



(12)发明专利申请

(10)申请公布号 CN 107197171 A

(43)申请公布日 2017.09.22

(21)申请号 201710482260.0

(22)申请日 2017.06.22

(71)申请人 西南大学

地址 400715 重庆市北碚区天生路2号

(72)发明人 周素梅 邓蜀国

(74)专利代理机构 北京国坤专利代理事务所

(普通合伙) 11491

代理人 姜彦

(51)Int.Cl.

H04N 5/238(2006.01)

H04N 5/243(2006.01)

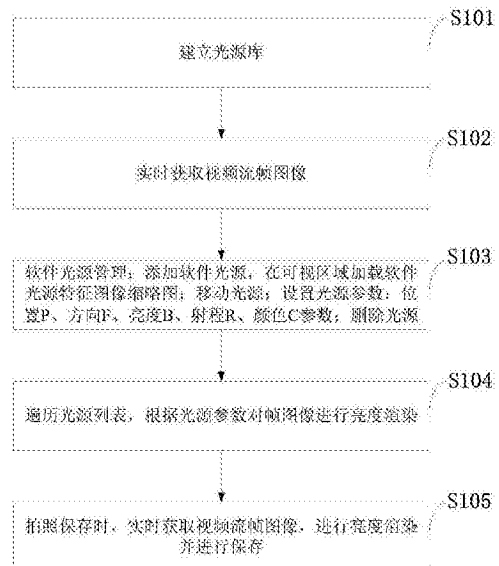
权利要求书1页 说明书5页 附图1页

(54)发明名称

一种添加智能软件光源的数码拍照处理方法

(57)摘要

本发明公开了一种添加智能软件光源的数码拍照处理方法,获取视频流帧图像,用户在可视区域指定位置置入软件光源,系统自动获取图像区域颜色值,提取(r,g,b)分量,根据光源参数计算渲染范围并叠加亮度。本发明可对视频任意区域添加一个或多个软件光源(数量在理论上没有限制),并可设置灯光方向F、灯光亮度B、灯光射程R、光线颜色C等参数,能够根据灯光参数智能、准确渲染指定区域亮度,达到提高数码拍照成像质量目的;在外部物理光源不能满足数码拍照条件时,可以通过软件实现添加多种类型光源,调整灯光方向、射程和亮度等参数,实现对视频图像指定区域进行智能渲染,从而提高照片成像质量。



1. 一种添加智能软件光源的数码拍照处理方法,其特征在于,所述添加智能软件光源的数码拍照处理方法获取视频流帧图像,用户在可视区域指定位置置入软件光源,系统自动获取图像区域颜色值,提取(r,g,b)分量,根据光源参数计算渲染范围并叠加亮度。

2. 如权利要求1所述的添加智能软件光源的数码拍照处理方法,其特征在于,所述添加智能软件光源的数码拍照处理方法包括以下步骤:

步骤一,建立光源库;

步骤二,实时获取视频流帧图像;

步骤三,软件光源管理;添加软件光源,在可视区域加载软件光源特征图像缩略图;移动光源;设置光源参数:位置P、方向F、亮度B、射程R、颜色C参数;删除光源;

步骤四,遍历光源列表,根据光源参数对帧图像进行亮度渲染;

步骤五,拍照保存时,实时获取视频流帧图像,进行亮度渲染并进行保存。

3. 如权利要求2所述的添加智能软件光源的数码拍照处理方法,其特征在于,所述在步骤四中软件光源管理包括:光源位置的移动;修改光源参数;删除指定光源。

4. 如权利要求2所述的添加智能软件光源的数码拍照处理方法,其特征在于,所述步骤五中根据光源参数对帧图像进行亮度渲染包括:

获取图像区域颜色值,提取(r,g,b)分量;

根据灯光参数位置P、亮度B、射程R、方向F、光线颜色C参数计算渲染范围和光线颜色叠加亮度,并在可视区域实时显示最终渲染效果图样。

5. 如权利要求2所述的添加智能软件光源的数码拍照处理方法,其特征在于,所述视频流帧图像的加密方法包括以下步骤:

步骤一、定义随机函数funA,根据视频文件的长度,产生若干对随机整数(x,y),存入集合S {}中,并将集合S保存到数据库中对应的视频文件属性字段中,控制生成视频加密文件大小的Y值<10;

步骤二、按字节读取视频文件流,从数据库中依据视频文件的信息,读取存入的集合S {},获取所有随机数对;

步骤三、根据数据对(x,y),在文件流中的X位置写入Y长度的随机字符串,将文件流写入加密视频文件流中;

步骤四、重复步骤三,直到文件读写完成。

一种添加智能软件光源的数码拍照处理方法

技术领域

[0001] 本发明属于数码拍照处理技术领域,尤其涉及一种添加智能软件光源的数码拍照处理方法。

背景技术

[0002] 目前,通过电脑进行人像数码拍照技术已经成熟,并且应用非常广泛。专业拍照一般在摄影棚中进行,摄影棚安装有专业的光源,一般需要主灯、辅灯、地灯、顶灯各1台,背景灯1到2台。这种布光方式使影像光线柔和,曝光均匀,能够去掉眼袋、消除鼻沟阴影,使人像更加唯美。在具体拍照操作时,特别是对一些高校、中小学、机关、企事业单位在大批量采集照片时,往往因为没有专业的摄影棚,并且时间仓促,一般就地选择办公室、会议室作为拍照地点,依赖房间顶部的日光灯或靠窗口、过道的散射光作为光源,由于光源不均衡,拍照时,常常出现人像局部曝光过量、不足或失衡,在一定程度上影响了最终照片效果。

[0003] 综上所述,目前大批量采集照片时存在光源不均衡,出现人像局部曝光过量、不足或失衡,影响最终照片效果。

发明内容

[0004] 本发明的目的在于提供一种添加智能软件光源的数码拍照处理方法,旨在解决目前大批量采集照片时存在光源不均衡,出现人像局部曝光过量、不足或失衡,影响最终照片效果的问题。

[0005] 本发明是这样实现的:在窗口中实时捕获并显示摄像头视频图像,选择光源库,拖动指定光源到视频指定位置,系统将以默认光源参数在该位置对视频进行渲染,用户可以通过鼠标或菜单对光源参数进行修改,系统根据用户修改后的光源参数对视频图像实时进行渲染。

[0006] 进一步,所述添加智能软件光源主要包括以下步骤:

[0007] 步骤一,建立光源库:即动态光源数组,每一数组元素代表一个光源,每一光源具有位置P、方向F、亮度B、射程R、颜色C等属性;

[0008] 步骤二,实时获取视频流帧图像;

[0009] 步骤三,光源管理;

[0010] 1、添加光源:在光源库中选择一个光源,将其拖动到可视窗口指定位置,即完成了软件光源的添加(此时,在可视窗口显示光源特征图像缩略图);

[0011] 2、移动光源:在光源图像区域建立的位置热区,在该区域内拖动鼠标即可直观完成对光源位置的移动;

[0012] 3、设置光源参数:

[0013] 方法一:通过右键菜单:设置位置P、方向F、亮度B、射程R、颜色C等参数精确数值;

[0014] 方法二:通过鼠标操作:在光源图像区域建立参数热区:包括方向F热区、亮度B热区、射程R热区,通过鼠标在热区内拖动相关操作杆即可直观完成参数设置;

[0015] 4、删除光源：根据视频图像亮度平衡需要，可删除多余光源，删除光源即删除动态光源数组中指定元素即可；

[0016] 步骤四，遍历光源列表，根据光源参数对帧图像进行亮度渲染：获取图像区域颜色原始值，提取 (r, g, b) 分量；根据光源位置P、亮度B、射程R、方向F、光线颜色C等参数计算渲染后应叠加的颜色值并提取其颜色分量 (r, g, b)，将原始颜色值的 (r, g, b) 分量同步叠加光源渲染颜色值的 (r, g, b)，即可在可视区域实时显示最终渲染效果图样；

[0017] 步骤五，拍照保存时，实时获取视频流帧图像，进行亮度渲染并进行保存。

[0018] 进一步，所述视频流帧图像的加密方法包括以下步骤：

[0019] 步骤一、定义随机函数funA，根据视频文件的长度，产生若干对随机整数 (x, y)，存入集合S {} 中，并将集合S保存到数据库中对应的视频文件属性字段中，控制生成视频加密文件大小的Y值<10；

[0020] 步骤二、按字节读取视频文件流，从数据库中依据视频文件的信息，读取存入的集合S {}，获取所有随机数对；

[0021] 步骤三、根据数据对 (x, y)，在文件流中的X位置写入Y长度的随机字符串，将文件流写入加密视频文件流中；

[0022] 步骤四、重复步骤三，直到文件读写完成。

[0023] 本发明提供的添加智能软件光源的数码拍照处理方法，可对视频任意区域添加一个或多个软件光源（数量在理论上没有限制），并可分别设置灯光方向F、灯光亮度B、灯光射程R、光线颜色C等，能够根据画面需要智能、准确渲染指定区域亮度，达到提高数码拍照成像质量目的；在外部物理光源不能满足数码拍照条件时，可以通过软件实现添加多种类型光源，调整灯光方向、射程和亮度等参数，实现对视频图像指定区域进行智能渲染，从而提高照片成像质量。

附图说明

[0024] 图1是本发明实施例提供的添加智能软件光源的数码拍照处理方法流程图。

具体实施方式

[0025] 为了使本发明的目的、技术方案及优点更加清楚明白，以下结合实施例，对本发明进行进一步详细说明。应当理解，此处所描述的具体实施例仅仅用以解释本发明，并不用于限定本发明。

[0026] 下面结合附图对本发明的应用原理作详细的描述。

[0027] 如图1所示，本发明实施例提供的添加智能软件光源的数码拍照处理方法包括以下步骤：

[0028] S101：建立光源库；

[0029] S102：实时获取视频流帧图像；

[0030] S103：软件光源管理；添加软件光源，在可视区域加载软件光源特征图像缩略图；移动光源；设置光源参数：位置P、方向F、亮度B、射程R、颜色C 参数；删除光源；

[0031] S104：遍历光源列表，根据光源参数对帧图像进行亮度渲染；

[0032] S105：拍照保存时，实时获取视频流帧图像，进行亮度渲染并进行保存。

[0033] 在步骤S103中:光源参数设置:(1)通过鼠标拖动相关热区操作杆完成;(2)通过右键菜单进行设置。

[0034] 所述视频流帧图像的加密方法包括以下步骤:

[0035] 步骤一、定义随机函数funA,根据视频文件的长度,产生若干对随机整数(x,y),存入集合S {}中,并将集合S保存到数据库中对应的视频文件属性字段中,控制生成视频加密文件大小的Y值<10;

[0036] 步骤二、按字节读取视频文件流,从数据库中依据视频文件的信息,读取存入的集合S {},获取所有随机数对;

[0037] 步骤三、根据数据对(x,y),在文件流中的X位置写入Y长度的随机字符串,将文件流写入加密视频文件流中;

[0038] 步骤四、重复步骤三,直到文件读写完成。

[0039] 下面结合具体实施例对本发明的应用原理作进一步的描述。

[0040] 本发明实施例提供的添加智能软件光源的数码拍照处理方法包括以下步骤:

[0041] 1、建立光源库

```
public struct Lamp
{
    public int number;           //光源序号
    public string name;         //光源名称
    public Image lampImage;     //光源图像
    public Point position;      //光源中心位置
    [0042] public Size size;      //光源原始尺寸大小
    public float resizeRat;     //光源显示缩放比例
    public float brightness;    //灯光亮度
    public float range;        //灯光射程:照射面的半径
    public float rayAngle1;    //灯光起始角度: 0-360 度
    public float rayAngle2;    //灯光终止角度: 0-360 度
    [0043] public Color rayColor; //灯光颜色
}
```

[0044] public static ArrayList Lamps;//动态光源结构数组:已经加载的光源

[0045] 其中:光源序号为编号流水如1,2,3……;光源名称为对光源的标记,对多个光源便于识别;光源图像为特征图片,比如灯泡、太阳……;光源中心位置为中心点坐标;光源原始尺寸为光源图片的实际大小;光源显示缩放比例即在窗口中缩放显示,便于对光源的操作;亮度、射程、起始角度、终止角度和颜色分别为光源的不同特征参数。

[0046] 光源库即为预定义一批常用光源列表,如太阳、日光灯……

[0047] 2、实时获取视频流帧图像。

```
Bitmap VideoFrameBitmap = (Bitmap)
videoSourcePlayer.GetCurrentVideoFrame();
```

[0049] 3、光源管理。

[0050] (1)添加光源:AddLamp(Bitmap LampBitmap,LampStyle S,Point P,float F1, float F2,float B,int R,Color C),即在激活的动态光源数组中增加一个元素,同时在窗口中显示出来。

[0051] (2)移动光源:MoveLamp(int LampNumber,Point P);即改变指定光源的坐标。

[0052] (3) 设置光源参数:EditLamp(int LampNumber),即设置指定光源的位置P、亮度B、射程R、方向角F1、F2、光线颜色C等参数。

[0053] 其中位置P的范围是视频可视区域内;亮度B范围是0至255;射程B范围是可视区域最大范围max(VideoFrameW,VideoFrameH);方向角F1(起始角)、方向F2(终止角):范围是 $F1 \geq 0 \& \& F1 <= 360, F2 \geq 0 \& \& F2 <= 360$ 。

[0054] (4) 删除光源>DeleteLamp(int LampNumber)。即在激活的动态光源数组中删除一个元素,同时在窗口中不再显示该光源。

[0055] 4、遍历光源列表,根据光源参数对帧图像进行亮度渲染:获取图像区域颜色原始值,提取(r,g,b)分量;根据光源位置P、亮度B、射程R、方向F、光线颜色C等参数计算渲染后应叠加的颜色值并提取其颜色分量(r,g,b),将原始颜色值的(r,g,b)分量同步叠加光源渲染颜色值的(r,g,b),即可在可视区域实时显示最终渲染效果图样。

[0056] (1):获取图像区域原始颜色值Color_Old(x,y),提取(r,g,b)分量。

[0057] (2):根据灯光参数位置P、亮度B、射程R、方向F、颜色C等参数计算光线颜色ColorBrightness(x,y),并提取其(r,g,b)分量。

[0058] (3) 叠加光线亮度:在原始颜色值的基础上叠加光线颜色值,即在r、g、b分量上同步叠加,在可视区域实时显示最终效果

[0059] $r += (\text{int}) \text{BrightnessR};$

[0060] $g += (\text{int}) \text{BrightnessG};$

[0061] $b += (\text{int}) \text{BrightnessB};$

[0062] 5、保存时,实时获取视频流帧图像,重复上述步骤进行亮度渲染并对指定的照片区域进行裁剪、保存。

[0063] $\text{Bitmap SavePhoto} = \text{CropBitmap}(\text{VideoFrameBitmap}, \text{SavePhotoRectangle});$

[0064] $\text{SavePhoto.Save}(\text{SavePhotoFileName}, \text{SavePhotoFileFormat.JPG}).$

[0065] 以上技术处理说明:

[0066] 1、在光源库区域内建立点击选择光源热区(Δ 、 ∇)、滚动选择光源热区和拖动添加光源热区。热区为一系列坐标数组,用于感应鼠标的位置和事件。

[0067] $\text{Point}[] \text{p_lampMode_DownArea};$ //上部 Δ 选择光源热区:用于选择上一个光源

[0068] $\text{Point}[] \text{p_lampMode_UpArea};$ //下部 ∇ 选择光源热区:用于选择下一个光源

[0069] $\text{Point}[] \text{p_lampMode_RollArea};$ //滚动选择光源热区

[0070] $\text{Point}[] \text{p_lampMode_Area};$ //拖动添加光源热区

[0071] 2、在光源特征图像区域内建立光源参数设置热区和移动热区,进入光源特征图像缩略区域,系统自动放大显示光源特征图像,并响应鼠标点击、滚动、拖动操作,触发光源位置移动、参数设置事件,并根据鼠标对光源参数操纵杆的位移量实时渲染可视区域。

[0072] 本发明提供的智能软件的实现原理:光源的亮度由光源中心沿灯光方向呈线性衰减,即以光源中心颜色为基准,提取(r,g,b)分量,通过同步线性改变其r、g、b分量的数值,即可实现光源亮度的衰减。在实时渲染视频亮度时放弃传统的像素法(通过开发环境提供的专门获取颜色的函数来获取指定位置像素的颜色),而是采用内存法,即通过获取图像内存变量的扫描宽度、像素宽度,对指定位置像素的颜色进行内存寻址,通过直接对内存的操作达到修改颜色亮度的目,部分关键代码如下:

```
[0073] int Stride=Math.Abs (bmpData.Stride); //扫描宽度;
[0074] int PixelWidth=Stride/bmpData.Width; //一个像素所占的位数4位:(R,G,B,
Y), 3位:(R,G,B)。
[0075] byte[] rgb Values=new byte[bytes]; //位图数据字节数组;
[0076] //原始颜色值
[0077] int r=rgbValues[y*Stride+x*PixelWidth+2]; //R为红色分量;
[0078] int g=rgbValues[y*Stride+x*PixelWidth+1]; //G为绿色分量;
[0079] int b=rgbValues[y*Stride+x*PixelWidth+0]; //B为蓝色分量。
[0080] //光源r,g,b叠加
[0081] //内存数据直接写入
[0082] rgb Values[y*Stride+x*PixelWidth+0] = (byte) (b);
[0083] rgb Values[y*Stride+x*PixelWidth+1] = (byte) (g);
[0084] rgb Values[y*Stride+x*PixelWidth+2] = (byte) (r);
[0085] Marshal.Copy (rgb Values,0,ptr,bytes);
[0086] 通过内存法进行渲染,处理、存储速度极快,在实际应用中几乎感觉不到时间延迟。
```

[0087] 在对视频进行渲染过程中,涉及到较大的图像内存变量,尽管开发语言提供自动内存管理,但是为提高内存利用效率和防止内存溢出,系统专门增加了内存变量管理进程,对废弃的变量及时进行清除,进一步提高了视频渲染的速度和稳定性。在长时间(连续3天不间断)、画质(1080P)、高速率(次/50ms)情况下对视频源、光源、头像区进行随机移动、360度范围内任意角度旋转等超强度压力测试,系统均能正常运行。软件在无专业摄影棚,特别是针对学校、机关、企事业等单位进行批量拍照时具有很强的实用性。相对软件光源,物理光源在成像效果上更具层次感和真实性,如条件许可,还是尽量搭建专门的物理光源。

[0088] 以上所述仅为本发明的较佳实施例而已,并不用以限制本发明,凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等,均应包含在本发明的保护范围之内。

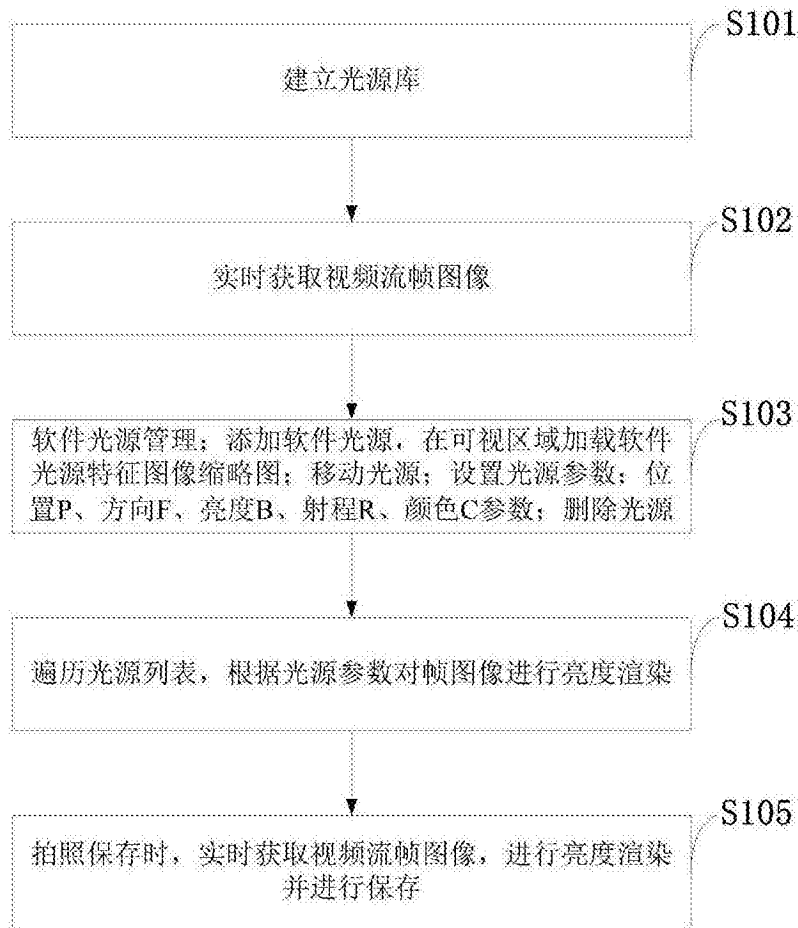


图1