一种基于拉普拉斯方程的图像彩色化方法

滕升华 谌安军 邹谋炎
[1](中国科学院电子学研究所,北京 100080) [2](中国科学院研究生院,北京 100049)

摘 要 黑白图像的彩色化是图像处理领域一个活跃的、有挑战性的研究课题。在总结现有图像彩色化方法的基础上,分析了彩色化由局部向全局扩展的本质,进而提出了一种通过求解拉普拉斯方程实现颜色扩展的彩色化方法。为提高边界定位能力,设计了加权形式的差分方案。与同是采用偏微分方程的 Saprio 彩色化方法相比,该方法与其控制颜色扩展的机制不同,图像彩色化效果要明显优于 Saprio 方法。

关键词 黑白图像 彩色化 拉普拉斯方程
中图法分类号: TP391 **文献标识码**: A **文章编号**: 1006-8961(2006)04-0545-04

Colorization Using Laplace Equation

TENG Sheng-hua, CHEN An-jun, ZOU Mou-yan
[1](*Institute of Electronics, Chinese Academy of Sciences, Beijing* 100080)
[2](*Graduate School of the Chinese Academy of Sciences, Beijing* 100049)

Abstract Colorization of grayscale image is in general an active and challenging area of research in image processing. In this paper, current colorization algorithms are summarized and the principle of color propagation is analyzed. Then a colorization algorithm based on Laplace equation is proposed. In order to determine the boundary accurately, a weighted difference scheme is applied. In contrast with Saprio's colorization algorithm which is also a partial differential equation based method, the principle of color propagation in our algorithm is different. Furthermore, our algorithm can achieve more vivid colorized result.

Keywords grayscale image, colorization, Laplace equation

1 引 言

彩色化是给黑白图像、电影或电视节目加上颜色的处理过程,一般认为是由 Wilson Markle 在 1970 年发明的,最初用于处理阿波罗登月计划获取的月球影像[1]。20 世纪 80 年代,国外的一些黑白老电影经过彩色化处理被重新搬上了荧幕,从而面目一新。尽管从艺术的角度来看人们对老电影的彩色化加工还存在一定争议,但时至今日,彩色化研究仍然是图像处理学界一个活跃的、有挑战性的课题,更多地应用于图像、视频编辑和图像通信等领域[2]。

2 图像彩色化的研究

这里所说的彩色化处理要求给黑白图像赋予自然的、接近于真实的色彩,严格地讲是一种假彩色处理[3]。与医学、遥感等领域广泛应用的伪彩色处理的不同之处在于,假彩色化力求再现影像场景的本来面目,而不仅仅是为了使图像内容更加醒目,后者可以完全不顾忌图像中真实的颜色。

早期电影的彩色化处理,一般是要依靠专门的画师对特定的帧图像进行手工着色,这种手工制作方式不仅对操作者自身水平要求较高,而且效率低下。近年来在图像彩色化方面出现了一些半自动的

收稿日期:2005-04-28;**改回日期**:2005-06-30

第一作者简介:滕升华(1977~),男。现为中国科学院电子学研究所信号与信息处理专业博士研究生。主要研究方向为图像、视频处理技术。E-mail: tengshua@163.com

处理方法，大致可归结为两类，即基于颜色转移的彩色化和局部彩色化向全局扩展的方法。

2.1 基于颜色转移的彩色化

图像彩色化的一大进步首先源于 Welsh 等人的工作[4]，其做法是为待处理的黑白图像设定一幅用来取色的彩色参考图像，然后根据亮度或纹理对应关系将彩色图像中的颜色转移到黑白图像中。尽管文献[4]给出了一些效果不错的彩色化实例，但是这类方法不能保证处理后图像颜色在空间上的连续性，即本来相邻的、颜色相近的区域可能由于亮度差异而被分配了截然不同的颜色。同时，这种方法的彩色化结果强烈依赖于所采用的参考图像，这使得参考图像的选择成了一个新的难题，由此带来的另一个问题是对彩色化结果的事先控制和局部调整都非常不便，而这一点往往又是很重要的。

2.2 局部彩色化的全局扩展

图像彩色化是一个“欠约束”的病态问题，问题的逐步解决是先由手工对图像局部进行粗略的彩色化，再结合图像的几何特征将局部的彩色化扩展到整幅图像。

基于这种考虑，Horiuchi 假定图像颜色服从局部的 Markov 性，即图像中一点的颜色仅与其邻近像素有关，进而设定一些彩色化的“种子像素”，把图像中所有像素与其邻近像素颜色差异的总和作为最小化目标，将图像彩色化转换为一个极值问题[5]。

从类似的前提出发，Levin 等人的做法是先由人工在图像中各个区域涂上适当的彩色线条，颜色扩展过程要使所有像素与其邻域颜色加权差的平方和最小[6]。Levin 算法相对于 Horiuchi 算法的重要进步是在求颜色差值的过程中采用了加权形式，这实际上考虑了颜色扩展中像素间的亮度关系，即相邻像素只有亮度相近时才能分配相近的颜色；权值的采用有助于物体边界的正确定位。

Sapiro 将图像修补(Inpainting)的思想引入到图像彩色化中[2]，将颜色作为修补对象，处理方法是在图像上涂上一些彩色线条，颜色根据由亮度的梯度决定的引导场扩展到整幅图像。Sapiro 率先将偏微分方程法应用到图像彩色化中，但其方法依然不能很好地解决物体边界的定位问题。

相对于第 1 类方法，颜色由局部向全局扩展的彩色化方法操作简便，而且处理效果更好一些。本文着眼于第 2 类处理方法，根据颜色的局部相似性，在由人工对图像粗略涂色之后，以拉普拉斯方程求解的形式实现颜色的全局扩展。

3 算法描述与数值解法

3.1 算法原理

一般来说，实际图像中除去物体边界，邻近像素间的颜色不会发生剧变，这意味着实际图像中颜色梯度的 L_2 范数在平坦区域内应趋于零。事先在各区域粗略地涂上颜色，在保持涂色像素颜色不变、整幅图像颜色梯度的 L_2 范数最小的前提下，不同颜色的彩色线条在一定范围内扩展，最后在物体边界汇合。

基于上述考虑，设计求解最小化问题 $\min J(C)$

$$J(C)=\iint_{\Omega}\|\nabla C\|^2 \tag{1}$$

式中，C 表示图像的色度(如 YUV 颜色空间的 U、V)，$\nabla=\left[\frac{\partial}{\partial x},\frac{\partial}{\partial y}\right]$ 是梯度算子，Ω 代表图像的支持域。利用变分法，最小化 $J(C)$ 的问题转化为求解 Euler-Lagrange 方程，带有 Dirichlet 边界条件

$$\Delta C=0 \tag{2}$$

$$C|_{\partial\Omega}=C^*|_{\partial\Omega} \tag{3}$$

其中，$\Delta=\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}$ 是拉普拉斯算子，$\partial\Omega$ 表示涂色区域，C^* 是涂色区域的颜色。这样，图像彩色化问题转化为对拉普拉斯方程的求解问题。

3.2 数值解法

拉普拉斯方程对应边界值问题，为求解引入一个人工时间将其改写成扩散方程

$$\frac{\partial C}{\partial t}=\Delta C \tag{4}$$

当 $t\to\infty$ 时，C 得到一个平衡解，这个平衡解对时间的导数为零，因此即是方程式(2)的解。

利用前向中心差分格式，方程式(4)演化为

$$C_{i,j}^{(n+1)}=C_{i,j}^{(n)}+\frac{\delta t}{(\delta x)^2}(C_{i+1,j}^{(n)}+C_{i-1,j}^{(n)}+C_{i,j+1}^{(n)}+C_{i,j-1}^{(n)}-4C_{i,j}^{(n)}) \tag{5}$$

其中，δx、δt 分别是空间和时间上的差分步长，C 的上标表示迭代次数。根据稳定性准则[7]，选取允许的最大时间步长 $\delta t=(\delta x)^2/4$，方程式(5)变为

$$C_{i,j}^{(n+1)}=\frac{1}{4}(C_{i+1,j}^{(n)}+C_{i-1,j}^{(n)}+C_{i,j+1}^{(n)}+C_{i,j-1}^{(n)}) \tag{6}$$

方程式(6)实际上是一种经典的方程迭代算法即雅可比方法，该方法因收敛较慢而并不实用，但它

是多种迭代算法的基础。上面的推导是意在说明如何将方程式(2)的抽象形式应用到数字图像中,实际处理中的后续迭代采用逐次超松弛法[7]。

对方程式(6),一旦右端 C 的新值可用即用它替换上一次迭代的值,即

$$C_{i,j}^{(n+1)}=\frac{1}{4}(C_{i+1,j}^{(n)}+C_{i-1,j}^{(n+1)}+C_{i,j+1}^{(n)}+C_{i,j-1}^{(n+1)}) \quad (7)$$

上式称为 Gauss-Seidel 迭代。方程式(7)的右端加减 $C_{i,j}^{(n)}$,或直接由式(5)变换得到

$$C_{i,j}^{(n+1)}=C_{i,j}^{(n)}+\frac{1}{4}(C_{i+1,j}^{(n)}+C_{i-1,j}^{(n+1)}+C_{i,j+1}^{(n)}+C_{i,j-1}^{(n+1)}-4C_{i,j}^{(n)}) \quad (8)$$

对方程式(8)进行过度修正,将修正项乘上一个超松弛参数 ω 得到逐次超松弛法的表达形式

$$C_{i,j}^{(n+1)}=C_{i,j}^{(n)}+\frac{\omega}{4}(C_{i+1,j}^{(n)}+C_{i-1,j}^{(n+1)}+C_{i,j+1}^{(n)}+C_{i,j-1}^{(n+1)}-4C_{i,j}^{(n)}) \quad (9)$$

要使方程式(9)收敛,超松弛参数 ω 取值范围是 $0<\omega<2$;取 $1<\omega<2$ 时可使收敛速度快于 Gauss-Seidel 迭代。

4　实验结果与加权差分方案

4.1　实验结果

采用上述方法进行图像彩色化的例子如图版 I 图 1 所示。可见,图版 I 图 1(a)中所涂的线条能够在各自区域内合理地扩展,不同颜色在边界处自然汇合。

由于同样是采用偏微分方程的处理方法,本文重点同 Saprio 方法进行比较,图版 I 图 1(c)给出了利用后者对图版 I 图 1(a)的处理结果。Saprio 方法处理多种颜色扩展的能力明显不足,在物体边界处颜色偏差很大。

4.2　加权差分方案

尽管采用本文提出的方法进行图像彩色化能够取得不错的效果,但是在物体边界处容易出现颜色偏差,这一点在图版 I 图 2(b)中更为明显。

考察 2.2 节所述几种彩色化方法,Horiuchi 算法存在边界定位不准的问题,而 Levin 算法通过对邻域像素分配一个决定于亮度的权值较好地解决了这个问题;类似权值先见于图像分割算法[8]。权值的选取原则是邻域像素与中心点亮度差异大则分配小权值,反之亦然。这样,与中心像素亮度相差大的点对中心像素的颜色分配影响较小,这就保证了邻域内的像素只有亮度相近才分配相近的颜色;即颜色扩展不仅考虑像素间的位置关系,而且要顾及相邻像素间的亮度关系。

针对同样问题,此处引入类似权值,定义加权形式的拉普拉斯算子

$$\left.\begin{aligned}\Delta_w C&=\sum_{m=-1}^{1}\sum_{n=-1}^{1}(w_{i+m,j+n}C_{i+m,j+n}-C_{i,j})\\ w_{i+m,j+n}&\propto e^{-(Y_{i,j}-Y_{i+m,j+n})^2/2\sigma_{i,j}^2}\\ \sum_{m=-1}^{1}\sum_{n=-1}^{1}w_{i+m,j+n}&=8\end{aligned}\right\} \quad (10)$$

其中,$Y_{i,j}$ 表示该像素的亮度,$\sigma_{i,j}^2$ 表示邻域内像素的亮度方差,$m,n\in\boldsymbol{Z}$,m,n 不能同时取 0。

采用以式(10)为基础的加权差分形式求解拉普拉斯方程,对图版 I 图 2(a)实施彩色化的结果如图版 I 图 2(c)所示,边界处的颜色偏差明显改善。作为对比,图版 I 图 2(d)给出了利用 Saprio 算法的处理结果。

4.3　结果分析

从物理意义上讲,本文的方法基于颜色的局部相似性,加权差分使得颜色扩展保证亮度相近的相邻像素颜色相近;从这一点上来说本文的思想更类似于 Levin 算法。Saprio 方法试图为颜色扩展寻求一个引导场,在其处理过程中假定亮度和颜色梯度一致,从而利用亮度的梯度场引导颜色的扩展;亮度和颜色梯度一致的结论有待严格的证明,而且从文献[2]及本文处理结果来看,该方法仅仅是对单一颜色的扩展较为有效。

从方程形式上来看,采用拉普拉斯方程,较 Saprio 采用的泊松方程更为简洁,更重要的是处理效果方面明显优于 Saprio 方法。

5　结　论

偏微分方程法应用到图像处理领域是图像处理发展史上中的一大进步,与随机场建模、小波分析等分析方法一起促成了图像处理技术从“技巧”到“科学”的渐变。本文在总结分析图像彩色化现有研究成果的基础上,抓住第 2 类彩色化方法颜色扩展的本质,以拉普拉斯方程求解的方式实现了整幅图像的彩色化,采用加权差分方案强化了边界定位能力。

尽管迄今涌现了多种图像彩色化方法,但各种方法都有其应用上的局限性,突出问题是在物体边

界附近容易出现着色偏差,因此图像彩色化问题的完全解决还有待于更多更深入的研究。

参考文献(References)

1 Burns G. Colorization[EB/OL]. http://www.museum.tv/archives/etv/C/htmlC/colorization/colorization.htm.

2 Sapiro G. Inpainting the Colors[EB/OL]. http://www.ima.umn.edu/preprints/may2004/1979.pdf, 2004-05.

3 Ruan Q Q. Digital Image Processing[M]. Beijing: Publishing House of Electronics Industry, 2001: 212 ~ 214.[阮秋琦著. 数字图像处理学[M]. 北京:电子工业出版社, 2001:212 ~ 214.]

4 Welsh T, Ashikhmin M, Mueller K. Transferring color to grayscale images[A]. In: Proceedings of ACM SIGGRAPH[C], San Antonio, USA, 2002:277 ~ 280.

5 Takahiko Horiuchi. Colorization algorithm using probabilistic relaxation[J]. Image and Vision Computing, 2004, **22**(3): 197 ~ 202.

6 Levin A, Lischinski D, Weiss Y. Colorization using optimization[A]. In: Proceedings of ACM SIGGRAPH[C], Los Angeles, USA, 2004:689 ~ 693.

7 William P, Saul T, William V, *et al*. Numerical Recipes in C: The Art of Scientific Computing[M]. New York, USA: Cambridge University Press, 1992:863 ~ 870.

8 Shi Jian-bo, Malik J. Normalized cuts and image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, **22**(8): 888 ~ 905.

(a) 涂上彩色线条的黑白图像 (b) 本文方法的处理结果 (c) Saprio方法的彩色化结果

图1 彩色化实例及与Saprio方法比较

Fig.1 Colorization example and comparison with Saprio's method

(a) 涂上彩色线条的黑白图像[6] (b) 本文方法的处理结果(简单差分形式)

(c) 本文方法的处理结果(加权差分形式) (d) Saprio方法的彩色化结果

图2 基于简单/加权差分形式及Saprio方法处理的结果

Fig.2 Comparison between our method with/without weighted difference and Saprio's method