(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 112861641 B
(45)授权公告日 2022.05.20

(21)申请号 202110051273.9

(22)申请日 2021.01.15

(65)同一申请的已公布的文献号
申请公布号 CN 112861641 A

(43)申请公布日 2021.05.28

(73)专利权人 复旦大学
地址 200433 上海市杨浦区邯郸路220号

(72)发明人 韩军 张永亮 李强 张辉
王威振 贾立兴 曾晓洋

(74)专利代理机构 上海正旦专利代理有限公司 31200
专利代理师 陆飞 陆尤

(51)Int.Cl.
G06V 40/20(2022.01)
G06V 10/94(2022.01)
G06T 5/30(2006.01)
G06T 7/194(2017.01)

(56)对比文件
CN 108846356 A,2018.11.20
WO 2018076484 A1,2018.05.03
US 2002006222 A1,2002.01.17
Yong-Liang Zhang;Jun Han,ect.An Ultra-low-power High-precision Dynamic Gesture Recognition Coprocessor Based On RISC-V Architecture.《2020 IEEE 15th International Conference on Solid-State & Integrated Circuit Technology (ICSICT)》.2020,
童欣.一种基于OpenCV的手势轮廓识别与指尖定位跟踪方法.《福建电脑》.2018,(第12期),

审查员 尤阳

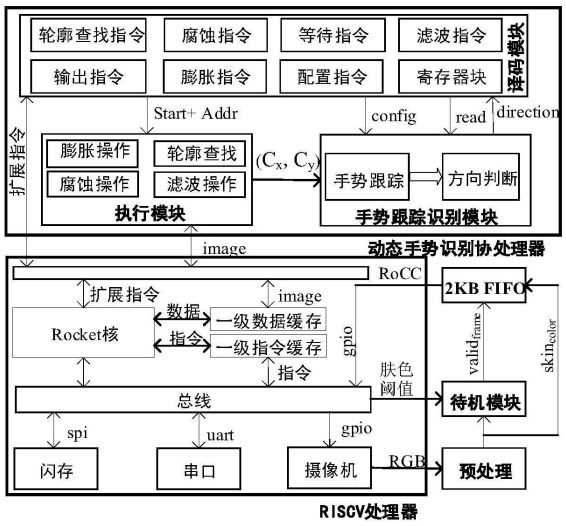
权利要求书3页 说明书7页 附图4页

(54)发明名称

一种面向人机交互领域的动态手势识别硬件加速器

(57)摘要

本发明属于集成电路技术领域,具体为一种面向人机交互领域的动态手势识别硬件加速器。本发明电路包括:图像预处理模块,待机模块,RISCV处理器,动态手势识别协处理器;该加速器采用低复杂度手势识别算法,将RGB图像转化为YCrCb图像,并根据YCrCb图像提取二值肤色图像,有效地降低片上存储容量,降低数据搬运功耗,进而提升整个芯片的能效比;该加速器采用外围轮廓查找手势重心,通过追踪重心的位移得到手势移动方向,具有优越的识别效果,可以广泛应用于无接触式人机交互领域;此外该加速器采用RISCV处理器进行控制,具有较强的灵活性,并为长期待机运行设置了待机模块,有效地降低长期工作的功耗。



CN 112861641 B

1.一种面向人机交互领域的动态手势识别硬件加速器，其特征在于，结构包括图像预处理模块、待机模块、2KB FIFO、RISCV处理器和动态手势识别协处理器；其中：

所述RISCV处理器，包括：闪存、串口、摄像机、RoCC、总线、rocket核、一级数据缓存、一级指令缓存组成；其中闪存中保存编译C产生的汇编指令，系统启动后首先rocket核初始化以后根据指令不断地访问4KB一级指令缓存，读取下一条指令；然而当一级指令缓存miss后，一级指令缓存会经过spi接口读取闪存保存的汇编代码并反馈给rocket核；rocket核获取指令后根据指令首先通过gpio端口配置摄像头，设置摄像头输出图像大小为320x240，帧频为30帧/秒；摄像头配置完成后rocket核向待机模块发送最小肤色阈值肤色总和；其后，rocket核通过RoCC向动态手势识别协处理器发送配置指令，设置参数：最小帧数/横向最小位移/纵向最小位移/横向最大位移/纵向最大位移；配置完成后，rocket核会通过总线读取2KBFIFO的empty信号，当empty＝0时，此时FIFO中含有数据，rocket核会通过gpio和总线读取数据并传输至24KB一级数据缓存；每读取一行数据以后，rocket核通过RoCC向动态手势识别协处理器发送执行指令，其中包括：滤波指令，膨胀指令，腐蚀指令，外围轮廓查找指令，等待指令；当完整的一张图像加载完毕后，RISCV处理器通过RoCC向动态手势识别协处理器发送输出指令，动态手势识别协处理器会根据该指令传输输出方向direction；RISCV处理器接收到direction会将其发送至UART模块进行输出；

所述图像预处理模块，用于接收来自摄像头采集的图像；并将RGB图像转化为YCrCb，再根据肤色区域在CrCb的分布情况，将YCrCb图像转化为二值肤色图像，二值肤色图像中肤色值为1，非肤色值为0；图像预处理模块发送二值肤色图像至待机模块和2KBFIFO；

所述待机模块，用于接收图像预处理模块发送的二值肤色图像，统计图像中所有的肤色值个数肤色总和；当肤色总和超过由RISCV处理器配置的肤色阈值，此时会打开拉升唤醒信号，并在下一帧图像保持唤醒信号为1；当肤色总和低于肤色阈值，唤醒信号置为0，并在下一帧保持为0；待机模块发送唤醒信号至2KBFIFO；

所述2KBFIFO，用于接收图像预处理模块发送的二值肤色图像和待机模块发送的唤醒信号；若唤醒信号为高时，2KBFIFO会保存二值肤色图像并拉低传输至RISCV处理器的empty信号；若唤醒信号为低时，2KBFIFO会忽略二值肤色图像并保持empty信号不变；当2KBFIFO中所有二值肤色图像传输至RISCV处理器，2KBFIFO中的empty信号会拉高；

所述动态手势识别协处理器，包括：译码模块、执行模块和手势跟踪识别模块；其中：

译码模块是由配置指令模块、配置寄存器块、膨胀指令寄存器、腐蚀指令寄存器、滤波指令寄存器、轮廓查找指令寄存器以及输出指令寄存器构成；

配置指令模块接收到RISCV处理器的配置指令，根据其中的序列号将配置参数config，包括最小帧数、横向最小位移、纵向最小位移、横向最大位移、纵向最大位移，分别写入配置寄存器块并发送到手势跟踪识别模块；滤波指令寄存器、膨胀指令寄存器、腐蚀指令寄存器、轮廓查寄存器分别用于接收RISCV处理器的滤波指令、膨胀指令、腐蚀指令、轮廓查找指令，对这些进行译码得到初始地址blur_addr、dila_addr、eros_addr、cont_addr和blur_start、dila_start、eros_start、cont_start，并发送至执行模块；输出指令寄存器用于接收输出指令，对输出指令进行译码并发送读取使能read至手势跟踪识别模块，并在下一周期发送读取的direction至RISCV处理器；

执行模块包括膨胀操作、腐蚀操作、滤波操作、轮廓查找操作；

所述手势跟踪识别模块,包括手势跟踪模块和方向判断模块;其中:

手势跟踪模块用于接收多个轮廓重心值C(x,y)和轮廓面积,然后取出面积最大的重心坐标作为当前坐标减去上一帧坐标,得到帧间位移(Δx,Δy)=C(x,y)-L(x,y),并用C(x,y)替换L(x,y);判断Δx是否处于横向最小位移和横向最大位移以及Δy是否处于纵向最小位移和纵向最大位移之间;

(一)若是,则:该重心坐标有效,选出历史位移(dest_x,dest_y)中绝对值最大的值进行临界判断:

①若dest_x绝对值大,则判断dest_x与Δx是否同号:

同号表示运动方向不变,dest_x=dest_x+Δx与dest_y=dest_y+Δy,累计帧数增加1,并继续下一帧检测;

若不同号,则:

(a)累计帧数大于等于最小帧数,拉高方向判断使能direct_en,并将direct_en与(dest_x,dest_y)发送至方向判断模块,下一周期将(dest_x,dest_y),累计帧数,direct_en置0,重新检测;

(b)累计帧数小于最小帧数,将(dest_x,dest_y),累计帧数,direct_en置0,重新检测;

②若dest_y绝对值大,判断dest_y与Δy是否同号:

同号表示运动方向不变,dest_x=dest_x+Δx与dest_y=dest_y+Δy,累计检测帧数累计帧数增加1,并继续下一帧检测;

若不同号,则:

(a)当累计帧数大于等于最小帧数,拉高方向判断使能direct_en并将direct_en与(dest_x,dest_y)发送至方向判断模块,下一周期将(dest_x,dest_y),累计帧数,direct_en均置为0,重新检测;

(b)当累计帧数小于最小帧数,将(dest_x,dest_y),累计帧数,direct_en均置为0,重新检测;

(二)若否,则:

(a)当累计帧数大于等于最小帧数,拉高方向判断使能direct_en并将direct_en与(dest_x,dest_y)发送至方向判断模块,下一周期将(dest_x,dest_y),累计帧数,direct_en均置为0,重新检测;

(b)当累计帧数小于最小帧数,将(dest_x,dest_y),累计帧数,direct_en均置为0,重新检测;

方向判断模块用接收direct_en与(dest_x,dest_y)信号,当direct_en为高电平时,根据(dest_x,dest_y)绝对值最大的数的方向进行判断:当dest_x绝对值大时:若dest_x为正,则运动方向为向右,direct=1;若为负,则运动方向为左,direct=2;当dest_y绝对值大时:若dest_y为正,则运动方向为向下,direct=3;若为负,则运动方向为上,direct=4;当direct_en为低电平时,direct=0;方向判断模块接收到read使能时,将direct发送至译码模块。

2.根据权利要求1所述的面向人机交互领域的动态手势识别硬件加速器,其特征在于,所述图像预处理模块中,将RGB图像转化为YCrCb,采用如下颜色域转化公式:

Y=(77*R+150*G+29*B)>>8,

Cr＝(-43*R-85*G+128*B)＞＞8+128，

Cb＝(128*R-107*G-21*B)＞＞8+128；

所述肤色区域在CrCb的分布情况为：$skin_{color}$＝(133<Cr<173)&&(77<Cb<127)；从而将YCrCb图像转化为二值肤色图像。

3.根据权利要求1所述的面向人机交互领域的动态手势识别硬件加速器，其特征在于，所述执行模块中：

所述滤波操作用于接收blur_addr和blur_start，并通过RoCC读取一级数据缓存中(blur_addr，blur_addr+5行)的5行图像，并在5行图像左右两边添加2列0值；对最左边5*5的图像块进行求和得到滤波总和，当滤波总和超过12输出为1，否则为0；其后按步长为1不断向右滑动最终得到一行完整的输出，将该输出替换一级数据缓存中以blur_addr为起点的一行图像；

所述膨胀操作用于接收dila_addr和dila_start，并通过RoCC读取一级数据缓存中(dilat_addr，dila_addr+5行)的5行图像；并在5行图像左右两边添加2列0值；对最左边5*5图像块进行按位或，图像块内含有值为1则输出为1，否则为0；其后按步长为1不断向右滑动最终得到一行完整的输出，将该输出替换一级数据缓存中以dilation_addr为起点的一行图像；

所述腐蚀操作用于接收eros_addr和eros_start，并通过RoCC读取一级数据缓存中(eros_addr，eros_addr+5行)的5行图像；并在5行图像左右两边添加2列0值；对最左边5*5图像块进行按位与，图像块内含有值为0则输出为0，否则为1；其后按步长为1不断从左向右滑动最终得到一行完整的输出，将该输出替换一级数据缓存中以erosion_addr为起点的一行图像；

所述外围轮廓查找操作用于接收cont_addr和cont_start，并通过RoCC读取一级数据缓存中以cont_addr为起点的完整一张图像；外围轮廓查找操作先从左到右，从上到下遍历二值图片；当像素点值为1时，设置当前坐标为断点坐标，并以断点坐标为起点进行八邻域轮廓查找；八邻域轮廓查找是以当前坐标为中心，上一个轮廓坐标为起点进行逆时钟旋转，第一个非0值为外围轮廓坐标；保存外围轮廓左右两边的坐标值直至查找的轮廓坐标与断点坐标一致，表示八邻域轮廓查找完成；根据外围轮廓左右两边的坐标值从一级数据缓存中读取区域并统计区域内非0值的横纵坐标总和以及非0值个数，并将所有数据置0替换一级数据缓存中相同位置数据；当所有的轮廓区域统计完成后，由横纵坐标总和除以非0值个数得到该轮廓重心值C(x,y)，同时计算轮廓面积并发送至手势跟踪识别模块；其后，返回断点坐标继续遍历，重复上述操作直至图像遍历完毕。

一种面向人机交互领域的动态手势识别硬件加速器

技术领域

[0001] 本发明属于集成电路技术领域，具体涉及一种面向人机交互领域的动态手势识别硬件加速器。

背景技术

[0002] 自然的、无障碍的、实效性高的、无接触的新型智能人机交互系统成为信息发展中必然趋势。手势作为人机交互领域极其重要的通道之一，具有应用广泛，操作简单，使用频率高等优点。手势识别分为静态手势识别和动态手势识别，其中静态手势识别是通过静态手势图像进行分类，例如“ok“、”比心“、”yes“等手势，动态手势识别是通过视频序列下的手势行为进行分析，例如”上“、”挥手“、”推“等手势。然而，对于动态手势识别，传统的基于模板训练、特征提取、手势分类等多种算法其算法复杂度过高，对于资源要求高，相应的功耗也会过高，无法部署于移动端、IoT、可穿戴设备中。此外，由于外界环境的复杂程度很高，以机器学习等智能算法在动态手势识别领域的实际应用面临着巨大挑战，并且智能算法实现的参数量大，复杂度高，也是难以满足移动端、IoT、可穿戴设备对于低功耗的要求。因而，为移动端、IoT、可穿戴设备等智能终端设计一款可靠性强、识别精度高、低功耗的动态手势识别的人机交互智能芯片已经成为迫切需要。

发明内容

[0003] 为了克服现有技术的不足，本发明提出一种面向人机交互领域的动态手势识别硬件加速器。以期能够实现算法复杂度低，资源消耗少，功耗低，可靠性强的动态手势识别硬件加速器，该加速器可以广泛应用于移动端、IoT、可穿戴设备等智能终端领域，具有超长待机功能和高精度的实时识别能力。

[0004] 本发明提供的面向人机交互领域的动态手势识别硬件加速器，其结构包括图像预处理模块、待机模块、2KB FIFO(先入先出队列)、RISCV处理器和动态手势识别协处理器；其中：

[0005] 所述RISCV处理器，包括：闪存、串口、摄像机、RoCC、总线、rocket核、一级数据缓存、一级指令缓存组成；其中闪存中保存编译C产生的汇编指令，系统启动后首先rocket核初始化以后根据指令不断地访问4KB一级指令缓存，读取下一条指令；然而当一级指令缓存miss后，一级指令缓存会经过spi接口读取闪存保存的汇编代码并反馈给rocket核；rocket核获取指令后根据指令首先通过gpio端口配置摄像头，设置摄像头输出图像大小为320x240，帧频为30帧/秒；摄像头配置完成后rocket核向待机模块发送最小肤色阈值肤色总和；其后，rocket核通过RoCC向动态手势识别协处理器发送配置指令，设置参数：最小帧数/横向最小位移/纵向最小位移/横向最大位移/纵向最大位移；配置完成后，rocket核会通过总线读取2KBFIFO的empty信号，当empty＝0时，此时FIFO中含有数据，rocket核会通过gpio和总线读取数据并传输至24KB一级数据缓存；每读取一行数据以后，rocket核通过RoCC向动态手势识别协处理器发送执行指令，其中包括：滤波指令，膨胀指令，腐蚀指令，外

围轮廓查找指令,等待指令;当完整的一张图像加载完毕后,RISCV处理器通过RoCC向动态手势识别协处理器发送输出指令,动态手势识别协处理器会根据该指令传输输出方向direction;RISCV处理器接收到direction会将其发送至UART模块进行输出;

[0006] 所述图像预处理模块,用于接收来自摄像头采集的图像。根据颜色域转化公式:Y=(77*R+150*G+29*B)>>8,Cr=(-43*R-85*G+128*B)>>8+128,Cb=(128*R-107*G-21*B)>>8+128,将RGB图像转化为YCrCb,其后根据肤色区域在CrCb的分布情况:$skin_{color}$=(133<Cr<173)&&(77<Cb<127),从而将YCrCb图像转化为二值肤色图像,二值肤色图像中肤色值为1,非肤色值为0;图像预处理模块发送二值肤色图像至待机模块和2KBFIFO;

[0007] 所述待机模块,用于接收图像预处理模块发送的二值肤色图像,统计图像中所有的肤色值个数肤色总和;当肤色总和超过由RISCV处理器配置的肤色阈值,此时会打开拉升唤醒信号,并在下一帧图像保持唤醒信号为1;当肤色总和低于肤色阈值,唤醒信号置为0,并在下一帧保持为0;待机模块发送唤醒信号至2KBFIFO;

[0008] 所述2KBFIFO,用于接收图像预处理模块发送的二值肤色图像和待机模块发送的唤醒信号;若唤醒信号为高时,2KBFIFO会保存二值肤色图像并拉低传输至RISCV处理器的empty信号;若唤醒信号为低时,2KBFIFO会忽略二值肤色图像并保持empty信号不变;当2KBFIFO中所有二值肤色图像传输至RISCV处理器,2KBFIFO中的empty信号会拉高;

[0009] 所述动态手势识别协处理器,包括:译码模块、执行模块和手势跟踪识别模块;其中:

[0010] 译码模块是由配置指令寄存器、配置寄存器块、膨胀指令寄存器、腐蚀指令寄存器、滤波指令寄存器、轮廓查找指令寄存器以及输出指令寄存器构成;

[0011] 配置指令模块接收到RISCV处理器的配置指令,根据其中的序列号将配置参数config(包括最小帧数、横向最小位移、纵向最小位移、横向最大位移、纵向最大位移)分别写入配置寄存器块并发送到手势跟踪识别模块;滤波指令寄存器、膨胀指令寄存器、腐蚀指令寄存器、轮廓查寄存器分别用于接收RISCV处理器的滤波指令、膨胀指令、腐蚀指令、轮廓查找指令,对这些进行译码得到初始地址blur_addr、dila_addr、eros_addr、cont_addr和blur_start、dila_start、eros_start、cont_start,并发送至执行模块;输出指令寄存器用于接收输出指令,对输出指令进行译码并发送读取使能read至手势跟踪识别模块,并在下一周期发送读取的direction至RISCV处理器;

[0012] 执行模块包括膨胀操作、腐蚀操作、滤波操作、轮廓查找操作;

[0013] 滤波操作接收blur_addr和blur_start后,通过RoCC读取一级数据缓存中(blur_addr,blur_addr+5行)的5行图像,并在5行图像左右两边添加2列0值;对最左边5*5的图像块进行求和得到滤波总和,当滤波总和超过12输出为1,否则为0;其后按步长为1不断向右滑动最终得到一行完整的输出,将该输出替换一级数据缓存中以blur_addr为起点的一行图像;

[0014] 膨胀操作接收dila_addr和dila_start后,通过RoCC读取一级数据缓存中(dilat_addr,dila_addr+5行)的5行图像;并在5行图像左右两边添加2列0值;对最左边5*5图像块进行按位或,图像块内含有值为1则输出为1,否则为0;其后按步长为1不断向右滑动最终得到一行完整的输出,将该输出替换一级数据缓存中以dilation_addr为起点的一行图像;

[0015] 腐蚀操作接收eros_addr和eros_start后,通过RoCC读取一级数据缓存中(eros_

addr,eros_addr+5行)的5行图像;并在5行图像左右两边添加2列0值;对最左边5*5图像块进行按位与,图像块内含有值为0则输出为0,否则为1;其后按步长为1不断从左向右滑动最终得到一行完整的输出,将该输出替换一级数据缓存中以erosion_addr为起点的一行图像;

[0016] 外围轮廓查找操作接收cont_addr和cont_start后,通过RoCC读取一级数据缓存中以cont_addr为起点的完整一张图像;外围轮廓查找操作先从左到右,从上到下遍历二值图片;当像素点值为1时,设置当前坐标为断点坐标,并以断点坐标为起点进行八邻域轮廓查找;八邻域轮廓查找是以当前坐标为中心,上一个轮廓坐标为起点进行逆时钟旋转,第一个非0值为外围轮廓坐标;保存外围轮廓左右两边的坐标值直至查找的轮廓坐标与断点坐标一致,表示八邻域轮廓查找完成;根据外围轮廓左右两边的坐标值从一级数据缓存中读取区域并统计区域内非0值的横纵坐标总和以及非0值个数,并将所有数据置0替换一级数据缓存中相同位置数据;当所有的轮廓区域统计完成后,由横纵坐标总和除以非0值个数得到该轮廓重心值C(x,y),同时计算轮廓面积并发送至手势跟踪识别模块;其后,返回断点坐标继续遍历,重复上述操作直至图像遍历完毕;

[0017] 所述手势跟踪识别模块,包括手势跟踪模块和方向判断模块;其中:

[0018] 手势跟踪模块用于接收多个轮廓重心值C(x,y)和轮廓面积,然后取出面积最大的重心坐标作为当前坐标减去上一帧坐标,得到帧间位移(Δx,Δy)=C(x,y)-L(x,y),并用C(x,y)替换L(x,y);判断Δx是否处于横向最小位移和横向最大位移以及Δy是否处于纵向最小位移和纵向最大位移之间;

[0019] (一)若是,则:该重心坐标有效,选出历史位移(dest_x,dest_y)中绝对值最大的值进行临界判断:

[0020] ①若dest_x绝对值大,则判断dest_x与Δx是否同号:

[0021] 同号表示运动方向不变,dest_x=dest_x+Δx与dest_y=dest_y+Δy,累计帧数增加1,并继续下一帧检测;

[0022] 若不同号,则:

[0023] (a)累计帧数大于等于最小帧数,拉高direct_en,并将direct_en与(dest_x,dest_y)发送至方向判断模块,下一周期将(dest_x,dest_y),累计帧数,direct_en置0,重新检测;

[0024] (b)累计帧数小于最小帧数,将(dest_x,dest_y),累计帧数,direct_en置0,重新检测;

[0025] ②若dest_y绝对值大,判断dest_y与Δy是否同号:

[0026] 同号表示运动方向不变,dest_x=dest_x+$delt_x$与dest-y=dest_y+Δy,累计检测帧数累计帧数增加1,并继续下一帧检测;

[0027] 若不同号,则:

[0028] (a)当累计帧数大于等于最小帧数,拉高direct_en并将direct_en与(dest_x,dest_y)发送至方向判断模块,下一周期将(dest_x,dest_y),累计帧数,direct_en均置为0,重新检测;

[0029] (b)当累计帧数小于最小帧数,将(dest_x,dest_y),累计帧数,direct_en均置为0,重新检测;

[0030] (二)若否,则:

[0031] (a)当累计帧数大于等于最小帧数,拉高方向判断使能direct_en并将direct_en与(dest_x,dest_y)发送至方向判断模块,下一周期将(dest_x,dest_y),累计帧数,direct_en均置为0,重新检测;

[0032] (b)当累计帧数小于最小帧数,将(dest_x,dest_y),累计帧数,direct_en均置为0,重新检测;

[0033] 方向判断模块用接收direct_en与(dest_x,dest_y)信号,当direct_en为高电平时,根据(dest_x,dest_y)绝对值最大的数的方向进行判断:当dest_x绝对值大时:若dest_x为正,则运动方向为向右,direct=1;若为负,则运动方向为左,direct=2;当dest_y绝对值大时:若dest_y为正,则运动方向为向下,direct=3;若为负,则运动方向为上,direct=4;当direct_en为低电平时,direct=0;方向判断模块接收到read使能时,将direct发送至译码模块。

[0034] 与现有技术相比,本发明的有益技术效果体现在:

[0035] 1、本发明使用RISCV处理器进行指令控制,使用RISCV处理器对外设进行配置和使用,加快了设计速度;采用RISCV的扩展指令集进行动态手势识别协处理器开发,减少协处理器的控制过程,有利于协处理器开发;此外使用高级语言对外设以及协处理器进行控制,灵活性高,通用性强,易于实现。

[0036] 2、预处理模块将输入的RGB图像转化为YCrCb后再通过肤色提取的方式分离出肤色与背景,将整个图像转化为二值黑白图像表示;该方式能够减少在计算过程中的数据搬运量,加快计算过程,降低功耗,提升整个芯片的性能。

[0037] 3、待机模块采用简单的计数模块,结合模块级门控时钟能够关闭后续的所有模块的运行,能够有效的降低功耗,使芯片在待机状态时以极低功耗运行,提升芯片的使用周期;

[0038] 4、协处理器对二值图像进行滤波操作和形态学闭操作,外围轮廓查找等方法将二维图像转化为连续运动点的分析,再通过对历史运动的轨迹进行分析,最后判断出移动方向;识别算法简单,硬件资源消耗小,功耗低,可靠性强,识别精度高。

附图说明

[0039] 图1为面向人机交互领域的动态手势识别加速器系统图。

[0040] 图2为预处理模块的图像变化。

[0041] 图3为待机模块的唤醒机制。

[0042] 图4为动态手势识别协处理器框图。

[0043] 图5为扩展指令集格式。

[0044] 图6为滤波操作。

[0045] 图7为膨胀操作。

[0046] 图8为腐蚀操作。

[0047] 图9为轮廓查找操作。

[0048] 图10为8邻域查找方法。

[0049] 图11为手势跟踪识别电路。

具体实施方式

[0050] 本发明设计的动态手势识别硬件加速器，系统结构如图1所示，其结构包括图像预处理模块、待机模块、2KB FIFO,RISCV处理器和动态手势识别协处理器;其中：

[0051] RISCV处理器，包括:闪存/串口/摄像机/RoCC/总线/rocket核/一级数据缓存/一级指令缓存组成;其中闪存中保存编译C产生的汇编指令，系统启动后首先rocket核初始化以后根据指令不断地访问4KB一级指令缓存，读取下一条指令;然而当一级指令缓存miss后，一级指令缓存会经过spi接口读取闪存保存的汇编代码并反馈给rocket核;汇编代码是对高级语言C代码进行编译，能够对所有的外设，协处理器，内部寄存器，一级缓存等多个模块进行控制和修改;rocket核获取指令后根据指令首先通过gpio端口配置摄像头，设置摄像头输出图像大小为320x240，帧频为30帧/秒，这里摄像头使用75MHz的始终发送RGB565图像，按照帧同步场同步的方式发送的数据，一个周期只能发送8bit的数据，2个周期才能得到一个像素点;摄像头配置完成后rocket核向待机模块发送最小肤色阈值肤色总和;其后，rocket核通过RoCC向动态手势识别协处理器发送配置指令，设置参数:最小帧数/横向最小位移/纵向最小位移/横向最大位移/纵向最大位移;配置完成后，rocket核会通过总线读取2KBFIFO的empty信号，当empty＝0时，此时FIFO中含有数据，rocket核会通过gpio和总线读取数据并传输至24KB一级数据缓存;每读取一行数据以后，Rocket核通过RoCC向动态手势识别协处理器发送执行指令，其中包括:滤波指令，膨胀指令，腐蚀指令，外围轮廓查找指令，等待指令;Rocket提供了扩展指令集，用户可以根据设计的协处理器需要添加需要的扩展指令;当完整的一张图像加载完毕后，RISCV处理器通过RoCC向动态手势识别协处理器发送输出指令，动态手势识别协处理器会根据该指令传输输出方向direction;此时direction＝0表示没有方向(没有动作或者动作没有达到要求)，direction＝1表示向右移动，direction＝2表示向左移动，direction＝3表示向上移动，direction＝4表示向下移动;RISCV处理器接收到direction会将其发送至UART模块进行输出;

[0052] 图像预处理模块，用于接收来自摄像头采集的图像，根据颜色域转化公式:Y＝(77*R+150*G+29*B)＞＞8,Cr＝(-43*R-85*G+128*B)＞＞8+128,Cb＝(128*R-107*G-21*B)＞＞8+128,将RGB图像转化为YCrCb，其后根据肤色区域在CrCb的分布情况:$skin_{color}$＝(133＜Cr＜173)&&(77＜Cb＜127)，从而将YCrCb图像转化为二值肤色图像，二值肤色图像中肤色值为1，非肤色值为0;单个像素从16bit转化为1bit的二值图像，降低了计算的数据量，并且可以分离手势和背景，如图2所示;图像预处理模块发送二值肤色图像至待机模块和2KBFIFO。

[0053] 待机模块，接收到图像预处理模块发送的二值肤色图像，统计图像中所有的肤色值个数肤色总和;当肤色总和超过由RISCV处理器配置的肤色阈值，此时会打开拉升唤醒信号，并在下一帧图像保持唤醒信号为1;当肤色总和低于肤色阈值，唤醒信号置为0，并在下一帧保持为0;待机模块发送唤醒信号至2KBFIFO。如图3所示，在待机模式下仅会运行预处理模块和待机模块，唤醒机制可以保证芯片能够在低功耗基础上长期待机工作，延长工作时间。

[0054] 2KBFIFO接收到图像预处理模块发送的二值肤色图像和待机模块发送的唤醒信号;若唤醒信号为高时，2KBFIFO会保存二值肤色图像并拉低传输至RISCV处理器的empty信号;若唤醒信号为低时，2KBFIFO会忽略二值肤色图像并保持empty信号不变;当2KBFIFO中

所有二值肤色图像传输至RISCV处理器，2KBFIFO中的empty信号会拉高；为了保证数据在每一行处理过程中数据不会因为FIFO的满状态导致的图像丢失，这里使用2KB的存储；

[0055] 动态手势识别协处理器，包括：译码模块，执行模块和手势跟踪识别模块，如图3所示。动态手势识别协处理器与RISCV处理器通过AXI4总线进行握手交互。RISCV处理器通过扩展指令读写协处理器的寄存器来控制协处理器的运行；协处理器多个模块访问一级数据缓存，因而需要使用数据读写仲裁进行判断。

[0056] 译码模块是由配置指令寄存器、配置寄存器块、膨胀指令寄存器、腐蚀指令寄存器、滤波指令寄存器、轮廓查找指令寄存器以及输出指令寄存器构成如图4所示。译码模块根据扩展指令中的标志位funct7进行区分，如图5所示：funct7＝1表示配置指令，funct7＝2表示膨胀指令，funct7＝3表示腐蚀指令，funct7＝4表滤波指令，funct7＝5表示轮廓查找指令，funct7＝6表示输出指令。

[0057] 配置指令模块接收到RISCV处理器的配置指令，会根据其中的存储于rs1代表的寄存器中的序列号将存储于rs2代表的寄存器中的配置参数config(最小帧数/横向最小位移/纵向最小位移/横向最大位移/纵向最大位移)分别写入配置寄存器块并发送到手势跟踪识别模块；滤波指令寄存器、膨胀指令寄存器、腐蚀指令寄存器、轮廓查寄存器接收到RISCV处理器的滤波指令、膨胀指令、腐蚀指令、轮廓查找指令，对这些进行译码得到初始地址blur_addr、dila_addr、eros_addr、cont_addr和blur_start、dila_start、eros_start、cont_start并发送至执行模块；输出指令寄存器接收到输出指令，对输出指令进行译码并发送读取使能read至手势跟踪识别模块；

[0058] 执行模块包括膨胀操作、腐蚀操作、滤波操作、轮廓查找操作；

[0059] 滤波操作接收blur_addr和blur_start后，通过RoCC读取一级数据缓存中(blur_addr，blur_addr+5行)的5行图像，并在5行图像左右两边添加2列0值；对最左边5*5的图像块进行求和得到滤波总和，当滤波总和超过12输出为1，否则为0；其后按步长为1不断向右滑动最终得到一行完整的输出，将该输出替换一级数据缓存中以blur_addr为起点的一行图像；其中的滑动过程我们采用向左移动1位进行代替，如图6所示。

[0060] 膨胀操作接收dila_addr和dila_start后，通过RoCC读取一级数据缓存中(dilat_addr，dila_addr+5行)的5行图像；并在5行图像左右两边添加2列0值；对最左边5*5图像块进行按位或，图像块内含有值为1则输出为1，否则为0；其后按步长为1不断向右滑动最终得到一行完整的输出，将该输出替换一级数据缓存中以dilation_addr为起点的一行图像；其中的滑动过程我们采用向左移动1位进行代替，如图7所示。

[0061] 腐蚀操作接收eros_addr和eros_start后，通过RoCC读取一级数据缓存中(eros_addr，eros_addr+5行)的5行图像；并在5行图像左右两边添加2列0值；对最左边5*5图像块进行按位与，图像块内含有值为0则输出为0，否则为1；其后按步长为1不断从左向右滑动最终得到一行完整的输出，将该输出替换一级数据缓存中以erosion_addr为起点的一行图像；其中的滑动过程我们采用向左移动1位进行代替，如图8所示。

[0062] 如图9所示，外围轮廓查找操作接收cont_addr和cont_start后，通过RoCC读取一级数据缓存中以cont_addr为起点的完整一张图像；外围轮廓查找操作先从左到右，从上到下遍历二值图片；当像素点值为1时，设置当前坐标为断点坐标，并以断点坐标为起点进行八邻域轮廓查找；八邻域轮廓查找是以当前坐标为中心，上一个轮廓坐标为起点进行逆时

钟旋转，第一个非0值为外围轮廓坐标，如图10所示；保存外围轮廓左右两边的坐标值直至查找的轮廓坐标与断点坐标一致，表示八邻域轮廓查找完成；根据外围轮廓左右两边的坐标值从一级数据缓存中读取区域并统计区域内非0值的横纵坐标总和以及非0值个数，并将所有数据置0替换一级数据缓存中相同位置数据；当所有的轮廓区域统计完成后，由横纵坐标总和除以非0值个数得到该轮廓重心值C(x,y)，同时计算轮廓面积并发送至手势跟踪识别模块；其后，返回断点坐标继续遍历，重复上述操作直至图像遍历完毕；

[0063] 手势跟踪识别模块包括手势跟踪模块和方向判断模块；

[0064] 手势跟踪模块接收到多个轮廓重心值C(x,y)和轮廓面积后，选通面积最大的重心坐标作为当前坐标减去上一帧坐标，得到帧间位移(Δx,Δy)＝C(x,y)-L(x,y)，并用C(x,y)替换L(x,y)，如图11所示；判断Δx是否处于横向最小位移和横向最大位移以及Δy是否处于纵向最小位移和纵向最大位移之间，若是：则该重心坐标有效，选出历史位移(dest_x,dest_y)中绝对值最大的值进行临界判断：若dest_x绝对值大，判断dest_x与Δx是否同号，同号表示运动方向不变，dest_x＝dest_x+Δx与dest_y＝dest_y+Δy，累计检测帧数累计帧数增加1，并继续下一帧检测；若否：当累计帧数大于等于最小帧数，拉高direct_en，并将direct_en与(dest_x,dest_y)发送至方向判断模块，下一周期将(dest_x,dest_y)，累计帧数，direct_en置0，重新检测；当累计帧数小于最小帧数，将(dest_x,dest_y)，累计帧数，direct_en置0，重新检测；若dest_y绝对值大，判断dest_y与Δy是否同号，同号表示运动方向不变，dest_x＝dest_x+$delt_x$与dest_y＝dest_y+Δy，累计检测帧数累计帧数增加1，并继续下一帧检测；若否：当累计帧数大于等于最小帧数，拉高direct_en并将direct_en与(dest_x,dest_y)发送至方向判断模块，下一周期将(dest_x,dest_y)，累计帧数，direct_en均置为0，重新检测；当累计帧数小于最小帧数，将(dest_x,dest_y)，累计帧数，direct_en均置为0，重新检测；若否：当累计帧数大于等于最小帧数，拉高方向判断使能direct_en并将direct_en与(dest_x,dest_y)发送至方向判断模块，下一周期将(dest_x,dest_y)，累计帧数，direct_en均置为0，重新检测；当累计帧数小于最小帧数，将(dest_x,dest_y)，累计帧数，direct_en均置为0，重新检测；初始阶段的时候，会对(dest_x,dest_y)，累计帧数以及(Δx,Δy)执行复位操作。

[0065] 方向判断模块接收到direct_en与(dest_x,dest_y)信号后，当direct_en为高电平时，根据(dest_x,dest_y)绝对值最大的数的方向进行判断：当dest_x绝对值大时：若dest_x为正，则运动方向为向右，direct＝1；若为负，则运动方向为左，direct＝2；当dest_y绝对值大时：若dest_y为正，则运动方向为向下，direct＝3；若为负，则运动方向为上，direct＝4；当direct_en为低电平时，direct＝0；方向判断模块接收到read使能时，将direct发送至译码模块。

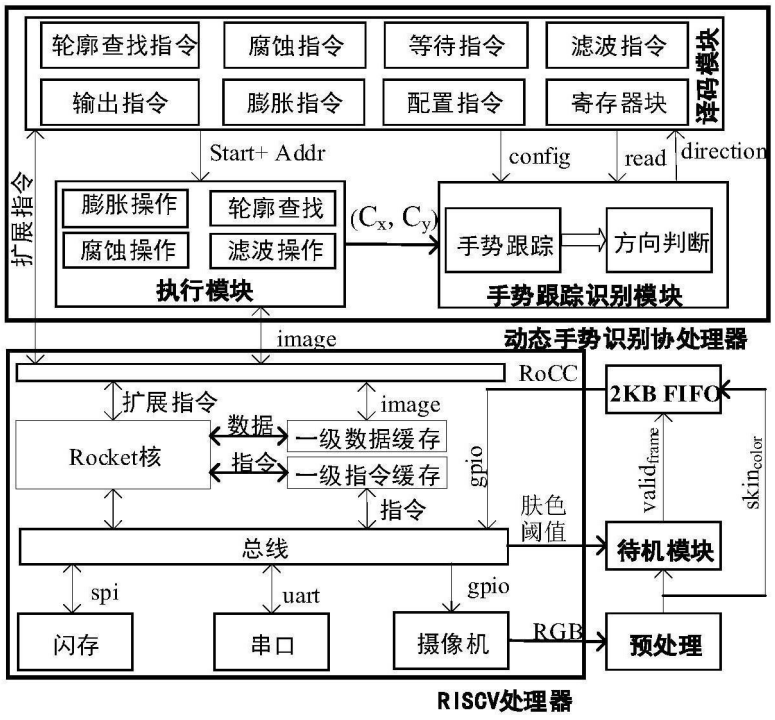

轮廓查找指令
腐蚀指令
等待指令
滤波指令
输出指令
膨胀指令
配置指令
寄存器块
译码模块
Start+ Addr
config
read
direction
扩展指令
膨胀操作
轮廓查找
腐蚀操作
滤波操作
执行模块
(C_x, C_y)
手势跟踪
方向判断
手势跟踪识别模块
image
动态手势识别协处理器
RoCC
2KB FIFO
扩展指令
image
数据
一级数据缓存
Rocket核
指令
一级指令缓存
gpio
肤色阈值
指令
$valid_{frame}$
$skin_{color}$
总线
待机模块
spi
uart
gpio
闪存
串口
摄像机
RGB
预处理
RISCV处理器

图1

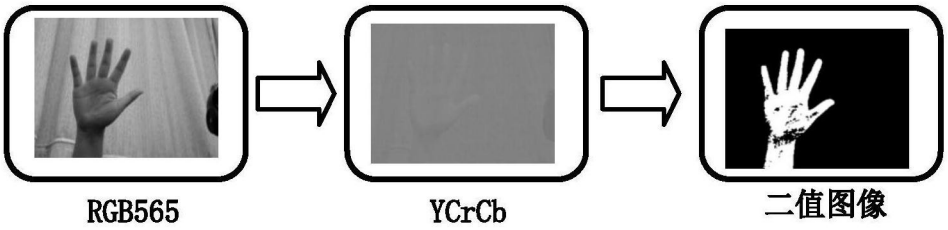

RGB565
YCrCb
二值图像

图2

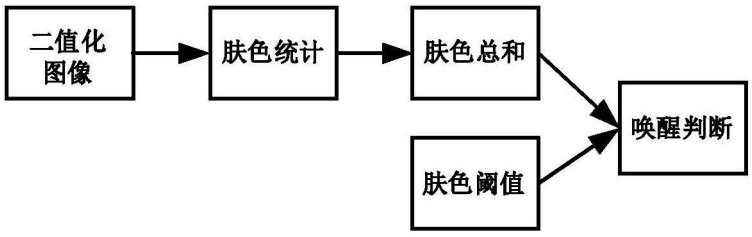

二值化图像
肤色统计
肤色总和
肤色阈值
唤醒判断

图3

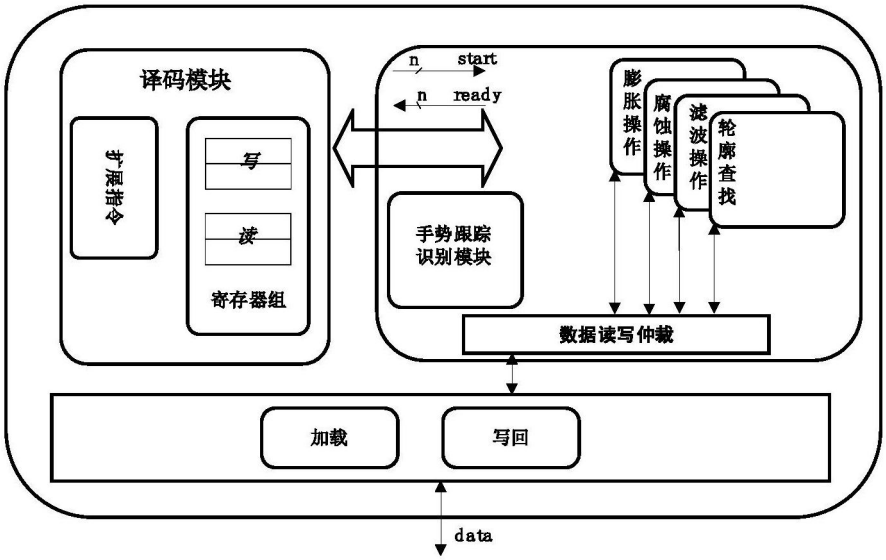

译码模块
扩展指令
写
读
寄存器组
n start
n ready
手势跟踪
识别模块
膨胀操作
腐蚀操作
滤波操作
轮廓查找
数据读写仲裁
加载
写回
data

图4

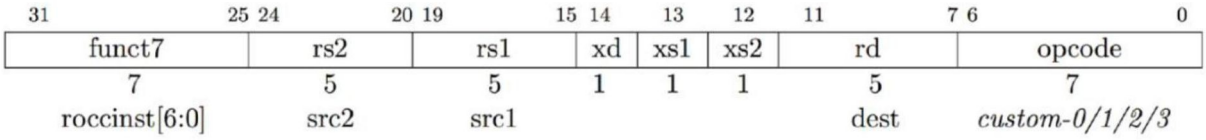

31 25 24 20 19 15 14 13 12 11 7 6 0
funct7 rs2 rs1 xd xs1 xs2 rd opcode
7 5 5 1 1 1 5 7
roccinst[6:0] src2 src1 dest custom-0/1/2/3

图5

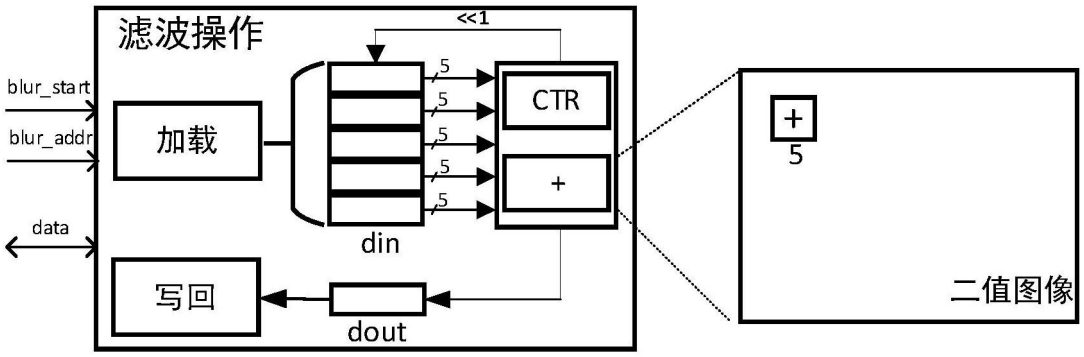

滤波操作
<<1
blur_start
blur_addr
data
加载
写回
din
dout
5
CTR
+
+
5
二值图像

图6

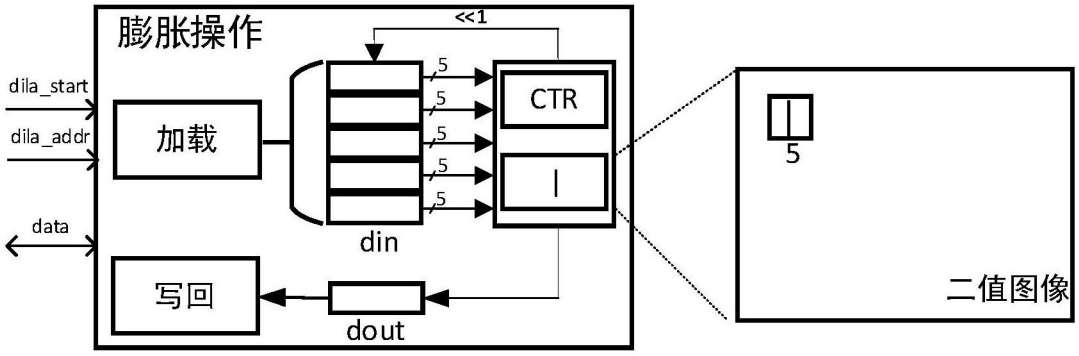

膨胀操作
<<1
dila_start
dila_addr
data
加载
写回
din
dout
CTR
|
5
5
5
5
5
|
5
二值图像

图7

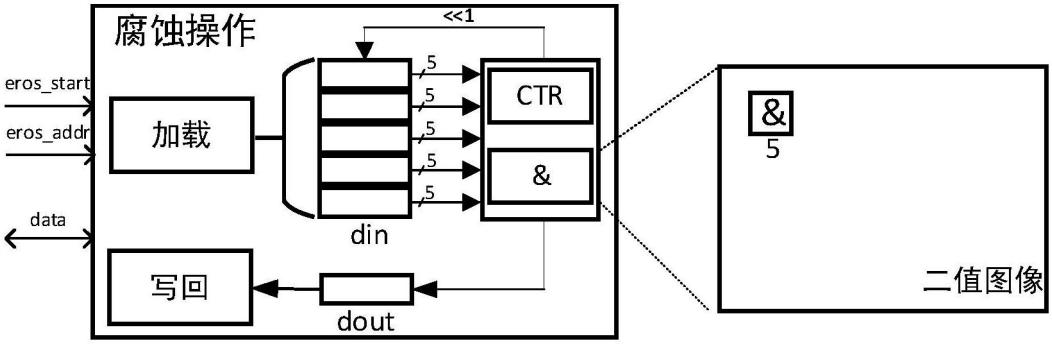

腐蚀操作
<<1
eros_start
eros_addr
data
加载
写回
din
dout
CTR
&
5
5
5
5
5
&
5
二值图像

图8

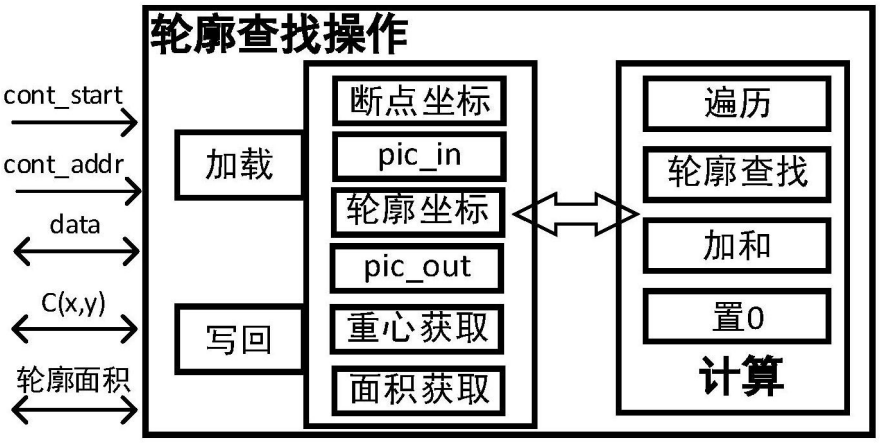

轮廓查找操作
cont_start
cont_addr
data
C(x,y)
轮廓面积
加载
写回
断点坐标
pic_in
轮廓坐标
pic_out
重心获取
面积获取
遍历
轮廓查找
加和
置0
计算

图9

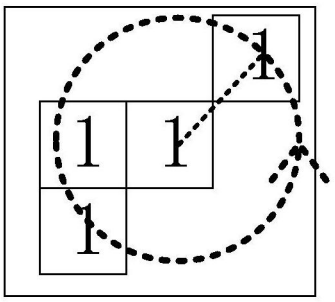

1
1
1
1

图10

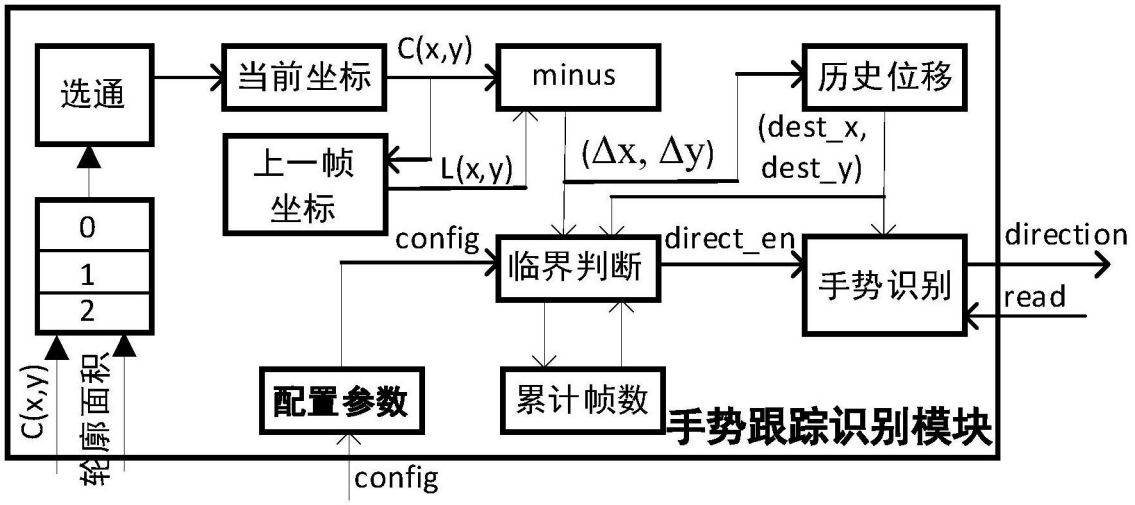

选通
当前坐标
C(x,y)
minus
历史位移
上一帧
坐标
L(x,y)
(Δx, Δy)
(dest_x,
dest_y)
0
1
2
config
临界判断
direct_en
手势识别
direction
read
C(x,y)
轮廓面积
配置参数
累计帧数
手势跟踪识别模块
config

图11