



(12) 发明专利

(10) 授权公告号 CN 114612523 B

(45) 授权公告日 2024. 11. 08

(21) 申请号 202210229821.7

G06T 5/70 (2024.01)

(22) 申请日 2022.03.10

G06T 5/30 (2006.01)

(65) 同一申请的已公布的文献号

G06T 1/60 (2006.01)

申请公布号 CN 114612523 A

G06T 1/20 (2006.01)

(43) 申请公布日 2022.06.10

G06V 40/20 (2022.01)

(73) 专利权人 东南大学

G06V 10/94 (2022.01)

地址 211102 江苏省南京市江宁区东南大学路2号

G06F 3/01 (2006.01)

(72) 发明人 陈龙 陈静

(56) 对比文件

CN 107516296 A, 2017.12.26

(74) 专利代理机构 南京瑞弘专利商标事务所

CN 113989683 A, 2022.01.28

(普通合伙) 32249

审查员 狄希

专利代理师 沈廉

(51) Int. Cl.

G06T 7/254 (2017.01)

G06T 7/246 (2017.01)

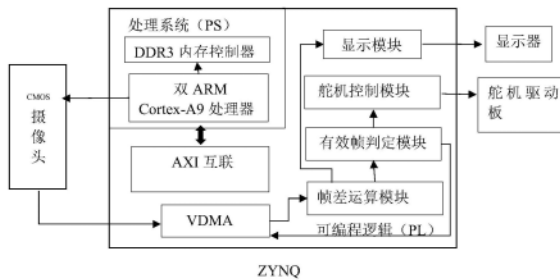
权利要求书3页 说明书8页 附图1页

(54) 发明名称

基于FPGA的动态目标跟踪系统及其检测方法

(57) 摘要

本发明是一种基于FPGA的动态目标跟踪系统及其检测方法,由CMOS摄像头采集实时视频数据,VDMA进行图像缓存,接着在可编程逻辑PL端进行数据处理:帧差处理模块通过相邻两帧图像完成目标检测,同时进一步处理图像数据,在图像中画出运动物体的最小外接矩形;接着建立数学模型,对帧差处理模块得到的数据结果进行有效帧判定;若判定结果为真,由舵机控制模块计算出二自由度舵机转动至目标位置所需PWM波占空比,并通过串口通信传输至舵机驱动板,同时更新有效帧判定模块中模型数据;若判定结果是否为,对接下来的视频数据重新进行计算;显示模块输出处理后的图像数据;实现提高图像处理速度,实时跟踪动态目标,并减少画面内其他物体的干扰影响。



1. 一种基于FPGA的动态目标跟踪系统的动态目标检测方法,其特征在于:该检测方法包括以下步骤:

步骤S1:由摄像头模块采集实时视频数据,数据流转换模块将RGB888数据转为axi4数据流格式,数据在PYNQ-Z2开发板的PL端传输,在VDMA数据缓存模块中缓存连续两帧图像,接着进行帧差处理;

步骤S2:将帧差处理后的图像数据先膨胀后腐蚀,实现填充物体内部细小空洞,连接邻近物体与平滑边界功能;

步骤S3:寻找经步骤S2处理后图像中的边界并绘制运动物体的外接矩形边框,计算运动物体中心;所得运动区域参数 $(X_{12}, X_{22}, X_{32}, X_{42})$ 与目标参数 $(X_{11}, X_{21}, X_{31}, X_{41})$;

步骤S4:建立相似性度量数学模型,根据外接矩形边框形状、面积以及中心与原追踪目标进行相似度判定,若判定结果为真,继续进行步骤S5,若为否,返回步骤S1;

步骤S41:利用步骤S3所得运动区域参数 $(X_{12}, X_{22}, X_{32}, X_{42})$ 与目标参数 $(X_{11}, X_{21}, X_{31}, X_{41})$ 的夹角余弦来作为相似性度量,记原目标变量为 $(X_{11}, X_{21}, X_{31}, X_{41})$,步骤S3所得运动区域变量为 $(X_{12}, X_{22}, X_{32}, X_{42})$,进行余弦相似度计算:

$$r_{12} = \frac{\sum_{i=1}^4 X_{i1}X_{i2}}{(\sum_{i=1}^4 X_{i1}^2 \sum_{i=1}^4 X_{i2}^2)^{\frac{1}{2}}} \quad (2)$$

其中 (X_{11}, X_{21}) 表示原目标中心坐标位置, X_{31} 表示原目标外接矩形的长, X_{41} 为原目标外接矩形的宽, (X_{12}, X_{22}) 表示步骤S3所得运动区域中心坐标位置 (x, y) , X_{32} 表示步骤S3所得运动区域外接矩形的长, X_{42} 为步骤S3所得运动区域外接矩形的宽; X_{i1} 表示原目标第i个变量, X_{i2} 表示运动区域的第i个变量;

若 r_{12} 大于阈值T3,则认为步骤S3所得运动区域与原目标相似,进入步骤S42,反之,判定S3所得结果为无效图像,返回步骤S1;

步骤S42:由步骤S3所得运动区域外接矩形框长宽计算面积,与原目标外接矩形框面积做面积判定:

$$r_2 = \left(\frac{X_{31}X_{41} - X_{32}X_{42}}{X_{32}X_{42}} \right)^2 \quad (3)$$

若 r_2 小于阈值T4,则可认为面积判定为真,进入步骤S5,反之,判定S3所得结果为无效图像,返回步骤S1;

步骤S5:更新原追踪物体外接矩形边框长宽、中心数据,并由S4所得物体中心位置计算舵机占空比,由串口通信传入舵机驱动板,调整舵机角度。

2. 根据权利要求1所述的动态目标检测方法,其特征在于:所述步骤S1包含以下具体步骤:

步骤S11:使能FPGA开发板的两个EMIO接口,令其模拟SCCB时序来控制SCCB总线的读写,进而完成摄像头寄存器的配置;

步骤S12:将8位摄像头输入数据拼接成16位RGB565数据,使用VIVADO自带Video in to axi4-stream IP核将RGB888数据转为axi4数据流格式,方便数据在PYNQ-Z2开发板的PL端传输;

步骤S13:在PYNQ-Z2开发板板载DDR3中缓存连续两帧图像;

步骤S14:从VDMA中读取第N帧与第N+1帧图像,将这两帧图像转为灰度图像,将两帧图

像相减,并进行二值化处理:

$$P_1(x,y) = \begin{cases} 255, & |Y_1(x,y) - Y_2(x,y)| > T \\ 0, & |Y_1(x,y) - Y_2(x,y)| \leq T \end{cases} \quad (1)$$

其中, $P_1(x,y)$ 表示第N+1帧与第N帧灰度图像之差; (x,y) 表示像素点的横纵坐标位置; $Y(x,y)$ 表示灰度图像YCbCr格式中的亮度分量; T 表示设定的阈值。

3. 根据权利要求1所述的动态目标检测方法,其特征在于:所述步骤S2先膨胀后腐蚀,完成闭运算;实现填充物体细小空洞,连接邻近物体,平滑物体边界且不明显改变物体面积,同时抑制比结构单元小的细节的功能,

所述膨胀,定义集合A为二值化后值为255的像素点坐标位置,结构元B为3*3的矩形,不断平移结构元B的原点,若B与A的交集不为空,记下B的原点位置,记为集合C;

所述腐蚀,定义结构元D为

0	255	0
255	255	255
0	255	0

不断平移结构元D的原点,记下D仍然包含于

C的集合,记为集合E,E便为闭运算结果。

4. 根据权利要求1所述的动态目标检测方法,其特征在于:所述步骤S3包含以下具体步骤:

步骤S31:对步骤S2的结果进行面积检测,去除面积小于阈值 T_2 的连通区域,得到完整的运动区域;

步骤S32:遍历步骤S31所得完整运动区域,得到图像像素点总个数;

步骤S33:对图像进行水平投影,记录在图像在X轴每行的像素点个数,并找出运动区域的左边界与右边界,右边界与左边界x值相减得运动区域外接矩形的长;

步骤S34:将步骤S33记录的X轴像素点个数进行加权累加,即每行像素点个数乘以对应X轴坐标,并相加;

步骤S35:计算运动区域的水平中心点,将步骤S34的加权累加值除以步骤S32遍历的图像像素点总个数,结果为运动区域的水平中心点坐标x;

步骤S36:对图像进行垂直投影,记录在图像在Y轴每行的像素点个数,并找出运动区域的上边界与下边界,上边界与下边界y值相减得运动区域外接矩形的宽;

步骤S37:将步骤S36记录的Y轴像素点个数进行加权累加,即每行像素点个数乘以对应Y轴坐标,并相加;

步骤S38:计算运动区域的垂直中心点,将步骤S37的加权累加值除以步骤S32遍历的图像像素点总个数,结果为运动区域的垂直中心点坐标y。

5. 根据权利要求1所述的动态目标检测方法,其特征在于:所述步骤S5包含以下具体步骤:

步骤S51:由步骤S3所得结果更新原追踪物体外接矩形边框长宽、中心位置数据;

步骤S52:舵机控制由数值转换和串口通信控制两部分组成,数值转换完成了由运动目标中心位置确定实现摄像头跟踪所需转动角度到驱动板控制输出PWM波占空比的转换,由于系统使用摄像头视角较小,使用一次曲线拟合图像中像素位置与舵机转动角度的关系,即占空比与中心位置横纵坐标关系,

$$\begin{cases} \text{angle}_x = A_1 X_{11} + C_1 \\ \text{angle}_y = A_2 X_{21} + C_2 \end{cases} \quad (4)$$

其中 A_1, A_2, C_1, C_2 是由上位机软件与舵机驱动板测量多组数据拟合得到的占空比与像素位置对应曲线参数;

步骤S53:使用串口通信,将由运动区域中心坐标位置 (X_{12}, X_{22}) 带入公式(4)计算好的占空比换算值以及自己设定的转动指定角度所需时间到驱动板,实现对动态目标的跟踪。

6.一种用于权利要求1所述的动态目标检测方法的基于FPGA的动态目标跟踪系统,其特征在于:该系统包括数据信号采集模块、数据信号预处理模块、目标识别模块、舵机控制模块和数据信号显示模块;

所述数据信号采集模块包括CMOS摄像头、数据流转换模块、VDMA数据缓存模块和采集模块控制单元;CMOS摄像头实现原始视频数据采集功能,将采集后的视频数据输入数据流转换模块;数据流转换模块实现将RGB888数据转为axi4数据流格式功能,方便数据在PYNQ-Z2开发板的PL端传输;VDMA数据缓存模块实现图像缓存功能,能存下两帧图像,以便后续图像处理;采集模块控制单元在处理系统PS端实现,完成摄像头初始化及DDR3内存控制功能;

所述数据信号预处理模块实现提取VDMA中图像数据,并将其转为灰度图像数据;

所述目标识别模块,包括帧差运算模块和有效帧判定模块;帧差运算模块对灰度图像进行帧差运算及二值化处理,并通过闭运算填充空洞、平滑边界;有效帧判定模块通过建立数学模型,对帧差运算模块所得结果进行判定;

所述舵机控制模块包括占空比计算模块及串口通信模块;占空比计算模块根据通过有效帧判定的数据计算舵机所需占空比,实现对物体的追踪;串口通信模块实现与舵机驱动板的通信,完成对二自由度舵机的控制;所述二自由度舵机可在X,Y方向转动,所转动的角度由输入PWM波占空比控制;

所述数据信号显示模块包括外接矩形框的处理模块及图像转换模块,实现将运动区域用外接矩形框标出,并转换RGB888图像数据为DVI视频格式,便于直观了解动态目标位置以及后续调试;

首先由数据信号采集模块录入实时视频数据,然后通过数据信号预处理模块转换数据格式,接着目标识别模块找出有效的运动区域,最后由舵机控制模块计算舵机转动所需PWM波占空比并将其传输到舵机驱动板,同时,数据信号显示模块会将目标识别模块处理后的数据输出,便于系统的调试。

基于FPGA的动态目标跟踪系统及其检测方法

技术领域

[0001] 本发明属于动态目标跟踪领域,具体涉及一种基于FPGA的动态目标检测方法及跟踪系统。

背景技术

[0002] 随着计算机技术的飞速发展,计算机处理视觉信息被人们广泛关注。计算机视觉是融合了图像处理、计算机图形学、模式识别、人工智能、人工神经网络、计算机、心理学、物理学和数学等领域的一门交叉性很强的学科。计算机视觉研究的目的是使计算机感知环境中的物体的几何信息,包括它的形状、位置、姿态、运动等,并对其进描述、存储、识别与理解。视频目标跟踪方法的研究和应用作为计算机视觉领域的一个重要分支,正日益广泛地应用到科学技术、国防建设、航空航天、医药卫生以及国民经济的各个领域,因而研究目标跟踪技术有着重大的实用价值和广阔发展前景。

[0003] 目前存在的动态目标跟踪系统存在以下问题:1) 由于缺乏并行运算能力,处理速度不理想;2) 实时性较低,容易丢失目标;3) 可拓展性与可移植性不理想;4) 识别误差较大,当画面中存在多物体时,对其他物体的干扰处理有待优化;5) 设备较为笨重,不便于移动。

[0004] 现场可编程门阵列(Field-Programmable Gate Array,FPGA)作为一种高性能、低功耗的可编程芯片,可以适用于小批量流式应用,很适合处理深度学习的推断阶段,能大大缩短开发周期。随着运动目标识别准确率和实时性的提高,深度学习算法的计算复杂度和内存需求也急剧增加,当前的通用处理器无法满足其计算需求,因此,将FPGA用于运动目标的识别可以更好的达到实时的检测速度和低功耗运行状态,从而适应运动目标识别的现实应用场景。

发明内容

[0005] 技术问题:本发明的目的是为了优化上述现有技术存在的问题而提供的一种基于FPGA的动态目标检测方法及跟踪系统,以实现提高图像处理速度,实时跟踪动态目标,并减少画面内其他物体的干扰的目的。

[0006] 技术方案:本发明的一种基于FPGA的动态目标跟踪系统包括数据信号采集模块、数据信号预处理模块、目标识别模块、舵机控制模块和数据信号显示模块;

[0007] 所述数据信号采集模块包括CMOS摄像头、数据流转换模块、VDMA数据缓存模块和采集模块控制单元;CMOS摄像头实现原始视频数据采集功能,将采集后的视频数据输入数据流转换模块;数据流转换模块实现将RGB888数据转为axi4数据流格式功能,方便数据在PYNQ-Z2开发板的PL端传输;VDMA数据缓存模块实现图像缓存功能,能存下两帧图像,以便后续图像处理;采集模块控制单元在处理系统PS端实现,完成摄像头初始化及DDR3内存控制功能;

[0008] 所述数据信号预处理模块实现提取VDMA中图像数据,并将其转为灰度图像数据;

[0009] 所述目标识别模块,包括帧差运算模块和有效帧判定模块;帧差运算模块对灰度

图像进行帧差运算及二值化处理,并通过闭运算填充空洞、平滑边界;有效帧判定模块通过建立数学模型,对帧差运算模块所得结果进行判定;

[0010] 所述舵机控制模块包括占空比计算模块及串口通信模块;占空比计算模块根据通过有效帧判定的数据计算舵机所需占空比,实现对物体的追踪;串口通信模块实现与舵机驱动板的通信,完成对二自由度舵机的控制;所述二自由度舵机可在X,Y方向转动,所转动的角度由输入PWM波占空比控制;

[0011] 所述数据信号显示模块包括外接矩形框的处理模块及图像转换模块,实现将运动区域用外接矩形框标出,并转换RGB888图像数据为DVI视频格式,便于直观了解动态目标位置以及后续调试;

[0012] 首先由数据信号采集模块录入实时视频数据,然后通过数据信号预处理模块转换数据格式,接着目标识别模块找出有效的运动区域,最后由舵机控制模块计算舵机转动所需PWM波占空比并将其传输到舵机驱动板,同时,数据信号显示模块会将目标识别模块处理后的数据输出,便于系统的调试。

[0013] 其中,

[0014] 该检测方法包括以下步骤:

[0015] 步骤S1:由摄像头模块采集实时视频数据,数据流转换模块将RGB888数据转为axi4数据流格式,数据在PYNQ-Z2开发板的PL端传输,在VDMA数据缓存模块中缓存连续两帧图像,接着进行帧差处理;

[0016] 步骤S2:将帧差处理后的图像数据先膨胀后腐蚀,实现填充物体内部细小空洞,连接邻近物体与平滑边界功能;

[0017] 步骤S3:寻找经步骤S2处理后图像中的边界并绘制运动物体的外接矩形边框,计算运动物体中心;所得运动区域参数 $(X_{12}, X_{22}, X_{32}, X_{42})$ 与目标参数 $(X_{11}, X_{21}, X_{31}, X_{41})$;

[0018] 步骤S4:建立相似性度量数学模型,根据外接矩形边框形状、面积以及中心与原追踪目标进行相似度判定,若判定结果为真,继续进行步骤S5,若为否,返回步骤S1;

[0019] 步骤S5:更新原追踪物体外接矩形边框长宽、中心数据,并由S4所得物体中心位置计算舵机占空比,由串口通信传入舵机驱动板,调整舵机角度。

[0020] 所述步骤S1包含以下具体步骤:

[0021] 步骤S11:使能FPGA开发板的两个EMIO接口,令其模拟SCCB时序来控制SCCB总线的读写,进而完成摄像头寄存器的配置;

[0022] 步骤S12:将8位摄像头输入数据拼接成16位RGB565数据,使用VIVADO自带Video in to axi4-stream IP核将RGB888数据转为axi4数据流格式,方便数据在PYNQ-Z2开发板的PL端传输;

[0023] 步骤S13:在PYNQ-Z2开发板板载DDR3中缓存连续两帧图像;

[0024] 步骤S14:从VDMA中读取第N帧与第N+1帧图像,将这两帧图像转为灰度图像,将两帧图像相减,并进行二值化处理;

$$[0025] \quad P_1(x, y) = \begin{cases} 255, & |Y_1(x, y) - Y_2(x, y)| > T \\ 0, & |Y_1(x, y) - Y_2(x, y)| \leq T \end{cases} \quad (1)$$

[0026] 其中, $P_1(x, y)$ 表示第N+1帧与第N帧灰度图像之差; (x, y) 表示像素点的横纵坐标位置; $Y(x, y)$ 表示灰度图像YCbCr格式中的亮度分量; T 表示设定的阈值。

[0027] 所述步骤S2先膨胀后腐蚀,完成闭运算;实现填充物体内部细小空洞,连接邻近物体,平滑物体边界且不明显改变物体面积,同时抑制比结构单元小的细节的功能,

[0028] 所述膨胀,定义集合A为二值化后值为255的像素点坐标位置,结构元B为3*3的矩形,不断平移结构元B的原点,若B与A的交集不为空,记下B的原点位置,记为集合C;

[0029] 所述腐蚀,定义结构元D为
$$\begin{matrix} 0 & 255 & 0 \\ 255 & 255 & 255 \\ 0 & 255 & 0 \end{matrix}$$
,不断平移结构元D的原点,记下D仍然

包含于C的集合,记为集合E,E便为闭运算结果。

[0030] 所述步骤S3包含以下具体步骤:

[0031] 步骤S31:对步骤S2的结果进行面积检测,去除面积小于阈值 T_2 的连通区域,得到完整的运动区域;

[0032] 步骤S32:遍历步骤S31所得完整运动区域,得到图像像素点总个数;

[0033] 步骤S33:对图像进行水平投影,记录在图像在X轴每行的像素点个数,并找出运动区域的左边界与右边界,右边界与左边界x值相减得运动区域外接矩形的长;

[0034] 步骤S34:将步骤S33记录的X轴像素点个数进行加权累加,即每行像素点个数乘以对应X轴坐标,并相加;

[0035] 步骤S35:计算运动区域的水平中心点,将步骤S34的加权累加值除以步骤S32遍历的图像像素点总个数,结果为运动区域的水平中心点坐标x;

[0036] 步骤S36:对图像进行垂直投影,记录在图像在Y轴每行的像素点个数,并找出运动区域的上边界与下边界,上边界与下边界y值相减得运动区域外接矩形的宽;

[0037] 步骤S37:将步骤S36记录的Y轴像素点个数进行加权累加,即每行像素点个数乘以对应Y轴坐标,并相加;

[0038] 步骤S38:计算运动区域的垂直中心点,将步骤S37的加权累加值除以步骤S32遍历的图像像素点总个数,结果为运动区域的垂直中心点坐标y。

[0039] 所述步骤S4包含以下具体步骤:

[0040] 步骤S41:利用步骤S3所得运动区域参数 $(X_{12}, X_{22}, X_{32}, X_{42})$ 与目标参数 $(X_{11}, X_{21}, X_{31}, X_{41})$ 的夹角余弦来作为相似性度量,记原目标变量为 $(X_{11}, X_{21}, X_{31}, X_{41})$,步骤S3所得运动区域变量为 $(X_{12}, X_{22}, X_{32}, X_{42})$,进行余弦相似度计算:

$$[0041] \quad r_{12} = \frac{\sum_{i=1}^4 X_{i1} X_{i2}}{(\sum_{i=1}^4 X_{i1}^2 \sum_{i=1}^4 X_{i2}^2)^{\frac{1}{2}}} \quad (2)$$

[0042] 其中 (X_{11}, X_{21}) 表示原目标中心坐标位置, X_{31} 表示原目标外接矩形的长, X_{41} 为原目标外接矩形的宽, (X_{12}, X_{22}) 表示步骤S3所得运动区域中心坐标位置 (x, y) , X_{32} 表示步骤S3所得运动区域外接矩形的长, X_{42} 为步骤S3所得运动区域外接矩形的宽; X_{i1} 表示原目标第i个变量, X_{i2} 表示运动区域的第i个变量;

[0043] 若 r_{12} 大于阈值 T_3 ,则可以认为步骤S3所得运动区域与原目标相似,进入步骤S42,反之,判定S3所得结果为无效图像,返回步骤S1;

[0044] 步骤S42:由步骤S3所得运动区域外接矩形框长宽计算面积,与原目标外接矩形框面积做面积判定:

$$[0045] \quad r_2 = \left(\frac{X_{31}X_{41} - X_{32}X_{42}}{X_{32}X_{42}} \right)^2 \quad (3)$$

[0046] 若 r_2 小于阈值 T_4 ,则可认为面积判定为真,进入步骤S5,反之,判定S3所得结果为无效图像,返回步骤S1。

[0047] 所述步骤S5包含以下具体步骤:

[0048] 步骤S51:由步骤S3所得结果更新原追踪物体外接矩形边框长宽、中心位置数据;

[0049] 步骤S52:舵机控制由数值转换和串口通信控制两部分组成,数值转换完成了由运动目标中心位置确定实现摄像头跟踪所需转动角度到驱动板控制输出PWM波占空比的转换,由于系统使用摄像头视角较小,使用一次曲线拟合图像中像素位置与舵机转动角度的关系,即占空比与中心位置横纵坐标关系,

$$[0050] \quad \begin{cases} \text{angle}_x = A_1 X_{11} + C_1 \\ \text{angle}_y = A_2 X_{21} + C_2 \end{cases} \quad (4)$$

[0051] 其中 A_1, A_2, C_1, C_2 是由上位机软件与舵机驱动板测量多组数据拟合得到的占空比与像素位置对应曲线参数;

[0052] 步骤S53:使用串口通信,将由运动区域中心坐标位置 (X_{12}, X_{22}) 带入公式(4)计算好的占空比换算值以及自己设定的转动指定角度所需时间到驱动板,实现对动态目标的跟踪。

[0053] 有益效果:与现有技术相比,本发明具有以下优点:

[0054] (1)本发明是在Xilinx公司的PYNQ-Z2可拓展平台上基于ZYNQ架构搭建,在PL端完成帧差处理、有效帧判定、图像处理输出、舵机控制及串口通信,PS端完成CMOS摄像头寄存器初始化、VDMA初始化。既充分利用了PYNQ-Z2强大的并行运算能力及逻辑资源,又结合ARM核在搭建操作系统与实现复杂算法方面的优点,提高了运算速度,减小了系统体积;

[0055] (2)本发明通过建立数学模型,结合相似度判定与面积判定,排除无效数据,减小了追踪单个目标,背景内存在多个物体时的干扰,提高了系统的稳定性。

[0056] (3)本发明实现小型化运动物体跟踪系统,功耗低,体积小,便携。

[0057] (4)本发明的硬件价格相较于PC具有优势。

附图说明

[0058] 图1为本发明的硬件加速数据流图。

[0059] 图2为本发明的运动物体检测与判定流程图。

具体实施方式

[0060] 为了更好地理解发明,下面结合实例和附图对本发明做进一步的说明。

[0061] 本发明为一种基于FPGA的动态目标检测方法及跟踪系统,该系统包含PYNQ-Z2开发板、电源线、OV5640CMOS摄像头、液晶显示器、舵机驱动板、二自由度舵机和HDMI线;

[0062] 所述PYNQ-Z2是一款支持PYNQ开源框架的开发平台,除支持传统ZYNQ开发方式外,还可支持Python进行APSoC编程。PYNQ-Z2开发板以ZYNQ XC7Z020 FPGA为核心,配备有以太网,HDMI输入/输出,MIC输入,音频输出,Arduino接口,树莓派接口,2个Pmod,板载DDR3,用户LED,按钮和开关。

[0063] 所述ASPSoC芯片包括可编程逻辑PL端与处理系统PS端,PS端部分由一个双核ARM Cortex-A9处理器和两个Neon协处理器等组成,其内部还集结大量外设及内存控制器,PL端为可编程逻辑器件,实现帧差处理、有效帧判定、舵机控制等功能。

[0064] 所述电源线用于给PYNQ-Z2开发板供电;

[0065] 所述系统硬件加速数据流图如图1所示;

[0066] 所述0V5640CMOS摄像头通过自己焊接的转接板与PYNQ-Z2开发板的PMOD接口连接,由两个EMIO接口模拟SCCB时序来控制SCCB总线的读写,进而完成摄像头寄存器的配置;

[0067] 所述舵机驱动板的Tx与Rx接口与PYNQ-Z2开发板的GPIO端口相连接,实现PYNQ-Z2开发板与舵机驱动板间的串口通信;

[0068] 所述HDMI线连接PYNQ-Z2开发板的HDMI OUT接口与液晶显示器;

[0069] 所述DDR3内存控制器与ASPSoC芯片和SD Card接口连接;

[0070] 所述ASPSoC芯片与HDMI OUT接口连接;

[0071] 所述处理系统PS端与DDR3内存控制器连接;

[0072] 所述AXI数据传输总线用于连接PL与PS端,AXI总线实现ARM与FPGA之间的数据通信,有效提高PL与PS端间传输速度;

[0073] 所述可编程逻辑PL端包括VDMA数据储存模块、帧差处理模块、有效帧判定模块、舵机控制模块、显示输出模块;

[0074] 所述VDMA数据储存模块和AXI数据传输总线连接,用于存取视频流数据;

[0075] 所述帧差处理模块用于对VDMA图像数据进行处理;

[0076] 所述有效帧判定模块用于根据帧差处理模块的结果进行有效帧判定,排除无效数据;

[0077] 所述舵机控制模块通过GPIO接口与舵机驱动板连接,用于计算舵机占空比并与舵机驱动板进行串口通信;

[0078] 所述显示输出模块用于输出处理后的图像,便于系统的调试;

[0079] 一种基于FPGA的动态目标检测方法,包括如下步骤:

[0080] S1、CMOS摄像头采集实时视频数据,然后进行帧差处理。首先使能FPGA开发板的两个EMIO接口,令其模拟SCCB时序来控制SCCB总线的读写,进而完成摄像头寄存器的配置。在本实施例中,设置0V5640输出像素为640*480,图像帧数为40帧/秒;

[0081] 接着将8位摄像头输入数据拼接成16位RGB565数据,使用VIVADO自带Video in to axi4-stream IP核将RGB888数据转为axi4数据流格式,方便数据在PYNQ-Z2开发板的PL端传输,在PYNQ-Z2开发板板载DDR3中缓存连续两帧图像。最后,从VDMA中读取第N帧与第N+1帧图像,将这两帧图像转为灰度图像。

$$[0082] \begin{cases} Y = 0.229R + 0.587G + 0.114B \\ Cb = 0.568(B - Y) + 128 \\ Cr = 0.713(R - Y) + 128 \end{cases}$$

$$[0083] \begin{cases} Y = (77 * R + 150 * G + 29 * B) >> 8 \\ Cb = (-43 * R - 85 * G + 128 * B) >> 8 + 128 \\ Cr = (128 * R - 107 * G - 21 * B) >> 8 + 128 \end{cases}$$

[0084] 将两帧图像相减,并进行二值化处理:

$$[0085] \quad P_1(x, y) = \begin{cases} 255, & |Y_1(x, y) - Y_2(x, y)| > T \\ 0, & |Y_1(x, y) - Y_2(x, y)| \leq T \end{cases}$$

[0086] 其中, $P_1(x, y)$ 表示第 $N+1$ 帧与第 N 帧灰度图像之差; (x, y) 表示像素点的横纵坐标位置; $Y(x, y)$ 表示灰度图像 YCbCr 格式中的亮度分量; T 表示设定的阈值 ($T=100$)。

[0087] S2、先膨胀后腐蚀,完成闭运算。填充物体内部细小空洞,连接邻近物体,平滑物体边界且不明显改变物体面积,同时抑制比结构单元小的细节,得到图像 Image1。

[0088] S3、寻找经步骤 S2 处理后图像中的边界并绘制运动物体的外接矩形边框,计算运动物体中心。首先对步骤 S2 的结果 Image1 进行面积检测,去除面积小于阈值 T_2 的连通区域,得到完整的运动区域 Image。然后遍历所得完整运动区域 Image,得到图像像素点总个数。接着对图像进行水平投影,记录在图像在 X 轴每行的像素点个数,并找出运动区域的左边界与右边界。再将 X 轴像素点个数进行加权累加,即每行像素点个数乘以对应 X 轴坐标,并相加。最后,计算运动区域的水平中心点,将水平投影的加权累加值除以遍历的图像像素点总个数,结果为运动区域的水平中心点坐标 x 。对图像进行垂直投影,记录在图像在 Y 轴每行的像素点个数,并找出运动区域的上边界与下边界。然后将记录的 Y 轴像素点个数进行加权累加,即每行像素点个数乘以对应 Y 轴坐标,并相加。最后,计算运动区域的垂直中心点,将垂直投影的加权累加值除以遍历的图像像素点总个数,结果为运动区域的垂直中心点坐标 y ;

[0089] S4、建立数学模型,根据外接矩形边框形状、面积以及中心与原追踪目标进行相似性判定。首先,利用步骤 S3 所得运动区域参数与目标参数的夹角余弦来作为相似性度量,记原目标变量为 $(X_{11}, X_{21}, X_{31}, X_{41})$, 步骤 S3 所得运动区域变量为 $(X_{12}, X_{22}, X_{32}, X_{42})$, 进行余弦相似度计算:

$$[0090] \quad r_{12} = \frac{\sum_{i=1}^4 X_{i1} X_{i2}}{(\sum_{i=1}^4 X_{i1}^2 \sum_{i=1}^4 X_{i2}^2)^{\frac{1}{2}}}$$

[0091] 其中 (X_{11}, X_{21}) 表示原目标中心坐标位置, X_{31} 表示原目标外接矩形的长, X_{41} 为原目标外接矩形的宽, (X_{12}, X_{22}) 表示步骤 S3 所得运动区域中心坐标位置, X_{32} 表示步骤 S3 所得运动区域外接矩形的长, X_{42} 为步骤 S3 所得运动区域外接矩形的宽;

[0092] 若 r_{12} 大于阈值 T_3 , 则可以认为步骤 S3 所得运动区域与原目标相似,反之,判定 S3 所得结果为无效图像,返回步骤 S1;

[0093] 接着,由步骤 S3 所得运动区域外接矩形框长宽计算面积,与原目标外接矩形框面积做面积判定:

$$[0094] \quad r_2 = \left(\frac{X_{31} X_{41} - X_{32} X_{42}}{X_{32} X_{42}} \right)^2$$

[0095] 若 r_2 小于阈值 T_4 , 则可认为面积判定为真,进入步骤 S5,反之,判定 S3 所得结果为无效图像,返回步骤 S1;

[0096] S5、更新原追踪物体外接矩形边框长宽、中心数据,并由 S4 所得物体中心位置计算舵机占空比,由串口通信传入舵机驱动板,调整舵机角度。

[0097] 首先,由步骤 S3 所得结果更新原追踪物体外接矩形边框长宽、中心位置数据。

[0098] 接着进行舵机控制,主要由数值转换和串口通信控制两部分组成。数值转换完成了由运动目标中心位置确定实现摄像头跟踪所需转动角度到驱动板控制输出 PWM 波占空比

的转换。由于系统使用摄像头视角较小,便可以使用一次曲线拟合图像中像素位置与舵机转动角度的关系,即占空比与中心位置横纵坐标关系:

$$[0099] \quad \begin{cases} \mathit{angle}x = A_1 X_{11} + C_1 \\ \mathit{angle}y = A_2 X_{21} + C_2 \end{cases}$$

[0100] 其中 A_1, A_2, C_1, C_2 由上位机软件与舵机驱动板测量多组数据拟合得到,在本实施例中,拟合结果为:

$$[0101] \quad \begin{cases} \mathit{angle}x = \frac{17}{16} X_{11} + 1250 \\ \mathit{angle}y = -\frac{33}{32} X_{21} + 1670 \end{cases}$$

[0102] 最后,使用串口通信,将计算好的角度换算值以及转动指定角度所需时间传到驱动板,实现对动态目标的跟踪。

[0103] 以下是利用本系统的具体实施例,下面实施例中系统对图像处理过程一致,通过改变二自由度舵机顶端所装设器件,实现不同功能,但本发明方案不限于以下应用,此处不一一举例赘述。

[0104] 实施例1:

[0105] 所述二自由度舵机由舵机驱动板输出PWM波控制,顶端装有USB摄像头,与电脑端连接,由电脑观察追踪结果;

[0106] 所述USB摄像头为LRCP1080P__0V2710,感光芯片1/2.7inch;Pixel Size:3um◇3um;

[0107] (1)由摄像头模块采集实时视频数据,数据流转换模块将RGB888数据转为axi4数据流格式,方便数据在PYNQ-Z2开发板的PL端传输,在VDMA数据缓存模块中缓存连续两帧图像,接着进行帧差处理。

[0108] (2)将帧差处理后的图像数据先膨胀后腐蚀,填充物体内黑洞,连接邻近物体与平滑边界;

[0109] (3)寻找经步骤(2)处理后图像中的边界并绘制运动物体的外接矩形边框,计算运动物体中心;

[0110] (4)建立数学模型,根据外接矩形边框形状、面积以及中心与原追踪目标进行相似判定,若判定结果为真,继续进行步骤(5),若为否,返回步骤(1);

[0111] (5)更新原追踪物体外接矩形边框长宽、中心数据,并由(4)所得物体中心位置计算舵机占空比,由串口通信传入舵机驱动板,调整舵机角度。

[0112] 物体始终位于舵机顶端的USB摄像头录入的视频中心,在与USB摄像头连接的电脑端可观察到对运动目标的实时监控。

[0113] 实施例2:

[0114] 所述二自由度舵机由舵机驱动板输出PWM波控制,顶端装有激光器,实现激光光点始终位于运动目标之上;

[0115] (1)由摄像头模块采集实时视频数据,数据流转换模块将RGB888数据转为axi4数据流格式,方便数据在PYNQ-Z2开发板的PL端传输,在VDMA数据缓存模块中缓存连续两帧图像,接着进行帧差处理。

[0116] (2)将帧差处理后的图像数据先膨胀后腐蚀,填充物体内黑洞,连接邻近物体与平

滑边界；

[0117] (3) 寻找经步骤(2)处理后图像中的边界并绘制运动物体的外接矩形边框,计算运动物体中心；

[0118] (4) 建立数学模型,根据外接矩形边框形状、面积以及中心与原追踪目标进行相似度判定,若判定结果为真,继续进行步骤(5),若为否,返回步骤(1)；

[0119] (5) 更新原追踪物体外接矩形边框长宽、中心数据,并由(4)所得物体中心位置计算舵机占空比,由串口通信传入舵机驱动板,调整舵机角度。

[0120] 无论运动目标以任何轨迹运动,舵机顶端的激光器始终照射于目标之上,实现对运动目标的实时跟踪。

[0121] 实施例3:

[0122] 所述二自由度舵机由舵机驱动板输出PWM波控制,顶端装有一支笔,可实现隔空手写。

[0123] (1) 由摄像头模块采集实时视频数据,追踪手部信息,数据流转换模块将RGB888数据转为axi4数据流格式,方便数据在PYNQ-Z2开发板的PL端传输,在VDMA数据缓存模块中缓存连续两帧图像,接着进行帧差处理。

[0124] (2) 将帧差处理后的图像数据先膨胀后腐蚀,填充物体内黑洞,连接邻近物体与平滑边界；

[0125] (3) 寻找经步骤(2)处理后图像中的边界并绘制运动物体的外接矩形边框,计算运动物体中心；

[0126] (4) 建立数学模型,根据外接矩形边框形状、面积以及中心与原追踪目标进行相似度判定,若判定结果为真,继续进行步骤(5),若为否,返回步骤(1)；

[0127] (5) 更新原追踪物体外接矩形边框长宽、中心数据,并由(4)所得物体中心位置计算舵机占空比,由串口通信传入舵机驱动板,调整舵机角度。

[0128] COMS摄像头追踪手部运动信息,由二自由度舵机转动带动其顶端的笔,实现隔空手写。

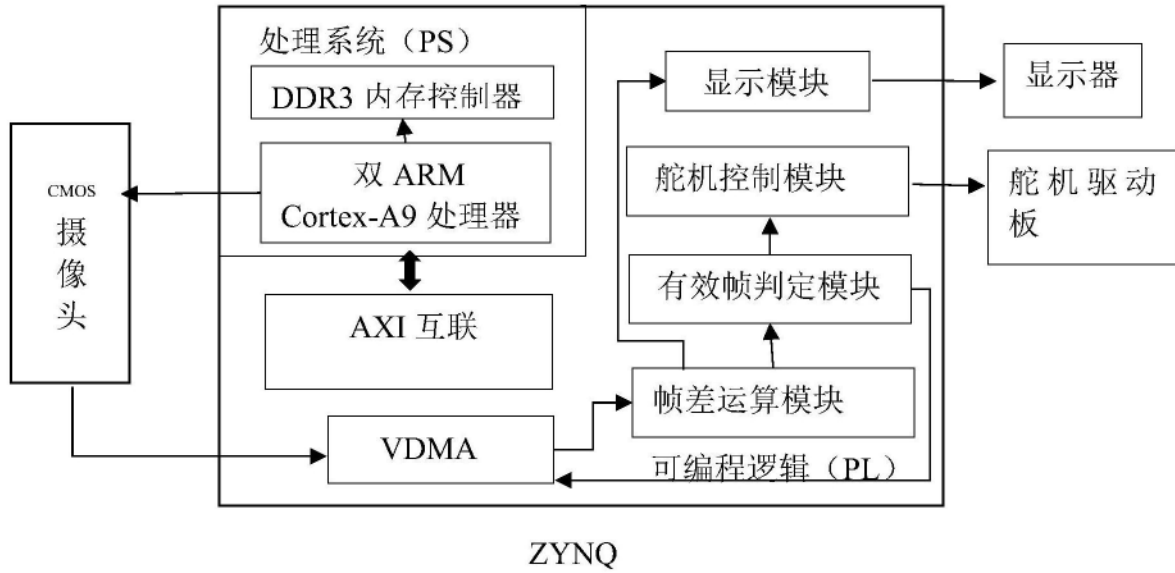


图1

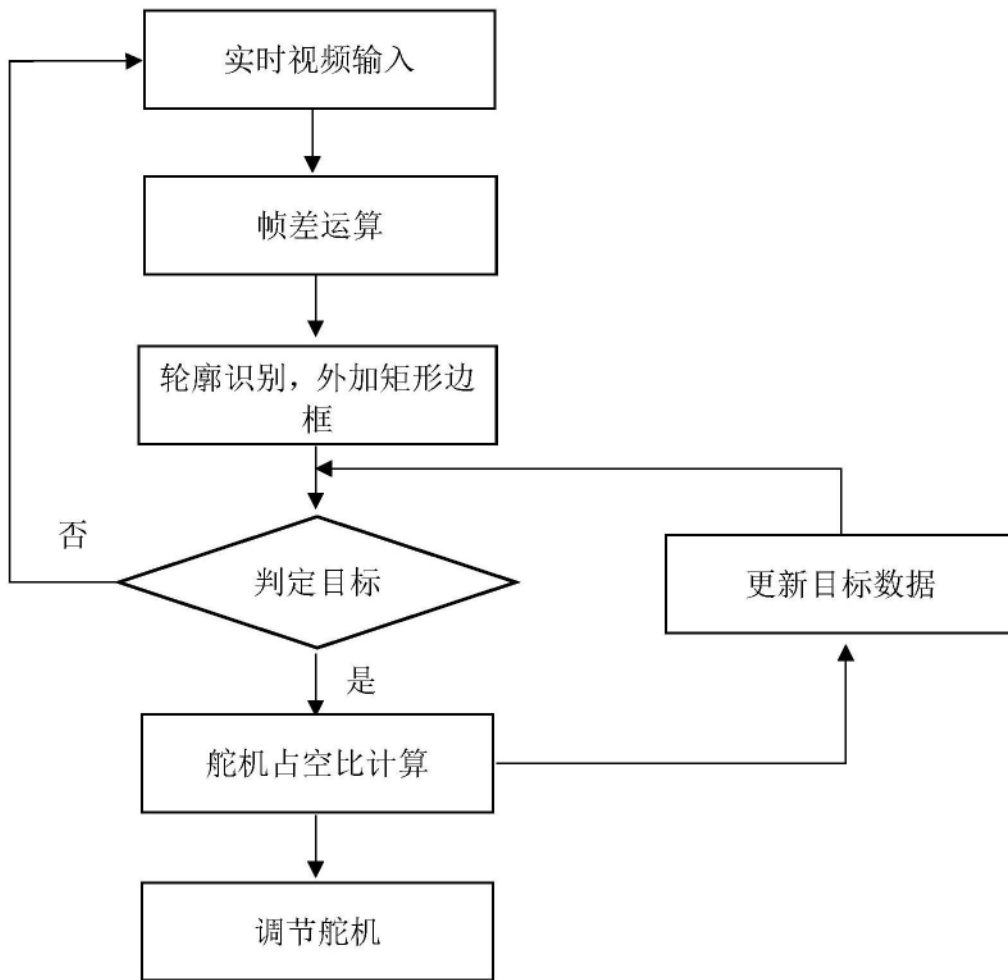


图2