



(12)发明专利

(10)授权公告号 CN 104240197 B

(45)授权公告日 2017. 12. 15

(21)申请号 201410424531.3

(22)申请日 2014.08.26

(65)同一申请的已公布的文献号

申请公布号 CN 104240197 A

(43)申请公布日 2014.12.24

(73)专利权人 浙江工商大学

地址 310018 浙江省杭州市下沙高教园
区学正街18号

(72)发明人 刘春晓 罗婷婷

(74)专利代理机构 杭州中成专利事务所有限公
司 33212

代理人 朱莹莹

(51)Int.Cl.

G06T 5/00(2006.01)

(56)对比文件

CN 101499167 A,2009.08.05,

US 7706606 B1,2010.04.27,

US 2010/0254601 A1,2010.10.07,

CN 101951523 A,2011.01.19,

王辉 等.改进的非线性全局映射灰度化方
法.《计算机辅助设计与图形学学报》.2013,第25
卷(第10期),1476-1479,1488.

Cewu Lu 等.Contrast Preserving
Decolorization.《Proceedings of IEEE
International Conference on Computational
Photography》.2012,1-7.

审查员 王爽

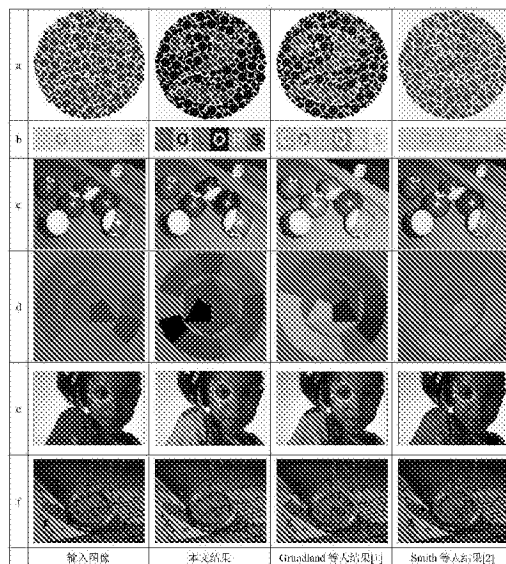
权利要求书3页 说明书9页 附图2页

(54)发明名称

一种保持对比度、颜色一致性和灰度像素特
征的消色方法

(57)摘要

本发明公开了一种保持对比度、颜色一致性和灰度像素特征的消色算法。我们通过双高斯模型构建像素对之间的对比度保持能量项，以此保持原始图像的结构和局部对比度信息；同时我们采用局部线性嵌入模型构建颜色一致性约束能量项，确保原始图像中颜色一致的像素在结果图像中也拥有一样的灰度级；为了保持灰度像素特征，我们先标记出原始图像中的灰度像素，并强制规定这些像素的灰度值是已知的且在消色变换的过程中始终等于它们的R值，然后用双高斯模型构建出灰度像素与其他像素之间的灰度特征保持能量项。我们线性结合这三个能量项，得到目标能量函数，然后再通过迭代法求解出使总能量值达到最小的灰度值，从而得到了最终的消色结果。



1. 一种保持对比度、颜色一致性和灰度像素特征的消色方法,其特征在於:包括如下步骤

(1) 读取一幅大小为 $M \times N$ 的彩色图像,并将彩色图像转换成双精度浮点型;

(2) 对整幅图像的每个像素,用公式(3)计算出其对应的颜色对比度;然后再用公式(1)计算出整幅图像的对比度保持能量项;

$$\delta_{xy} = \sqrt{(L_x - L_y)^2 + (a_x - a_y)^2 + (b_x - b_y)^2} / 100 \quad (3)$$

式(3)中,

x 为图像上的像素, y 为 x 的4领域相邻像素, $(x, y) \in NP$, NP 表示相邻像素集;

L , a 和 b 分别表示彩色图像在Lab颜色空间中的亮度通道和两个颜色通道;

L_x , a_x 和 b_x 分别表示像素 x 在Lab颜色空间上的三个颜色值;

L_y , a_y 和 b_y 分别表示像素 y 在Lab颜色空间上的三个颜色值;

$$E_1 = - \sum_{(x,y) \in NP} \ln \{ \alpha_{xy} G(\delta_{xy}, \sigma^2) + (1 - \alpha_{xy}) G(-\delta_{xy}, \sigma^2) \} \quad (1)$$

式(1)中, $G(\delta_{xy}, \sigma^2)$ 表示高斯函数,其具体公式如式(2)

$$G(\delta_{xy}, \sigma^2) = \exp \left\{ - \frac{|g_x - g_y - \delta_{xy}|^2}{\sigma^2} \right\} \quad (2)$$

其中 $\sigma \in [0, 1]$ 是一个用于控制平滑程度的变量, g_x 和 g_y 为待求的 x 和 y 像素的灰度值;

式(1)中, α_{xy} 表示像素 x 和 y 所对应的 δ_{xy} 的符号权值,其计算方法如公式(6)

$$\alpha_{xy} = \begin{cases} 1.0 & \{(R_x \leq R_y \ \& \ G_x \leq G_y \ \& \ B_x \leq B_y) \mid (R_x \geq R_y \ \& \ G_x \geq G_y \ \& \ B_x \geq B_y)\} \\ 0.5 & \text{否则} \end{cases} \quad (6)$$

其中, R_x , G_x 和 B_x 分别表示像素 x 在RGB颜色空间上的三个颜色值; R_y , G_y 和 B_y 分别表示像素 y 在RGB颜色空间上的三个颜色值;

如果 $\alpha_{xy} = 1$,那么像素对服从单高斯分布函数 $G(\delta_{xy}, \sigma^2)$ 或者 $G(-\delta_{xy}, \sigma^2)$,否则,我们令像素对之间的差异服从双高斯分布;

(3) 用LLE算法找出每个像素 x 所对应的 K 个最近邻 x_j , $j = 1, 2, L, K \dots$,然后用公式(8)计算出每个最近邻在拟合原像素时所占的权重值,再用公式(7)计算出图像对应的颜色一致性约束能量项;

$$\sum_{x=1}^N \left\| C_x - \sum_{j=1}^K w_{x_j} C_{x_j} \right\|^2 \quad (8)$$

其中, w_{x_j} 表示在重建像素 x 时第 j 个最近邻所占的权重,对于任意固定的像素 x 满足

$$\sum_{j=1:K} w_{x_j} = 1;$$

$$E_2 = - \sum_{x \in U} \ln \left\{ \exp \left\{ - \frac{|g_x - \sum_{j=1:K} w_{x_j} g_{x_j}|^2}{2\sigma^2} \right\} \right\} \quad (7)$$

U 表示整幅图像的所有像素, g_{x_j} 为待求的像素 x 的第 j 个最近邻 x_j 的灰度值;

(4) 在整幅图像中标识RGB颜色空间上的三个颜色值相等的灰度像素 t ,然后再用公式

(9) 计算出灰度像素与其他像素 x 之间的误差能量函数,即灰度像素特征能量项;

$$E_3 = -\sum_{x \in U} \left\{ \sum_{t \in P} \ln \{ \alpha_{xt} G(\delta_{xt}, \sigma^2) + (1 - \alpha_{xt}) G(-\delta_{xt}, \sigma^2) \} \right\} \quad (9)$$

其中,

P 表示图像中所有灰度像素的集合;

δ_{xt} 的计算公式(3-1),

$$\delta_{xt} = \sqrt{(L_x - L_t)^2 + (a_x - a_t)^2 + (b_x - b_t)^2} / 100 \quad (3-1)$$

区别在于公式(3)中的 L_y , a_y 和 b_y 分别被像素 t 在Lab颜色空间上的三个颜色值 L_t , a_t 和 b_t 所替代;

α_{xt} 的计算公式(6-1),

$$\alpha_{xt} = \begin{cases} 1.0 & \{(R_x \leq R_t \& G_x \leq G_t \& B_x \leq B_t) \mid (R_x \geq R_t \& G_x \geq G_t \& B_x \geq B_t)\} \\ 0.5 & \text{否则} \end{cases} \quad (6-1)$$

区别在于公式(6)中的 R_y , G_y 和 B_y 分别被像素 t 在RGB颜色空间上的三个颜色值 R_t , G_t 和 B_t 所替代;

$G(\delta_{xt}, \sigma^2)$ 的计算公式(2-1),

$$G(\delta_{xt}, \sigma^2) = \exp\left\{-\frac{|g_x - g_t - \delta_{xt}|^2}{\sigma^2}\right\} \quad (2-1)$$

区别在于公式(2)中的 g_t 表示灰度像素的灰度值,它是一个已知量,其值等于像素 t 所对应的 R_t 值;

(5) 最后将上面得到的三个能量项进行线性结合,得到公式(11)所示的总能量函数,并通过迭代求解法求出使能量值达到最小的图像灰度值,从而得到消色后的灰度图像

$$E = \lambda_1 E_1 + \lambda_2 E_2 + \lambda_3 E_3 \quad (11)$$

其中 E_1 表示对比度保持能量项, E_2 表示颜色一致性约束能量项, E_3 表示灰度像素特征能量项, λ_1 、 λ_2 和 λ_3 分别是三个能量项的权重系数。

2. 根据权利要求1所述的保持对比度、颜色一致性和灰度像素特征的消色方法,其特征在于:所述步骤(2)中,式(2)高斯函数在 δ_{xy} 处达到了最值,而 δ_{xy} 的大小不仅与像素对之间的对比度有关,也与像素对的灰度级符号权值 α_{xy} 相关;由于灰度级符号权值 α_{xy} 在公式中并没有准确的物理意义,因此我们通过调整它的正负来提高对比度保持约束的灵活性;

对于这种能够明确区分出次序的颜色对,我们采用公式(2)的单高斯函数;

根据式(4)定义判断条件判断哪些颜色对是可区分的:

$$R_x \leq R_y \& G_x \leq G_y \& B_x \leq B_y \quad (4)$$

其中, R_x , G_x 和 B_x 分别表示像素 x 在RGB颜色空间上的三个颜色值; R_y , G_y 和 B_y 分别表示像素 y 在RGB颜色空间上的三个颜色值;

若一对像素对 x, y 能满足公式(4)中的条件,则这对像素对的颜色次序能被明确区分,采用式(2)所示单高斯函数;此时 δ_{xy} 的符号直接取决于 $g_x - g_y$;

若一对像素对 x, y 不能满足公式(4)中的条件,则不明确指定 δ_{xy} 的符号,采用式(5)双高斯模型来自动寻找一个合适的颜色次序

$$\frac{1}{2} \{ G(\delta_{xy}, \sigma^2) + G(-\delta_{xy}, \sigma^2) \} \quad (5)$$

3. 根据权利要求1或2所述的保持对比度、颜色一致性和灰度像素特征的消色方法,其特征在于:所述步骤(3)在处理中,将 $\sum_{j=1}^K w_{x_j} g_{x_j}$ 当成是一个整体,因此, g_x 和 $\sum_{j=1}^K w_{x_j} g_{x_j}$ 的差异服从高斯分布函数;又由于K个最近邻能够很好地重建像素x,因此忽略了它们之间的颜色对比度差,无需考虑符号问题。

一种保持对比度、颜色一致性和灰度像素特征的消色方法

技术领域

[0001] 本发明属于计算机领域,涉及彩色图像灰度化算法设计技术领域,具体是指一种保持对比度、颜色一致性和灰度像素特征的消色方法。

背景技术

[0002] 近年来随着科学技术水平的不断发展,现如今的数字打印技术已经没办法满足人们对原图像保持程度的要求。虽然现在市场上出现了彩色打印技术,可以保持更多的细节信息,但彩色打印技术有需要很多的能耗,价格比较昂贵,在我们的日常生活中暂时还无法普及;同时目前很多的数码相机也根据了客户的需求设计了很多不同的风格,其中图像灰度化风格却损失了很多重要的细节信息,无法满足对图像要求严格的艺术研究人员的要求。针对这个问题,如果能够提出了新的图像灰度化算法,使结果图像能够尽可能多地保持原图像的对比度信息,可以使数字打印技术和图像风格化技术更加符合当前市场的需求。

[0003] 在计算机领域中,灰度图像是每个像素只有一个采样颜色的图像。这类图像通常显示为从最暗的黑色到最亮的白色的灰度,其灰度值的取值范围为[0,255]。而彩色图像包含R、G、B三个通道值,将彩色图像转换到灰度图像的任务是降低图像的维度,这样必然会损失一部分的信息,因而如要得到一个好的结果,必须尽可能多地保持原图像的对比度。

[0004] 而目前针对人类视觉感知系统的研究表明,人类视觉系统不能精确地感知到色调和亮度的变化,且每个人对亮度变化的敏感程度也不相同。同时由消色变换的原理可知,有限的灰度级是不可能一一对应地表示彩色空间中的每一种颜色,因此我们只能是尽可能多地保留视觉感知最敏感的对比度变化部分。又因为人眼对于图像相邻区域的颜色变化最为敏感,所以我们认为相邻像素之间的信息在灰度变换中起着重要的作用。而且当前的灰度化算法中少有考虑到灰度像素特征的问题,更没有一个算法能够同时很好地保持原始图像的颜色一致性,对比度信息和灰度像素特征。

[0005] 根据前面的缺陷和所得出来的结论,我们提出了保持对比度、颜色一致性和灰度像素特征的消色方法。与已有的算法相对比,该算法不仅能够更好地保持原图像的结构和局部对比度信息,也能够同时保持图像的颜色一致性以及灰度像素特征。

发明内容

[0006] 为克服以上问题,本发明的目的在于提供一种保持对比度、颜色一致性和灰度像素特征的消色方法。

[0007] 一种保持对比度、颜色一致性和灰度像素特征的消色方法,包括如下步骤

[0008] (1) 读取一幅大小为M×N的彩色图像,并将彩色图像转换成双精度浮点型;

[0009] (2) 对整幅图像的每个像素,用公式(3)计算出其对应的颜色对比度;然后再用公式(1)计算出整幅图像的对比度保持能量项;

$$[0010] \quad \delta_{xy} = \sqrt{(L_x - L_y)^2 + (a_x - a_y)^2 + (b_x - b_y)^2} / 100 \quad (3)$$

[0011] 式(3)中,

[0012] x 为图像上的像素, y 为 x 的4领域相邻像素, $(x,y) \in NP$, NP 表示相邻像素集;

[0013] L , a 和 b 分别表示彩色图像在Lab颜色空间中的亮度通道和两个颜色通道;

[0014] L_x , a_x 和 b_x 分别表示像素 x 在Lab颜色空间上的三个颜色值;

[0015] L_y , a_y 和 b_y 分别表示像素 y 在Lab颜色空间上的三个颜色值;

$$[0016] \quad E_1 = - \sum_{(x,y) \in NP} \ln \{ \alpha_{xy} G(\delta_{xy}, \sigma^2) + (1 - \alpha_{xy}) G(-\delta_{xy}, \sigma^2) \} \quad (1)$$

[0017] 式(1)中, $G(\delta_{xy}, \sigma^2)$ 表示高斯函数,其具体公式如式(2)

$$[0018] \quad G(\delta_{xy}, \sigma^2) = \exp \left\{ - \frac{|g_x - g_y - \delta_{xy}|^2}{\sigma^2} \right\} \quad (2)$$

[0019] 其中 $\sigma \in [0, 1]$ 是一个用于控制平滑程度的变量, g_x 和 g_y 为待求的 x 和 y 像素的灰度值;

[0020] 式(1)中, α_{xy} 表示像素 x 和 y 所对应的 δ_{xy} 的符号权值,其计算方法如公式(6)

[0021]

$$\alpha_{xy} = \begin{cases} 1.0 & \{(R_x \leq R_y \ \& \ G_x \leq G_y \ \& \ B_x \leq B_y) \mid (R_x \geq R_y \ \& \ G_x \geq G_y \ \& \ B_x \geq B_y)\} \\ 0.5 & \text{否则} \end{cases} \quad (6)$$

[0022] 其中, R_x , G_x 和 B_x 分别表示像素 x 在RGB颜色空间上的三个颜色值; R_y , G_y 和 B_y 分别表示像素 y 在RGB颜色空间上的三个颜色值;

[0023] 如果 $\alpha_{xy} = 1$,那么像素对服从单高斯分布函数 $G(\delta_{xy}, \sigma^2)$ 或者 $G(-\delta_{xy}, \sigma^2)$,否则,我们令像素对之间的差异服从双高斯分布;

[0024] (3)用LLE算法找出每个像素 x 所对应的 K 个最近邻 x_j , $j=1,2,L,K \dots$,然后用公式(8)计算出每个最近邻在拟合原像素时所占的权重值,再用公式(7)计算出图像对应的颜色一致性约束能量项;

$$[0025] \quad \sum_{x=1}^N \left\| C_x - \sum_{j=1}^K w_{x_j} C_{x_j} \right\|^2 \quad (8)$$

[0026] 其中, w_{x_j} 表示在重建像素 x 时第 j 个最近邻所占的权重,对于任意固定的像素 x 满足

$$\sum_{j=1:K} w_{x_j} = 1;$$

$$[0027] \quad E_2 = - \sum_{x \in U} \ln \left\{ \exp \left\{ - \frac{|g_x - \sum_{j=1:K} w_{x_j} g_{x_j}|^2}{2\sigma^2} \right\} \right\} \quad (7)$$

[0028] U 表示整幅图像的所有像素, g_{x_j} 为待求的像素 x 的第 j 个最近邻 x_j 的灰度值;

[0029] (4)在整幅图像中标识RGB颜色空间上的三个颜色值相等的灰度像素 t ,然后再用公式(9)计算出灰度像素与其他像素 x 之间的误差能量函数,即灰度像素特征能量项;

$$[0030] \quad E_3 = - \sum_{x \in U} \left\{ \sum_{t \in P} \ln \{ \alpha_{xt} G(\delta_{xt}, \sigma^2) + (1 - \alpha_{xt}) G(-\delta_{xt}, \sigma^2) \} \right\} \quad (9)$$

[0031] 其中,

[0032] P 表示图像中所有灰度像素的集合;

[0033] δ_{xt} 的计算公式(3-1),

$$[0034] \quad \delta_{xt} = \sqrt{(L_x - L_t)^2 + (a_x - a_t)^2 + (b_x - b_t)^2} / 100 \quad (3-1)$$

[0035] 区别在于公式(3)中的 L_y, a_y 和 b_y 分别被像素 t 在Lab颜色空间上的三个颜色值 L_t, a_t 和 b_t 所替代;

[0036] α_{xt} 的计算公式(6-1),

$$[0037] \quad \alpha_{xt} = \begin{cases} 1.0 & \{(R_x \leq R_t \& G_x \leq G_t \& B_x \leq B_t) \mid (R_x \geq R_t \& G_x \geq G_t \& B_x \geq B_t)\} \\ 0.5 & \text{否则} \end{cases} \quad (6-1)$$

[0038] 区别在于公式(6)中的 R_y, G_y 和 B_y 分别被像素 t 在RGB颜色空间上的三个颜色值 R_t, G_t 和 B_t 所替代;

[0039] $G(\delta_{xt}, \sigma^2)$ 的计算公式(2-1),

$$[0040] \quad G(\delta_{xt}, \sigma^2) = \exp\left\{-\frac{|g_x - g_t - \delta_{xt}|^2}{\sigma^2}\right\} \quad (2-1)$$

[0041] 区别在于公式(2)中的 g_t 表示灰度像素的灰度值,它是一个已知量,其值等于像素 t 所对应的 R_t 值;

[0042] (5)最后将上面得到的三个能量项进行线性结合,得到公式(11)所示的总能量函数,并通过迭代求解法求出使能量值达到最小的图像灰度值,从而得到消色后的灰度图像

$$[0043] \quad E = \lambda_1 E_1 + \lambda_2 E_2 + \lambda_3 E_3 \quad (11)$$

[0044] 其中 E_1 表示对比度保持能量项, E_2 表示颜色一致性约束能量项, E_3 表示灰度像素特征能量项, λ_1, λ_2 和 λ_3 分别是三个能量项的权重系数。

[0045] 优选的,所述步骤(2)中,式(2)高斯函数在 δ_{xy} 处达到了最值,而 δ_{xy} 的大小不仅与像素对之间的对比度有关,也与像素对的灰度级符号权值 α_{xy} 相关;由于灰度级符号权值 α_{xy} 在公式中并没有准确的物理意义,因此我们通过调整它的正负来提高对比度保持约束的灵活性;

[0046] 对于这种能够明确区分出次序的颜色对,我们采用公式(2)的单高斯函数;

[0047] 根据式(4)定义判断条件判断哪些颜色对是可区分的:

$$[0048] \quad R_x \leq R_y \& G_x \leq G_y \& B_x \leq B_y \quad (4)$$

[0049] 其中, R_x, G_x 和 B_x 分别表示像素 x 在RGB颜色空间上的三个颜色值; R_y, G_y 和 B_y 分别表示像素 y 在RGB颜色空间上的三个颜色值;

[0050] 若一对像素对 x, y 能满足公式(4)中的条件,则这对像素对的颜色次序能被明确区分,采用式(2)所示单高斯函数;此时 δ_{xy} 的符号直接取决于 $g_x - g_y$;

[0051] 若一对像素对 x, y 不能满足公式(4)中的条件,则不明确指定 δ_{xy} 的符号,采用式(5)双高斯模型来自动寻找一个合适的颜色次序

$$[0052] \quad \frac{1}{2} \{G(\delta_{xy}, \sigma^2) + G(-\delta_{xy}, \sigma^2)\} \quad (5)$$

[0053] 优选的,所述步骤(3)在处理中,将 $\sum_{j=1}^K w_{x_j} g_{x_j}$ 当成是一个整体,因此, g_x 和 $\sum_{j=1}^K w_{x_j} g_{x_j}$ 的差异服从高斯分布函数;又由于 K 个最近邻能够很好地重建像素 x ,因此忽略了它们之间的颜色对比度差,无需考虑符号问题。

[0054] 下面对本发明做进一步的描述,本发明解决其技术问题采用的技术方案如下:

[0055] 1) 对比度保持能量项

[0056] 为了保持原始图像的局部和全局对比度,我们通过双高斯模型构建了如下所示的能量函数:

$$[0057] \quad E_1 = - \sum_{(x,y) \in N} \ln \{ \alpha_{xy} G(\delta_{xy}, \sigma^2) + (1 - \alpha_{xy}) G(-\delta_{xy}, \sigma^2) \} \quad (1)$$

[0058] 在我们的方法中,我们将图像的每个像素点都当成是一个变量。对每个像素 x ,我们首先找出它的相邻像素 y , $(x, y) \in N$, N 表示相邻像素集;然后根据双高斯模型创建对应的能量函数。这里的 $G(\delta_{xy}, \sigma^2)$ 表示高斯函数,其具体公式如下

$$[0059] \quad G(\delta_{xy}, \sigma^2) \propto \exp \left\{ - \frac{|g_x - g_y - \delta_{xy}|^2}{\sigma^2} \right\} \quad (2)$$

[0060] 其中 $\sigma \in [0, 1]$ 是一个用于控制平滑程度的变量; δ_{xy} 表示颜色对比度,其具体表达式如下所示:

$$[0061] \quad \delta_{xy} = \sqrt{(L_x - L_y)^2 + (a_x - a_y)^2 + (b_x - b_y)^2} / 100 \quad (3)$$

[0062] L , a 和 b 分别表示彩色图像在Lab空间中的亮度通道和两个颜色通道。

[0063] 由等式(2)可得,高斯函数在 δ_{xy} 处达到了最值,而 δ_{xy} 的大小不仅与像素对之间的对比度有关,也与像素对的灰度级符号权值 α_{xy} 相关。由于灰度级符号权值在公式中并没有准确的物理意义,因此我们通过调整它的正负来提高对比度保持约束的灵活性。下面将详细说明如何选取该符号。

[0064] 在彩色图像中,我们很容易辨别出颜色对中哪种颜色的亮度更强一些。比如,在正常人看来,任何其他颜色都比黑色亮一些。对于这种能够明确区分出次序的颜色对,我们采用单高斯函数,如公式(2)。而为了更好地判断哪些颜色对是可区分的,我们如下定义判断条件:

$$[0065] \quad c_x \leq c_y \Leftrightarrow r_x \leq r_y, g_x \leq g_y, b_x \leq b_y \quad (4)$$

[0066] 一对像素对 x, y 如果能满足公式(4)中的条件,我们便认为这对像素对的颜色次序能被明确区分,此时 δ_{xy} 的符号直接取决于 $g_x - g_y$ 。否则我们不明确指定 δ_{xy} 的符号,而是采用双高斯模型来自动寻找一个合适的颜色次序。其中双高斯模型如下所示:

$$[0067] \quad \frac{1}{2} \{ G(\delta_{xy}, \sigma^2) + G(-\delta_{xy}, \sigma^2) \} \quad (5)$$

[0068] 根据前面定义的颜色次序判断方法,我们将判断条件函数化:

$$[0069] \quad \alpha_{xy} = \begin{cases} 1.0 & \text{if } r_x \leq r_y, g_x \leq g_y, b_x \leq b_y \\ & \text{or } r_y \leq r_x, g_y \leq g_x, b_y \leq b_x \\ 0.5 & \text{otherwise} \end{cases} \quad (6)$$

[0070] 如果 $\alpha_{xy} = 1$,那么像素对采用单高斯函数 $G(\delta_{xy}, \sigma^2)$ 或者 $G(-\delta_{xy}, \sigma^2)$ 。否则,我们令颜色对之间的差异服从双高斯分布,如公式(5),以此来自动地选择符号。

[0071] 2) 颜色一致性约束项

[0072] 为了保持图像的颜色一致性特征,即颜色相同的像素在灰度变换后拥有一样的灰度值,我们构建了如下所示的能量项:

$$[0073] \quad E_2 = -\sum_{x \in U} \ln \left\{ \exp \left\{ -\frac{\left| g_x - \sum_{j=1:K} w_{xj} g_j \right|^2}{2\sigma^2} \right\} \right\} \quad (7)$$

[0074] U 表示整幅图像的所有像素； w_{xj} 表示在重建像素 x 时，第 j 个最近邻所占的权重，其值可通过局部线性嵌入(LLE)算法计算得到。

[0075] LLE算法是一种将一组高维数据投影到低维空间上的非线性降维方法，它能够使降维后的数据较好地保持原有的流型结构。假设给定一组数据集 X_1, \dots, X_N ，其中 X_x 表示像素 x 的一组特征向量(在我们的方法中，用的是R、G、B三个特征值)。对每个 X_x ，我们找出它的 K 个最近邻，即 X_{x1}, \dots, X_{xk} 。然后，通过最小化下面的能量函数计算出一组能够很好地重建出 X_x 的权重 w_{xj} 。

$$[0076] \quad \sum_{x=1}^N \left\| X_x - \sum_{j=1}^K w_{xj} X_{xj} \right\|^2 \quad (8)$$

[0077] 权重 w_{xj} 还需要满足一个约束条件，即 $\sum_{j=1:K} w_{xj} = 1$ 。根据Roweis和Saul所描述的方法，可计算出权重矩阵 W (w_{xj} 表示 W 的第 xj 个元素)，该矩阵包含了数据集在特征空间中的流型信息。在我们的消色方法中，我们希望保持这种流型结构，以此保证图像在消色后能够保持颜色一致性特征。但要保持这种流型结构，就必须使得 $g_x = \sum_{j=1}^K w_{xj} g_{xj}$ ， g_x 表示像素 x 的消色结果。在处理中，我们将 $\sum_{j=1}^K w_{xj} g_{xj}$ 当成是一个整体，因此， g_x 和 $\sum_{j=1}^K w_{xj} g_{xj}$ 的差异服从高斯分布函数。又由于 K 个最近邻能够很好地重建像素 x ，所以我们忽略了它们之间的颜色对比度差，无需考虑符号问题。根据这些因素，我们如公式(7)所示创建能量项 E_2 。

[0078] 3) 灰度特征保持能量项

[0079] 在彩色图像中存在一些RGB三个值相等的像素，这种像素我们称为灰度像素。理论上，在进行灰度变换时，灰度像素的灰度值固定为R值。但若我们的目标函数中仅包含 E_1 和 E_2 项，会导致灰度像素的灰度值在图像消色的过程中发生改变。因此，我们定义了灰度特征保持能量项。

[0080] 首先，我们从原始图像的所有像素中标记出灰度像素；然后，将这些灰度像素看作是已知量，并构建能量函数。但如果我们仅是将这些像素作为已知像素点进行计算的话，由于灰度像素和其他像素在数量级上的不同，会导致灰度像素在结果图像中特别地凸显出来。因此，为了使结果图像能够比较和谐，同时能够保持我们所需要的对比度，在创建能量函数时，我们将灰度像素作为一个硬性的约束条件，使得其他像素的灰度值能根据灰度像素的灰度值进行调整。其能量函数如下所示：

$$[0081] \quad E_3 = -\sum_{x \in U} \left\{ \sum_{t \in P} \ln \left\{ \alpha_{xt} G(\delta_{xt}, \sigma^2) + (1 - \alpha_{xt}) G(-\delta_{xt}, \sigma^2) \right\} \right\} \quad (9)$$

[0082] 该能量项表示每个像素均需与所有的灰度像素进行一次能量计算。其中 P 表示图像中所有灰度像素的集合； α_{xt} 表示像素 x 所对应的 δ_{xt} 的符号权值，其计算方法与公式(6)一样。 $G(\delta_{xt}, \sigma^2)$ 表示像素 x 和灰度像素 t 之间的差异所服从的高斯函数，其公式如下所示：

$$[0083] \quad G(\delta_{xt}, \sigma^2) \propto \exp \left\{ -\frac{\left| g_x - g_t - \delta_{xt} \right|^2}{\sigma^2} \right\} \quad (10)$$

[0084] 其中 g_t 表示灰度像素，是一个已知量，其值等于像素 t 所对应的R值。

[0085] 将上述的三个能量进行线性结合,得到如下所示的总能量函数

$$[0086] \quad E = \lambda_1 E_1 + \lambda_2 E_2 + \lambda_3 E_3 \quad (11)$$

[0087] 其中 E_1 表示对比度保持能量项, E_2 表示颜色一致性约束项, E_3 表示灰度特征保持能量项。在建立总的目标函数后,我们通过迭代法求出使能量值达到最小的图像灰度值,从而得到灰度图像。

[0088] 本发明不仅能够更好地保持原图像的结构和局部对比度信息,也能够同时保持图像的颜色一致性以及灰度像素特征。

[0089] 本发明公开了一种保持对比度、颜色一致性和灰度像素特征的消色方法。我们算法的核心是创建和求解一个目标能量函数,该能量函数主要包含三个能量项——对比度保持能量项,颜色一致性约束项和灰度特征保持能量项。在我们的算法中,我们通过双高斯模型构建像素对之间的对比度保持能量项,以此保持原始图像的结构和局部对比度信息;同时我们采用局部线性嵌入模型构建颜色一致性约束能量项,确保原始图像中颜色一致的像素在结果图像中也拥有一样的灰度级;为了保持灰度像素特征,我们先标记出原始图像中的灰度像素(即R、G、B三个值相等的像素),并强制规定这些像素的灰度值是已知的且在消色变换的过程中始终等于它们的R值,然后用双高斯模型构建出灰度像素与其他像素之间的灰度特征保持能量项。我们线性结合这三个能量项,得到目标能量函数,然后再通过迭代法求解出使总能量值达到最小的灰度值,从而得到了最终的消色结果。通过我们的方法得到的图像能够同时保持原始图像的对比度,颜色一致性和灰度像素特征。

附图说明

[0090] 图1为对6组不同图像分别按本发明的算法与Grundland和Dodgson的算法^[1]和Smith等人的算法^[2]的对比图。

[0091] 图2为本发明的算法与Grundland和Dodgson的算法^[1]和Lu等人的算法^[3]的对比图及其细节放大对比图;(a)原始的彩色图像,(b)本发明算法,(c)Grundland and Dodgson的算法[1],(d)Lu等人[3]的算法,(e-h)是(a-d)中白色框所选择区域的放大效果。

[0092] 附图中的原图像为彩色图像,由于专利申请说明书限制文章中不能出现彩色图像,因此,在这里我们直接通过matalab中的rgb2gray函数将原图像灰度化后的结果作为原图像显示。

具体实施方式

[0093] 下面结合具体实施例进一步阐述本发明,应理解,以下实施例仅用于说明本发明而并不用于限制本发明的保护范围。

[0094] 实施例

[0095] 1.先读取一幅大小为 $M \times N$ 的彩色图像,由于图像本身的数据是整型的,而我们的计算是针对浮点型数据的,因此我们首先要将其转换成双精度浮点型;

[0096] 2.对整幅图像的每个像素,用公式(3)计算出其对应的颜色对比度;然后再用公式(1)计算出整幅图像的对比度保持能量项;

$$[0097] \quad \delta_{xy} = \sqrt{(L_x - L_g)^2 + (a_x - a_g)^2 + (b_x - b_g)^2} / 100 \quad (3)$$

[0098] 式(3)中,

[0099] x 为图像上的像素, y 为 x 的4领域相邻像素, $(x,y) \in NP$, NP 表示相邻像素集;

[0100] L , a 和 b 分别表示彩色图像在Lab颜色空间中的亮度通道和两个颜色通道;

[0101] L_x , a_x 和 b_x 分别表示像素 x 在Lab颜色空间上的三个颜色值;

[0102] L_y , a_y 和 b_y 分别表示像素 y 在Lab颜色空间上的三个颜色值;

$$[0103] \quad E_1 = - \sum_{(x,y) \in N} \ln \{ \alpha_{xy} G(\delta_{xy}, \sigma^2) + (1 - \alpha_{xy}) G(-\delta_{xy}, \sigma^2) \} \quad (1)$$

[0104] 式(1)中, $G(\delta_{xy}, \sigma^2)$ 表示高斯函数,其具体公式如式(2)

$$[0105] \quad G(\delta_{xy}, \sigma^2) \propto \exp \left\{ - \frac{|g_x - g_y - \delta_{xy}|^2}{\sigma^2} \right\} \quad (2)$$

[0106] 其中 $\sigma \in [0, 1]$ 是一个用于控制平滑程度的变量, g_x 和 g_y 为待求的 x 和 y 像素的灰度值;

[0107] 式(1)中, α_{xy} 表示像素 x 和 y 所对应的 δ_{xy} 的符号权值,其计算方法如公式(6)

$$[0108] \quad \alpha_{xy} = \begin{cases} 1.0 & \{(R_x \leq R_y \ \& \ G_x \leq G_y \ \& \ B_x \leq B_y) \mid (R_x \geq R_y \ \& \ G_x \geq G_y \ \& \ B_x \geq B_y)\} \\ 0.5 & \text{否则} \end{cases} \quad (6)$$

[0109] 其中, R_x , G_x 和 B_x 分别表示像素 x 在RGB颜色空间上的三个颜色值; R_y , G_y 和 B_y 分别表示像素 y 在RGB颜色空间上的三个颜色值;

[0110] 如果 $\alpha_{xy} = 1$,那么像素对服从单高斯分布函数 $G(\delta_{xy}, \sigma^2)$ 或者 $G(-\delta_{xy}, \sigma^2)$,否则,我们令像素对之间的差异服从双高斯分布;

[0111] 3.用LLE算法找出每个像素所对应的 K 个最近邻 x_j , $j=1, 2, \dots, K$,然后用公式(8)计算出每个最近邻在拟合原像素时所占的权重值,再用等式(7)计算出图像对应的颜色一致性约束能量项;

$$[0112] \quad \sum_{x=1}^N \left\| X_x - \sum_{j=1}^K w_{xj} X_{xj} \right\|^2 \quad (8)$$

[0113] 其中, w_{xj} 表示在重建像素 x 时,第 j 个最近邻所占的权重,权重 w_{xj} 满足

$$[0114] \quad \sum_{j=1:K} w_{xj} = 1;$$

$$[0115] \quad E_2 = - \sum_{x \in U} \ln \left\{ \exp \left\{ - \frac{|g_x - \sum_{j=1:K} w_{xj} g_j|^2}{2\sigma^2} \right\} \right\} \quad (7)$$

[0116] U 表示整幅图像的所有像素, g_{xj} 为待求的像素 x 的第 j 个最近邻 x_j 的灰度值;

[0117] 4.在整幅图像中标识RGB三个值相等的像素,然后再用公式(9)计算出灰度像素与非灰度像素之间的误差能量函数,即灰度像素特征能量项;

$$[0118] \quad E_3 = - \sum_{x \in U} \left\{ \sum_{t \in P} \ln \{ \alpha_{xt} G(\delta_{xt}, \sigma^2) + (1 - \alpha_{xt}) G(-\delta_{xt}, \sigma^2) \} \right\} \quad (9)$$

[0119] 其中,

[0120] P 表示图像中所有灰度像素的集合;

[0121] δ_{xt} 的计算公式(3-1),

$$[0122] \quad \delta_{xt} = \sqrt{(L_x - L_t)^2 + (a_x - a_t)^2 + (b_x - b_t)^2} / 100 \quad (3-1)$$

[0123] 区别在于公式(3)中的 L_y, a_y 和 b_y 分别被像素 t 在Lab颜色空间上的三个颜色值 L_t, a_t 和 b_t 所替代;

[0124] α_{xt} 的计算公式(6-1),

$$[0125] \quad \alpha_{xt} = \begin{cases} 1.0 & \{(R_x \leq R_t \& G_x \leq G_t \& B_x \leq B_t) \mid (R_x \geq R_t \& G_x \geq G_t \& B_x \geq B_t)\} \\ 0.5 & \text{否则} \end{cases} \quad (6-1)$$

[0126] 区别在于公式(6)中的 R_y, G_y 和 B_y 分别被像素 t 在RGB颜色空间上的三个颜色值 R_t, G_t 和 B_t 所替代;

[0127] $G(\delta_{xt}, \sigma^2)$ 的计算公式(2-1),

$$[0128] \quad G(\delta_{xt}, \sigma^2) = \exp\left\{-\frac{|g_x - g_t - \delta_{xt}|^2}{\sigma^2}\right\} \quad (2-1)$$

[0129] 区别在于公式(2)中的 g_t 表示灰度像素的灰度值,它是一个已知量,其值等于像素 t 所对应的 R_t 值。

[0130] 5.最后将上面得到的三个能量项进行线性结合,得到公式(11)所示的总能量函数,并通过迭代求解法求出使能量值达到最小的图像灰度值,从而得到消色后的灰度图像

$$[0131] \quad E = \lambda_1 E_1 + \lambda_2 E_2 + \lambda_3 E_3 \quad (11)$$

[0132] 其中 E_1 表示对比度保持能量项, E_2 表示颜色一致性约束项, E_3 表示灰度特征保持能量项, λ_1, λ_2 和 λ_3 分别是三个能量项的权重系数。

[0133] 为了更直接地凸显出本算法的优越性,我们将算法的结果与三种算法的结果进行比较,分别是经过Čadik^[4]的试验得出的两种性能较好地算法—Grundland和Dodgson的算法^[1]和Smith等人的算法^[2],以及近年来在彩色图像灰度化领域中做的较好的Lu等人的算法^[3]。

[0134] 图1展示了6组不同图像的对比图,从图中我们可以发现Smith等人的结果^[2]损失了大部分的局部信息。而Grundland和Dodgson^[1]虽然能保持较多的局部对比度信息,但是它具有一定的局限性。该方法适用于像(e)、(f)这种内容比较丰富的图像,而对于(a)、(b)、(c)、(d)这种内容单一、颜色变化明显的图像,其局部对比度保持的效果并不理想。我们的方法不仅适用于内容丰富的图像也适用于内容单一的图像,而且能够同时很好地保持原始图像的全局结构和局部对比度信息。

[0135] 对于保持颜色一致性,我们算法使原图中颜色一样的像素在变换后得到相同的灰度值,当然这一特征在另外两种方法中也能体现。图(a)、(b)的灰度像素部分便是图中白色边缘部分,在我们的结果中,这些灰度像素的灰度值始终等于它们原本的R值,而另外两种算法的结果看起来虽然也是白色的,但它们的灰度值实际上已经发生了变化,因此他们的算法并没有保持原始图像的灰度像素特征。

[0136] 图2是细节对比图,在原始彩色图像(a)中我们可以清楚地看到蓝色衣服上的褶皱。若用Grundland和Dodgson^[1]和Lu等人^[3]的算法进行消色变换,我们得到如图(c)和(d)所示的结果,这种褶皱会变平滑,说明这两种方法会损失重要的局部细节信息。相对比而言,我们的方法能够清晰地保持这些细节信息,如图(b)所示。图2:(a)原始的彩色图像,(b)本文算法,(c)Grundland and Dodgson的算法^[1],(d)Lu等人^[3]的算法,(e-h)是(a-d)中白色框所选择区域的放大效果。比起Grundland and Dodgson的算法^[1]和Lu等人的算法^[3],我们的算法能够更好地保持更多细节信息。

[0137] 参考文献:

[0138] [1] 克卢兰. 马克, 道奇森. 尼尔. 安东尼. 消色: 快速, 对比度增强的彩色到灰度的转换[J]. 模式识别, 2007, 40 (11): 2891-2896.

[0139] M.Grundland, N.A.Dodgson. Decolorize: Fast, contrast enhancing, color to grayscale conversion[J]. Pattern Recognition, 2007, 40 (11): 2891-2896.

[0140] [2] 史密斯. 凯勒居, 兰德斯. 皮埃尔, 特隆特. 居尔勒, 麦斯科维斯基. 卡罗尔. 显著灰度化: 一个简单且快速的感知准确图像视频转换方法[J].

[0141] 计算机图形学论坛, 2008, 27 (2): 193-200.

[0142] K.Smith, P.Landes, J.Thollot, and K.Myszkowski. Apparent greyscale: A simple and fast conversion to perceptually accurate images and video[J].

[0143] Computer Graphics Forum, 2008, 27 (2): 193-200.

[0144] [3] 陆策武, 徐立, 贾佳亚. 对比度保持的消色算法[C]. 美国电气和电子工程师协会计算摄影学国际会议论文集, 2012, 1-7.

[0145] Cewu Lu, Li Xu and Jiaya Jia. Contrast Preserving Decolorization[C]. Proceedings of IEEE International Conference on Computational Photography, 2012, 1-7.

[0146] [4] 卡蒂克. 马丁. 彩色到灰度图像转换的感知评价[J]. 计算机图形学论坛, 2008, 27 (7): 1745-1754.

[0147] M. Čadík. Perceptual Evaluation of Color-to-Grayscale Image Conversions [J]. Computer Graphics Forum, 2008, 27 (7): 1745-1754.

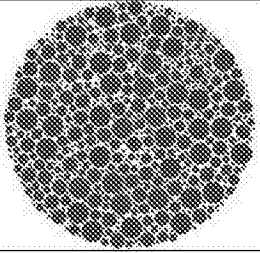
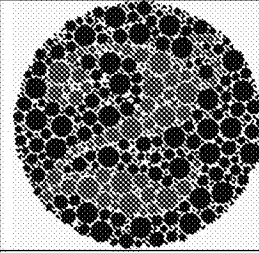
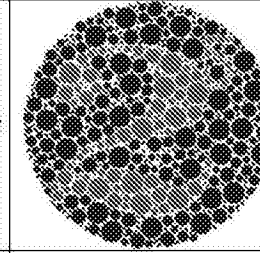
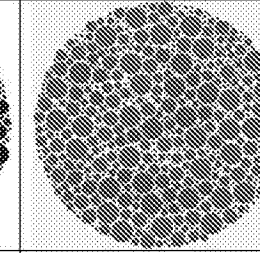
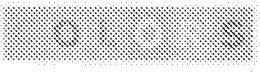


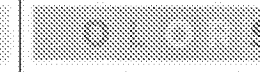
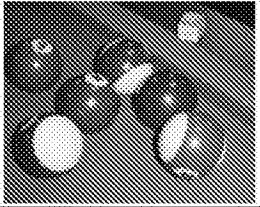
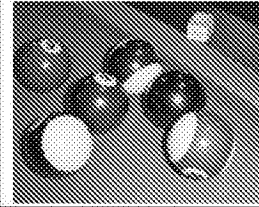
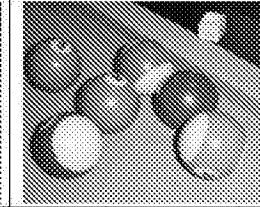
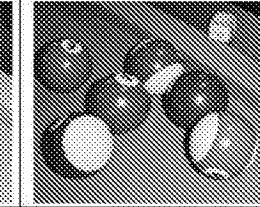
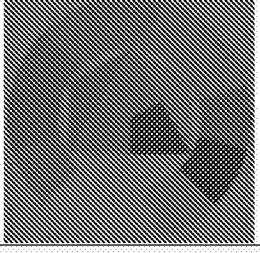
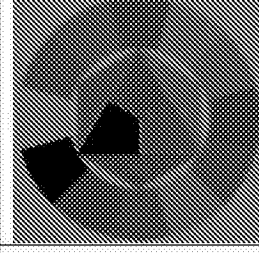
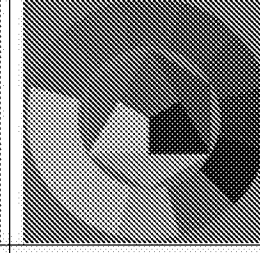
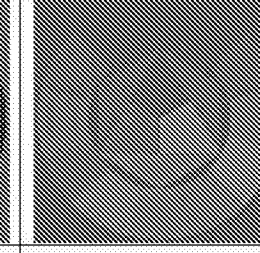
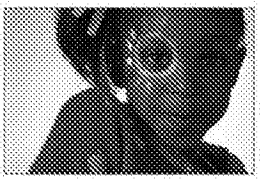
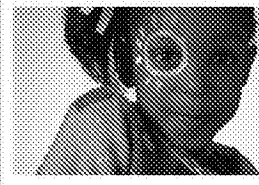
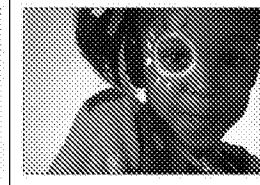
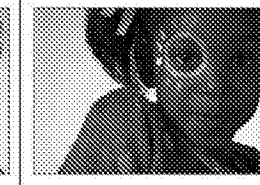
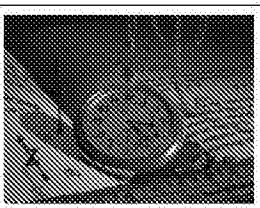
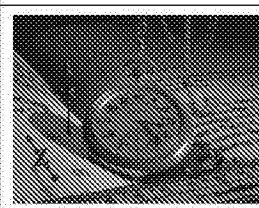
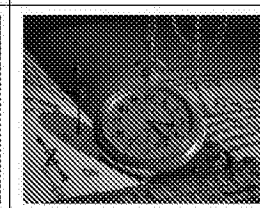
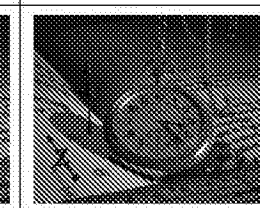
a				
b				
c				
d				
e				
f				
	输入图像	本文结果	Grundland 等人结果[1]	Smith 等人结果[2]

图1





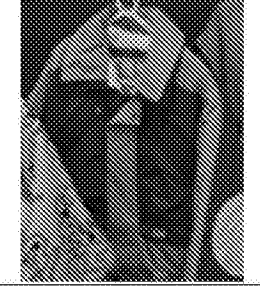
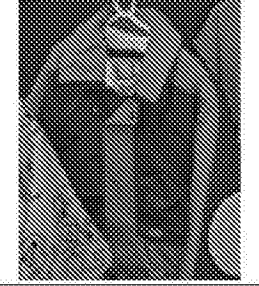
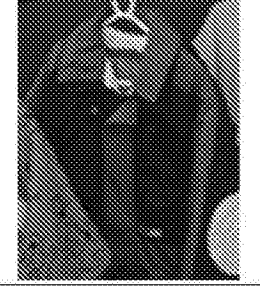
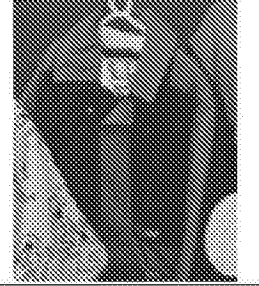
输入图像	本文结果	Grundland 等人结果[1]	Lu 等人结果[3]
			
a	b	c	d
			
e	f	g	h

图2