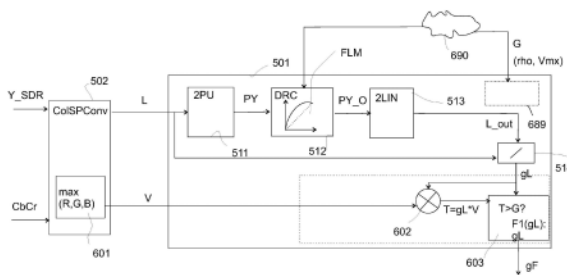




审查员 范慧慧

权利要求书3页 说明书25页 附图13页



1. 一种发光亮度处理器 (501), 其被布置为根据具有以第一峰值亮度 (PB\_HDR) 为特征的第一发光亮度动态范围的输入图像 (MAST\_HDR) 的空间共位的像素的输入发光亮度 (L) 来计算具有以第二峰值亮度 (PB\_LDR; PB\_MDR) 为特征的第二发光亮度动态范围的输出图像 (Im\_LDR; Im3000nit) 的像素的输出发光亮度, 其特征在于, 所述发光亮度处理器包括:

增益计算单元 (514), 其被布置为基于发光亮度映射函数 (FLM), 通过将乘法因子计算为所述发光亮度映射函数针对所述输入发光亮度的输出除以所述输入发光亮度的除法  $gL = FLM(L) / L$  来计算所述乘法因子 (gL), 所述发光亮度映射函数 (FLM) 指示如何调节所述输入发光亮度以变为对应的中间输出发光亮度, 所述乘法因子针对所述输入图像的图像像素的任何输入颜色的所述输入发光亮度被定义;

溢出计算器 (602), 其被布置为计算指示所述中间输出发光亮度在所述输入颜色的色度处的上色域边界上方多远处的溢出度量 (T), 所述中间输出发光亮度是通过将所述乘法因子 (gL) 应用于定义所述输入颜色的相应的颜色分量而得到的;

增益因子修改单元 (603), 其被布置为在所述溢出度量大于阈值 (G) 的情况下确定较低的替代增益因子 (F1 (gL)), 并且被布置为在其他情况下保持所述乘法因子 (gL), 并且被布置为在所述溢出度量大于所述阈值 (G) 的情况下输出所述较低的替代增益因子 (F1 (gL)) 作为最终增益因子 (gF) 或者在其他情况下输出所述乘法因子 (gL) 作为最终增益因子 (gF); 以及

乘法器 (530), 其用于将所述输入颜色 (R'G'B'\_nrm) 的相应的分量乘以所述最终增益因子 (gF) 以获得具有所述输出发光亮度的输出颜色 (R'G'B'\_HDR) 的相应的分量。

2. 根据权利要求1所述的发光亮度处理器 (501), 包括: 数据接收单元 (689), 其被布置为通过网络 (690) 从内容的创建者接收所述阈值 (G)。

3. 根据权利要求1或2所述的发光亮度处理器 (501), 包括: 数据接收单元 (689), 其被布置为通过网络 (690) 接收函数 (FADAP) 以根据所述内容的所述创建者确定所述替代增益因子。

4. 根据权利要求1或2所述的发光亮度处理器 (501), 包括: 数据接收单元 (689), 其被布置为通过网络 (690) 从内容的创建者接收修剪参数 (Vmx), 所述修剪参数指示颜色位于色域上边界上方多远处, 所述修剪参数被定义为所述颜色中的红色分量、绿色分量和蓝色分量中的最大的一个, 指定针对高于所述修剪参数的颜色发光亮度允许色调保留的颜色修剪。

5. 根据权利要求1或2所述的发光亮度处理器 (501), 包括: 部分映射计算器 (903), 其被布置为确定达到可实现的色域内颜色所需的所述输入颜色朝向色域边界的第一校正分数, 所述第一校正分数是要通过与最终乘法因子的乘法来处理的, 并且所述部分映射计算器被布置为确定用于确定用于驱动颜色饱和度计算的饱和度乘法器 (S) 的剩余第二校正分数。

6. 根据权利要求5所述的发光亮度处理器 (501), 包括: 数据接收单元 (689), 其被布置为通过网络 (690) 从内容的创建者接收分数值 (rho), 所述分数值确定所述第一校正分数和所述第二校正分数。

7. 根据权利要求1或2所述的发光亮度处理器 (501), 其中, 所述溢出度量 (T) 被计算为像素颜色中的红色分量、绿色分量和蓝色分量中的最高值乘以所述乘法因子 (gL)。

8. 一种高动态范围视频编码器 (1201), 包括:

输入部, 其用于从图像源 (1202) 接收输入图像;

编码器,其用于将所述输入图像编码为输出图像并对至少一个发光亮度映射函数(F<sub>Lt</sub>)进行编码;其特征在于,所述编码器包括图像评价处理器(1210),所述图像评价处理器被布置为分析所述视频的图像的颜色属性,确定由根据权利要求1所述的发光亮度处理器应用的阈值(G),并且所述编码器被布置为输出所述阈值(G)作为元数据。

9.一种高动态范围视频编码系统,包括:

用户接口(1203),其允许人类颜色分级者(1204)指定根据权利要求1所述的阈值(G)的值;

根据权利要求1所述的发光亮度处理器(501),其还被布置为接收并使用由所述人类颜色分级者指定的所述阈值(G)的所述值,并且被连接到显示器(1220)以显示所述发光亮度处理器的所述输出图像;并且所述高动态范围视频编码系统还包括编码器,所述编码器被布置为编码并输出所述阈值(G)。

10.一种高动态范围视频解码器(1300),包括:

输入部,其用于接收输入图像和发光亮度映射函数(FLM);

第二输入部,被布置为接收阈值(G)的值;

根据权利要求1至7中的任一项所述的发光亮度处理器(501),其被布置为通过以下操作来计算输出图像:使用所述发光亮度映射函数(FLM)和所述阈值(G)来改变所述输入图像的像素的发光亮度而产生输出图像;以及

输出部,其用于输出所述输出图像。

11.一种发光亮度处理方法,其用于根据具有以第一峰值亮度(PB<sub>HDR</sub>)为特征的第一发光亮度动态范围的输入图像(MAST<sub>HDR</sub>)的空间共位的像素的输入发光亮度来计算具有以第二峰值亮度(PB<sub>LDR</sub>;PB<sub>MDR</sub>)为特征的第二发光亮度动态范围的输出图像(Im<sub>LDR</sub>;Im3000nit)的像素的输出发光亮度,所述发光亮度处理方法包括:

基于发光亮度映射函数(FLM),通过将乘法因子计算为所述发光亮度映射函数针对所述输入发光亮度的输出除以所述输入发光亮度的除法 $gL = FLM(L) / L$ 来计算所述乘法因子(gL),所述发光亮度映射函数(FLM)指示如何调节所述输入发光亮度以变为中间输出发光亮度,所述乘法因子针对所述输入图像的图像像素的任何输入颜色的所述输入发光亮度被定义;

计算指示所述中间输出发光亮度在所述输入颜色的色度处的上色域边界上方多远处的溢出度量(T),所述中间输出发光亮度是通过将所述乘法因子(gL)应用于定义所述输入颜色的相应的颜色分量而得到的;

在所述溢出度量大于阈值(G)的情况下确定较低的替代增益因子(F1(gL)),并且在其他情况下保持所述乘法因子(gL),并且在所述溢出度量大于所述阈值(G)的情况下输出所述较低的替代增益因子(F1(gL))作为最终增益因子(gF)或者在其他情况下输出所述乘法因子(gL)作为最终增益因子(gF);并且

将所述输入颜色(R'G'B'<sub>nrm</sub>)的相应的分量乘以所述最终增益因子(gF)以获得具有所述输出发光亮度的输出颜色(R'G'B'<sub>HDR</sub>)的相应的分量。

12.根据权利要求11所述的发光亮度处理方法,包括:通过任何连接的网络从内容的创建者接收所述阈值(G),以将所述阈值应用于对所述输出颜色的所述计算中。

13.根据权利要求11或12所述的发光亮度处理方法,包括:通过任何连接的网络从内容

的创建者接收函数 (FADAP) ,以确定所述替代增益因子。

## 一种发光亮度处理器及方法、高动态范围视频编码器、解码器及编码系统

### 技术领域

[0001] 本发明涉及用于对高动态范围图像进行编码的方法和装置,特别涉及改变图像像素发光亮度以将图像转换为不同发光亮度动态范围的方法和装置。

### 背景技术

[0002] 大约5年前,高动态范围视频编码的新技术被引入。

[0003] 这种新技术与传统的视频编码形成对比,在此之前,根据传统的视频编码,所有视频都被编码,这在现在被称为标准动态范围(SDR)视频编码(又名低动态范围视频编码;LDR)。Rec.709的SDR的luma代码定义由于其近似平方根函数形状( $\text{luma: } Y = \sqrt{\text{发光亮度 } L}$ )而只能对大约1000:1的发光亮度动态(又名光电传递函数OETF)范围进行编码(具有8或10位luma),luma被编码用于当时所有显示器的典型渲染函数,其发光亮度大致在0.1nit到100nit之间,后者是所谓的峰值亮度(PB)。

[0004] 第一个被引入市场的HDR编解码器是HDR10,它例如用于创建新的黑带HDR蓝光,该新的黑带HDR蓝光仅将OETF改变为在SMPTE 2084中标准化的更具对数形状的感知量化器函数,该感知量化器函数允许定义亮度在1/10000nit到10000nit之间的lumas,这样的亮度足以用于实际的HDR视频生产。在计算了lumas后,一个lumas只有一个10位的像素平面(或者具有两个色度平面:3位的Cb平面和Cr平面),可以在流水线上进一步对其进行经典处置而“好像”它们在数学上是SDR图像,例如是MPEG压缩图像。当然,接收侧现在应该得到HDR图像而不是SDR图像,否则将发生错误的渲染。如果仅线性映射(编码图像中最大的白色)(即,将峰值亮度PB\_C编码为SDR显示峰值亮度PB\_D),则PB\_C=1000nit的图像会看起来暗了10倍,这意味着夜景变得无法观看。由于PQ OETF的对数性质,HDR10图像是可观看的,但是对比度却非常差,致使其看起来像被洗过且亮度错误。

[0005] 这样的单纯的HDR视频图像的问题(又名HDR分级,其中,词语分级指示:在PB\_C例如为1000nit(或更高)的图像表示中,为了使HDR场景在该表示中看起来最优,各种场景对象应该具有什么亮度)在于,它只能在相同的PB\_D=1000nit的显示器上得到正确显示,并且因此在类似的周围光照条件下会很快发现这样的蓝光光盘并不总是能完美显示,并且在这种情况下夜景可能无法观看。

[0006] 为此,更高级的HDR视频编码器对HDR场景的两个不同分级(较高动态范围的图像,例如,PB\_C=5000nit的图像;以及较低动态范围的图像,通常是PB\_C=100nit的SDR图像,因为它然后能够立即在传统的LDR显示器上显示)进行编码。

[0007] 现实世界场景(但是均匀照射的场景会因对象反射的比率为100:1而具有较小的动态范围)能够具有相当高的动态范围。例如,对于外部阳光照射具有小开口的洞穴可以在10000nit的PB\_C参考表示上,该PB\_C参考表示包含适用于家庭电视观看的场景的HDR分级,其中,洞穴像素的发光亮度远低于1nit,而户外像素的发光亮度则高达10000nit。如此具有挑战性的HDR图像无法被轻易转换到较低的动态范围(例如,在进行SDR时至少为100倍),特

别是在内容创建者希望在SDR渲染中仍然传达与利用图1阐明的内容相当相似的HDR外观的情况下,这在原理上是能够完成的。

[0008] 这里应该很好理解的是,如果我们在谈论对HDR图像和对应的较低动态范围图像的编码,则可能存在约束。代替实际传送两幅单独分级的图像(原始场景的HDR和100nit的PB\_CSDR分级图像),通常一幅分级图像(将利用图2进一步阐明)传送一对极值端点得到不同分级的不同PB\_C图像中的仅一幅分级图像,并且作为元数据函数而将接收到的图像的像素的颜色(特别是它们的发光亮度)映射成其他分级图像的颜色。

[0009] 应当理解,有两类图像编码器:第一类(模式1)将HDR图像传送到接收器,HDR10也是如此,因此能够例如重新使用PQ OETF对这些图像进行编码,但是添加了规定如何将该HDR图像的发光亮度降级到较低的动态范围(例如根据接收到的HDR图像来计算SDR 100nit PB\_C图像)的元数据。第二类(模式2)传达该对的SDR图像以及反向形状的函数以升级到HDR图像,这对于服务大量已经安装的传统电视很有用(这涉及对任何附加技术的另外的技术约束)。

[0010] 有一些技术机制可以重新分级到中间PB图像,这被称为中等动态范围(MDR)图像,因此代替将接收到的SDR图像转换到内容创建者在创建侧或传输侧创建的原始(例如,5000nit)PB\_C图像,接收器(例如,机顶盒、电视、计算机、电影院装备等)能够计算例如700nit PB\_C图像。这种所谓的显示适配很重要,因为并非每个人都具有精确的5000nit PB\_D的显示器(即,与内容自然匹配的显示器(所谓的参考显示器,其用于与接收侧的任何实际显示器相区别)),因此必须计算适合于例如700nit显示器的图像(这意味着最优地传达5000nit HDR分级中的艺术意图以及创建侧的发光亮度或一般的颜色分级函数)。如果不是这样,则渲染的行为是未定义的,并且通常不会示出或者至少较差地示出内容的较亮部分(这可能导致例如低对比度的人在明亮的薄雾中完全消失,这意味着某些人会看到完全不同的电影故事,这是非常不希望的情况)。

[0011] 我们将非限制性地假定HDR图像分级和SDR图像分级是由人类颜色分级者创建的(例如,首先是HDR分级,然后是SDR分级),然而,在某些应用中,自动系统会确定分级图像和颜色映射函数(例如当前在实时广播中优选是这种情况,在这种情况下,现有系统的重新使用与人类分级并不兼容)。

[0012] 为了方便读者并让他在所涉及的一些方面快速入门,图1示出了许多可能的HDR场景的几个原型说明示例,未来的HDR系统(例如被连接到1000nit PB\_D显示器)可能需要得到正确处理,即,通过为图像中的所有对象/像素渲染适当的发光亮度。例如,ImSCN1是来自西方电影的晴天室外图像(该电影大部分是明亮区,在理想情况下应将其渲染得比100nit显示器更亮一些,以提供比雨天看起来更为阳光明媚的晴天外观,例如平均发光亮度为500nit),而ImSCN2是夜间图像。

[0013] 是什么使这样的图像变得阳光明媚,却使另一幅图像变得暗淡?不一定是相对发光亮度,至少不是在SDR范例中。HDR图像渲染方式与仅在几年前结束的SDR时代一直进行的渲染方式的不同之处在于,SDR具有有限的动态范围(大约PB=100nit,黑色水平近似为0.1nit到1nit),使得大多数对象的固有反射率(其范围将落入90%(对于良好的白色)至1%(对于良好的黑色)之间)只能在SDR中示出。这对于在均匀的技术控制照明下识别对象(具有一定量的反射光亮度,当然还有其色度)将是非常好的,但是对于自然场景中的照明

本身的变化来说并不漂亮,这将影响到观看者。如果显示器允许的话,那么图像编码和处理技术也应如此,在林间小道上的人将能够真正看到阳光穿过树木,也就是说,不是像在SDR显示器上的黄色印象,当人从阴影中走入阳光时将会看到明亮鲜艳的阳光照射的衣服。至少在PB<sub>D</sub>允许的范围内,火灾和爆炸也应具有最优的视觉效果。

[0014] 在SDR中,可以使夜间图像在luma直方图中在一定程度上变得更暗,但又不过分,否则会将图像渲染得太暗且太丑。而且,在100nit的电视或100nit的编码下,也没有任何空间可用于任何过亮的东西。因此,必须示出对象而与其照明无关,并且同时不能忠实地示出场景中可能发生的所有有时出现的强烈对比的照明。实际上,这意味着必须以与沉闷的雨天场景近似相同的显示发光亮度(0-100nit)来渲染高亮度的晴天场景。而且,即使夜间场景也不能渲染得太暗,否则观看者将无法很好地辨别图像的最暗部分,因此,将再次跨越0nit到100nit之间的范围来渲染那些夜间亮度。对此,常规的解决方案是将夜景着色为蓝色,以使观看者理解他不是在看白天的场景。当然,现在在现实生活中,人的视力也会针对可获得的光量而进行调整,但是调整的范围并不是那么大(在现实生活中,大多数人的确能够识别出它正在变暗,或者他们处于更黑暗或相当明亮的环境中)。因此,希望利用能够在其中艺术设计的所有壮观的局部和临时光照效果来渲染图像,以至少在可获得HDR显示器的情况下得到更加逼真的渲染图像。对于在暗室中的光剑来说,适当的发光亮度到底是多少?我们将留给创建(一个或多个)主分级的颜色分级器来决定,并且本申请将重点关注创建和处理这样的图像所需的技术可能性。

[0015] 在图1的左轴上是对5000nit PB<sub>D</sub>显示器而言人希望在5000nit PB主HDR分级中看到的对象发光亮度(即,分级器制作图像,假定家用的典型的高质量HDR电视会有5000nit PB<sub>D</sub>,并且他实际上可能坐在这样的家庭观看室的表示中并且在这样的分级显示器上分级)。如果要传达的不仅是幻觉,还想传达牛仔在明亮的阳光照射环境中的真实感觉,则必须那些像素发光亮度指定并渲染得足够明亮(但是也不会令人讨厌地太亮,这是HDR图像创建和处理的典型困难),例如指定并渲染在500nit左右。对于夜景,人在大多数情况下想要获得暗的发光亮度,但是摩托车的主要特性应该是能易于识别的(即,不要太暗(例如在5nit左右)),并且同时可能存在发光亮度很高的像素(例如,路灯的像素,其发光亮度在5000nit显示器上例如在3000nit左右,或者在任何HDR显示器上在峰值亮度(例如,1000nit)左右)。第三示例ImSCN3示出了现在在HDR显示器上也能够实现的情况:人能够同时渲染非常亮的像素和非常暗的像素。它示出了黑暗的洞穴,其上面有小开口,通过小开口人能够看到外面阳光明媚的景象。对于该场景,人可能想要使像树这样的阳光照射的对象的亮度比想要渲染明亮的阳光风景的印象的场景中的亮度(例如,400nit左右)低一些,这样应该与洞穴内部的本质上的深色特性更加协调。颜色分级者可能想要最优地协调所有对象的发光亮度(已经在PB<sub>HDR</sub>=5000nit主HDR图像中),以使任何事物看起来都不会不适当地过暗或过亮并且对比度很好,例如,在该洞穴中的黑暗中站立的人可能会被编码在0.05nit左右的主HDR分级图像中(假定HDR渲染不仅能够渲染亮光,而且还能渲染黑暗区域)。

[0016] 因此,人发现针对重新分级可能有多种典型行为:例如,有时人希望在所有发光亮度范围内保持发光亮度基本相等,而有时对发光亮度进行缩放(例如随着PB<sub>C</sub>的减小而线性缩放)。或者,人可能想要使一些对象(如太阳)总是映射到PB<sub>C</sub>。

[0017] 必须理解,在编码框架中存在一些限制。例如,在模式1系统中,人不能随心所欲地做任何事情,而是通常必须根据有限的标准化发光亮度映射函数的集合进行映射(因为解码器需要了解在创建侧进行的操作,并且解码器需要满足实际的市场需求,例如不要过于复杂和昂贵)。因此,人必须设计根据系统用户和内容创建者的具体需求正确充分地运行的函数集合,这正是申请人在前几年发明的项目。对于模式2系统,还必须保证函数是可逆的,特别是SDR图像中不会丢失太多信息,因此,通过在试图获得原始创建侧HDR图像的接近的重建的接收器中进行重新分级将永远无法从数学上恢复该信息。

[0018] 针对最优重新分级的函数通常取决于内容,因此它们通常至少在每次拍摄相同、相似外观的图像和相似发光亮度分布场景的图像时都会进行传送。

[0019] 仅为了说明用于阐明对理解很重要的一些部件的一些技术视频编码可能性,我们描述了示例性HDR视频编码系统,申请人已经为HDR图像(特别是HDR视频编码)设计了这种HDR视频编码系统(其中,读者应当理解,本发明的原理除了适用于用于说明的示例性系统以外还适用于其他系统)。

[0020] 该视频编码系统不仅能够处理针对现场的典型单一类型的显示器(例如,被定义为在假定每一个终端观看者都具有1000nit PB\_D显示器的情况下具有PB\_C=1000nit的图像)的仅单个标准化HDR视频的通信(编码)(例如,10位感知量化器用作定义针对编码的EOTF的luma代码),而且还能够同时对针对具有现场的各种其他峰值亮度的各种可能的其他类型的显示器具有最优外观/分级的视频(特别是针对100nit PB\_D SDR显示器的SDR图像)进行传送和处理。

[0021] 也就是说,虽然在这样的HDR视频通信系统中,实际上仅传送一种类型的分级图像作为传输的像素化图像,但是通常在该示例中并不是唯一地传送SDR图像(或替代地,唯一地传送HDR图像),因为它还会在元数据中添加定义HDR图像的像素颜色(特别是来自这些SDR图像的发光亮度)的一个或多个函数,同时它还传送了针对场景的HDR图像外观(实际上并不需要像在双图像传送中那样传送HDR图像,或者至少不需要第二层像素化HDR图像数据)。

[0022] 为此,如图2所示,定义了适当的可逆颜色变换函数 $F_{ct}$ 的集合。为了解释基本概念,图2非限制性地示出了SDR通信类型(即,模式2)的典型系统。这些函数可以由人类颜色分级者定义,以得到与HDR主图像MAST\_HDR相对应的合理外观的SDR图像( $Im\_LDR$ ),同时还确保通过使用反函数 $IF_{ct}$ 能够以足够的准确度将原始主HDR(MAST\_HDR)图像重建为重建的HDR图像( $Im\_RHDR$ )。能够根据传送的正向的HDR到SDR的映射 $F_{ct}$ 函数来确定 $IF_{ct}$ 函数,或者系统甚至可以直接传送(一个或多个) $IF_{ct}$ 函数。

[0023] 颜色变换器202通常应用主HDR图像(MAST\_HDR)像素的相对发光亮度的 $F_{ct}$ 发光亮度映射,我们将假定要对该 $F_{ct}$ 发光亮度映射进行归一化,使得最大发光亮度为1.0。为了以简单的方式理解本发明的概念,为简单起见,可以假定它使用4次幂发光亮度映射函数( $L_{out\_SDR} = power(L_{in\_HDR}; 1/4)$ )来导出100nit PB\_C SDR输出图像 $Im\_LDR$ (即,图1的右侧)的像素的归一化SDR输出发光亮度,即,这样的函数针对场景的主HDR图像给出了对SDR分级的对应图像的合理外观(对于特定场景而言具有合理含义,例如,大部分阴影区不会显得暗淡,即使在SDR图像中,灯和其他发光对象也会因与较暗的图像区域之间仍然具有合理的区域间对比度而会如所期望的那样显现,至少在SDR发光亮度动态范围允许的范围



内等;对于其他图像,其他因素也可能有贡献,但是这样的细节不是必需的,也不会对阐明本发明的技术部件产生限制)。

[0024] 由于接收器必须能够根据接收到的对应SDR图像来重建主HDR图像,或者至少进行接近的重建但具有某些与压缩有关的伪影,因此除了实际的像素化图像之外,颜色映射函数还必须进入视频编码器203。在不进行限制的情况下,我们可以假定视频是用MPEG HEVC视频压缩器压缩的,并且函数例如借助于SEI机制或类似技术被存储在元数据中。

[0025] 因此,在内容创建装置221的动作之后,从图像传送技术的角度来看,视频编码器203假装它得到了正常的SDR图像作为输入,并且更重要的是:在技术上是SDR图像(编码图像Im\_COD)的输出遵循Rec.709标准SDR luma规范。因此,另外的技术(例如,应用所有必要的变换来格式化数据以通过某种传输介质205(例如,编码以存储在BD盘上,或者频率编码以进行线缆传输等)的传输格式化器204)能够在SDR编码范例中仅应用其曾经执行过的所有典型步骤。

[0026] 随后,图像数据例如根据ATSC 3.0或DVB或任何视频信号通信原理在某种传输介质205(例如,卫星或线缆或互联网传输)上传播到一个或多个接收侧。

[0027] 在任何消费者或专业人员侧,可以被并入各种物理装置(例如,机顶盒、电视或计算机)中的接收器去格式化器206通过应用去格式化和通道解码来解除通道编码。然后,视频解码器207应用例如HEVC解码以产生经解码的SDR图像Im\_RLDR以及颜色变换函数元数据F\_ct。然后,颜色变换器208被布置为将SDR图像变换为任何非SDR动态范围(即,PB\_C高于100nit,并且通常至少高6倍)的图像。例如,可以通过应用在编码侧使用的颜色变换F\_ct的逆颜色变换IF\_ct以根据MAST\_HDR制作Im\_LDR来重建5000nit原始主图像Im\_RHDR。或者,可以包括显示适配单元209,显示适配单元209将SDR图像Im\_RLDR变换到不同的动态范围,例如,在显示器210是3000nit PB显示器的情况下im3000 nit得到最优分级,或者1500nit或1000nit PB图像等。我们以非限制性方式假定视频解码器和颜色变换器位于单个视频重新确定装置220中。本领域读者能够理解,人能够类似地设计传送例如PB\_C=10000nit的HDR图像的拓扑结构,并且颜色变换器制作例如针对对应的电视或监视器PB\_C=2500nit的输出HDR图像。

[0028] 图3阐明了在理论颜色空间中的HDR到SDR颜色映射(特别是包括发光亮度映射)的外观。

[0029] 颜色空间属于圆柱体类别,并且以色度平面为基底。二维色度表示颜色的“自身性质”,例如是具有一定的饱和橙色,还是具有不饱和红色(即,粉红色),而与该颜色的发光亮度或亮度无关。作为亮度度量,我们使用发光亮度,它是该空间的第三个垂直维度(实际上使用相对发光亮度,假定其最大停止在1.0处)。由于色度(Chrom)坐标,我们使用饱和度(Sat)(即,距中心的消色差的发光亮度轴的距离)和色相角(H),但是也可以使用CIE(u,v)坐标,或者 $r=R/(R+G+B)$ 和 $g=G/(R+G+B)$ ,其中,R、G和B分别是红色、绿色和蓝色对颜色的线性贡献量(可以简单地说是红色等的光子的量)等。

[0030] 根据HDR颜色(Col\_in)来计算SDR颜色(Col\_out)。读者应当理解,我们在归一化的颜色空间(PB\_C=1.0)中表示了两种像素颜色,因此通常SDR颜色应该更亮而仍可见(能够理解,这可以通过考虑以下技术来实现,其中,相对发光亮度L对应于背光灯的LCD透射百分比,该LCD透射百分比针对HDR显示器比针对SDR显示器要亮得多,因此要在显示器的正面看

到相同的最终颜色,HDR显示器必须透射较少的光,反之亦然)。应当迅速注意,在相同的归一化色域中示出HDR图像发光亮度和SDR图像发光亮度这两者时可能会在某些方面产生误导,因为最亮的HDR颜色刚好是在SDR中不可能实现的颜色,因此美术师或自动机必须对其选择合理的近似结果,但是对于本专利申请来说,这种简单的解释就足够了,它涉及在HDR图像或视频处理链中发生的技术计算,例如根据接收到的主HDR图像来创建较低动态范围图像的方式。

[0031] 颜色变换的表现非常理想(特别是正交变换,这意味着发光亮度变化不会影响色度,反之亦然),因为能够看到相对变亮的确不涉及颜色的色度(Chrom)的变化。

[0032] 这并不是从一开始就如此不重要。最典型的传统变换是通过在三个附加颜色通道上单独应用函数来使图像变亮(例如,幂( $L; 1/3$ )函数将很好地使最暗的颜色变亮,同时将1.0映射到1.0作为输出,从而不会产生溢出),这将引入色调错误,因此:不是仅使像素颜色变亮,而是它们还会在可能的颜色的整个色域上改变其色调,这是因为在添加混合物中R、G和B颜色分量的相对量之间具有非线性改变关系。

[0033] 即使在理论颜色空间中,仍然存在改变色度(特别是其饱和度)的另一可能性。

[0034] 在自然界中,所有颜色的空间将是圆柱体,因为实际上可以用超新星的光来照亮选定的不饱和黄色。

[0035] 然而,显示器具有局限性,并且还有任何实际的颜色编码(这就是我们引入归一化色域的原因:即使 $PB_C=1.0$ 可能对应于nit中的相当高(例如,10000nit)的实际绝对 $PB_C$ ,仍然有这样的限制):通过自然构造,最亮的颜色是白色,而任何色度颜色(例如,亮黄色)必须具有较小的发光亮度。

[0036] 因此,即使在这种理论颜色空间中,也可能发生以下情况:例如在通常比较暗的蓝色(例如,白色的发光亮度的10%)附近,简单的变亮映射会映射到可实现的颜色(GAMU)的色域之上,这实际上意味着某种颜色分量(在该示例中为蓝色)必须被修剪到其最大值,从而在上色域边界上产生通常过于去饱和并且常常具有错误色调(例如,浅蓝色)的颜色。

[0037] 然而,申请人确实设法构建了表现符合所有要求的简单系统:特别是没有色域外问题,从而使得针对所有颜色都具有正确的色度,但代价是其发光亮度可能比理想情况降低了一些。在W0 2014056679中描述的这种HDR编码技术的原理(即,对其 $F_{ct}$ 发光亮度映射元数据函数的确定、传送和使用)被如下地简短迭代(以用于稍后理解与本教导的区别)。

[0038] 如图5所示,通过将三个颜色分量与确定的增益因子 $g$ 均等地相乘来实际应用变亮(或变暗)的发光亮度变换(请记住,发光亮度是线性颜色分量R、G和B的线性加法组合),即, $L=a*R+b*G+c*B$ ;系数 $a$ 、 $b$ 和 $c$ 取决于视频定义中使用的RGB原色的色度,例如,针对Rec.709,它们大致为 $a=0.3$ ;  $b=0.6$ ;  $c=0.1$ ),这确保了它们的相对比例不会改变,即,在发光亮度处理之前和之后保持颜色的色调和饱和度相同。

[0039] 该教导进一步规定,基于三个RGB分量中的哪个分量最大来确定针对每种可能的颜色的增益因子,这保证了映射可以利用局部 $L_{max}(\text{chrom})$ 进行正确缩放,即,永远不会出现色域问题。因此,这样的颜色数学实施方式实现了理想的线性和分离的颜色/亮度处理行为,同时还考虑了色域(特别是显示器的色域)的局限性。

[0040] 然而,出于各种原因,人想要也能够其他颜色表示(即,其他颜色空间)中工作的颜色处理。

[0041] 由于模拟视频(PAL、NTSC)的时代已经到来,因此定义了YCbCr(或对应的YIQ)颜色空间,这对于视频通信非常有用,但是人们也想要在其中进行颜色处理(例如,利用模拟电视电路通过调节Y通道来改变亮度,或者通过类似地改变Cb通道和Cr通道来改变饱和度)。

[0042] 图4示出了它的外观及其属性。首先,RGB(处于0-1.0中)立方体在其顶部旋转,使得现在 $R=G=B$ 的对角线轴变为消色差轴(即,无色亮度轴)。

[0043] 然而,现在新的亮度度量是luma。

[0044] 它是利用与发光亮度相同的a、b、c系数来定义的,但是现在是根据非线性(伽玛预先校正)RGB系数来定义的,根据Rec.709,非线性(伽玛预先校正)RGB系数大致为 $R = \sqrt[R\_linear]{}$ 等(请注意,在HDR时代,已经通过 $R\_nonlin = OETF\_HDRcodec(R\_lin)$ 等定义了另外的非线性RGB分量)。

[0045] 人会倾向于认为 $Luma;2$ 等于发光亮度,而在消色差轴上确实如此(因为 $R=G=B$ ,所以 $power((a+b+c=1)*\sqrt[R]{R};2) = power(Y;2) = L$ )。然而,对于色度颜色来说,luma不仅仅与发光亮度一维相关,而且还与色度相关,这被称为非恒定发光亮度泄漏。

[0046] 这会产生一些后果:如果人想要通过应用函数(例如, $Y\_out = power(Y\_in;1/3)$ )使图像颜色变亮,那么输入Luma  $Y\_in$ 取决于色度,因此归因于(选择简单的而易于理解的) $1/3$ 幂函数的Y依赖作用,变亮的强度也取决于色度。注意,在我们的上述现有技术系统中的任何这样的luma映射函数(无论是简单的示例性幂函数,还是具有若干仔细选择的局部较高或较低斜率(其对应于场景中的特定感兴趣对象的发光亮度)的更为复杂和感兴趣的函数)都对应于取决于luma的增益 $g(Y)$ ,将这种取决于luma的增益 $g(Y)$ 乘以(在相应的可能的实施例中对线性或非线性的)RGB分量(因为人在线性RGB的幂函数重新定义(例如,YCbCr)中也能够实施分量乘法)。因此,如果针对颜色(其具有在理论上与需要变亮的某种消色差灰色相同的发光亮度)估计出“错误的”发光亮度作为不同的luma,则这样的颜色可能会得到“错误的”增强,从而导致饱和颜色(如红色)的荧光外观。

[0047] 与图3的理论颜色空间的第三个主要区别在于颜色分量CbCr不是色度,这能够通过以下事实看出:(除了上半部分中的顶部的色域限制之外)该颜色空间是圆锥形的而不是圆柱形的。

[0048] 饱和度不会随着Cb的增加而变化,而是会随着与消色差luma轴的夹角而变化。这能够通过以下操作来理解:采用例如在这种取决于显示器的颜色系统中可重现的最大饱和度的蓝色原色(ColBlu),并且朝向最明亮的可能的蓝色(Blu)增大其发光亮度。这是相同种类的饱和蓝色,其色度及其饱和度不会改变,但是其Cb值会随着luma(或者对应地随着发光亮度)而增加。这种取决于发光亮度的性质就是将Cb分量和Cr分量称为色度的原因,第一部分指示将颜色方面(颜色中的蓝色)进行整理的事实,而第二部分指示其随非线性luma发生的线性变化。能够通过以下公式将色度与恒定的饱和度乘法器( $S_m$ )均等地相乘来改变这样的YCbCr表示形式的饱和度:

[0049]  $Cb\_out = S_m * Cb\_in; Cr\_out = S_m * Cr\_in$

[0050] 然后,luma不会改变,因为它不受该处理的影响,但是颜色的发光亮度会由于luma的非恒定发光亮度属性而改变(在很多情况下,颜色处理系统的用户不会觉得有问题,但是在某些更关键的情况下颜色处理系统的用户可能会感觉到有问题)。然而,仅改变luma部分不会引起恒定的颜色色度行为,因为CbCr分量对应于“错误的”发光亮度,因此对应于不同

饱和度的颜色。但是,通过类似的luma差(或基于发光亮度的处理中的发光亮度,例如,线性YCbCr空间,或被并入在如下所述的非线性拓扑结构中的线性版本等)来反向操控CbCr系数,能够相对容易地进行校正。

[0051] 色域的顶部的饱和度问题因其朝向最大白色变窄而在任何系统中仍然存在(并且可能导致色调错误),而这正是当前的应用的新的实施例和教导所要解决的问题。

[0052] 图5阐明了最后一件事,它是可用于全面理解以下各种教导的有趣的背景信息。

[0053] 图5重新总结了申请人共同开发的Technicolor-Philips(又名SLHDR1和SLHDR2)基本HDR解码器硬件(编码器是镜像处理)。它能够类似地以两种方式进行驱动:模式1,在这种情况下,输入基于PQ的YCbCr,即,它是基于利用作为SMPTE 2084感知量化器函数的OETF根据线性R'G'B'坐标计算得出的非线性R'G'B'坐标来计算得出的;以及我们在这里阐明的模式2版本,在这种情况下输入正常的(即Rec.709可解码的或近似于基于sqrt-OETF的)SDR YCbCr图像颜色。

[0054] 注意,类似于该示例,我们的实施例既能够在线性(即,基于线性RGB的)YCbCr颜色空间计算中实施,也能够基于幂律的非线性YCbCr颜色空间计算中实施,例如,我们在这里描述的基于 $R' = \sqrt{R}$ 的示例。

[0055] 这是因为能够使线性空间中的计算等效于基于幂的计算,特别是基于sqrt-power的计算,例如: $k * \text{power}(a * \sqrt{R} + b * \sqrt{G}; 2) = \text{power}(a * \sqrt{k * R} + b * \sqrt{k * G}; 2)$ 。

[0056] 因此,这意味着即使我们将通用增益因子 $g_{NL}$ 应用于非线性归一化RGB分量 $R'G'B'_{nrm}$ 以获得正确变暗的HDR像素颜色,我们实际上仍然能够进行基于线性、完全独立的发光亮度的处理(即,避免由于基于非恒定发光亮度的亮度处理导致的荧光颜色亮度错误)。

[0057] 这能够通过以下事实看到:发光亮度处理器501进行的整个SDR到HDR颜色映射的发光亮度处理部分(图3中的示例颜色的向上移动)作为针对当前处理的像素的发光亮度L的输入。

[0058] 通过将颜色空间转换器502布置为将非恒定发光亮度luma<sub>Y</sub>转换为发光亮度(这能够通过各种方式来完成),颜色空间转换器502能够处理该问题(例如,通过应用适当的逆OETF(即,所谓的EOTF)将非线性R'G'B'值转换为线性R'G'B'值,然后如上所述通过基于RGB三元组值的适当线性定义方程来计算发光亮度L)。

[0059] 通过将归一化输出发光亮度 $L_{out}$ 除以归一化输入发光亮度L,可以获得针对三个颜色分量所需的线性乘法因子 $gL$ 。

[0060] 但是非常有趣的是,虽然可以对发光亮度进行线性处理,但是实际上能够在非线性版本的发光亮度中指定(并进行)处理(由于分离出仅发光亮度信息通道,因此能够等效地表示)。

[0061] 在该示例中,我们示出了在感知均匀域中的处理,这对于人类分级者(或自动机)指定其动态范围重新分级的需求更为自然。为此,域转换器511将线性发光亮度转换成在感知上更均匀的感知luma<sub>SPY</sub>(例如通过应用PQ OETF或飞利浦在WO 2015007505中获得专利的函数来完成该任务,这些教导内容作为本申请的潜在实施例教导内容而并入本文)。发光亮度映射器512执行SDR感知luma<sub>SPY</sub>到HDR输出感知luma<sub>SPY\_0</sub>的映射。可以假定继续进行以上的简化阐述,即,通过应用简单的三次幂律 $PY_0 = \text{power}(PY; 3)$ 可以获得HDR图像的正确感知luma<sub>SPY</sub>,但是一般而言,内容创建者能够定义任何复杂的最为优化的曲线形状,该

最为优化的曲线形状可以从协同传送的元数据SEI (F<sub>ct</sub>) 中被读取,并且例如被加载到LUT或类似文件中。能够看到(与利用图3解释的内容相反,图3是关于如何为动态范围较小的图像(特别是与输入图像相比其编码峰值亮度PB<sub>C</sub>较小的图像)创建对应的对象像素发光亮度(在这里实际上被计算为其对应的亮度))需要使大多数相对发光亮度变暗以获得较高动态范围(例如PB<sub>C</sub>=1000nit)的正确的对应的外观图像。

[0062] 此后,第二域转换器513将感知lumas重新线性化,即,获得正确的HDR发光亮度L<sub>out</sub>。

[0063] 因此,这种拓扑结构是可以根据需要以各种方式实现保持色度保留的纯发光亮度变化的强大方法(至少是:只要公共乘法因子不会将R'G'B'值中的任何值提升到其最大值以上即可,这不限于我们将假定为1.0,这是如上所述的色域修剪的情况)。电路的其余部分是对理解本发明的以下方面不是很关键的实施例。

[0064] 能够(通过色度处理导出器520)从元数据中加载另一颜色LUT B(Y),从而为每个像素luma<sub>Y</sub>给出B因子以(通过乘法器521)乘以接收到的CbCr值,以获得更合适的输出值CbCr<sub>COR</sub>。这能够是简单的策略,也能够是更复杂的策略(其还考虑颜色的V值,其中,在线性场景中, $V = \max(R'G'B')$ 或 $V = \max(R, G, B)$ )。最后,矩阵会导出归一化的R'G'B'值R'G'B'<sub>nrm</sub>。这些值在归一化的伽马中,如图3和图4所示,但是平方根分布的,因为它们仍然处于输入图像颜色的非线性域(Y<sub>SDR</sub>,Cb,Cr)中。实际上,这些值是位于其SDR位置处的颜色(参见图3:Col<sub>out</sub>(SDR),因为在该示例中,SDR颜色是输入颜色,而不是输出颜色)。它们将需要通过针对每个像素适当映射对应的gNL因子,使得乘法器530能够将这三个分量类似地乘以该因子以获得非线性归一化HDR颜色R'G'B'<sub>HDR</sub>。

[0065] 因此,乘法器530执行: $R'_{HDR} = gNL * R'_{nrm}$ ;  $G'_{HDR} = gNL * G'_{nrm}$ ;  $B'_{HDR} = gNL * B'_{nrm}$  [公式1]

[0066] 也能够替代地在其他基于幂函数的导数的颜色表示上实施相同的发光亮度改变,例如, $Y'_{HDR} = gNL * Y'_{nrm}$ ;  $Cb_{HDR} = gNL * Cb_{nrm}$ ;  $Cr_{HDR} = gNL * Cr_{nrm}$ ,其中,Cb和Cr是根据基于R'G'B'的常规公式定义的,即,在该示例中这将是SDR色度,但被归一化为1。注意,在本文中,素数符号指示我们具有基于1/2的幂的量(即,非线性lumas的经典Rec.709定义),并且没有素数通常指示正常的线性颜色分量,而双素数指示颜色分量的另一种可能的HDR定义,其中,R''通常指示 $R'' = OETF\_PQ(R)$ 。有经验的读者能够理解在哪个定义的系统定义了相应的Cb和Cr。

[0067] 最后,显示调节器540能够调节颜色以用于一般参考(例如在存储的情况下)或特定的连接的显示器。在前一个示例中,处理可以很简单以将伽马1/(2.0)定义的R'G'B'值转换为PQ OETF定义的lumas R''G''B''<sub>PQ</sub>。在后一个示例中,显示调节器540可以应用任何特定的显示适配算法来获得例如800nit PB<sub>D</sub>优化的R''G''B''值而不是获得例如5000nit PB<sub>C</sub>定义的R'G'B'<sub>HDR</sub>(不应误解为在其他实施例中这种显示适配可能已经在处理过程本身中(特别是在发光亮度处理器501中)发生。实际上,显示调节器能够产生所连接的显示器所需的任何信号格式,并且可以进行传送以从显示器接收有关该信号格式的信息,然后再形成所需的输出颜色和将它们进行整理的信号(例如通过HDMI线缆或无线方式等来完成该任务)。

[0068] 读者能够看到,这种拓扑结构具有多种迄今为止互斥的颜色处理愿景的优点:它

具有YCbCr结构,特别是它接受正常的YCbCr输入颜色,它还示出了线性颜色处理的所有出色属性,例如正常颜色没有色调移位(即,在HDR图像中,大多数颜色都位于色域顶部以下,这是因为这些颜色通常会整理超明亮颜色(例如,灯的颜色),这些超明亮颜色通常是无色的白色,或者当他们是去饱和状态时至少不令人讨厌)。然而,色域以上的颜色映射问题仍然存在,因为任何圆锥形顶部的颜色空间(无论是圆锥形基底还是圆柱形基底)都存在该问题。它似乎不像单独的通道非线性RGB映射那样在颜色立方体处理中发生,至少在这样的表示中很容易避免溢出,但是正如在这样的表示中所说的,因颜色分量比例改变而会导致颜色错误返回。

[0069] 增益计算单元514能够将任何发光亮度映射函数(例如,由人类分级者期望和创建的函数形状)或算法转换成单个乘法增益因子的方式如下所示:假定输入发光亮度例如为0.2并且单元501中的所有函数的应用(其通常能够在两个端点框之间进行域转换,包括任何数量的处理,从而得到最终结果)产生0.4的输出发光亮度。然后,增益计算单元514能够是将(线性域)gL因子产生为 $L_{out}/L$ 的简单除法器。

[0070] 输入发光亮度L将对应于输入颜色分量,例如,线性R,G,B。在示例中将gL因子乘以2.0即可产生正确的输出颜色分量,该输出颜色分量也具有正确的期望的输出发光亮度。如果计算(即,IC拓扑结果或软件)不在线性域中发生的,则会需要增益因子域转换器515,在该示例中,处理发生在经典伽玛2.0SDR域中,因此应当应用平方根函数以获得最终所需的非线性增益因子 $gL = \sqrt{gL}$ 。在构想亮度映射以发光亮度作为输入而发生的情况下,能够类似地体现以luma作为输入的对应公式,为此通常将使用经典的伽玛2.0luma。(注意,事先没有教导单元520的颜色处理的V依赖性,但是它与本发明结合时是有用的)。颜色矩阵器523为了根据CbCr\_COR导出归一化R'G'B'\_nrm坐标所应用的颜色数学运算对于颜色技术领域的人员来说应该不难理解,并且它也不是很重要,因为我们目前的教导适用于各种颜色计算实施例,即,无论是否存在该部分,只要能够导出三个能够通过通用g因子进行发光亮度调节的颜色分量即可,以表示部分颜色处理已经充分存在,即,虽然我们示出了完整的HDR解码器颜色处理拓扑结构,但是对于当前的教导,它是感兴趣的501的上部。

[0071] WO 2017/157977还教导了本申请人对HDR视频发光亮度重新分级的优选方法的各个方面,特别是关于如何处置超深黑色的一些细节。但与本讨论内容的共同点是,它教导了如何能够有利地定义发光亮度映射函数,然后能够将其用作像素的三个颜色分量(线性或非线性)的对应的g因子的公共乘法器,例如通常为YCbCr。该专利申请没有教导的是用于任何随之而来的超出色域的行为的简明解决方案,如果在实践中发现这是令人反感的(请注意,HDR图像可能常常包含相对较暗的颜色,这在被过度增强时可能不会在SDR中产生问题,并且常常出现一些消色差的高光,如果将其修剪就不会造成心理视觉问题)。专家通常会做的是选择稍微不太“激进”(即,从HDR到SDR的发光亮度提升较少)的曲线,例如为WO'977的图12的动态范围调节器(DRA)所应用的发光亮度映射函数的最亮图像发光亮度选择更合适的斜率。虽然通常会带来一些其他心理视觉图像效果(例如,所述上部线性段斜率的变化通常也会因连续性的需求而对下面函数的形状产生影响),但是通常会充分缓解任何引起人充分关注的问题。

[0072] 因此,发明人设想到提供一种新的动态范围变化的颜色处理的技术解决方案,同时保留一些现有颜色空间计算的一些良好属性,特别是能够与如以下(和以上)所描述的若

干实用的动态范围变化的硬件或软件拓扑结构或类似的拓扑结构相匹配的解决方案。

## 发明内容

[0073] 如上所述, (显示器的 (但通常不是排他性的)) 色域顶部附近的颜色映射问题是令人讨厌的问题, 该问题很容易引入例如色调错误 (例如, 颜色变得更绿色), 这就是为什么本申请教导了这样的颜色处理的新方法的原因, 该方法适合用于高动态范围图像处理的新出现领域以及不同动态范围 (可能为了简单起见, 假定其主要由峰值亮度PB\_C确定) 的图像的对应衍生领域, 该方法特别包括一种发光亮度处理器 (501), 所述发光亮度处理器被布置为根据具有以第一峰值亮度 (PB\_HDR) 为特征的第一发光亮度动态范围的输入图像 (MAST\_HDR) 的空间共位的像素的输入发光亮度 (L) 来计算具有以第二峰值亮度 (PB\_LDR; PB\_MDR) 为特征的第二发光亮度动态范围的输出图像 (Im\_LDR; Im3000nit) 的像素的输出发光亮度, 其特征在于, 所述发光亮度处理器包括:

[0074] 增益计算单元 (514), 其被布置为基于指示如何调节所述输入发光亮度以变为对应的中间输出发光亮度的发光亮度映射函数 (FLM), 通过将乘法因子计算为所述发光亮度映射函数针对所述输入发光亮度的输出除以所述输入发光亮度的除法  $gL = FLM(L) / L$  来计算所述乘法因子 (gL), 所述乘法因子针对所述输入图像的图像像素的任何输入颜色的所述输入发光亮度被定义;

[0075] 溢出计算器 (602), 其被布置为计算指示所述中间输出发光亮度在所述输入颜色的色度处的上色域边界上方多远处的溢出度量 (T), 所述中间输出发光亮度是通过将所述乘法因子 (gL) 应用于所述输入颜色而得到的;

[0076] 增益因子修改单元 (603), 其被布置为在所述溢出度量大于阈值 (G) 的情况下确定较低的替代增益因子 (F1 (gL)), 并且被布置为在其他情况下保持原始增益因子, 并且被布置为输出所述替代增益因子和所述原始增益因子中的一个作为最终增益因子 (gF); 以及

[0077] 乘法器 (530), 其用于将所述输入颜色 (R'G'B'\_nrm) 乘以所述最终增益因子 (gF) 以获得具有所述输出发光亮度的输出颜色 (R'G'B'\_HDR)。

[0078] 输出图像通常会具有较小的动态范围 (即, 较低的PB\_C值), 但也不都是这种情况, 因为在某些场景中, 色域问题的顶部 (因此其解决方案) 也存在于动态范围升级中, 并且无论如何都能够将处理 (例如, 相同的IC) 应用于这两种场景, 因为作为一种保护策略, 它不会在不需要时启用 (由内容创建侧导出的颜色映射函数F\_ct或在一般情况下例如由接收者本身基于图像分析以任何方式导出的颜色映射函数指定仅需要处理的无问题颜色)。

[0079] 增益计算单元确定乘法增益因子 (例如, gL或与其有关的因子), 因为该机制通常是通过类似于按比例缩放的发光亮度的三个颜色分量 (其自然是用于加色重现系统 (如电视) 的线性的红色分量、绿色分量和蓝色分量; 或者其幂函数 (如Y'CbCr), 其能够基于线性RGB分量的平方根来计算, 这是对传统Rec. 709视频OETF的非常好的近似) 来工作的。

[0080] 能够根据发明人的洞察通过以下操作来良好的判断这种解决方案的潜在危险: 开始于计算 (在数值上略有不同但根据该方法的原理的行为相似的) 输入颜色 (或max (R', G', B')) 的  $V = \max(R, G, B)$ , 然后基于此定义溢出度量 (因为, 一方面, 由于例如HDR到SDR图像计算, 颜色更可能映射到色域边界上方, 如果颜色已经很明亮, 而且如果它们是特定类型的颜色 (像总是具有低发光亮度的蓝色), 则基于输入发光亮度的任何发光亮度映射, 像素颜色



针对这样的蓝色会有风险)。虽然能够定义各种溢出度量来(在正确的位置和以所需的程度)应用本发明的相同技术原理,但是其中的一些不需要使用V(例如经由包含针对每个色度的较高色域发光亮度的表并计算距离公式),特别是通过将gL与V值相乘来将gL应用于V值是体现溢出度量的简明方式。读者不应误解这种机制,因为它总是基于该强度值V来应用发光亮度映射(即, $V_{out}=F_{Lt}(V_{in})$ ),就像在W0 2014056679中讲授的那样。根据重新分级规定(其对V作用而不是对被连续不断处理的像素颜色的发光亮度L作用)对例如RGB分量计算乘法增益因子始终是在色域内部自动映射的,因此不存在本解决方案所要解决的问题,但是这是产生较暗的SDR图像(可能比优选图像更暗)的缺点。当然,在本系统的实施例中,还能够制作保守的更暗的基于发光亮度的映射函数 $F_{ct}$ (/FLM的反函数),但是实施例还可能制作更亮的函数并且以其他方式解决上色域区的发光亮度映射问题。最基本的变体可能还想要将某些最亮的颜色修剪到上色域边界上的颜色,但是通常可以保证其色度(或至少为色调)与输入颜色的相同。这仅创建了差分变亮,仅适用于图像的一些颜色以及一些图像(特别是在进入与输入图像的动态范围相差不大但通常较低的动态范围时),因此可能只有少数像素被如此修剪。实际上,可以将图6右下角的虚线矩形中的内容视为某种保护机制。在根据最初的g因子产生经校正的g因子的意义上说,它实际上是否是保护机制取决于实施例类型。在第一类实施例中,内容创建侧的自动机或人类分级者轻松地确定了一些HDR到SDR的发光亮度映射函数(对于大多数图像颜色,它的行为都适当,例如,它对当前场景图像的较暗颜色给出了很好的正确明亮外观,并且对消色差轴附近的去饱和色也很适用),并且依靠该机制将充分解决针对(少数)有问题的颜色的问题,例如一些明亮的带红色和带橙色的日落天空颜色。这具有以下优点:这样的机制能够“快速且肮脏地”确定该函数,而不是需要规定要考虑应该确切发生的针对少数几个关键日落天空颜色的逐颜色映射确切的情况的细节,并且可以例如用于实时内容编码。这样的机制仍然可被分成另外两个子类别:其中的一个子类别,编码器为内容本身确定合理的值(或者甚至具有前缀值,在减少对应于不执行任何操作的修剪错误严重性的意义上,这通常在大多数关键颜色场景中都可以正常工作);以及第二个子类别,其中,内容创建侧本身确定最优策略。因此,简单的实施例能够使用非常简单的变体,例如具有固定的G阈值(例如,0.75)作为替代方案,较低的增益分配将发光亮度线性地分布在G(通常低于1)与例如1.5(例如,针对该色度或者甚至针对所有色度的最大中间输出颜色,为[G,1])之间,或者具有可变的、创建者可优化的G阈值。并且,例如利用图11,我们阐明第二次校正如何传递给他最初指定的(一个或多个)重新分级函数,这大致上是好的,即,对于大多数图像像素和图像外观而言,例如人类分级者能够指定几个其他参数来定义色域的上部区中的映射(所需的目标通常是将重新分级单独留在输出色域的下部部分中,即,由内容创建者将其指定在FLM函数中)。实际上,G阈值可以用作对输出色域的上部区域中最适合对原始重新分级进行校正的点的可优化的或至少合理的确定。

[0081] 但是,应当看到,作为针对“校正”实施例的替代方案,至少像rho等的参数确定了对初始确定的g因子(即,例如基于函数FLM和当前颜色的发光亮度的gL)的重新计算,至少针对有问题的颜色(显著)映射在色域顶部边界之上,编码器还能够将所有颜色变换成最终的颜色处理函数(或甚至与之相对应的g因子)的集合以应用于接收侧。特别是在内容创建侧协作传送函数以将颜色(基于正常的未校正行为“有区别地”)映射在上色域区中的情况下能够这样做。在这种情况下,增益因子修改单元会根据内容创建者在其函数FADAP中指定



的标准对是否应用“标准的”粗略发光亮度映射或更高级的发光亮度映射进行测试(并且读者应当理解,接收侧(例如,电视)甚至能够在该规定上做一些细微的变化,但是为了简化教导,我们将假定它只是遵循如在各种上色域区发光亮度或一般颜色映射参数中传送的创建侧的重新分级智慧。)

[0082] 溢出计算器(602)计算溢出度量 $T$ ,该溢出度量 $T$ 指示情况的关键程度,即,接近色域边界的程度,因此对溢出所映射的颜色(即,在将发光亮度映射应用为其 $g$ 因子之后)特别感兴趣:因此,所映射的颜色在色域顶部上方有多远,需要进行多大程度的校正(因为如上所述,校正可能仅包括变暗,但这对于所有不同类型的HDR图像可能不是最佳选择,特别是如果需要大量变暗,则如下所示,实施例还能够决定以不同的更高级的方式进行校正,例如,除了确定一些较少量的变暗之外,还通过确定最优的颜色去饱和量来完成校正)。特别地,低于实际溢出的溢出度量(例如,90%)在需要还通过对本身并没有真正溢出的一些相邻颜色进行重新分级为溢出的颜色留出空间的实施例中是有用的,并且这在需要可逆颜色处理(如模式2编码,其将HDR图像实际编码为对应的重新分级的SDR图像)的情况下是特别感兴趣的(因此,两幅图像都必须包含足够的细节,才能通过本身也足够好的函数被映射到足够质量的HDR图像)。

[0083] 有利的是,所述发光亮度处理器(501)包括数据接收单元(689),所述数据接收单元被布置为通过网络(690)从内容的创建者接收所述阈值( $G$ )。发光亮度处理器的其他实施例可以自己确定阈值,例如,在HDR图像接收器中例如通过分析该图像的属性(例如通过查看存在哪种类型的HDR效果,例如有多少像素,其中的结构类型(例如具有纹理表征物等))来确定阈值。更简单的版本甚至可以利用固定的或至少标准的初始值(例如 $G=0.9$ )来工作。然而,如果内容的创建侧能够为整个电影、同一场景的 $N$ 个相继图像的镜头或甚至是每个单个瞬间图像指定 $G$ 的最优值,那将是非常有利的。然后,不仅能够在创建侧进行更复杂的图像分析,而且人类分级者还能够指定他自己的最佳工作 $G$ 值,例如利用UI,如图11所示。基于单个 $G$ 值,能够设计 $g$ 因子重新确定的策略,例如具有线性函数段,如图10所示。

[0084] 有利的是,所述发光亮度处理器(501)包括数据接收单元(689),所述数据接收单元被布置为通过网络(690)接收函数(FADAP)以根据所述内容的所述创建者确定所述替代增益因子。以这种方式,可以在内容创建侧创建相当复杂的 $g$ 因子重新确定函数,该 $g$ 因子重新确定函数考虑了内容的特定需求,这在上色域区中存在非常特定的非常关键的内容的情况下会很有用(例如,在蓝色商业TL管上书写的柔和对比度的图案,然后可以对重新确定函数进行整形,使得其至少在那些文字字符的发光亮度附近保持较高的发光亮度对比度斜率)。人类分级器(或甚至是自动机)能够例如在校正映射到最终输出颜色之后识别图像的最亮部分中需要足够对比度的关键区域,并且调谐FADAP函数的形状,这相当于对最终 $g$ 因子将如何分布上述意图的色域进行调谐,但是无法实现的颜色会在上色域区域上分布。当云的部分开始显得太淡时,能够校正这样的FADAP函数的形状(或者能够将某些校正移至饱和度处理)。

[0085] 我们在这里教导了允许确定、传送和最终应用这样的FADAP函数的框架方面,因为本领域技术人员从教导中能够想象到,关于如何为每个特定图像(相继的图像的集合)或图像类别等设计特定的FADAP函数形状能够有很多详细方面(例如,在一些情况下,内容创建者可能会重点强调最亮发光亮度的一些子集的发光亮度对比度,这会以FADAP函数的形状

反映出来,从而确定输入颜色在输出色域的顶部区域上分配,其中,一些特定的场景对象或图像区域颜色会占用更多可用色域量,但会牺牲其他一些颜色的准确性)。利用阳光灿烂的黄色夜空的简单的说明性示例,读者能够想象到,如果需要减少一些局部对比度(根据其灰度值来定义云的形状),可能低于可见光或者至少给出降低的视觉印象,改变第一灰度值集合(为简单起见,假定将其映射为单个值)对云的形状的影响能够与强加另一发光亮度子集的影响不同(例如,因为第一发光亮度发生在一些重要的云的明亮点亮的边界周围,而第二发光亮度确定了一些次要云的内部外观)。FADAP的实施例是描述了正常发光亮度函数的至少较高部分(其类似于FLM能够通过除法被转换为g因子),还是在g因子上进行定义( $g_{out} = FADAP(g_{in})$ ),在细节上对于允许在当前的新框架的属性内设置好的替代方案的原则无关紧要。

[0086] 即使传送了函数,对于若干实施例(例如,那些允许在接收器处改变规定的重新确定机制的实施例;或函数的部分表征),对于仍然传送G因子仍然是有用的。

[0087] 有利的是,所述发光亮度处理器(501)包括数据接收单元(689),所述数据接收单元被布置为通过网络(690)从内容的创建者接收修剪参数( $V_{mx}$ ),所述修剪参数指示允许高于哪个色调保留的颜色修剪,所述修剪参数被定义为红色分量、绿色分量和蓝色分量中的最大的一个。在这种情况下,能够允许一些颜色最终在输出图像中变为相同的颜色,这对于不可逆的场景可能是很好的最优选择,即,无需从已经变为单一颜色的颜色中重新确定原始颜色。这能够与经修剪的颜色的子集下方的其他策略(例如,利用去饱和平衡的发光亮度调光)一起工作。在理想情况下(对于一些应用来说是当然的情况),根本不用修剪,但是有时它是最优图像重新分级的良好分量,因为它能够允许例如针对一些其他图像颜色的较小地变暗或去饱和度,这在存在一些关键区域(例如在云中,这就需要高于平均水平的注意力来观察上输出色域区域的大子体积)时特别有用。

[0088] 取决于配置参数的方式,图像上可能会发生许多原型动作,其严重程度取决于HDR场景的类型,因此人想要在选项之间进行最优选择。

[0089] 在最简单的情况下,能够将G设置为等于色域顶部值( $G=1.0$ ),这仅意味着以保持色调和饱和度的方式对所有碰巧映射到色域顶部以上的颜色进行硬修剪。当然,这可能意味着在输入图像(通常是HDR图像)中最初不同的几种颜色被映射到(通常)动态范围较低的输出图像中的相同颜色。如果唯一违反色域内条件的对象是例如有颜色的TL管,则上述情况是可接受的,因为无论如何跟随电影或视频故事的发光亮度差异在语义上都是没有意义的(特别是在这只是背景中的某个装饰灯的情况下,而动作发生在其他地方)。知道人无论如何也无法在SDR中制作出完美的饱和红色且非常明亮的TL管(这就是HDR图像具有优点的原因),对于这样的场景,可能最好的做法是仅将SDR中的整个TL管表示为最大亮度原红色,而不是例如显著降低饱和度,从而仍在光管区域内部保留一些发光亮度差异(这会使管变成丑陋的粉红色)。对于具有许多红灰色云的结构的美丽红色日落,最优决策可能会大不相同。在这种情况下,并不想进行过多的硬修剪而可能破坏过多的美丽的云结构。能够通过使所有云像素颜色更均匀地变暗来解决该问题,但这可能导致G值较低,或者(也会)引入一些去饱和度。然而,即使当“保护”较低动态范围图像中的大多数云结构时, $V_{mx}$ 参数也允许指定还存在一些修剪。在使用可逆图像处理方法(如基于模式2的通信系统,其中需要根据从任何接收侧接收到的SDR图像来重建HDR图像)的情况下, $V_{mx}$ 值通常应当使得仅几个像素被

修剪而失去分化的可能性(例如,能够在SDR图像中为灯的内部赋予所有相同的值1023,并且这样仍然允许进行HDR重建,虽然在数学上与创建侧的主HDR图像并不完全相同,但是在视觉上给出了足够接近的近似结果)。

[0090] 非常有用的是,根据前述权利要求中的任一项所述的发光亮度处理器(501)包括部分映射计算器(903),所述部分映射计算器被布置为确定用于与(gL)的乘法的第一校正分数,并且被布置为确定第二校正分数,所述第二校正分数用于确定用于驱动颜色饱和度计算的饱和度乘法器(S)。根据我们教导的新颖方法,技术人员能够理解,从理想状态的上述色域原始重新分级的颜色朝向色域边界必须保持一定距离。参见图8,本领域技术人员能够学习并理解,能够定义这样的距离的分数(例如其一半),并且能够设计替代g因子,其映射到这样的距离的一半,即,不进行所需的完全变暗以向下达到针对精确输入色度的上色域边界((色调,饱和度)或(u,v))。因此,仍然存在某种距离(或更确切地说是溢出)区,该区将以某种方式被覆盖,否则,色度保留修剪可能会作为默认回退而起作用,但是对于该图像中受此影响的对象可能无法给出最佳的视觉效果。能够在正交方向上覆盖朝向上色域边界颜色的剩余距离,即通过进行对应的去饱和处理来完成上述操作。

[0091] 发明人发现,所需的衰减A(即,针对 $gL \cdot V$ 的对应乘法因子,以使其充分降低到色域内部或在其边界上)能够非常用地包括第一部分 $A_{rho}$ 和其余部分 $A_{rem}$ ,用参数rho定义如下: $A_{rho} = power(A; rho)$ ;  $A_{rem} = power(A; 1 - rho)$  [公式2]。

[0092] 能够基于某种最坏情况下的颜色来确定这样的规定,然后对其他颜色进行类似的处置。

[0093] 这能够大大缓解发光亮度下映射的复杂性,即,函数重新确定最终g因子gF,以在较低动态范围图像的上范围中得到许多关键的HDR效果颜色,特别是在需要函数反转(如模式2HDR编码)的情况,因为这样就能够解决剩下的问题,即,中间颜色仍然利用去饱和操作而不是进一步变暗或者在使用另一重新确定函数FADAP形状时的更极端的对比度降低而仍在色域边界之上。这再次允许更大的自由度来选择每种特殊的HDR效果,例如日落时太阳附近的明亮点亮的云,或者在明亮的薄雾中略微可见的人(影子人),或者从窗户看到的阳光照射的外界,或者在迪斯科舞厅或杂货市场中的灯等,而不管是以下哪种情况都是如此:更好地减少有问题的近色域顶部颜色中的最暗的亮度,或者更大地减少那些区中的纹理的对比度,或者更好地减少饱和度。任选地,代替该算法的简单全局操作(这对于人类分级者而言实现起来相对简单,不需要花费太多昂贵的分级时间,甚至对自动机也是如此),在非常棘手的情况下,甚至能够发送大量处理参数集,以及在其上应用它们的颜色空间的区域的规定(例如,针对红色日落的不同方法,而不是针对在前景中的contre jour房屋的立面的蓝色TL管区的方法)。

[0094] 再次,能够相当有利的是,所述发光亮度处理器(501)包括数据接收单元(689),所述数据接收单元(689)被布置为通过网络(690)从内容的创建者接收上述公式的分数值(rho),而不是接收者必须自己确定最优的rho值,因为通常对图像进行技术处理的最优值本质上可能是艺术性的,即,人类创建者是否对第一美丽日落的接受度比较少修剪的第二美丽日落的接受度更高可能是个品味问题(一些分级者只想要外观有力的颜色,而其他分级者想要更柔和、更细腻的模式)。

[0095] 在从创建侧接收到算法参数的情况下,接收侧发光亮度处理器能够或最终应当采

取什么措施以获得较低的动态范围图像颜色的技术镜像主张是大量制作这些参数(例如,零G阈值或FADAP函数等)的系统。任何或所有这样的参数都能够由人或自动机设置(例如,建议良好设置的自动机在需要的情况下能够由人进行改进,或者人仅指定任何值而不会出现图像分析自动机等)。自动机能够将这样的属性识别为区域连贯性(例如,如果从远处看到的像花圃这样的详细纹理中有很多值,或者很少),在若干区域(例如在图像的中心,这可能在视觉上更重要)中在若干方向上计算一些连续对比度,进行一些可识别性算法,例如,经由模糊拟合,能够容易地将椭圆拟合到具有许多灰度值的平滑演化边界上,而如果所述边界是任意色调分离的,则很难获得良好的椭圆,从而产生锯齿状边界等)。

[0096] 例如,一种高动态范围视频编码器用于通常将所需参数编码在元数据中,所述元数据要与表示HDR视频的图像一起发送或者能够与该图像一起获得,所述高动态范围视频编码器可以包括:

[0097] 输入部,其用于从图像源(1202)接收输入图像;

[0098] 编码器,其用于将所述输入图像编码为输出图像并对至少一个发光亮度映射函数(F\_Lt)进行编码;其特征在于,所述编码器包括图像评价处理器(1210),所述图像评价处理器被布置为:分析所述视频的图像的颜色属性,确定由根据权利要求1所述的发光亮度处理器应用的阈值(G),并且所述编码器被布置为输出所述阈值(G)作为元数据。可以包括上面(和下面)描述的类型发光亮度处理器,以例如辅助人了解他在解码器侧的选择,但是可能不一定被包括在自动编码器中(虽然通常会包括在内,因为自动图像分析随后可以在得自一个或多个候选参数(例如,最优G阈值)选择的一幅或多幅图像上运行,这一个或多个候选参数组合是通过将输入和测试候选输出图像中的一些图像质量度量(例如,对比度、纹理度量等)进行比较而得到的。

[0099] 通常,一种高动态范围视频编码器(1201)可以包括:

[0100] 输入部,其用于从图像源(1202)接收输入图像;

[0101] 编码器,其用于将所述输入图像编码为输出图像并对至少一个发光亮度映射函数(F\_Lt)进行编码;其特征在于,所述编码器包括图像评价处理器(1210),所述图像评价处理器被布置为:分析所述视频的图像的颜色属性,确定以下参数中的至少一个:a)根据针对最优阈值的使用而给定的示例的阈值(G)(将颜色留在何处,以及在何处最佳地开始应用缓解算法及其对应的非理想颜色)、函数(FADAP)、修剪参数(Vmx)以及指示要通过调光与去饱和来处理的分割和校正量的分数值(rho),并且编码器被布置为输出这样的至少一个参数作为元数据,使得任何接收器的发光亮度处理器都能够应用对应的重新分级。

[0102] 再次取决于实施例,也许实际上仅确定和传送了一个参数(例如,Vmx),在这种情况下,接收侧只能确定接近针对与强度值 $V \leq V_{mx}$ 相对应的颜色的色域顶部的发光亮度范围内重新分配颜色的映射。但是,如果传送三个或更多个参数,则能够在接收侧使用更复杂的平衡方程。将参数的数量限制为几个最重要的参数,这当然具有优点,例如,具有滑块或轨迹球的少许滑动的人类分级者能够设置对最终结果的质量影响最大的情况值。不过,较简单的系统将具有自动机来确定至少一个参数,因为在一些场景中,将没有人力来做这项任务,或者至少不想打扰生产人员。这样的自动机能够进行图像分析,例如检测最明亮的颜色的区(高L检查,特别是高V检查)是否具有明显的空间发光亮度变化结构,例如通过采用纹理分析器来完成这项任务。而且,若干实用的实施例能够与图像劣化质量评估一起工作。

这能够如以下情况一样简单,在一些场景中对修剪像素的量或空间分布进行计数,或者计算如在校正性重新确定g因子处理之前和之后对云的发光亮度变化进行连续对比度测量的度量。

[0103] 当然,如果该创建侧具有包括所描述的高动态范围视频编码器的高动态范围视频编码系统,则该系统是特别感兴趣的,该编码器被连接到用户接口单元(1203),所述用户接口单元允许人类颜色分级者(1204)指定参数(G、rho或向我们指定更复杂的g因子重新确定函数的参数等)中的至少一个参数作为图像评估处理器(1210)的输出,其中,高动态范围视频编码器包括发光亮度处理器(501),所述发光亮度处理器被布置为计算能够在所连接的显示器(1220)上检查的输出图像(Im\_LDR; Im3000nit)。本领域技术人员理解这能够以各种方式来体现,例如在半自动确定中,有利的是,图像评估处理器已经一起提供了关于rho、G等的提议,但是人能够对其进行调谐,或者在一些操作中他甚至可能想要完全绕过在输出通道本身上的评估值和设置值等。在这些情况下,发光亮度处理器将位于编码器内部,以允许人能够在接收侧查看该动作实际上将要执行的操作,并且在那些工作比当前场景更好的情况下校正其他参数。显示器1220可以是高PB\_D(例如,10000nit)的参考显示器,使得它能够示出结果得到的SDR图像以及许多可能的结果得到的MDR图像,例如在PB\_MDR=3000nit的情况下,它能够示出高达3000nit的像素发光亮度。

[0104] 另外的有利的实施方式特别是:

[0105] 一种高动态范围视频解码器(1300),包括:

[0106] 输入部,其用于接收输入图像和发光亮度映射函数(FLM);

[0107] 第二输入部,被布置为接收阈值(G)的值;

[0108] 根据权利要求1至7中的任一项所述的发光亮度处理器(501),其被布置为通过以下操作来计算输出图像:使用所述发光亮度映射函数(FLM)和所述阈值(G)来改变所述输入图像的像素的发光亮度而产生输出图像;以及

[0109] 输出部,其用于输出所述输出图像。

[0110] 一种高动态范围视频解码器(1300),包括根据所教导的发光亮度处理器实施例中的任一个的发光亮度处理器(501);

[0111] 一种发光亮度处理方法,其用于根据具有以第一峰值发光亮度(PB\_HDR)为特征的第一发光亮度动态范围的输入图像(MAST\_HDR)的空间共位的像素的输入发光亮度来计算具有以第二峰值发光亮度(PB\_LDR; PB\_MDR)为特征的第二发光亮度动态范围的输出图像(Im\_LDR; Im3000nit)的像素的输出发光亮度,所述发光亮度处理方法包括:

[0112] 计算作为输入发光亮度和发光亮度映射函数(FLM)的函数的乘法因子(gL);

[0113] 计算强度值(V),所述强度值是输入图像的像素的颜色中的红色分量、绿色分量和蓝色分量这三个分量中的最大的一个,其中,那些分量是线性的红色分量、绿色分量和蓝色分量或者是那些线性的红色分量、绿色分量和蓝色分量的幂;

[0114] 计算溢出度量(T),所述溢出度量指示输出发光亮度与上色域边界的接近程度;

[0115] 在溢出度量大于阈值(G)的情况下确定替代增益因子(F1(gL)),在其他情况下保持原始增益因子,并且输出替代增益因子和原始增益因子中的一个作为最终增益因子(gF);并且

[0116] 将输入颜色(R'G'B'\_nrm)与最终增益因子(gF)相乘以获得具有输出发光亮度的

输出颜色(R'G'B'\_HDR)。

[0117] 一种发光亮度处理方法,其用于根据具有以第一峰值发光亮度(PB\_HDR)为特征的第一发光亮度动态范围的输入图像(MAST\_HDR)的空间共位的像素的输入发光亮度来计算具有以第二峰值发光亮度(PB\_LDR;PB\_MDR)为特征的第二发光亮度动态范围的输出图像(Im\_LDR;Im3000nit)的像素的输出发光亮度,所述发光亮度处理方法包括:

[0118] 基于指示如何调节所述输入发光亮度以变为中间输出发光亮度的发光亮度映射函数(FLM),通过将乘法因子计算为所述发光亮度映射函数针对所述输入发光亮度的输出除以所述输入发光亮度的除法 $gL = FLM(L) / L$ 来计算所述乘法因子(gL),所述乘法因子针对所述输入图像的图像像素的任何输入颜色的所述输入发光亮度被定义;

[0119] 计算指示所述中间输出发光亮度在所述输入颜色的色度处的上色域边界上方多远处的溢出度量(T),所述中间输出发光亮度是通过将所述乘法因子(gL)应用于所述输入颜色而得到的;

[0120] 在所述溢出度量大于阈值(G)的情况下确定较低的替代增益因子(F1(gL)),并且在其他情况下保持原始增益因子,并且输出所述替代增益因子和所述原始增益因子中的一个作为最终增益因子(gF);并且

[0121] 将所述输入颜色(R'G'B'\_nrm)乘以所述最终增益因子(gF)以获得具有所述输出发光亮度的输出颜色(R'G'B'\_HDR)。

[0122] 一种发光亮度处理的方法,包括通过任何连接的网络从内容的创建者接收所教导的以下参数中的至少一个:a)阈值(G),高于该阈值则开始进行重新确定处理,并且其中,必须将重新确定的颜色挤压到输出色域中;b)进行g因子确定所需的特定函数(FADAP)(或其任何等效物,例如,强度值V的函数)的形状的参数整理;c)修剪参数(Vmx),其仍然允许对最具违背性的颜色(即,那些落在色域顶部上方最远处的颜色因此最难于重新映射在色域中)进行一定程度的修剪;以及d)分数值(rho),其允许在两种部分重新映射策略中指定拆分(一种部分重新映射策略重新计算像素发光亮度,并且一种部分重新映射策略重新计算像素饱和度),并且在发光亮度计算中应用这样的至少一个参数。

[0123] 一种HDR图像信号,包括:

[0124] 包括图像宽度乘以图像高度像素颜色的集合;

[0125] 至少一个发光亮度映射函数,其定义如何根据像素颜色的发光亮度来计算输出发光亮度,并且参数中的至少一个参数是:a)阈值(G),b)修剪参数(Vmx),c)分数值(rho);

[0126] 一种HDR图像信号,包括:

[0127] 包括图像宽度乘以图像高度像素颜色的集合;

[0128] 至少一个发光亮度映射函数,其定义如何根据像素颜色的发光亮度来计算输出发光亮度,并且还包括函数(FADAP)。

[0129] 显然,这些信号在这样的实施例中体现了本发明,其中,例如G的值在将要发生的特定发光亮度重新映射中起主要作用。这样的元数据的含义、格式、唯一可识别性不是与该阐述无关的细节,因为技术人员将容易理解,整个链的正确工作(该信号仅在创建装置或方法与使用装置或方法之间建立了联系)通常通过适当标准化的元数据来实现,例如利用由标准化机构(例如,ETSI)同意的占位符来实现,并且创建装置确保了接收装置将通过例如在标头中传送的格式来理解信号以ETSI SL\_HDRxyz格式等被传送。

## 附图说明

[0130] 参考下文描述的实施方式和实施例并且参考附图,根据本发明的方法和装置的这些方面和其他方面将变得显而易见,附图仅用作非限制性特定图示来例示更一般的概念,并且其中的虚线用于指示部件是任选的,非虚线部件不一定是必不可少的。虚线也能够用于指示那些被解释为必不可少但隐藏在对象内部的元件,或者用于诸如对对象/区域(以及它们可以如何在显示器上示出)的选择之类的无形的东西。

[0131] 在附图中:

[0132] 图1示意性地图示了当将高动态范围图像最优地映射到对应的最优颜色分级并且看起来相似的(如期望和可行地相似,给出了第一动态范围DR<sub>1</sub>相对于第二动态范围DR<sub>2</sub>的差异)较低动态范围图像(例如,峰值亮度为100nit的标准动态范围图像)的多个典型的颜色变换,在可逆性(模式2)的情况下,它 also 对应于接收到的实际上对HDR场景进行编码的SDR图像到该场景的重建的HDR图像的映射;

[0133] 图2示意性地图示了对高动态范围图像进行编码的技术的卫星视图示例,即,通常能够具有至少700nit(即,SDR图像的PB<sub>C</sub>的至少7倍)的发光亮度或更高(通常为1000nit或更多)的图像。该申请人最近开发的技术能够实际上将(一幅或多幅)HDR图像作为SDR图像加上元数据编码的颜色变换函数进行传送,该元数据编码的颜色变换函数至少包括适当确定的针对像素颜色的发光亮度变换,以供解码器用于将接收到的(一幅或多幅)SDR图像转换成(一幅或多幅)HDR图像,该(一幅或多幅)HDR图像是对在图像创建侧创建的(一幅或多幅)原始主HDR图像的忠实重建;

[0134] 图3示出了在这样的HDR解码(或编码)和处理中发生的动态范围变换的仅发光亮度变化(即,没有不想要的色度变化)的发生方式,例如当确定图像的动态范围低于输入图像时,如理论颜色空间所示,具有色度平面和正交发光亮度L轴;

[0135] 图4示意性地阐明了颜色和颜色处理在另一种颜色空间(具有另一种几何形状)中的行为,该空间是视频的典型颜色编码空间:YCbCr(其中,Y原则上能够是线性发光亮度L,或者是非线性伽玛2.01uma Y',或者甚至是利用另一OETF定义的某种其他luma,如基于PQ的luma Y'')。

[0136] 图5(并不旨在是限制性的)示出了申请人通常如何喜欢执行图2的单元202或208的颜色变换的更详细的示例;

[0137] 图6示出了本发明的主要方面的一般性顶层阐述,本发明将其添加到例如在图2和图5中示出的系统的发光亮度处理中;

[0138] 图7示出了这样的系统的部分行为,特别是在图5中没有图6的校正元件的情况下在色度-发光亮度颜色空间中的部分行为。

[0139] 图8示出了高级实施例的行为,该高级实施例借助于去饱和操作部分解决了在输出颜色空间的色域的顶部附近的校正映射的问题;

[0140] 图9大体上且示意性地示出了发光亮度处理器的示例,该发光亮度处理器是还具有饱和度处理器的颜色处理器的部分。

[0141] 图10示意性地示出了用于确定色域的上部中的颜色的校正颜色映射策略的一些示例,并且基于强度值V的对应函数被定义为颜色的红色分量、绿色分量和蓝色分量中的最大的一个,仅对于色域顶部附近的颜色而言,这能够根据本教导而被形成为对应的增益因



子gF,特别是作为初始增益因子的偏差的结果得到的增益因子;

[0142] 图11示出了用户接口,该用户接口允许内容创建侧的人类分级者指定几个简单但功能强大的参数,这些参数近似地指导或精确地指定发光亮度,并且通常通过根据本发明的教导的变体中的任一个的接收侧发光亮度处理器对色域顶部颜色附近的明亮区域进行颜色处理;并且

[0143] 图12示意性地示出了可能的自动、半自动或手动编码器的示例,其产生根据所教导的实施例的解码侧发光亮度处理器变化所需的参数。

[0144] 图13阐明了根据本发明原理的与图9的编码器相对应的模式2解码器。

## 具体实施方式

[0145] 现在,我们对各种实施例的阐述集中在图5的发光亮度处理器(501)部分,以更精细的配置对其进行了重新绘制,以在图6中阐明本发明的新原理。

[0146] 首先,虽然我们将阐明假定处理是在线性发光亮度上进行的,即,发光亮度L进出( $L_{out}$ ),即使在该分支内部的处理也可能会转换为另一个诸如PQ之类的luma域,但是也能够其他luma域中这样做,例如通常利用经典的Rec.709luma,即, $\sqrt{L}$  luma  $Y'$ (即,在这样的场景中,发光亮度映射器512对动态范围转换的发光亮度映射可以被直接指定为 $Y'_{out} = FLM(Y'_{in})$ )。

[0147] 然而,在这样的场景中,必须注意,不是计算非恒定发光亮度的luma,而是计算对应的恒定发光亮度的luma(以确保发光亮度处理正确地表现为线性发光亮度处理,即使它被实施在平方根域中,也不会因在不正确的非恒定发光亮度的luma中发生发光亮度估计错误而导致错误,例如荧光红色涂层等)。

[0148] 回想一下,luma的标准定义为 $Y' = a \cdot \sqrt{R} + b \cdot \sqrt{G} + c \cdot \sqrt{B}$ ,而恒定发光亮度的luma的定义为 $Y'_{CL} = \sqrt{L}$ ,其中, $L = a \cdot R + b \cdot G + c \cdot B$ 。

[0149] 通常,视频解码器的发光亮度处理器会得到YCbCr输入,因此,根据该输入自然导出的颜色分量是非线性 $R'G'B'$ 。导出恒定发光亮度luma的一种方法是让颜色空间转换器502通过以下操作来计算线性RGB分量:首先通过应用适当的EOTF(即, $R = EOTF\_rec709(R') = \text{power}(R'; 2)$ 等),然后计算L,并且然后计算 $Y'_{CL} = \sqrt{L}$ 。

[0150] 由于硬件面积或处理成本(/功率)不是无限的,因此可能会有捷径,例如:

[0151] 如果 $\max(K, L) > 4 \cdot \min(K, L)$ ,则 $\text{Sqrt}(\text{power}(K; 2) + \text{power}(L; 2)) \approx \max(K, L)$ ,

[0152] 并且 $\approx (7/8) \cdot \max(K, L) + (1/2) \cdot \min(K, L)$ ,否则,对于任何数字K和L。

[0153] 因此,能够使用该公式来根据 $R'G'B'$ 值直接计算 $Y_{CL}$ 。

[0154] 虽然V是随发光亮度L增长的量(或者在具有luma  $Y'$ 的根号幂表示中),但是它并不是微不足道的,但是它确实具有有用的属性,它是与上色域边界的接近程度的度量,特别是色域上边界上的所有颜色均具有最大强度值 $V = 1.0$ (在变亮的情况下高于该边界,颜色分量中的最大的一个(例如,蓝色)将需要超出其归一化最大值,例如, $B = 1.1$ ,这在数学上是可能的,但是在物理上是不可能的)。附图标记为500s的部件与图5中的部件相似,即,它们特别会进行“正常”的颜色变亮,即,按照发光亮度处理函数FLM中的指定要求,并且尚未具有g因子确定的上色域区处理机制。

[0155] 假定图6的发光亮度处理器位于HDR解码器中,它将得到利用YCbCr像素颜色编码



的5000nit PB\_C HDR作为输入(如果它们是基于PQ进行编码的,则转换器将进行正确的数学运算以转换为发光亮度,并且在示例中转换为线性RGB),并且导出例如900nit的MDR图像及其发光亮度。还计算了强度值V,并且利用使用线性颜色分量的示例进行阐明,因此 $V = \max(R, G, B)$ 。

[0156] 现在有趣的是,发明人认识到,虽然应该将由创建侧确定的映射FLM应用于像素颜色的发光亮度L(或者实际上执行了等效性质的颜色处理,例如将线性RGB分量与相同的发光亮度改变g因子gL相乘),将其应用于强度值V是感兴趣的,因为这样,对于每种颜色色度,就能够看到它是否会映射到色域边界上方。

[0157] 这能够在图7中看到。发光亮度映射应该基于它们的发光亮度来映射颜色,而与它们的色度无关,因此我们能够在发光亮度轴上检查它是否为消色差的颜色。假定我们有HDR输入图像的某种颜色的平均发光亮度Col\_in,它会变亮为Col\_out(以例如创建SDR图像或MDR图像),因为所有颜色都需要变亮,特别是较暗的颜色仍然能够合理地看到在较低动态范围渲染中在影片中发生的变化,但是由于需要将发光亮度关系保持在两个分级(即,输出图像与输入图像)中,因此那些变亮的颜色也会将所有更多的发光颜色向上推。

[0158] 对于消色差灰色来说,这种变亮是没有问题的,因为Col\_out似乎仍在色域内部。然而,如果绘制等发光亮度的水平线,我们会看到将与灰色HDR颜色Col\_in相同发光亮度的蓝色ColBluH映射为其LDR等效的ColBluL,我们将在色域上方进行映射,因此需要在我们的处理中进行一些操作,否则修剪器将会将至少一个颜色分量自动修剪为1.0,这通常会涉及相当大的色度错误,包括潜在的令人讨厌的色调错误。

[0159] 然而,通过绘制等V线(用粗虚线表示),能够看到这种情况,输入蓝色ColBluH在色域中仍然很好(例如, $V=0.8$ ),但是输出颜色ColBluL具有例如 $V=1.2$ ,因此,其高于1.0。

[0160] 现在有趣的是,线性V度量的行为确实与L类似地成比例,因此能够看到它被相同的g因子变换:

[0161] 如果 $R_{out} = gL * R_{in}$ ;  $G_{out} = gL * G_{in}$ ;  $B_{out} = gL * B_{in}$ ,那么 $\max(R_{out}, G_{out}, B_{out}) = gL * V_{in}$ 。

[0162] 因此,实际上,准则看到gL的值\*由图6中的502计算的V的(输入)值,并且在示例性实施例中,这是通过乘法器602计算的,并且其结果被称为测试值T。

[0163] 随后,增益因子修改单元603在测试中使用该值来根据一些图像元数据编码公式修改根据分级函数FLM导出的初始增益值,该分级函数FLM是由接收器本地确定的,或者是从通常与实际接收到的图像在一起的内容创建侧接收到的。

[0164] 在最简单的实施例中,阈值G等于1.0(例如,硬固定在发光亮度处理器中)。

[0165] 然后,如果颜色映射为 $T = gL(\text{color}) * V(\text{color}) \leq 1.0$ ,则它们实际上是在色域内映射的,这没有问题,因此,映射很好,并且通过了未修改的初始增益gL作为最终增益gF以用于在RGB分量上进行实际的HDR到MDR颜色改变。请注意,我们添加了(颜色),以强调计算的gL因子和V值均取决于当前像素颜色的颜色分量值。

[0166] 然而,如果 $T > G = 1.0$ ,则存在色域溢出问题,例如当 $gL * V$ 等于1.8时。

[0167] 在这种情况下,必须应用函数F1(gL),这样确保了至少对于大多数颜色,映射的颜色(以及其V值)映射在色域内部(除非确定了V<sub>mx</sub>值,否则只有少数几种颜色仍然能够修剪)。

[0168] 对于所有可能的色域映射 $g_L(\text{color}) * V(\text{color})$ ,能够由解码器确定的简单函数(例如被固定在其发光亮度处理器软件或电路中)可以如下所示:

[0169] 如果 $T > G = 1.0$ ,则 $g_F = g_L * (1/T)$  [公式3]

[0170] 在这种情况下,确实能够看到,如果对于一些颜色,当利用原始 $g_L$ 因子映射输出颜色时, $T$ 值例如为1.3,然后利用另一 $g_F$ (其为 $g_L/1.3$ )映射相同的输入颜色(及其 $V$ 值)将完全映射到一个(并且对另一颜色应用 $g_F/1.8$ ,另一颜色映射到1.8将进行相同的操作等)。

[0171] 当然,虽然保留了色度,但是这是相当粗糙的策略(但是易于开始解释),即使在一些情况下也适用,但是在处理更关键种类的HDR图像时并未最优地工作。

[0172] 因此,使用更复杂的映射策略可能是有利的,该策略通过引入(至少)阈值 $G$ 而仍然保留最初映射在色域边界上方的所有颜色的原始发光亮度差异的可区分性。能够由接收侧发光亮度处理器再次确定该阈值,或者有利地,由数据接收单元(689)接收该阈值并将其传送到增益因子修改单元603,所述数据接收单元被布置为通过网络(690)从内容的创建者接收阈值( $G$ ),熟练的读者能够将其理解为任何当前或未来的图像或数据递送系统,例如,卫星电视频道、互联网或前缀打包存储设备,甚至可以将针对相继的视频图像的 $G$ 值存储在例如在蓝光光盘上并经由BD阅读器等访问该 $G$ 值。

[0173] 图10阐明了能够如何体现各种 $g_L$ 重新确定策略/函数。在图10A中,我们示出了一个示例,其中,阈值 $G$ 被设置为0.9,并且线性映射达到 $V_{mx}$ 。实际上,可能有几种颜色(可能只有几十个像素)的 $g_L * V$ 值变得高于 $V_{mx}$ 而达到图像中的最高值,或者运行了相继的图像 $V_{maxSCN}$ ,但是这些颜色可能很难被修剪为色域顶部边界上与HDR输入颜色具有相同色度的颜色。

[0174]  $T_{in}$ 是具有初始 $g_L$ 因子的映射颜色的 $V$ 值,并且我们应当利用 $g_F$ 进行最终映射,该最终映射将产生高达1.0的 $T_{out}$ 值,与色域边界相对应。低于0.9,增益因子不变,因此 $T_{out}$ 值也将与 $T_{in}$ 值(SEGPSS)相同。 $G$ 与 $V_{mx}$ 之间的重新映射能够通过多种方式来制定,并且能够像线性段(SEGADPT)一样简单。能够根据该轴系统中的表示来轻松地计算出衰减因子 $A$ ,例如如果颜色2.5必须映射为1.0,则需要将 $g_L$ 除以2.5,以此类推。

[0175] 针对衰减 $A$ (仅在发光亮度方向上的总衰减)的示例计算公式为:

[0176]  $A(T) = (T > G) ? 1 / \{1 + [(V_{mx} - 1) / (V_{mx} - G)] * (T - G)\} : 1$  [公式4]

[0177] 能够看出,如果 $T = V_{mx}$ ,则 $A(T)$ 变为 $1/V_{mx}$ ,以此类推。

[0178] 有趣的是,图10B示出了上述色域问题的部分通过去饱和得到解决的情况。在那种情况下,最初映射到 $V_{mx}$ 的颜色能够映射到 $y.x > 1.0$ ,例如1.9(可能在色域边界上方不太远而不需要太多的去饱和),因为溢出的其余部分被示出在饱和方向上。

[0179] 这被示出在图8的色域标绘图上。总衰减 $A$ 将把映射的颜色带到色域边界。然而,通过如上所述地定义 $\rho$ 参数,能够映射到两者之间的位置,然后通过去饱和将其水平移动到 $Col_{out}$ 。

[0180]  $\rho$ 参数能够被权衡为在变暗的有害影响与去饱和的有害影响之间的最优量,例如通过人类分级者将滑块1101定位在他的0至1之间的优选位置上来进行权衡。或者,一些实施例可以具有固定的设置,例如, $\rho = 1/2$ 等。

[0181] 去饱和和所需的乘法参数的计算公式能够如下:

[0182]  $S = (V_{interm} * A_{rem} - Y) / (V_{interm} - Y) = (1 - Y) / (V_{interm} - Y)$  [公式5],其中,

Vinterm是在进行发光亮度调光部分之后仍高于色域边界的高度,并且Y为例如线性示例中的中间发光亮度(即,图9中的Li)。当然,如果需要的话,能够使用更复杂的去饱和策略(对于非常数S,即,能够在颜色空间上变化的去饱和),特别是如果它们被传送到接收侧(至少是可逆系统,如模式2HDR作为SDR通信应该知道所有要重建的内容)。

[0183] 图9中示出了能够实施这样的校正策略的编码器中的处理架构的示例(以及连接到饱和单元的高级发光亮度处理器实施例)。原则上,能够选择根据需要的相反顺序进行饱和处理和发光亮度处理,但是对于选择的顺序,解码器应将该顺序反转(因此,如果编码器首先进行发光亮度映射,然后进行去饱和,则反向解码器将首先进行对应的重新饱和,然后进行发光亮度映射)。请注意,如果知道Vinterm和Y,则就知道S(无论是去饱和还是重新饱和),如果知道Arem,就例如借助于rho参数从创建侧传送Arem。

[0184] 解码器侧的衰减A(T)能够被计算为:

[0185]  $A(T) = (T > G) ? \{1 - [((V_{mx} - 1) / (V_{mx} - G)) * G] / \{1 - [((V_{mx} - 1) / (V_{mx} - G)) * T]\} : 1$  [公式6]

[0186] 现在,部分映射计算器903确定在乘法器902中乘以gL的衰减因子Arho,以获得用于部分发光亮度调光的适当的最终gF,该过程由乘法器910在RGB三元组上执行,并且必须变为中间发光亮度L1的发光亮度通过去饱和器911在稍后执行去饱和步骤。饱和因子确定单元901能够例如根据公式5来确定所需的饱和因子S(V,Y,Arem),在根据公式7的发光亮度保持去饱和的情况下,饱和处理器911应用该饱和确定因子:

[0187]  $R_o = L_i + S * (R_i - L_i); G_o = L_i + S * (G_i - L_i); B_o = L_i + S * (B_i - L_i)$

[0188] 这些输出颜色分量然后是针对例如根据HDR输入图像颜色导出的SDR颜色(或MDR颜色)的校正颜色分量。

[0189] 请注意,读者必须理解,在编码器和解码器中都进行降级时(即,实际上将HDR图像本身传送给接收器的模式i情况,通常利用PQ-OETF编码的非线性R”G”B”值,或者通常是因为视频编码将通过标准例如HEVC压缩对应于YCbCr颜色分量),编码器和解码器都将在相同的降级方向上变换颜色,因此解码器中的颜色处理顺序不会逆转(能够想象到在这种情况下编码器仅利用选定的参数选择进行变换,以检查以后的解码器行为)。

[0190] 然而,在可逆的HDR图像或视频编码(即,将图像编码为SDR图像的模式2)中,编码器通过降级来创建图像,而解码器则通过在相反的方向上升级来重新创建主HDR图像的近似结果。

[0191] 然后,解码器会有所变化,如图13所示。

[0192] 请注意,饱和(911)和变亮(乘法器910)的相反顺序(现在以1/gF发生,但是它是相同的部分g因子)将初始的高于色域的颜色ColBluL降低为最终颜色ColOut的中间发光亮度,这将再次成为以相同因子提高发光亮度的中间结果。现在,我们使用后缀s来指示在这种场景中输入颜色RsGsBs和发光亮度Ls是SDR颜色,并且输出颜色RoH、GoH、BoH是重建的HDR图像(根据对接近色域顶部的关键颜色的新颖优化处置)。还要注意额外的乘法器1301,由于发光亮度的中间位置不是初始位置,因此现在在g因子确定路径中使用额外的乘法器1301。

[0193] 图11示出了用户接口的示例,该用户接口能够用于人类分级者创建参数,该参数能够仅与三个值G、rho和Vmx一样简单。Rho能够例如由滑块设置,例如半自动方法的自动机

确定最优设置为0.25,但用户更喜欢0.2或0.3。

[0194] 他还有另外两个移位器1103resp.1103以设置Vmx和G值,例如在色域标绘图上进行设置。也可能存在正在处理的图像的视图1110,例如,在相继的视频图像的镜头中拍摄的HDR场景的表示性图像,在该图像上可能会自动生成指示符1111,以迅速吸引分级者的注意。例如,自动机已经计算出在该区域(阳光照射的云)中,纹理的对比度显著降低(甚至可能是硬修剪),并且用户接口能够例如在该区域周围示出闪烁的红色边界。

[0195] 图12示出了具有编码器1201的编码系统的示例(编码器1201既能够根据任何先前描述的模式对HDR图像进行编码,又能够确定当前的新参数rho、G、Vmx的输出或用于在接收侧重新确定g因子的函数),内容创建颜色分级器1204能够对其进行分级工作,至少根据本发明确定一个参数,例如,G或rho。他能够在显示器1220上检查发生了什么,因为发光亮度处理器501应用了处理(即,在模式1中直接降级,并且通常整个编码+解码链包括在模式2中的反向升级)来例如经由用户接口单元1203(其通常可以是分级面板)和软件(其将例如rho的变化分配给轨迹球)或者在一些情况下经由实时流处理来示出分级者选择的结果。图像评估处理器1210可以进行各种图像分析以得出rho、G等的提议,并且在自动实施例中,这将是确定那些参数的唯一单元。

[0196] 高动态范围视频编码器(1201)能够使用发光亮度处理器来帮助人类颜色分级者为控制其发光亮度改变算法的任何实施例的各种参数确定合适的值(例如被包括在解码器中,该解码器通过一些数据或视频连接将这些参数接收在元数据中作为控制参数)。为了传送任何这样的控制参数,HDR视频编码器可以包括:

[0197] 输入部,其用于从图像源(1202)接收输入图像;

[0198] 编码器,其用于将输入图像编码为输出图像并且对至少一个发光亮度映射函数(F<sub>Lt</sub>)进行编码;

[0199] 图像评估处理器(1210),其被布置为分析视频的图像的颜色属性,以确定以下参数中的至少一个:a) 阈值(G),其指示应在其之上使用一些替代最终增益gF,并且在其以下,在给定其对象或像素发光亮度分布细节的情况下与适合用于当前图像的动态范围重新分级的发光亮度映射函数FLM相对应的原始增益;b) 函数(FADAP),其能用于确定针对要处理的图像的较亮颜色的替代增益;c) 修剪参数(Vmx),其指示(在发光亮度重新分级之后)允许从哪个输入发光亮度进行修剪,但保留色度的修剪(如果需要这种效果的情况);以及d) 分数值(rho),其确定校正间隔朝向要通过变暗与去饱和而处理的色域的划分,例如在施加了最终gF之后剩余的发光亮度距离的40%并要进行去饱和处理,以至少保证像素的输入颜色的色调保持不变,并且编码器被布置为将至少一个参数作为元数据输出。技术人员能够从我们目前的所有教导中理解如何类似地例如确定合适的G值,在G值之上必须对原始的基于FLM的g因子确定的发光亮度变化应用一些校正(无论是完全保留色调还是保留一些色调错误修剪,但通常较小且远小于不应用目前的色域校正原理的情况),例如通过人类观看,在设置的方式下,有云的阳光照射的夜空有多少会变坏(例如利用预先约定的简单策略,编码侧知道在没有更好的FADAP被传送的情况下解码侧将应用该简单策略),还能够确定其他参数。例如,人类颜色分级者能够使用UI来绘制FADAP函数的上部的形状,以用于确定针对最亮颜色的gF值,使得例如不要使一些图像颜色变暗太多,或者不要降低云的一些部分中的一些像素的至少一些发光亮度,这会使这样的云的可见纹理变差太多等。而且,自动机可以

判定基于计算来应用一些修剪,例如被估计为感兴趣区域或关键区域(例如,所述云)中连接的像素的各个集合的对比度(可以基于例如其他纹理度量(例如,低业务度量)进行识别,这表明,与高频多色纹理区域(例如从远处看到的花坛等)相比,天空中的平滑渐变可能会更容易示出伪影)。但是如果编码器仅使用自动图像分析算法,则无需包括发光亮度处理器即可将任何选择的结果实际显示给视频内容创建人。

[0200] 类似地,发光亮度处理方法可以包括:通过任何连接的网络(网络显然能够在最广泛的意义上使用)从内容的创建者接收以下参数中的至少一个:a) 阈值(G)、函数(FADAP)、修剪参数(Vmx),以及分数值(rho),参数能够以单独或组合方式使用,这取决于特定的发光亮度改变实施例的需求,并且将这样的至少一个参数应用于其发光亮度计算。

[0201] 在实践中,本文中公开的算法部件可以(全部或部分)被实现为硬件(例如,专用IC的部分)或运行在特殊数字信号处理器或通用处理器等上的软件。

[0202] 从我们的介绍中,本领域技术人员应当理解,哪些部件可以是任选的改进并且能够与其他部件组合实现,以及方法的(任选)步骤如何对应于装置的相应单元,反之亦然。在本申请中,词语“装置”以其最广泛的含义使用,即,一组允许实现特定目标的单元,并且因此能够例如是IC(的小电路部分),也能够是专用设备(例如,带有显示器的设备),也能够是联网系统的部分等。“布置”也应在最广泛的意义上使用,因此可以特别包括单个装置、装置的部分,协作装置(的部分)的集合等。

[0203] 应将计算机程序产品的符号理解为涵盖使得通用或专用处理器在一系列加载步骤(可能包括中间转换步骤,例如翻译成中间语言和最终处理器语言)之后能够执行的命令集合以将命令输入到处理器中并运行本发明的任何特性功能的任何物理实现方式。特别地,该计算机程序产品可以被实现为载体(例如,磁盘或磁带)上的数据,存储器中存在的数数据,经由有线或无线网络连接传输的数据或纸上的程序代码。除了程序代码之外,程序所需的特性数据也可以被体现为计算机程序产品。

[0204] 该方法的操作所需的一些步骤可能已经存在于处理器的功能中,而不是在计算机程序产品中描述的内容,例如,数据输入和输出步骤。

[0205] 应当注意,上述实施例说明而非限制本发明。在技术人员能够容易地实现将所呈现的示例映射到权利要求的其他区域的情况下,为简洁起见,我们没有深度提及所有这些选项。除了如权利要求中所组合的本发明的元件的组合之外,这些元件的其他组合也是可能的。元件的任何组合能够在单个专用元件中实现。

[0206] 权利要求中的括号内的任何附图标记都不旨在限制权利要求。词语“包括”不排除权利要求中未列出的元件或方面的存在。元件之前的词语“一”或“一个”并不排除多个这样的元件的存在。

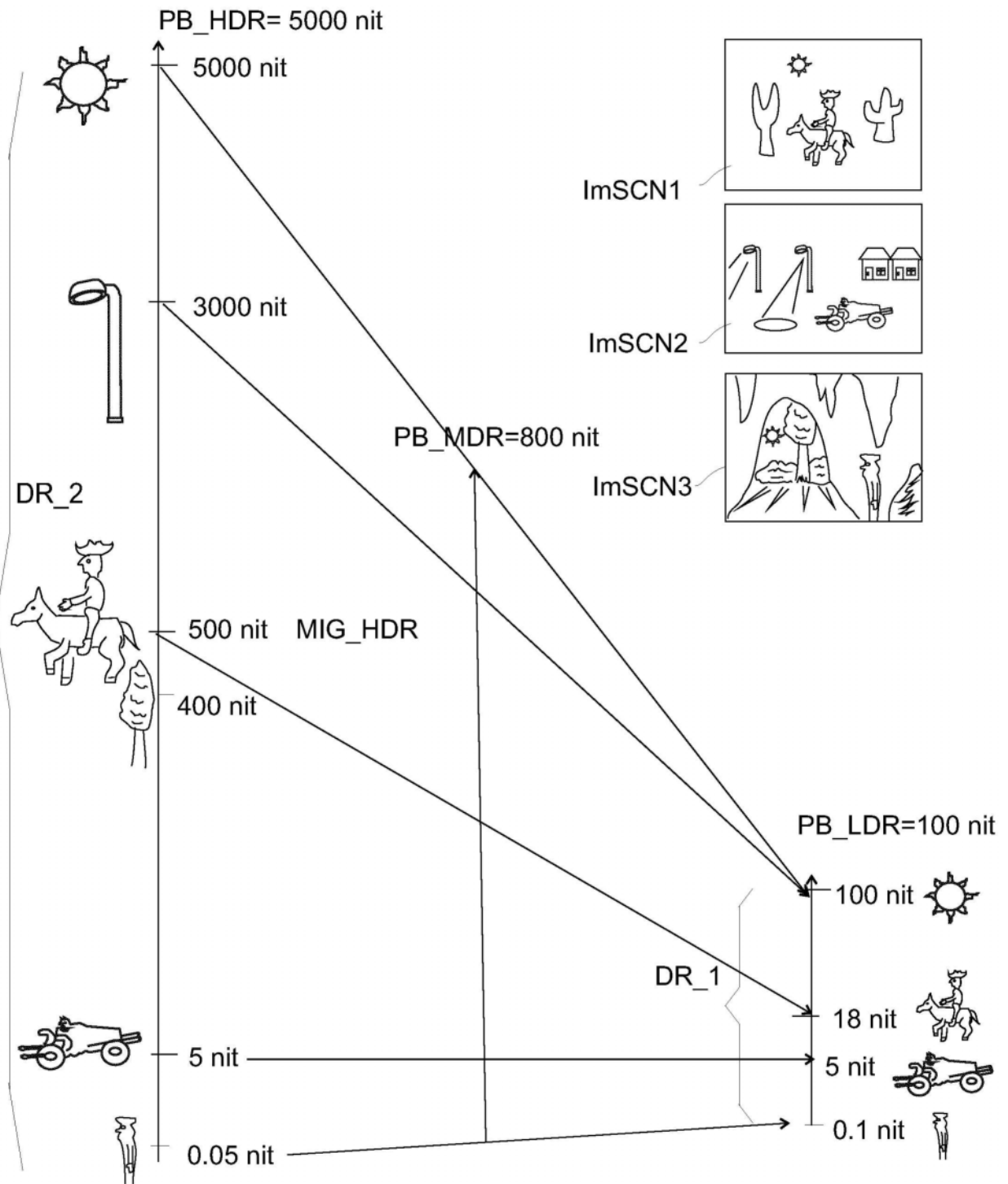


图1

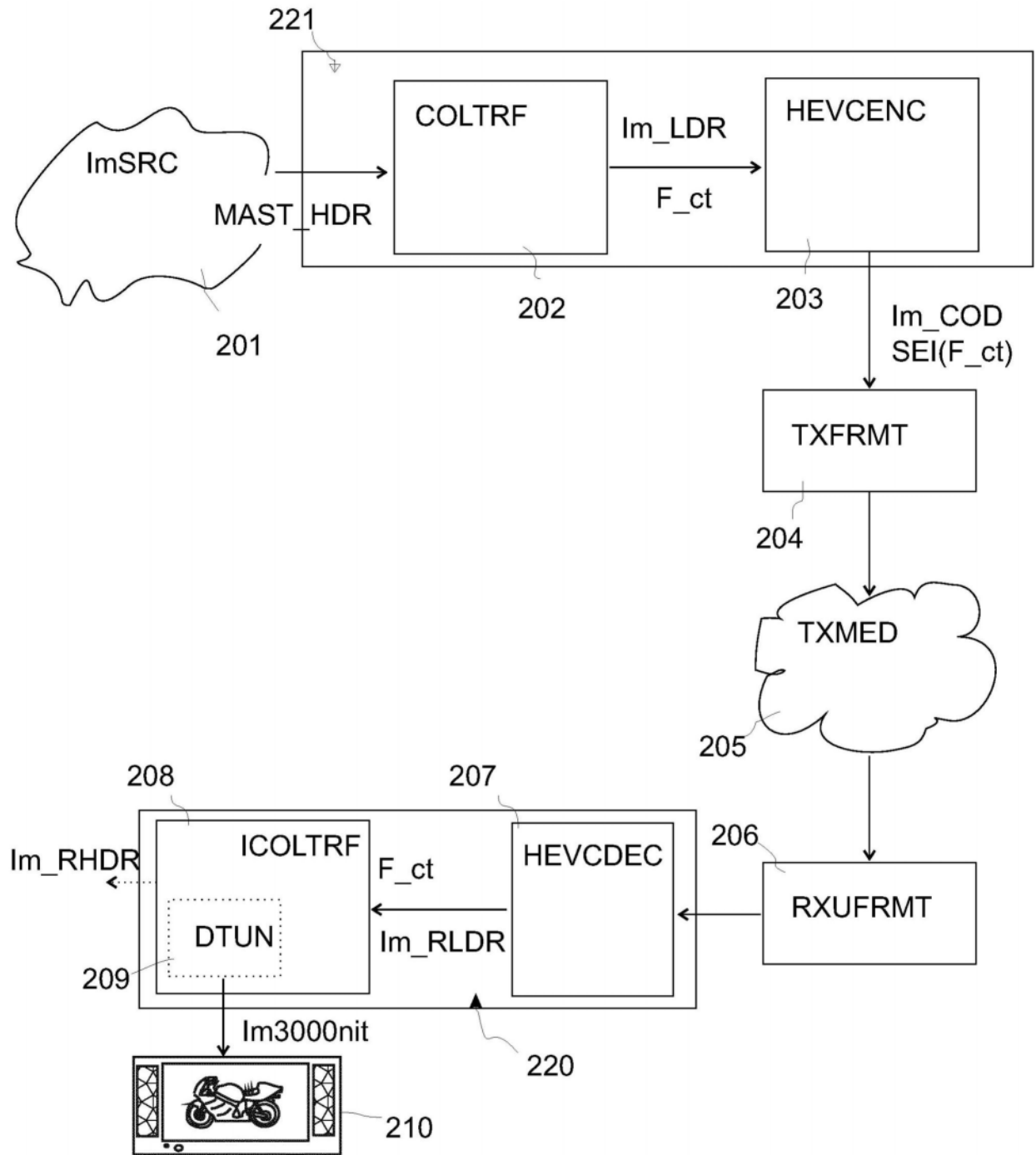


图2

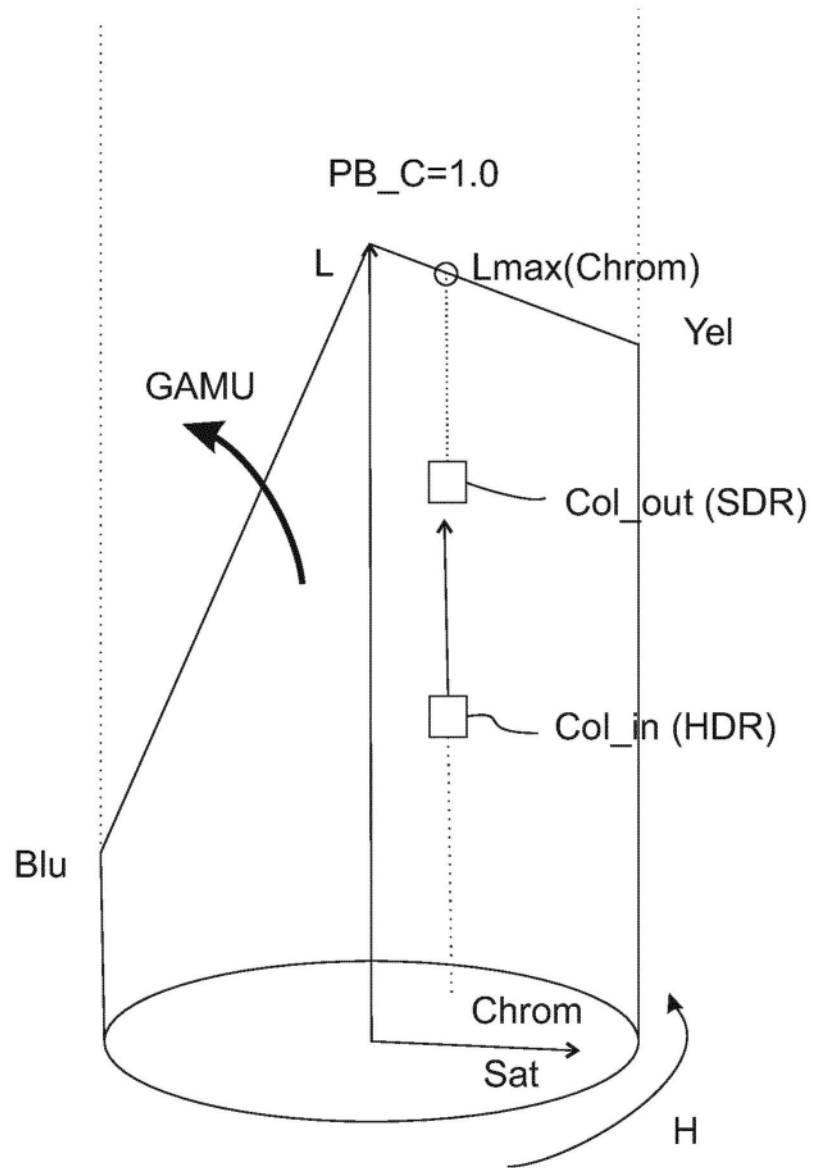


图3



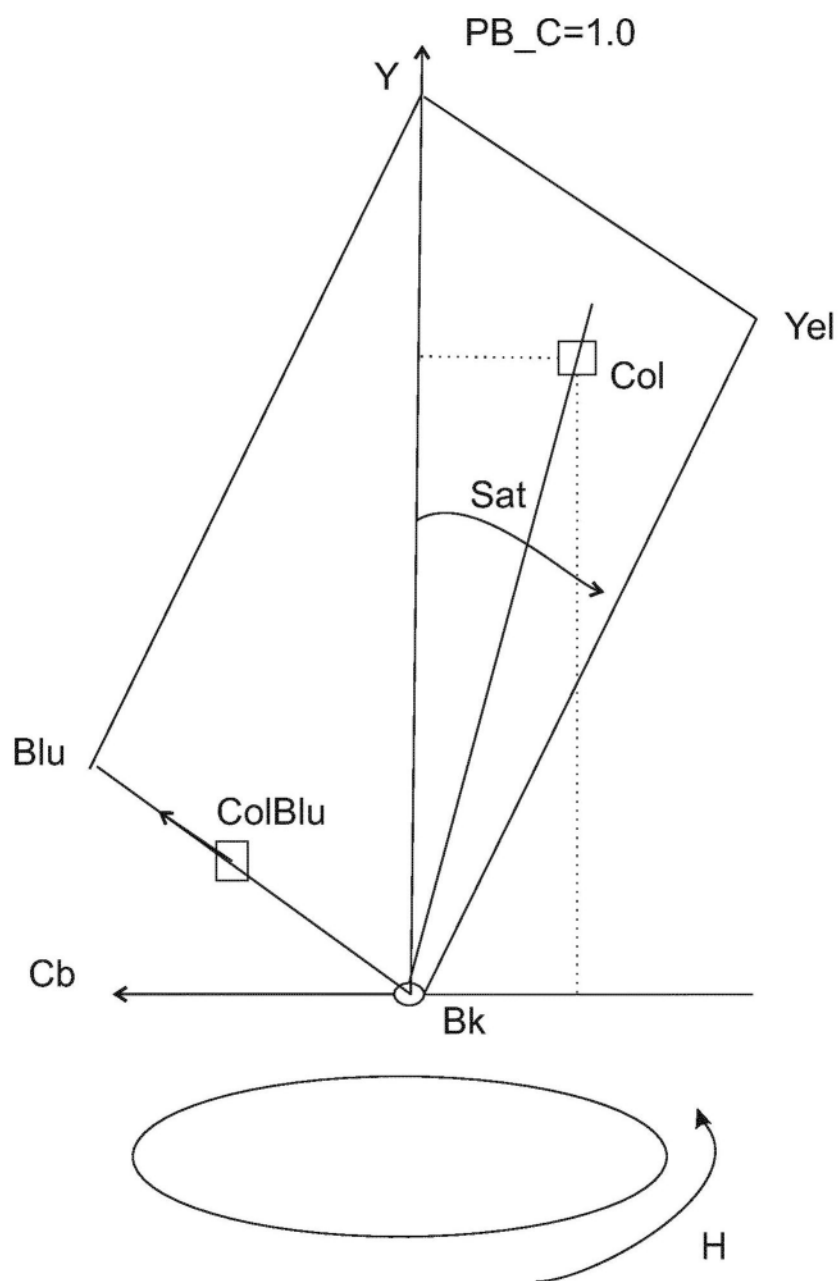


图4

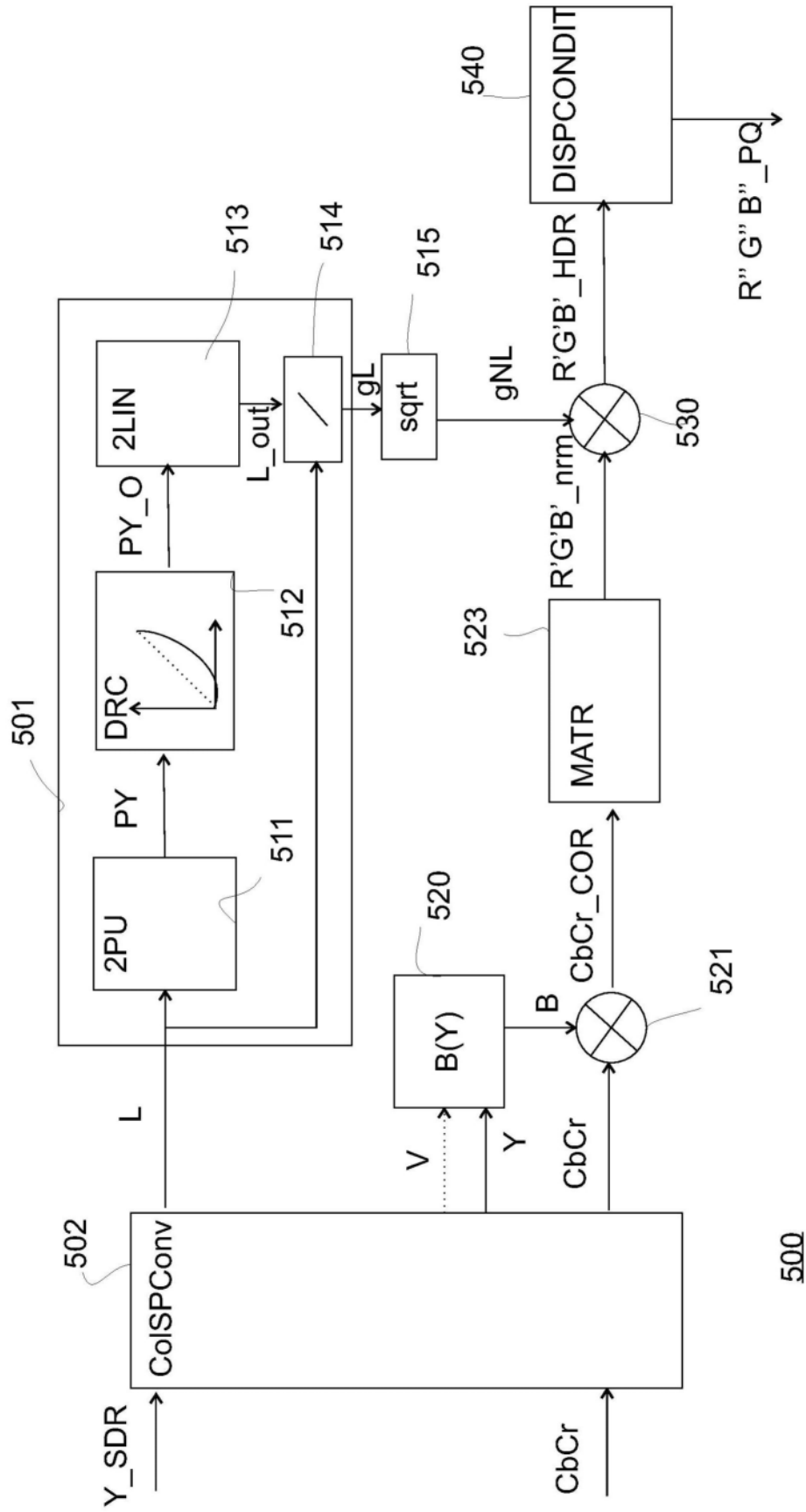


图5

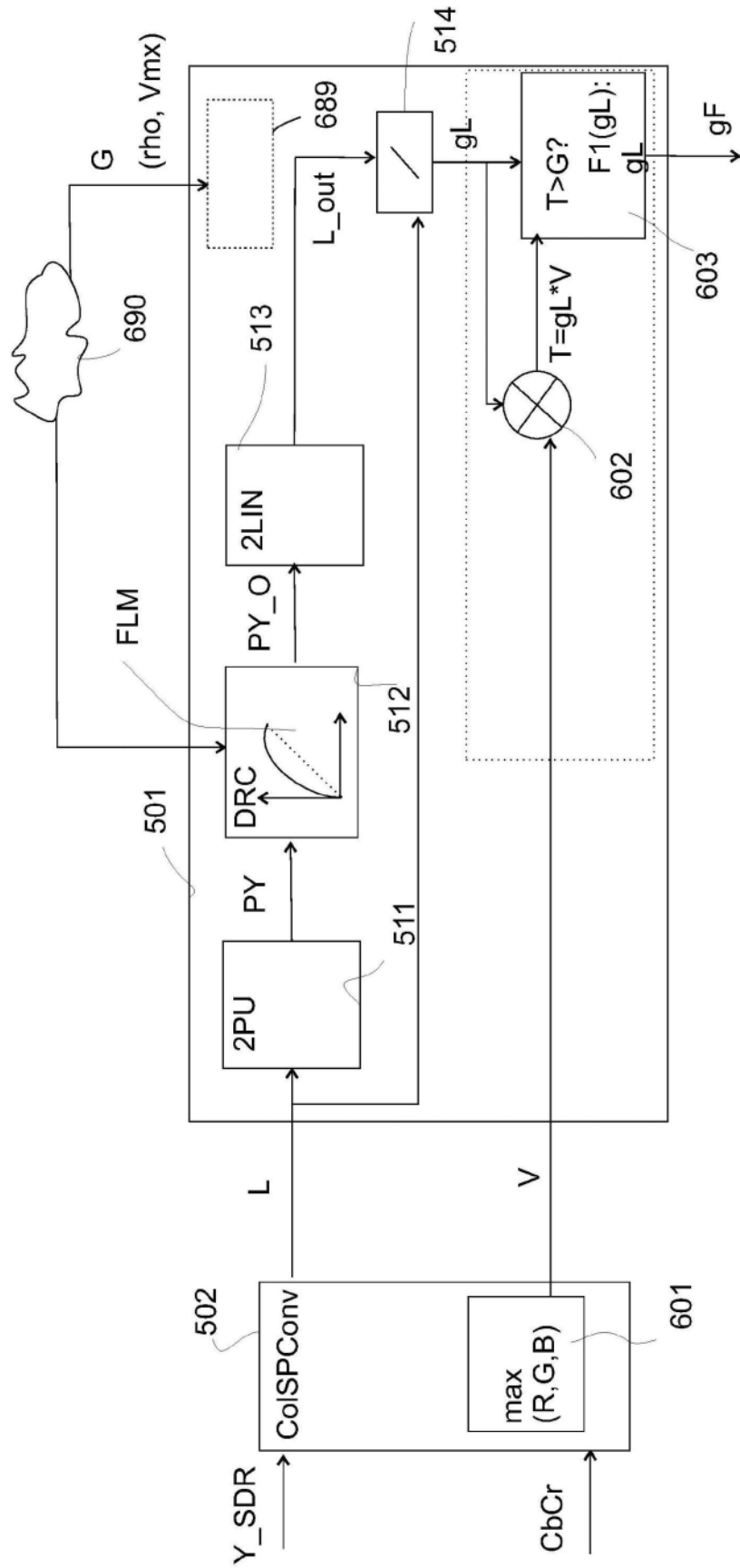


图6

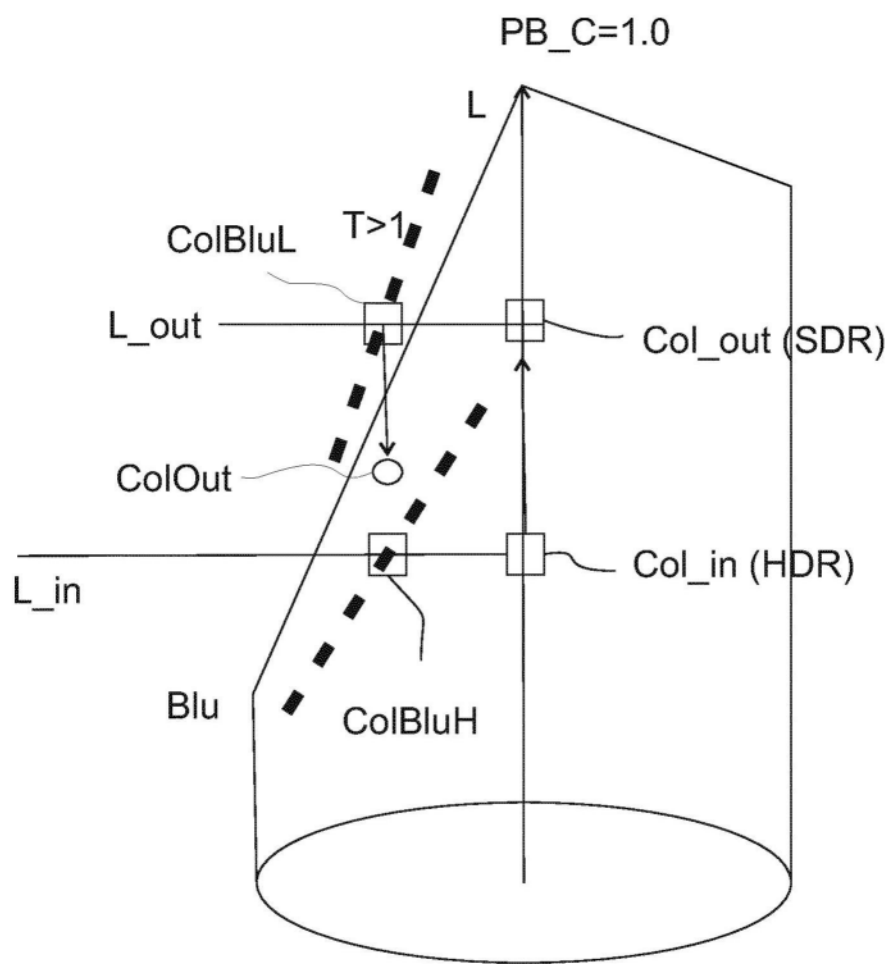


图7

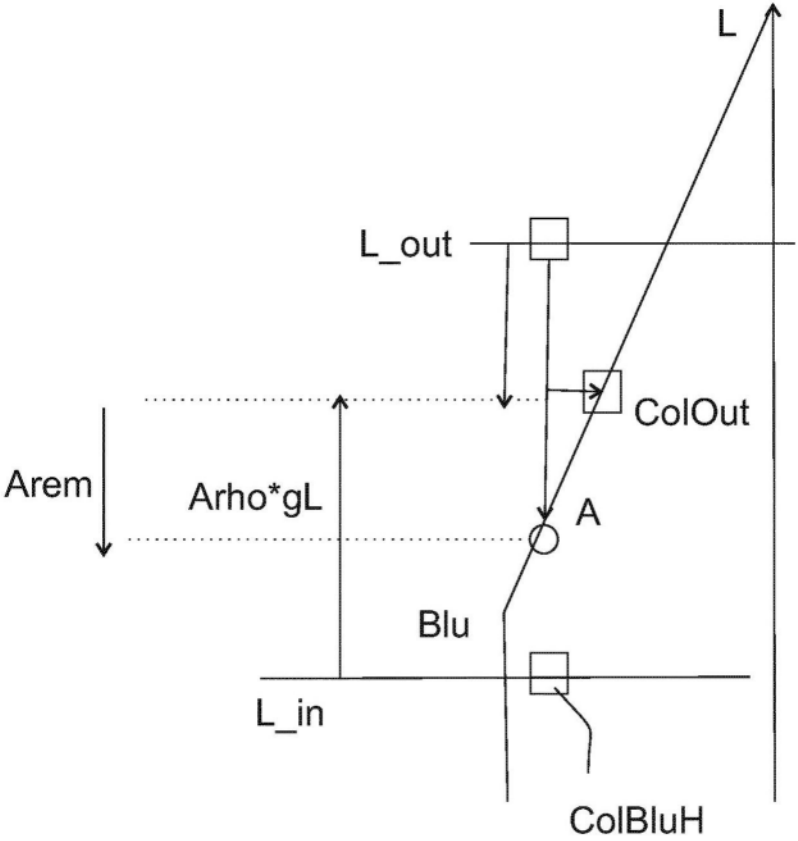


图8

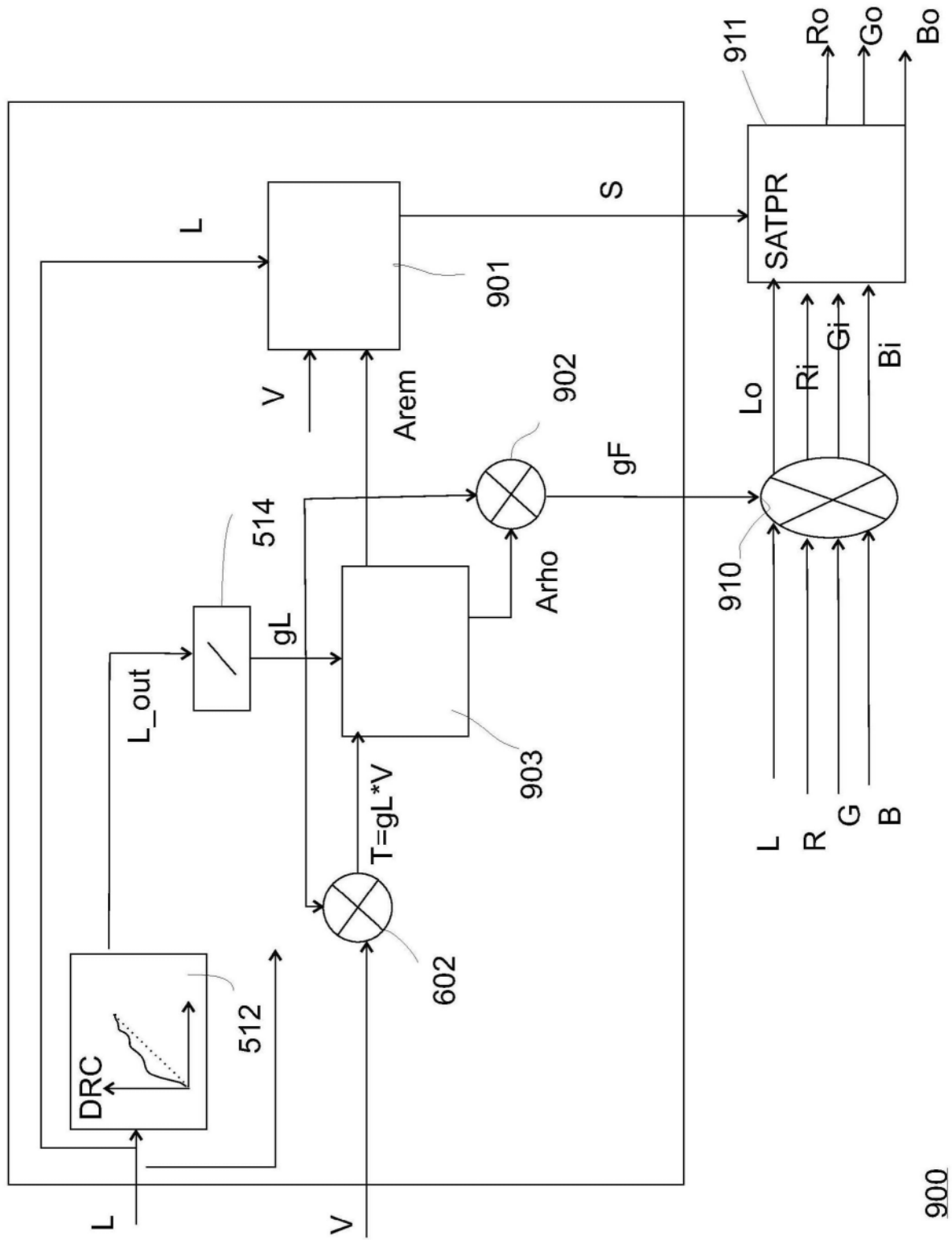


图9

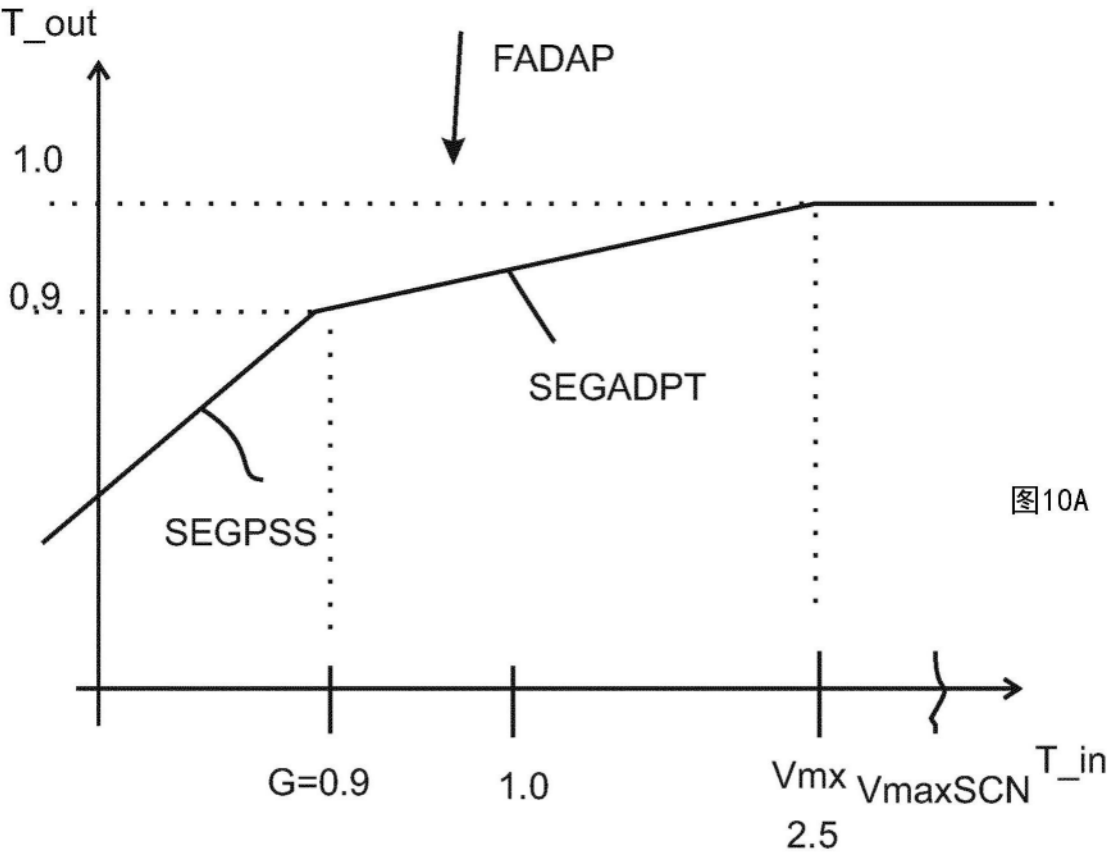


图10A

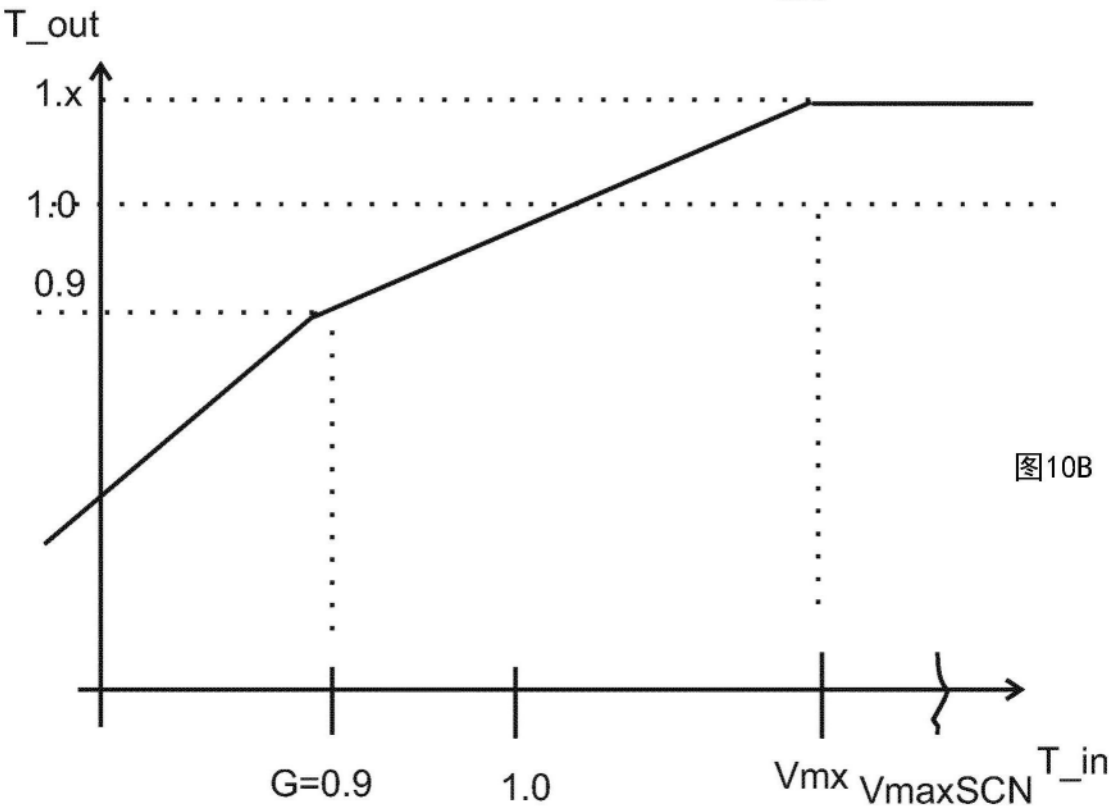


图10B

图10

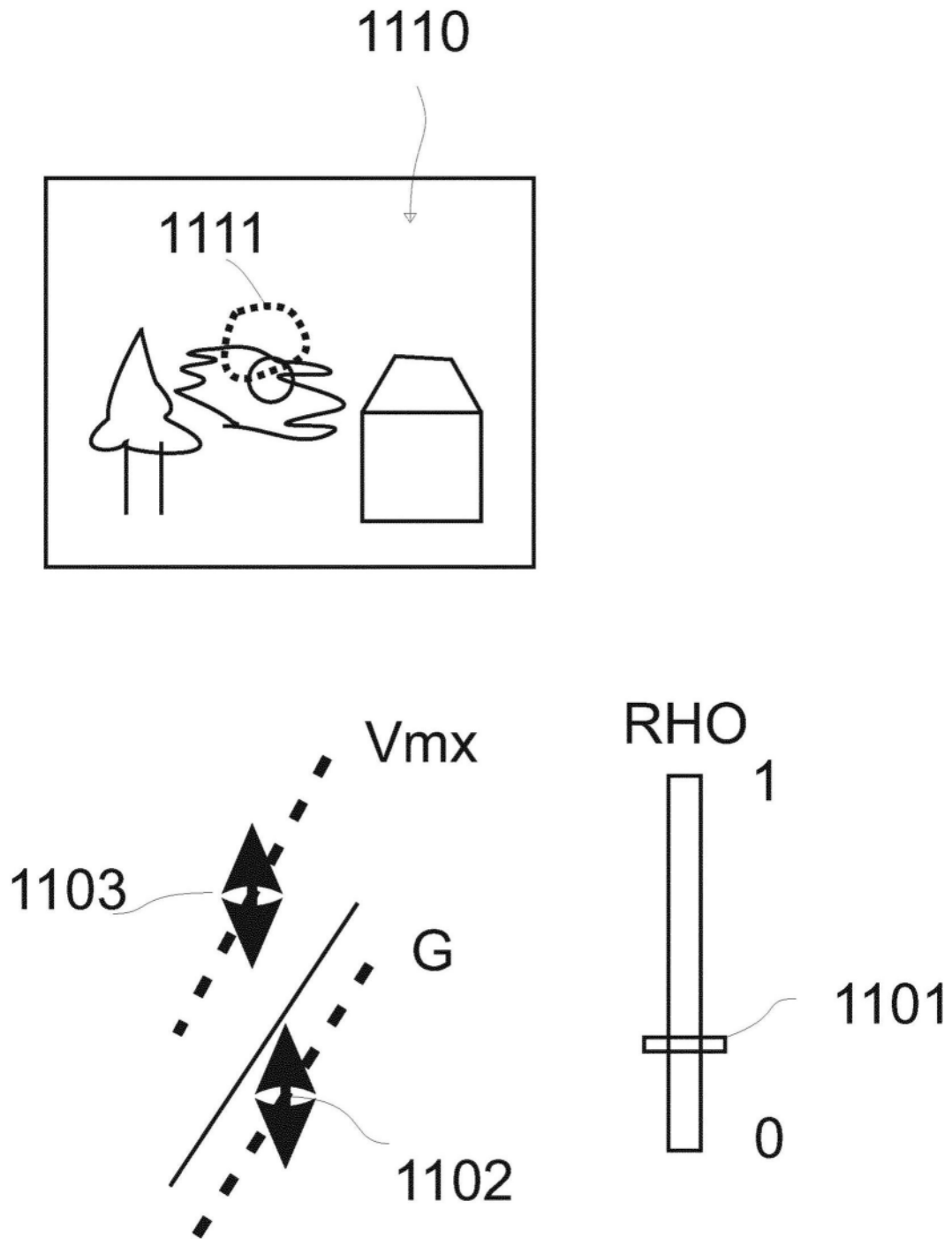


图11



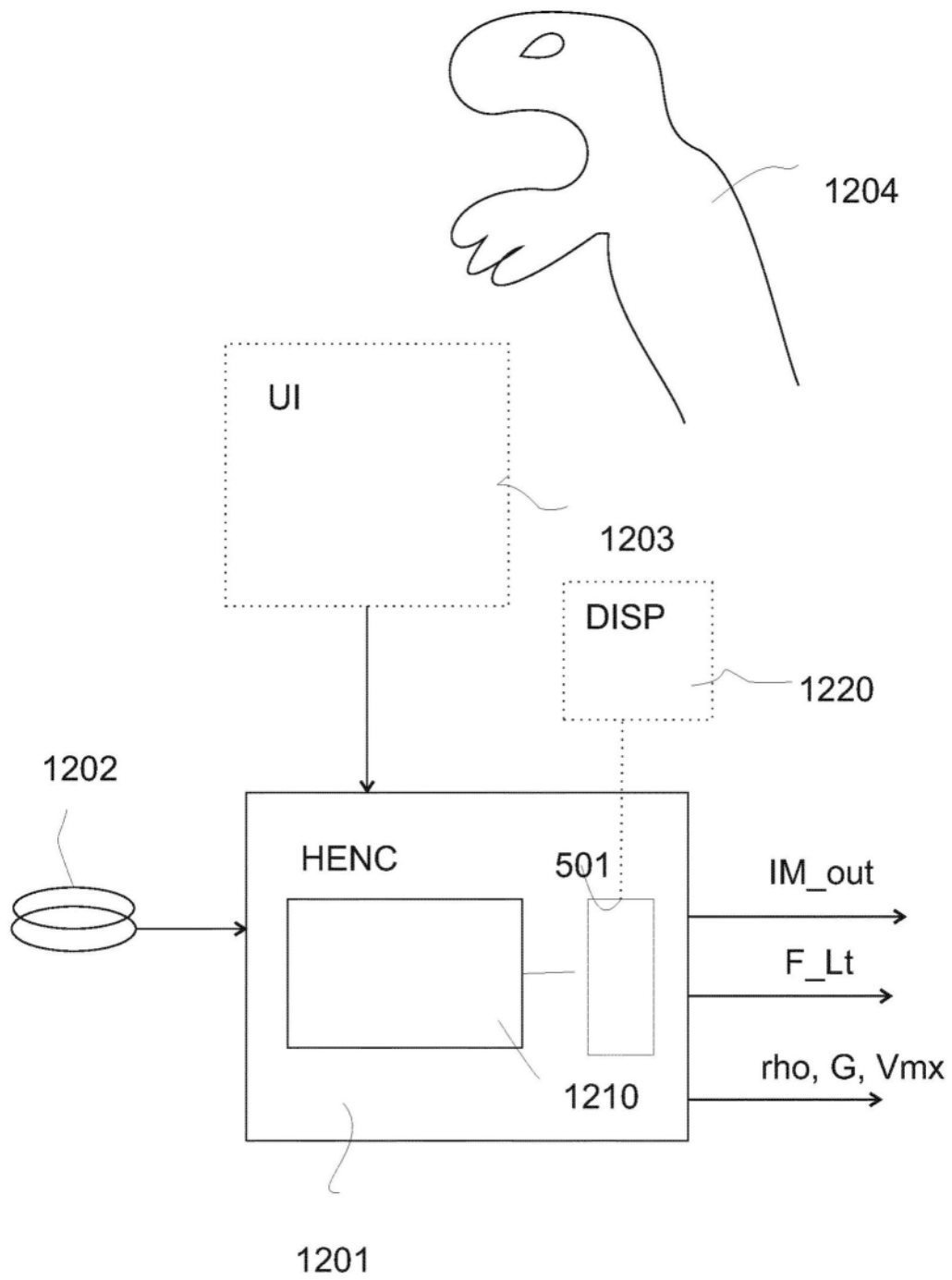


图12

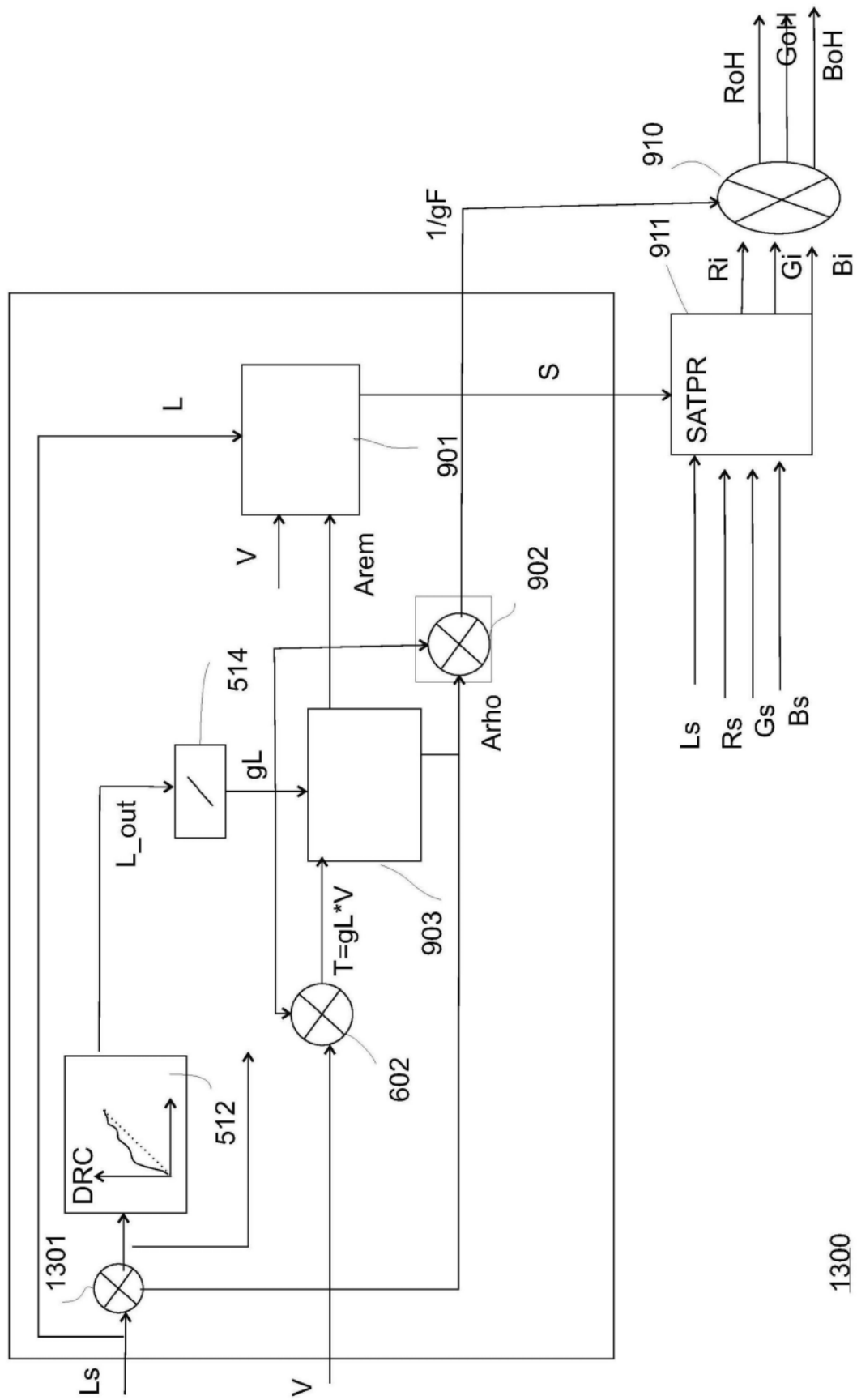


图13