

一种图像融合的基于行的硬件实现方法

申请号：[201410590976.9](#)

申请日：2014-10-29

申请(专利权)人 [复旦大学](#)

地址 [200433 上海市杨浦区邯郸路220号](#)

发明(设计)人 [范益波](#) [黄磊磊](#) [程魏](#) [金怡泽](#) [曾晓洋](#)

主分类号 [H04N5/262\(2006.01\)I](#)

分类号 [H04N5/262\(2006.01\)I](#) [H04N5/265\(2006.01\)I](#)

公开(公告)号 [104363385A](#)

公开(公告)日 [2015-02-18](#)

专利代理机构 [上海正旦专利代理有限公司](#) [31200](#)

代理人 [陆飞](#) [盛志范](#)



(12) 发明专利申请

(10) 申请公布号 CN 104363385 A

(43) 申请公布日 2015. 02. 18

(21) 申请号 201410590976. 9

(22) 申请日 2014. 10. 29

(71) 申请人 复旦大学

地址 200433 上海市杨浦区邯郸路 220 号

(72) 发明人 范益波 黄磊磊 程魏 金怡泽
曾晓洋

(74) 专利代理机构 上海正旦专利代理有限公司
31200

代理人 陆飞 盛志范

(51) Int. Cl.

H04N 5/262 (2006. 01)

H04N 5/265 (2006. 01)

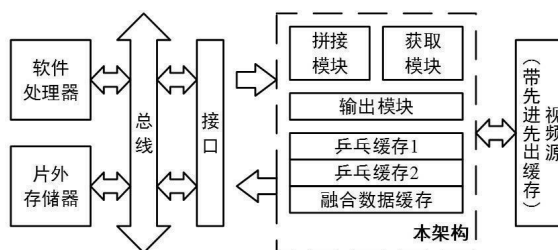
权利要求书3页 说明书4页 附图1页

(54) 发明名称

一种图像融合的基于行的硬件实现方法

(57) 摘要

本发明属于数字视频技术领域,具体为一种图像融合的基于行的硬件实现方法。对于一个普适的视频拼接应用,需要执行获取、投影、匹配、缩放和旋转、校正、缝合、拼接、输出等一系列操作。本发明将匹配操作交由软件执行一次,剩余的获取、校正、缝合、拼接和输出操作交由硬件执行;图像获取使用带有先入先出缓存的摄像头加以配合,图像校正,使用三个乘法器来完成;图像缝合使用基于行的缝合算法,图像拼接使用基于行的拼接算法,图像输出采用两种模式。本发明可以将处理单位从一帧像素缩减到一行像素,有效地减少硬件实现下的处理速度,从而高效地实现数字视频的实时融合操作。



1. 一种图像融合的基于行的硬件实现方法,在视频融合过程中,以一帧图像为单位执行获取、投影、匹配、缩放和旋转、校正、缝合、拼接、输出一系列操作;具体针对具有以下特征的图像融合:可以忽略尺寸失配、可以忽略角度失配、固定摄像头位置和角度,基于上述特征,投影、缩放和旋转不必执行,匹配操作只执行一次;其特征在于:

将匹配操作交由软件执行一次,剩余的获取、校正、缝合、拼接和输出操作交由硬件执行;

对于图像的获取,使用带有先入先出缓存(FIFO)的摄像头加以配合,使得对于图像数据的获取是间断的,并由获取模块控制,这也使得对于图像的获取操作基于行进行;

对于图像的校正,使用三个乘法器来完成;乘法器的一组因数是像素在红绿蓝(RGB)通道中的数值,另一组因数是各通道的校正系数;该系数同样由软件计算得到,用于调整红绿蓝(RGB)的相对权重和绝对权重;

对于图像的缝合,使用如下的基于行的缝合算法:

依据公式(1)计算第1行内重叠部分所有对应像素的绝对差,取值最小的一对像素点的位置作为缝合线的起点,并记为 i_1 :

$$d_{i,j} = |B_{1,i,j}^{ov} - B_{2,i,j}^{ov}|, i = 1, 2, 3, \dots, m; \quad j = 1 \quad (1)$$

其中, i 是像素所处的列坐标, j 是像素所处的行坐标,以视频源重叠的部分作为起点, $d_{i,j}$ 是重叠部分第 j 行第 i 列的对应像素的绝对差, $B_{1,i,j}^{ov}$ 和 $B_{2,i,j}^{ov}$ 分别是第1个视频源和第2个视频源重叠部分第 j 行第 i 列的像素,而所谓重叠部分指的是两个视频源所拍摄到的相同的一部分镜头,需要在融合时合并为一个镜头, m 是重叠部分的总列数;

依据公式(2)计算第2行直至最后1行的在 $[i_{j-1}-2, i_{j-1}+2]$ 范围内的所有对应像素的绝对差,取差值最小的一对像素点的位置作为该行的缝合点,并根据当前行数 j 记为 i_j :

$$d_{i,j} = |B_{1,i,j}^{ov} - B_{2,i,j}^{ov}|, i = i_{j-1} - 1, i_{j-1}, i_{j-1} + 1; \quad j = 2, 3, 4, \dots, n \quad (2)$$

其中, i 是像素所处的列坐标, j 是像素所处的行坐标,以视频源重叠的部分作为起点, $d_{i,j}$ 是重叠部分第 j 行第 i 列的对应像素的绝对差, $B_{1,i,j}^{ov}$ 和 $B_{2,i,j}^{ov}$ 分别是第1个视频源和第2个视频源重叠部分第 j 行第 i 列的像素, n 是重叠部分的总行数;

最后,将缝合线的起点 i_1 和后续的缝合点 $i_2, i_3, i_4, \dots, i_n$ 连接,即可得到缝合线;

对于图像的拼接,使用如下基于行的拼接算法:

根据公式(3)计算对应点拼接后的值:

$$F(p) = \alpha * I_1(p) + (1 - \alpha) * I_2(p), p \in [i_j - l_{blend}, i_j + l_{blend}] \quad (3)$$

其中, F 表示融合后图像, p 表示当前点, I_l 表示被融合图像, $l=1, 2$, i_j 是当前行的缝合点, l_{blend} 是执行拼接的范围, α 表示融合系数,由公式4计算都得到:

$$\alpha = \begin{cases} \max \left(1, d_1 / (d_1 + d_2) * (1 + 0.1 \partial I_1(p)) \right) & , p \leq i_j \\ \min \left(0, d_1 / (d_1 + d_2) * (1 - 0.1 \partial I_1(p)) \right) & , p > i_j \end{cases} \quad (4)$$

其中, d 表示当前点 p 到拼接范围的边界的距离;

对于图像的输出, 采用两种模式:

模式一, 将视频源的数据直接输出, 以供软件处理以完成只需要执行一次的匹配操作和对于校正数据的计算;

模式二, 将融合后的图像输出;

上述两种模式也都是基于行完成, 即在模式一情况下, 输出紧跟着对于一行原始数据的获取之后立刻进行; 模式二情况下, 输出紧跟这对于一行数据的拼接操作之后立刻进行。

2. 根据权利要求 1 所述的硬件实现方法, 其特征在于采用基于行的乒乓操作, 即在执行对于当行像素的获取的同时, 执行对于前一行像素的校正、缝合和拼接, 执行对于当行像素的校正、缝合和拼接的同时, 执行对于下一行像素的获取。

3. 根据权利要求 2 所述的硬件实现方法, 其特征在于对于视频源 A 和 B, 进行融合操作, 其对应的系统包括: 软件处理器, 片外存储器, 接口, 输送总线, 获取模块, 缝合模块, 拼接模块, 输出模块, 乒乓缓存 1, 乒乓缓存 2, 融合数据缓存; 其中, 获取模块、缝合模块、拼接模块、输出模块等分别对应实现获取、缝合、拼接、输出操作; 乒乓缓存 1、乒乓缓存 2 用于缓存基于行的乒乓操作过程中数据; 融合数据缓存用于缓存拼接后数据; 融合操作具体过程为:

(1) 令输出模块工作在模式一下, 依次将视频源的一帧图像数据和视频源的一帧图像数据, 通过接口和总线输送到片外存储器上;

(2) 软件处理器执行匹配算法计算出两个视频源的重叠部分, 并得到对于色彩的校正系数;

(3) 软件处理器将得到的结果输送到本硬件架构;

(4) 启动获取模块, 根据处理器计算的结果, 从视频源中读入对应的第一行数据, 通过校正用的三个乘法器, 最终存入乒乓缓存 1;

(5) 启动缝合模块, 根据处理器计算的结果, 寻找乒乓缓存 1 中的重叠部分的缝合点;

(6) 启动拼接模块, 根据缝合点对乒乓缓存 1 中对应的数据进行拼接, 得到拼接后数据, 存入融合数据缓存;

(7) 启动输出模块, 并令其工作在模式二下, 将融合数据输出;

(8) 在 5-7 执行的过程中, 同时启动获取模块, 根据处理器计算的结果, 从视频源中读入对应的第二行数据, 通过校正用的三个乘法器, 最终存入乒乓缓存 2;

(9) 启动缝合模块, 根据处理器计算的结果, 寻找乒乓缓存 2 中的重叠部分的缝合点;

(10) 启动拼接模块, 根据缝合点对乒乓缓存 2 中对应的数据进行拼接, 得到拼接后数据, 存入融合数据缓存;

(11) 启动输出模块, 并令其工作在模式二下, 将融合数据输出;

(12) 在 9-10 执行的过程中, 同时启动获取模块, 根据处理器计算的结果, 从视频源中读入对应的第二行数据, 通过校正用的三个乘法器, 最终存入乒乓缓存 1;

依次类推，
.....。

一种图像融合的基于行的硬件实现方法

技术领域

[0001] 本发明属于数字视频处理技术领域,具体涉及一种适用于图像融合的基于行的硬件实现方法。

背景技术

[0002] 随着电子多媒体的长足发展,人们对于宽屏乃至全景视频的需求也越来越旺盛。不管是看电影、玩游戏、还是视频会议、车载监控,人们都在追求更宽更大的视频体验。这种追求来源于宽屏全景能够提供普通视频所不能提供的沉浸式体验。在沉浸式的体验中,从情感上,人们能够更为身临其境地感受视频的氛围;从功能上,人们能够从视频中获取更多的信息。

[0003] 为了实现宽屏乃至全景,传统的方法是利用广角镜头进行拍摄。但这种方法会不可避免地引入至少以下三种问题。其一、由于拍摄范围过大所导致的细节分辨率下降;其二、由于广角镜头所引入的边缘扭曲甚至畸形;其三、昂贵的镜头和摄像机费用。

[0004] 作为广角镜头的候选方法,视频拼接慢慢地为人们所关注。该方向致力于从若干个分辨率较低的摄像头中获取源视频,通过拼接处理,从而产生出一个分辨率较高的融合后视频。由于每个摄像头只用于拍摄拼接后视频中的一部分区域,因此细节分辨率较高;普通镜头避免了广角引入的畸变;而由于实际使用的摄像头分辨率较低,因此成本相较低廉(此处并未考虑拼接成本)。

[0005] 对于一个普适的视频拼接应用,如图 1 所示,需要执行以下操作:

- 1、获取,包含了对于处理前视频的格式转换乃至视频解码;
- 2、投影,实际上,摄像头在摄影过程当中隐式地包含了由三维空间到二维空间的投影,这一投影会或多或少地导致边界处的扭曲以及相对于视频中心的失配。为了更好地进行匹配操作,需要将视频投影到一个更为合适的平面上,例如一个圆柱面,以减少扭曲或者失配所引入的影响;
- 3、匹配,利用 SIFT、SURF 或者其他算法,寻找视频间的特征点,并生成对应的转换矩阵和相对位移;
- 4、缩放和旋转,根据转换矩阵完成对于视频的缩放和旋转操作;
- 5、校正,消除由于摄像头之间的失配所导致的色差;
- 6、缝合,在被拼接视频中寻找最佳的缝合线;
- 7、拼接,沿着缝合线拼接视频;
- 8、输出,包含了对于处理后视频的格式转换乃至视频编码。

[0006] 这些操作往往都是以帧做单位的,即对于一帧图像,在完成了投影之后才能对其进行匹配,在完成了缝合之后才能进行拼接等,直接导致了图像融合模块对于存储空间和数据带宽的大量占用。这种占用最终会体现为效率低下和成本昂贵。

[0007] 本发明可以将处理单位从一帧像素缩减到一行像素,有效地减少硬件实现下的处理速度,从而高效地实现数字视频的实时融合操作。

发明内容

[0008] 本发明的目的在于提出一种可以克服现有技术不足的、能有效进行图像融合的基于行的硬件实现方法。

[0009] 本发明方法针对具有以下特征的图像融合：可以忽略的尺寸失配、可以忽略的角度失配、固定的摄像头位置和角度。该特征的典型应用为视频会议、可视电话、全景监控等。由于上述特征，投影、缩放和旋转可以不必执行，匹配操作可以只执行一次。由于这个原因，本发明将匹配操作交由软件执行一次，将剩余的获取、校正、缝合、拼接和输出操作交由硬件执行。

[0010] 对于图像的获取，本发明使用带有先入先出缓存(FIFO)的摄像头加以配合，这使得对于图像数据的获取可以是间断的，由获取模块控制的，而不必由于图像数据源源不断的进入而使用片外存储器，并占用数据带宽。同时，这也使得对于图像的获取操作可以是基于行进行的。

[0011] 对于图像的校正，本发明使用三个乘法器来完成。乘法器的一组因数是像素在红绿蓝(RGB)通道中的数值，另一组因数是各通道的校正系数，如图2所示。该系数同样由软件计算得到，用于调整RGB的相对权重和绝对权重。由于摄像头是固定的，这一数值也只需要被计算一次。值得注意的是这一校正算法实际是基于像素点的，因此，可以被扩充为基于行的。

[0012] 对于图像的缝合，本发明使用如下的基于行的缝合算法：

依据公式(1)计算第1行内重叠部分所有对应像素的绝对差，取值最小的一对像素点的位置作为缝合线的起点，并记为 i_1 ：

$$d_{i,j} = |B_{1,i,j}^{ov} - B_{2,i,j}^{ov}|, i = 1, 2, 3, \dots, m; \quad j = 1 \quad (1)$$

其中， i 是像素所处的列坐标， j 是像素所处的行坐标，以视频源重叠的部分作为起点， $d_{i,j}$ 是重叠部分第 j 行第 i 列的对应像素的绝对差， $B_{1,i,j}^{ov}$ 和 $B_{2,i,j}^{ov}$ 分别是第1个视频源和第2个视频源重叠部分第 j 行第 i 列的像素，而所谓重叠部分指的是两个视频源所拍摄到的相同的一部分镜头，需要在融合时合并为一个镜头， m 是重叠部分的总列数。

[0013] 依据公式(2)计算第2行直至最后1行的在 $[i_{j-1}-2, i_{j-1}+2]$ 范围内的所有对应像素的绝对差，取差值最小的一对像素点的位置作为该行的缝合点，并根据当前行数 j 记为 i_j ：

$$d_{i,j} = |B_{1,i,j}^{ov} - B_{2,i,j}^{ov}|, i = i_{j-1} - 1, i_{j-1}, i_{j-1} + 1; \quad j = 2, 3, 4, \dots, n \quad (2)$$

其中， i 是像素所处的列坐标， j 是像素所处的行坐标，以视频源重叠的部分作为起点， $d_{i,j}$ 是重叠部分第 j 行第 i 列的对应像素的绝对差， $B_{1,i,j}^{ov}$ 和 $B_{2,i,j}^{ov}$ 分别是第1个视频源和第2个视频源重叠部分第 j 行第 i 列的像素， n 是重叠部分的总行数。

[0014] 最后，将缝合线的起点 i_1 和后续的缝合点 $i_2, i_3, i_4, \dots, i_n$ 即可得到缝合线。所谓基于行指的是对于当前行缝合点的计算不需要依托于一整帧图像数据，即完成对于一整帧数据的处理后才能得到，而只需要当前行的数据即可。

[0015] 对于图像的拼接，本发明使用如下基于行的拼接算法：

根据公式(3)计算对应点拼接后的值：

$$F(p) = \alpha * I_1(p) + (1 - \alpha) * I_2(p), p \in [i_j - l_{blend}, i_j + l_{blend}] \quad (3)$$

其中, F 表示融合后图像, p 表示当前点, I_l 表示被融合图像, $l=1,2$, i_j 是当前行的缝合点, l_{blend} 是执行拼接的范围, α 表示融合系数, 由公式 4 计算都得到:

$$\alpha = \begin{cases} \max(1, d_1 / (d_1 + d_2) * (1 + 0.1 \theta I_1(p))) & , p \leq i_j \\ \min(0, d_1 / (d_1 + d_2) * (1 - 0.1 \theta I_1(p))) & , p > i_j \end{cases} \quad (4)$$

其中, d 表示当前点 p 到拼接范围的边界的距离。

[0016] 对于图像的输出, 本发明可以有两种模式。模式一, 将视频源的数据直接输出, 以供软件处理以完成只需要执行一次的匹配操作和对于校正数据的计算; 模式二, 将融合后的图像输出。上述两种模式也都是基于行完成的, 即在模式一情况下, 输出可以紧跟着对于一行原始数据的获取之后立刻进行; 模式二情况下, 输出可以紧跟这对于一行数据的拼接操作之后立刻进行。

[0017] 为了进一步优化速度, 本发明采用基于行的乒乓操作, 即在执行对于当行像素的获取的同时, 执行对于前一行像素的校正、缝合和拼接, 执行对于当行像素的校正、缝合和拼接的同时, 执行对于下一行像素的获取。由于单位为行, 极大地节省了存储空间和数据带宽, 使得乒乓式的处理流程是廉价而高效。

[0018] 综合上述说明, 可以得到本方法最后架构图, 如图 3 所示。

附图说明

[0019] 图 1: 普适的视频融合流程。

[0020] 图 2: 乘法校正器。

[0021] 图 3: 本发明方法的架构图。

具体实施方式

[0022] 下面通过实例, 进一步具体描述本发明方法。

[0023] 如对于视频源 A 和 B, 使用本发明完成对于其的融合操作, 具体过程为:

1、令输出模块工作在模式一下, 依次将视频源的一帧图像数据和视频源的一帧图像数据, 通过接口和总线输送到片外存储器上;

2、软件处理器执行匹配算法计算出两个视频源的重叠部分, 并得到对于色彩的校正系数;

3、软件处理器将得到的结果输送到本硬件架构;

4、启动获取模块, 根据处理器计算的结果, 从视频源中读入对应的第一行数据, 通过校正用的三个乘法器, 最终存入乒乓缓存 1;

5、启动缝合模块, 根据处理器计算的结果, 寻找乒乓缓存 1 中的重叠部分的缝合点;

6、启动拼接模块,根据缝合点对乒乓缓存 1 中对应的数据进行拼接,得到拼接后数据,存入融合数据缓存;

7、启动输出模块,并使之工作在模式二下,即将融合数据输出,而不是输出视频源的图像数据;

8、在 5-7 执行的过程中,同时启动获取模块,根据处理器计算的结果,从视频源中读入对应的第二行数据,通过校正用的三个乘法器,最终存入乒乓缓存 2;

9、启动缝合模块,根据处理器计算的结果,寻找乒乓缓存 2 中的重叠部分的缝合点;

10、启动拼接模块,根据缝合点对乒乓缓存 2 中对应的数据进行拼接,得到拼接后数据,存入融合数据缓存;

11、启动输出模块,并使之工作在模式二下,即将融合数据输出,而不是输出视频源的图像数据;

12、在 9-10 执行的过程中,同时启动获取模块,根据处理器计算的结果,从视频源中读入对应的第二行数据,通过校正用的三个乘法器,最终存入乒乓缓存 1。

[0024] 依次类推,

.....。

[0025] 其对应的系统包括:软件处理器,片外存储器,接口,输送总线,获取模块,缝合模块,拼接模块,输出模块,乒乓缓存 1,乒乓缓存 2,融合数据缓存;其中,获取模块、缝合模块、拼接模块、输出模块等分别对应实现获取、缝合、拼接、输出操作;乒乓缓存 1、乒乓缓存 2 用于缓存基于行的乒乓操作过程中数据;融合数据缓存用于缓存拼接后数据。

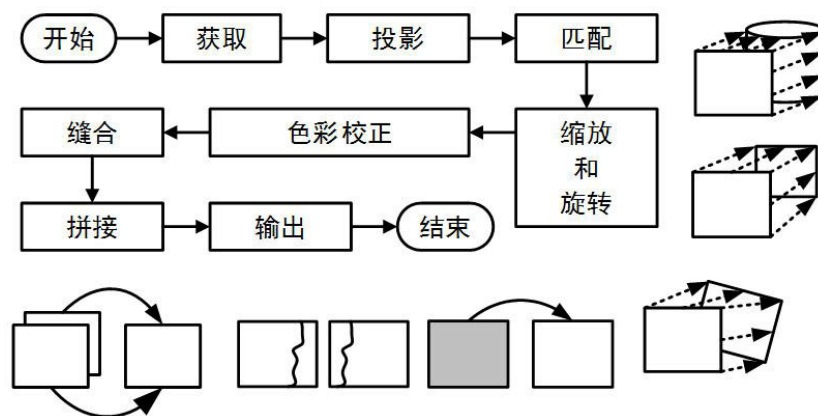


图 1

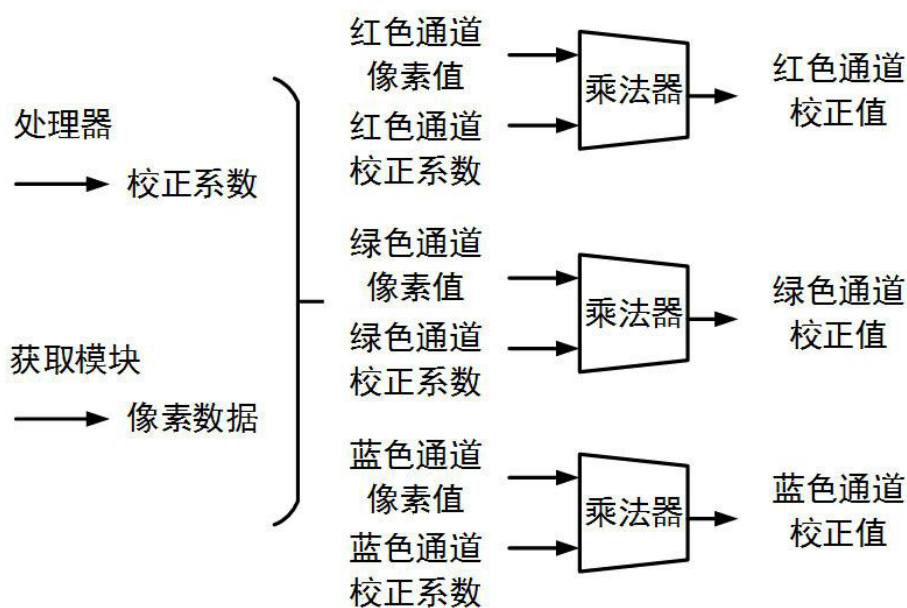


图 2

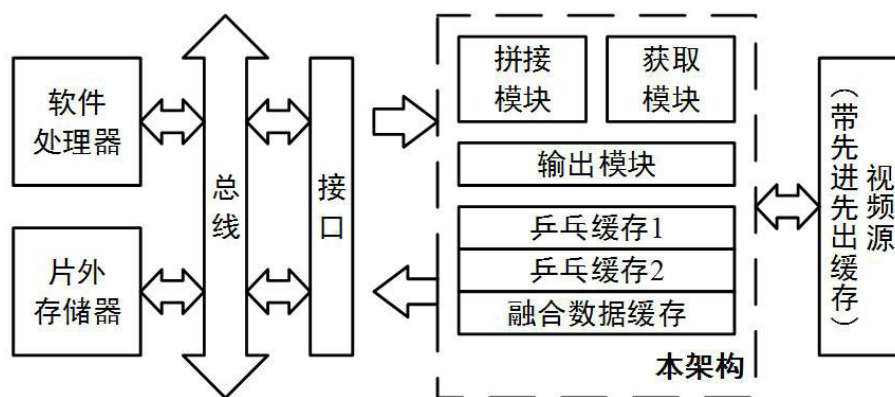


图 3