

因为有两个训练样本，所以我们需要有两个获取训练结果的函数，分别保存彼此的值

function *GetResult1*(error, results) {  
 if (error) {  
 *console*.error(error);  
 } else {  
 *console*.log(results);  
 *value1* = results;  
 *fill*([255, 0, 0]);  
 *textSize*(30);  
 }  
}  
function *GetResult2*(error, results) {  
 if (error) {  
 *console*.error(error);  
 } else {  
 *console*.log(results);  
 *value2* = results;  
 *fill*([255, 0, 0]);  
 *textSize*(30);  
 }  
}

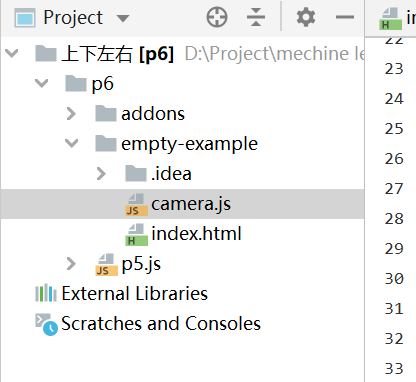
当我们点击训练按钮触发事件，就可以开始训练了。

对于视频上的小白块，我们可以通过两个训练出来的value值计算他们的位置，然后就可以让滑块根据位置来移动了。

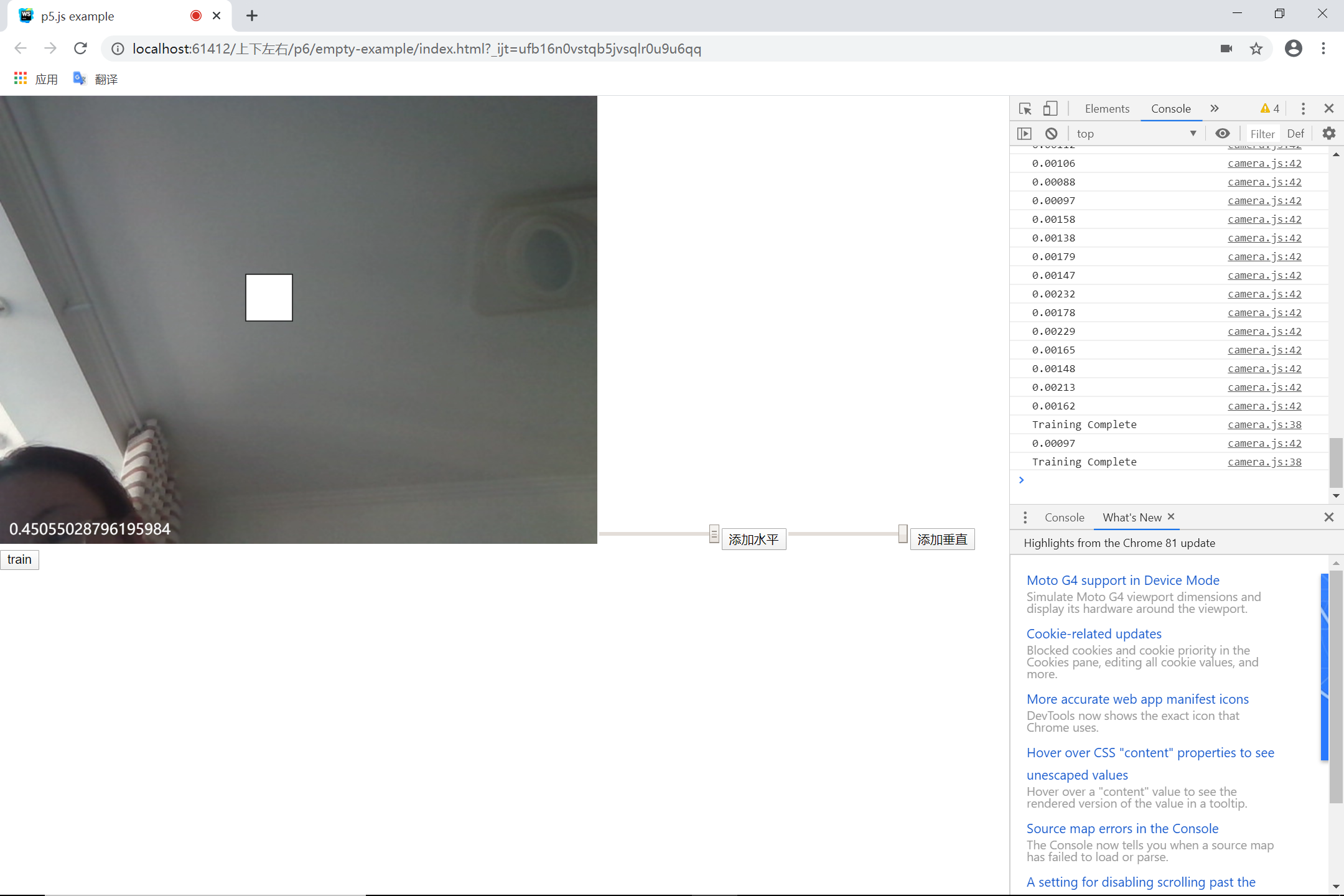
*rect*(*value1* \* width, height \* *value2*, 50, 50);

## 目录结构

本次实验的目录结构如下图：



# 运行截图



主要运行见文件中的录屏。

# 实验总结

通过这次实验，我实现了小滑块的移动，了解了深度迁移学习以进行回归训练的相关知识。