**基于手势识别的虚拟拖放拍照技术**

姓名

班级

指导老师：

**[摘要]：**手势是一种可以传达信号的人际交流模式，通过手势识别可以实现人机交互的功能。本文是基于MediaPipe框架下实现的虚拟拖放技术，并把虚拟拖放技术应用拍照的四个模块上，分别是：控制音量、双边滤波处理、自适应环境处理和漫画风处理。处理结果显示，该技术具有良好的处理性能，可以搭载在移动端适应于不同环境下的手势拍照，为远程拍照提供了一种新型可交互形式。

**[Abstract]：**Gesture is an interpersonal communication mode that can convey signals, and human-computer interaction can be achieved through gesture recognition. This paper is based on the virtual drag-and-drop technology implemented under the MediaPipe framework, and applies the virtual drag-and-drop technology to four modules of photo-taking, namely: volume control, bilateral filter processing, adaptive environment processing and comic wind processing. The processing results show that the technology has good processing performance and can be mounted on the mobile side to adapt to gesture photography in different environments, providing a new form of interactivity for remote photography.

**关键词：**手势识别; 双边滤波; 直方图均衡化; 边缘检测

1. **设计思想**

若要实现在电脑端的手势识别，首先需要调用电脑端的视频流，基于视频流的每一帧进行实时的手势识别；其次需要对虚拟拖放技术进行封装，每一次对虚拟方块的拖动都是一次接口的调用，实现软件的开闭原则，否则会大大增加耦合度；然后在虚拟拖放的基础上实现抓取某一帧视频照片拍照并对该帧做出相应的图像处理；最后将处理过后的图片保存下来。将上述的四个步骤导出到流程图如图1所示。

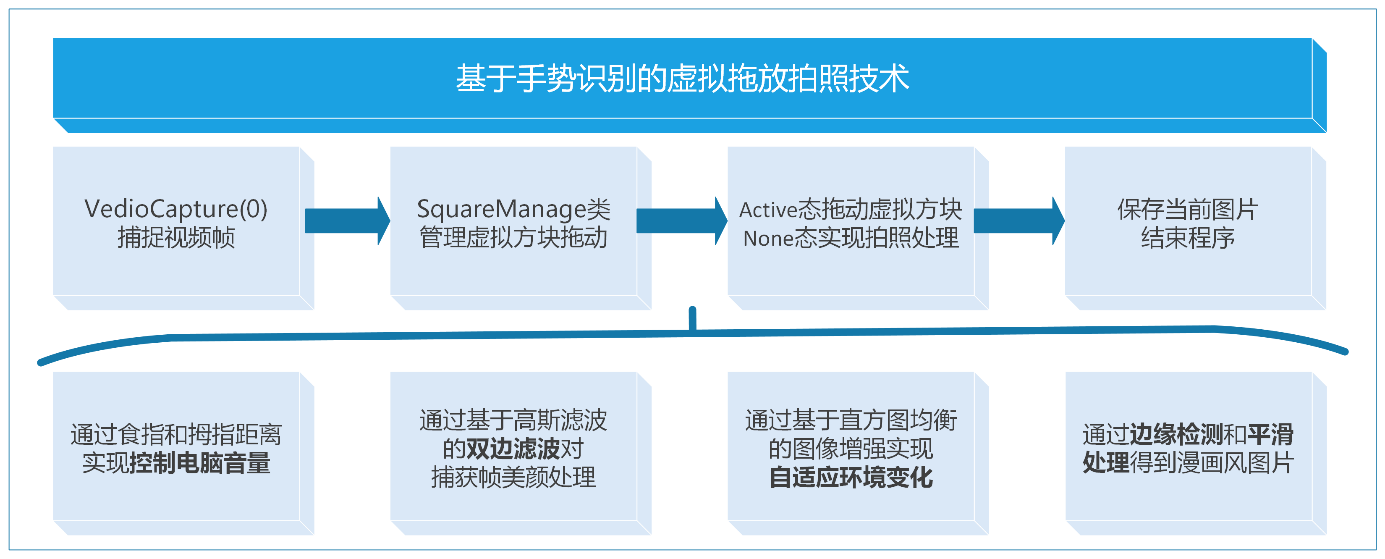


图1 技术实现组成

如图1中的流程图所示，第一行是每次软件运行时的步骤，打开电脑端或移动端视频软件捕捉视频帧，对SquareManage类中方框位置更新接口调用实现虚拟方框拖动，当虚拟方框选中时为Active（激活态），未选中时为None（未激活态），处于激活态的方框可以随着手势跟踪而移动，由激活态跳变到非激活态时（即松开虚拟方框）抓取当前视频帧，对不同的虚拟方框做出第二行的不同处理。下面将依次解释四个模块的具体功能及实现。

**二、手势识别的实现**

**2.1实现流程**

实现手势识别的方法有很多，最初的方法是通过肤色检测出手部区域再轮廓处理找到手的轮廓[1]，具体方法如下：

Step1：RGB空间转换到YCrCb空间；

Step2：在Cr空间内对图像进行高斯算子锐化；

Step3：将图像转成灰色图像；

Step4： 利用OTSU阈值算法处理边缘和掩膜计算；

Step5：通过二值化处理寻找最大轮廓区域即为手势区域。

实现方法如图2算法流程图所示。

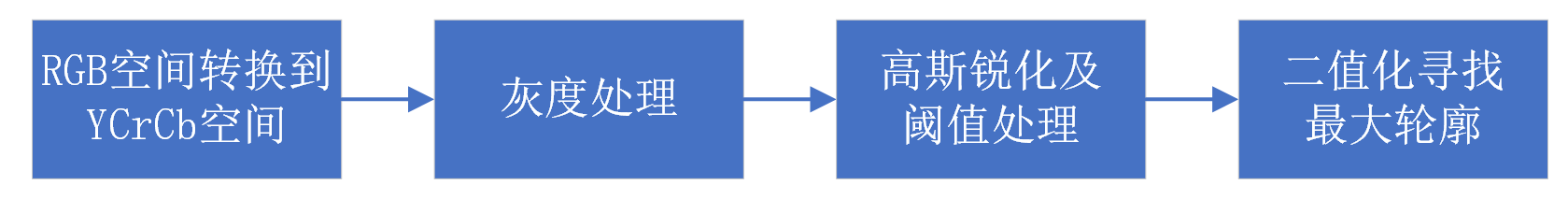


图2 手势识别算法流程

**2.2空间转换**

之所以要把RGB空间转换到YCrCb空间内，是因为在RGB空间里人脸的肤色受亮度影响相当大，所以肤色点很难从非肤色点中分离出来，也就是说在此空间经过处理后，肤色点是离散的点，中间嵌有很多非肤色，如果把RGB转为YCrCb空间的话，可以忽略Y(亮度)的影响，因为该空间受亮度影响很小，肤色会产生很好的类聚。这样就把三维的空间将为二维的CrCb，肤色点会形成一定得形状，如：人脸的话会看到一个人脸的区域，手臂的话会看到一条手臂的形态。

**2.3 OTSU阈值处理**

为将一张灰度图片的前景和背景区分开，我们可以合理的假设为如果将图像的像素分布图画出来，那么图像上应该有两个峰，即前景色和背景色。在这两个峰之间肯定有一个谷，那么我们就可以将阈值设在这里，从而对图像达到一个良好的分割效果。OTSU算法可以求出用这个阈值分割后的两个图像的类间方差[2]。对于每一个可能的阈值，我们计算并取出类间方差最大的那个像素值，此时这个值就可以较好的对图像进行分割。

**2.4二值化与寻找轮廓**

二值化是为了简化寻找轮廓，OTSU阈值处理后的灰度图像进行二值化操作，再将寻找到的轮廓进行排序，最大的轮廓值即为我们需要的手势轮廓。结果如图3、4所示。

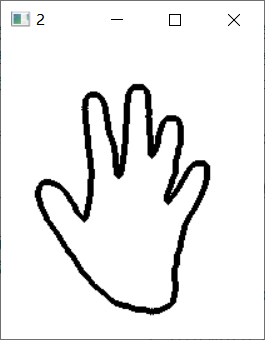
**图片** 

图3 原视频帧 图4 手势识别结果

**2.5基于MediaPipe库的手势识别**

目前Google基于 MediaPipe框架实现了一种全新的手部感知方法，该方法利用机器学习 (ML) 推断出单帧内的21个3D手部关键点，本文以下的图像处理都是在MediaPipe框架下所做。

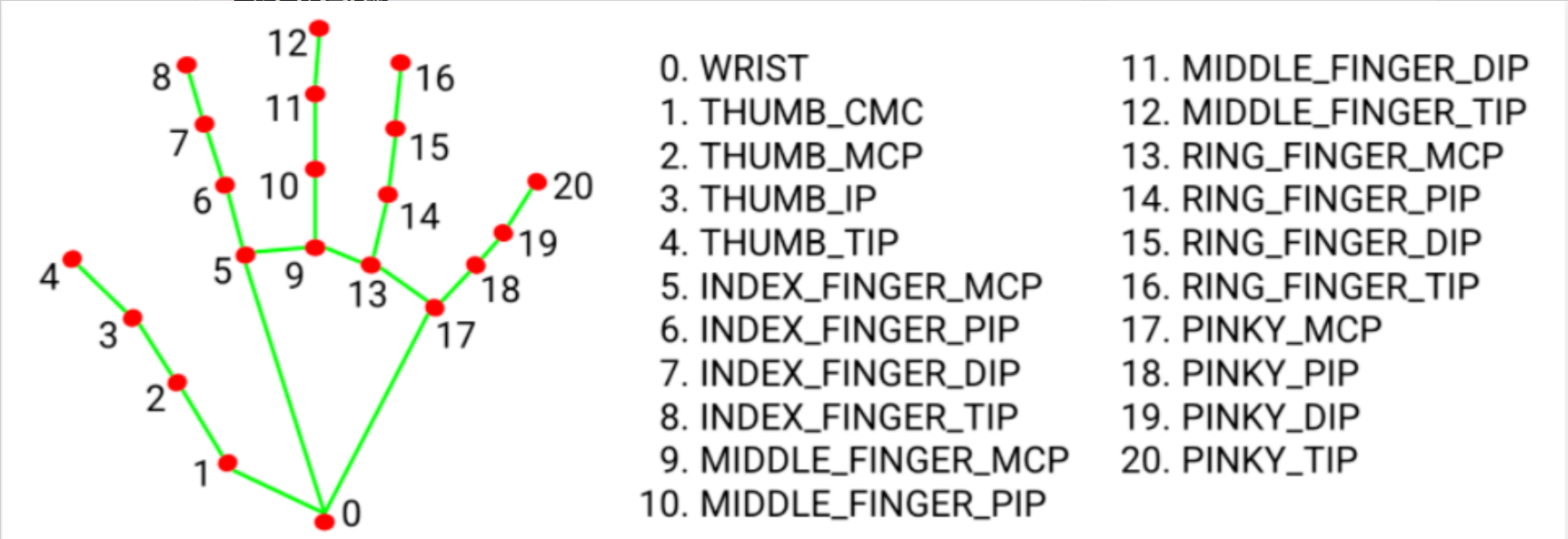


图5 Media Pipe框架下的21个手部关键点

**三、虚拟方块管理**

虚拟方块是通过OpenCV的retangle方法实现的，初始化时开辟一个list记录当前创建方块的长(Length)和宽(Width)，之所以记录每个方块的长和宽是为了在虚拟拖动时可以跟随视频帧实时的更新方块位置。

为实现虚拟拖动方块技术，需要定义何时可以拖动，何时不可拖动，标准如下所示：



：中指尖点的12号的像素点坐标，：食指尖点的8号的像素点坐标。通过勾股定理计算出两点间的距离||，当||大于50时表示不可拖动，||小于等于50时表示可以选中可以拖动，这样就避免了选中后无法取消选择状态的错误。当选中时可返回当前选中虚拟方块的ID号，以此来判断该进行哪种图像处理。

**四、基于MediaPipe框架的模块实现**

**4.1调节电脑音量模块**

模块一是通过大拇指和食指间的距离映射来控制电脑音量，首先在大拇指和食指最大距离时的中点上设置为参考点，以后的每次使用时，若大拇指和食指之间的距离小于此参考点的距离，则为减低音量；若大拇指和食指之间的距离大于此参考点的距离，则为升高音量。

其次为获得大拇指和食指间的距离，可以通过空间的欧几里得范数计算出两点间的距离，再通过pycaw库中找到电脑端中手指的长度范围[20,2000]，系统声音的范围[-65,0](0为最大音量)，这样就可以找到一个从手指距离到系统声音的映射函数，这里可以采用numpy库的interp接口即采用一位线性插值把系统声音范围细分成无数个线段与手指距离范围对应。具体操作界面如图6所示。

**图片**

图6 手势控制音量大小示意图

**4.2双边滤波美颜照片模块**

**4.2.1双边滤波简介**

在模块二部分中采用双边滤波对实时捕获的视频帧进行噪声滤波处理，获得更加平滑的图片，进而实现人脸的美颜祛痘效果。之所以采用双边滤波，是因为中值滤波，高斯滤波，维纳滤波等等这些降噪方法容易模糊图片的边缘细节，对于高频细节的保护效果并不明显。相比较而言，双边滤波器可以很好的边缘保护，即可以在去噪的同时，保护图像的边缘特性。双边滤波（Bilateral filter）是一种非线性的滤波方法，是结合图像的**空间邻近度**和**像素值相似度**的一种折衷处理，同时考虑**空域信息**和**灰度相似性**，达到保边去噪的目的。

**4.2.2双边滤波原理**

双边滤波器之所以能够做到在平滑去噪的同时还能够很好的保存边缘（Edge Preserve），是由于其滤波器的核由两个函数生成：**空间域核**和**值域核**[2]。

1. 空间域核：由像素位置欧式距离决定的模板权值



为当前模板像素的坐标点，为当前模板窗口的中心点，为高斯函数的标准差。该公式生成的滤波器模板和高斯滤波器使用的模板是没有区别的。

1. 值域核：由像素值的差值决定的模板权值



其中，表示图像在点的像素值。不管是空间域核还是值域核的范围均在[0,1]。双边滤波的模板权值是上述两个核的乘积，即：



简化得：

用python实现双边滤波如下图7、8所示，可见原图中的噪音消失效果明显。图9为手势识别下的结果图。

图7 双边滤波处理**前**图像 图8 双边滤波处理**后**图像

图9 手势识别中的双边滤波结果

由实际使用时可以看出滤波在消除噪音的情况下不仅做到局部的模糊且不会损失太多的边缘信息，如红色框内的手，在实际拍照情况下手指上会有很多皱纹和毛孔，经过双边滤波处理之后消除了大部分的皱纹和毛孔，起到美化效果。

**4.3自适应环境模块**

自适应环境拍照是指在拍照环境较暗或者较亮的情况下可以适当加强或者减弱环境亮度，其原理是图像增强中的直方图均衡化：对在图像中像素个数多的灰度值（即对画面起主要作用的灰度值）进行展宽，而对像素个数少的灰度值（即对画面不起主要作用的灰度值）进行归并，从而增大对比度，使图像清晰，达到增强的目的。

直方图均衡化的原理课堂上已经讲授过，这里则不再赘述，下面图10、11为均衡化前和均衡化后的效果图。

**图片**

图10 均衡化前的实景图

**图片**

图11 手势识别下均衡化后的实景图

可见图10中的光照强度越高，图11均衡化后光照亮度会降低。

**4.4漫画风模块**

对比现实中的漫画风，一般是先画出边缘轮廓使整体规划好，再填充颜色使其完整，因此可以依照这种思路实现漫画风格，不过和现实中不同的是本文是把图片一分为二分别进行边缘轮廓处理和平滑处理，再把两张图片拼接起来。下面展示实现细节。

首先提取边缘轮廓。由于漫画中不管是人物还是风景，刻画的细节有限，因此需要把重要以及有特色的部分体现出来，数量要适当。因此先把图片灰度化再二值化，这里通过二值化的方式将轮廓提取出来，采用自适应阈值二值化(adaptiveThreshold)，基于像素周围的小区域确定像素的阈值，可以将有区别的部分的界限提取出来，恰如漫画对象中黑色粗体轮廓，且细节得当。因阈值处理只能针对灰度图像，因此需要先将彩色图像转换为单通道的灰度图像，且为了去除描绘对象内部的冗余细节，还要对图像进行平滑处理，使颜色过度得缓慢一些，这样得出的轮廓就比较好。

其次进行填充颜色。按照现实中绘制漫画风，边缘轮廓描绘之后只需要在轮廓中填充颜料即可，因此这里我们采用图像金字塔对原图降低分辨率，再用双边滤波多次平滑处理去除噪声。这里需要注意的是分辨率降低后图像会变小，因此最后要将图像放大为原来的大小，虽然图像金字塔有专门的方法可以将图像放大，但是尺寸可能会有一两点变化，合并过程中要两个图像完全一样大。

最后把两次处理过的图片拼接起来即是漫画风后的图片。结果图如图12、13、14所示。

图12 原图 图13 漫画风后

**图片**

图14 手势识别下的实景漫画风

**五、总结**

此次数字图像处理大作业讲手势识别和图像处理相结合，实现了虚拟方块拖放技术下的各类图片处理。经过大作业的练习，初步掌握了MediaPipe框架下的手势识别部分，当然MediaPipe不仅仅局限于手势识别还有其他更多的肢体识别，如脸部特征点、四肢特征点等等，可以在MediaPipie基础上实现更多天马行空的工程，未来，若从事机器视觉研究，可以把MediaPipe框架和深度学习中的图神经网络结合起来，做更深层次的图像处理、识别、预测等工作。

本人实现的手势识别下的虚拟拖放技术仅能算一次抛砖引玉的实验，希望未来能有更多的同学在手势识别的启发下做出更多更妙的项目。

**参考文献**

[1]王如斌,窦全礼,张淇,周诚.基于MediaPipe的手势识别用于挖掘机遥操作控制[J/OL].土木建筑工程信息技术:1-8[2021-12-31].

[2]刘健庄,栗文青.灰度图象的二维Otsu自动阈值分割法[J].自动化学报,1993(01):101-105.DOI:10.16383/j.aas.1993.01.015.

[3]王玉灵. 基于双边滤波的图像处理算法研究[D].西安电子科技大学,2010.