

基于抠像技术的图像无缝融合算法

傅新元 郭禾 王宇新 刘天阳 李寒

(大连理工大学计算机系, 大连 116024)

摘要 为了更快更好的进行图像融合,在用自由融合算法对泊松图像编辑与抠像技术进行改进的基础上,实现了一套图像自由无缝融合算法。该融合算法首先利用抠像技术来提取图像中的物体边缘,然后迭代求解泊松方程,从而取得了自然的融合效果。该算法与最新的无缝融合算法相比,其优点在于:①对于前景与背景的颜色变化复杂的图像,仍可得到准确的边缘;②可以使用户能引导前景映射图的走向;③无论图像前景层中存在多少个洞,融合效果将不受任何影响。这使得该算法的应用具有相当的灵活性。对于目前的图像融合技术而言,该算法可以获得更高的融合质量、更快的处理速度以及多样的融合效果。

关键词 抠像技术 泊松方程 梯度域 自由融合

中图分类号:TP391.41 文献标识码:A 文章编号:1006-8961(2008)06-1082-08

Seamless Cloning Algorithm Based on Image Matting Technique

FU Xin-yuan, GUO He, WANG Yu-xin, LIU Tian-yang, LI Han

(Department of Computer Science and Technology, Dalian University of Technology, Dalian 116024)

Abstract In this paper, we present a method called arbitrary cloning in which image matting technique and Poisson image editing are applied to accomplish the image seamless cloning. First, the object boundary is extracted through image matting technique. And Poisson equation is solved to achieve natural result for seamless cloning. Compared with state-of-art techniques, the algorithm shows advantages that great performance can be reached in bad conditions when there is complicated color variation in the foreground and background image. The α matte can be guided by the user input and the algorithm is not affected even when there are many holes in the foreground image. The algorithm is very flexible in application. Within current cloning technique, the algorithm guarantees better output, higher speed and diverse results.

Keywords image matting, Poisson equation, gradient field, arbitrary cloning

1 引言

图像融合技术与抠像技术是图像处理领域里两个不同的分支。其中图像无缝融合技术在图像编辑领域处于核心地位,它的目的是先将目标物体从一幅源图像中分离出来,然后将目标物体嵌入到目标图像中。图像的无缝融合技术的发展比较缓慢,现存的主流技术有以下3种:一种是Adobe Photoshop中使用的技术(尚未公开),此外还有两种公开算法。其中一种是通过构造拉普拉

斯金字塔进行插值运算来融合两幅图像;另一种是文献[1]提出的图像融合解决方案——泊松图像编辑技术,其已经成为了近几年诸多融合算法的基础,它是通过解带有狄利克雷边界条件的泊松方程来实现颜色的无缝融合。该融合方法是先使用拉普拉斯有限差分公式把泊松方程离散化,再利用高斯塞德尔迭代求解其最优值。这样不同的初始值、迭代次数以及引导梯度域就可以产生不同的融合效果。虽然泊松编辑的方法拥有简洁的表达式,且便于理解和计算,但是,这种算法的最大弱点是它需依赖于用户手动来提取图像边缘,

收稿日期:2002-09-14; 改回日期:2006-11-10

第一作者简介:傅新元(1982~),男,2007年获大连理工大学计算机系硕士学位。研究方向为图像处理、计算机视觉。E-mail: fuxinyuan@gmail.com; xifu@microsoft.com

这要求用户即便对于很复杂的图像也要仔细地画出边缘,从而给用户带来了极大的不便。

文献[2]同时得到了高质量的边缘提取与无缝融合效果。它是先采用最短路径算法来寻找一条最优的物体边缘,再利用混合梯度域来优化边缘效果,最后利用泊松方程来实现融合。实验表明,这种算法可以得到非常理想的融合效果,它的最大优点是只需要用户在目标物体周围粗略地画一条封闭的曲线,再把这个区域拖拽到目标图像中即可。但是这种算法的缺点是:①对于前景和背景颜色接近,或前景与背景的颜色变化都很复杂混乱的情况,则可能会产生不准确的边界;②用户不能按照个人意愿,灵活地控制边界的走向;③如果原图像中间有很多洞,则会大大增加程序的复杂度与运行时间,因为对于每一个洞都需要重复进行边缘提取的所有步骤。

在抠像技术方面,近年来诸多学者提出了各种各样的技术来实现精确的抠像。其中主流的抠像技术一般是从一张图片中,根据额外的用户约束来解析出前景层 F 、背景层 B 和前景映射图 α ^[3]。另外,还有通过多张图像或者视频片段进行抠像以及通过主动光源或闪光灯图像进行抠像的技术。文献[4]的算法是基于用户输入的提示信息,但是它的运算引入了耗时的非线性迭代计算方法。文献[5]是使用三分图的边界信息来解泊松方程,从而可从图像梯度中解析出前景映射图 α 。为了得到准确的抠像结果,该文还引入了交互式局部泊松抠像技术,这使得这项技术需要大量的人机交互。文献[6]被看作一种经典算法,它把抠像技术转化成贝斯定理框架,再解析出前景映射图 α 。文献[7]、[8]也得到了不错的抠像效果。但这些算法都需要使用三分图作为输入,而用户手动构造三分图本身就是一项繁琐的工作。近年出现了一些方便的图像前景背景分离算法^[9,10],它们可以帮助用户构造三分图,它们都是先通过解决最小切割问题来实现前景层与背景层的硬分离,并不提取 α 信息,然后再对抠像结果进行处理即可得到粗略的三分图,但这两种算法的稳定性不高,而且对于结构复杂的图像很容易产生错误的抠像结果。

本文给出了一种新的图像融合的方法,它将抠像技术和图像融合技术结合成一个整体,给出了一个完整的图像融合解决方案。该方法是根据用户输入的提示信息,先通过提取出前景映射图

α ,即前景映射图来作为泊松方程的梯度域标识,然后通过解泊松方程实现图像的无缝融合。而该算法与大家看到的最新的图像融合算法进行的对比实验,结果证明,图像融合效果更加完美。因此,该算法可进行高质量的无缝融合,同时可得到最优的计算速度。

2 前景映射图的提取

2.1 基本理论

抠像技术假设图像满足下面的模型:

$$I_i = \alpha_i F_i + (1 - \alpha_i) B_i \quad (1)$$

其中, α_i 是前景层第 i 个像素的透明度, F_i 为前景层第 i 像素灰度值, B_i 为背景层第 i 像素灰度值。抠像技术的目的是提取出 α , F 和 B , 对于 RGB 彩色图像,每一个像素都有 7 个未知数。但由于输入图像的每一个像素只有 3 个已知分量,因此必须添加约束才能解出未知数。现行的添加约束的方法是由用户输入一个三分图作为初始值,一幅三分图由前景区域、背景区域和未知区域组成,它一般是由用户手动构造的。这些方法一般都是通过非线性迭代来同时计算出 α , F 和 B 。这就意味着三分图中的未知区域越小越好,否则会花费大量的计算时间,而对于前景层和背景层非常复杂的图像来说,则需要通过大量的人机交互来构造一个好的三分图。至关重要的一点是,对于本文的图像融合算法来说,只需要提取出前景映射图 α ,即前景映射图,用来作为泊松方程的梯度域标识,而并不需要花费额外的计算代价去求出 F 和 B 。文献[3]给出了一种新的求前景映射图的闭式解决方案。它是先根据在 F 和 B 上的平滑度基本假设,导出一个代价方程;然后根据理论推导,把 F 和 B 消去,产生一个只有 α 的方程。那么前景映射图就是这个函数的全局最优解,它可以通过解一个稀疏线性方程组来求得。这个方法可直接计算 α ,并不依赖 F 和 B ,而且用很少的用户提示信息就可以解析出高质量的前景映射图。

首先需要对其 α , F 和 B 做一些基本假设。

对于灰度图像,假设在每一个像素周围的小窗口内, F 和 B 近似不变,那么就可以把抠像模型重写为

$$\alpha_i \approx a I_i + b, \forall i \in w \quad (2)$$

其中, $a = \frac{1}{F_i - B_i}$, $b = \frac{B_i}{F_i - B_i}$, w 是任一个小窗口。最终目标是找到 α , a , b , 用来最小化下面的方程^[3]:

$$J(\boldsymbol{\alpha}, \mathbf{a}, \mathbf{b}) = \sum_{j \in I} \left(\sum_{i \in \mathbf{w}_j} (\alpha_i - a_j I_i - b_j)^2 + \varepsilon a_j^2 \right) \quad (3)$$

其中, \mathbf{w} 是第 i 个像素周围的小窗口, 一般大小取 3×3 , 而参数 ε 的作用是保证数据稳定性。对于参数 \mathbf{w} 与 ε 的详细分析, 请参见文献[3]。这个方程的物理意义是在满足上面基本约束的前提下, 使得最终解尽量符合模型(式(1))。这个方程是关于 $\boldsymbol{\alpha}$, \mathbf{a} , \mathbf{b} 的二次函数, 它很难被直接最优化。根据文献[3]中的推导, \mathbf{a} , \mathbf{b} 可以被消去, 化简为只有 $\boldsymbol{\alpha}$ 的二次方程: $J(\boldsymbol{\alpha}) = \min_{\mathbf{a}, \mathbf{b}} J(\boldsymbol{\alpha}, \mathbf{a}, \mathbf{b}) = \boldsymbol{\alpha}^T \mathbf{L} \boldsymbol{\alpha}$, 其中 $\boldsymbol{\alpha}$ 为 $N \times 1$ 维的向量, \mathbf{L} 是一个 $N \times N$ 大小的矩阵, N 为像素个数, 它的第 (i, j) 项是

$$\sum_{kl \in (i, j) \in \mathbf{w}_k} \left(\delta_{i, j} - \frac{1}{|\mathbf{w}_k|} \left(1 + \frac{1}{\frac{\varepsilon}{|\mathbf{w}_k|} + \sigma_k^2} \times (I_i - \mu_k)(I_j - \mu_k) \right) \right) \quad (4)$$

其中, 当 i, j 相等时, $\sigma_{i, j}$ 为 1, 否则 $\sigma_{i, j}$ 为 0, μ_k, σ_k^2 是第 k 个像素周围的窗口 \mathbf{w}_k 内像素点的灰度均值和方差。 $|\mathbf{w}_k|$ 是这个窗口内的像素个数。

对于彩色图像来说, 可以把上面的结论扩展为 RGB 通道, 也可以放松上面的基本假设。这样做的好处是, 由于这个颜色模型会符合大部分的彩色图像, 因而可远远扩大该算法的适用范围。新的假设为在任一个小的窗口之内, \mathbf{F} 是两个颜色 F_a, F_b 的线性组合, \mathbf{B} 同理。这样就可以得到

$$\alpha_i = \sum_c a^c I_i^c + b, \forall i \in \mathbf{w} \quad (5)$$

其中, c 代表 3 个颜色分量 (c 代表 R、G、B), 使用这个模型就可以定义与前面类似的代价函数:

$$J(\boldsymbol{\alpha}, \mathbf{a}, \mathbf{b}) = \sum_{j \in I} \left(\sum_{i \in \mathbf{w}_j} (\alpha_i - \sum_c a_j^c I_i^c - b_j)^2 + \varepsilon \sum_c a_j^{c^2} \right)$$

同理, 可以通过消去 \mathbf{a}^c 和 \mathbf{b} 来得到一个只有 $\boldsymbol{\alpha}$ 的二次函数, 即

$$J(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T \mathbf{L} \boldsymbol{\alpha} \quad (6)$$

\mathbf{L} 是一个 $N \times N$ 大小的矩阵, 它的第 (i, j) 项是

$$\sum_{kl \in (i, j) \in \mathbf{w}_k} \left(\delta_{i, j} - \frac{1}{|\mathbf{w}_k|} \left(1 + (\mathbf{I}_i^c - \boldsymbol{\mu}_k)^T \times \left(\boldsymbol{\Sigma}_k + \frac{\varepsilon}{|\mathbf{w}_k|} \mathbf{E}_3 \right)^{-1} (\mathbf{I}_j^c - \boldsymbol{\mu}_k) \right) \right) \quad (7)$$

式中, $\boldsymbol{\Sigma}_k$ 是 3×3 大小的协方差矩阵, $\boldsymbol{\mu}_k$ 是 3×1 维的颜色均值向量, \mathbf{E}_3 是 3×3 大小的单位矩阵。在文献[3]中参数 ε 与 $|\mathbf{w}_k|$ 的作用有完整的分析。另

外, 与 \mathbf{L} 的次小特征值对应的特征向量之所以可以引导用户提示信息的放置位置, 是因为 \mathbf{L} 的特征向量可以进行粗略的图像分割^[11]。

2.2 添加用户约束

这个算法需要用户先在源图像上粗略地标识前景层与背景层, 分别用白色与黑色表示(如图 1(a)、图 1(b)所示), 再把这些约束添加到最优化问题中; 最终, 还需要最优化下面的二次函数:

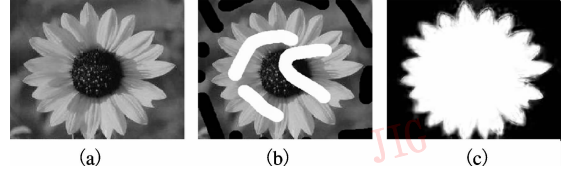


图 1 前景映射层的提取

Fig. 1 Extraction of α matte

$$C(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T \mathbf{L} \boldsymbol{\alpha} + \lambda (\boldsymbol{\alpha}^T - \mathbf{b}_s^T) \mathbf{D}_s (\boldsymbol{\alpha} - \mathbf{b}_s) \quad (8)$$

其中, 第 2 项是用户添加的约束, λ 为一个很大的数, 它用于决定用户约束所起作用的大小。 \mathbf{D}_s (下标 s 代表 constrains) 是一个对角阵, 对于用户标识了的像素, 它的相应对角元素为 1, 其余元素都为 0。 \mathbf{b}_s 为一个向量, 对于用户标识了的像素, 它的相应元素为用户标识的值(文中取 0 或 1), 其余元素均为 0。本文把这个函数所有偏导数置 0, 则可以通过以下线性方程组解出前景映射图 $\boldsymbol{\alpha}$ (如图 1(c)所示):

$$(\mathbf{L} + \lambda \mathbf{D}_s) \boldsymbol{\alpha} = \lambda \mathbf{b}_s \quad (9)$$

3 融合理论

3.1 基本原理

在讨论融合理论时可只考虑某一纯颜色分量, 再扩展到 RGB 颜色通道。融合算法的目的是把源图像 $\mathbf{I}_{\text{source}}$ 中的区域 Ω 融合到目标图像 $\mathbf{I}_{\text{target}}$ 中, f 是定义在 $\Omega - \partial\Omega$ 上的未知函数。 \mathbf{v} 是定义在源图像 $\mathbf{I}_{\text{source}}$ 中区域 Ω 上的已知向量域。文献[1]给出了融合的方法是为了解决下面的最优化问题:

$$\min_f \iint_{\Omega} |\nabla f - \mathbf{v}|^2 \text{ with } f|_{\partial\Omega} = \mathbf{I}_{\text{target}}|_{\partial\Omega} \quad (10)$$

其中, $\nabla = \left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right]$ 是梯度运算符。根据变分法理论中的奥氏方程^[12], 它的解就是带有狄利克雷边界条件的泊松方程的解:

$$\nabla^2 f = \text{div } \mathbf{v} \text{ over } \Omega, \text{ with } f|_{\partial\Omega} = \mathbf{I}_{\text{target}}|_{\partial\Omega} \quad (11)$$

$\text{div } \mathbf{v} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$ 是 $\mathbf{v} = (u, v)$ 的散度, $\nabla^2 = \frac{\partial^2}{\partial x^2} +$

$\frac{\partial^2}{\partial y^2}$ 是拉普拉斯运算符。

3.2 离散表示

对于图像来说,很显然应该通过像素点来把问题离散化。下面先定义一些基本变量,对于每个像素 p, N_p 表示由它的上下左右 4 个邻居组成的集合, $\langle p, q \rangle$ 表示一个像素对,使 $q \in N_p$ 。设 f_p 为在点 p 的 f 值,则算法的目标是求出 $f|_{\Omega} = \{f_p, p \in \Omega\}$ 。

运算时,首先确定梯度域函数,按照上面的符号,可取 $\mathbf{v} = \nabla I_{\text{source}}|_{\Omega}$,其离散形式为 $\mathbf{v}_{p,q} = I_p^{\text{source}} - I_q^{\text{source}}$ 。那么式(11)就变成了

$$\nabla^2 f = \Delta I_{\text{source}} \text{ over } \Omega, \text{ with } f|_{\partial\Omega} = I_{\text{target}}|_{\partial\Omega} \quad (12)$$

根据拉普拉斯 5 点有限差分公式,就可以把式(12)离散化为

$$|N_p| f_p - \sum_{q \in N_p \cap \Omega} f_q = \sum_{q \in N_p \cap \partial\Omega} I_q^{\text{target}} + \sum_{q \in N_p} \mathbf{v}_{p,q}, \text{ for all } p \in \Omega \quad (13)$$

当 Ω 包含边缘上的像素时,则这些像素的 4 邻域就减少为 3 或 2 邻域,此时 $|N_p| < 4$ 。而对于在 Ω 内部的像素,则由于式(13)的右侧并没有边界条件,因此式(13)可改写成

$$|N_p| f_p - \sum_{q \in N_p} f_q = \sum_{q \in N_p} \mathbf{v}_{p,q} \quad (14)$$

由于边界条件的任意性,很难利用这个方程建立一个整体的线性方程组,因此只能采用经典的线性迭代的方法来求解,例如高斯塞德尔迭代。

最终实现的效果可以把某图像中的任意形状的一块区域自然地融入另一幅图像之中。由于随着高斯塞德尔迭代次数的不同和迭代初始值的不同,实现的融合效果也多种多样,因而可以满足不同的应用。

4 无缝融合的实现

本文设计了一个交互式的系统来实现无缝融合,用户可以在任意的画图软件上先粗略地标识出前景层与背景层,作为程序输入,然后系统就可以自动地解析出它的前景映射图。用户再选定该区域在目标图像中的位置,就可实际图像的无缝融合。

由于本文提取前景映射图的计算只需要构造矩阵 L ,再解一个稀疏线性方程即可,因此速度非常的快。提取出来的前景映射图相当于一个灰度图像,其

值在 0 和 1 之间,它表示前景层的透明度,也可以把它看作一个标识表,只有源图像中与它对应的那些像素被无缝融合到目标图像中。利用前景映射图从源图像中取出梯度域的方法有多种。文献[1]、[2]分别提出了两种混合梯度的方法,前者利用 α 值把源图像与目标图像混合作为梯度域,后者取源图像与目标图像对应像素点的梯度绝对值中的大者作为梯度域。

通过实验证明,对于图像融合,直接选取与 Ω 所对应的源图像区域作为梯度域,效果非常好。图像融合时,首先确定区域 Ω 的范围,其定义为

$$x \in \Omega, \text{ 如果 } \alpha(x) > T_{\text{alpha}} \quad (15)$$

其中,阈值 T_{alpha} 由用户给定,文中一般取 0.3。那么图像融合的梯度域为

$$\mathbf{v}(x) = \nabla I_{\text{source}}(x), x \in \Omega \quad (16)$$

而对于图像混合来说,其目的并不是把原图像完全融入目标图像,而是要把两幅图像进行颜色叠加。传统的图像混合方法只是直接把两幅图像的 RGB 值分别加权求和,但这样会产生很不理想的结果,并且会在边缘处产生不自然的衔接。对于这类应用,可采用如下的梯度域形式:

$$\mathbf{v}(x) = \begin{cases} \nabla I_{\text{source}}(x), & \text{如果 } |\nabla I_{\text{source}}(x)| > \\ |\nabla I_{\text{target}}(x)|, & x \in \Omega \\ \nabla I_{\text{target}}(x), & \text{否则, } x \in \Omega \end{cases} \quad (17)$$

图 2 给出了这两种情况的对比,图 2(a) ~ 图 2



图 2 图像融合与混合
Fig. 2 Image cloning and blend

(d) 分别是源图像、用户提示图像、前景映射图和目标图像。图 2(e)、图 2(f) 是采用第 1 种梯度域的融合结果,图 2(g)、图 2(h) 是采用第 2 种梯度域的混合结果。由此可以看到图 2(f) 与图 2(h) 的差别。由于图 2(f) 是完全以原图像作为梯度域,因此融合结果中并不混有目标图像。由于图 2(h) 是采用混合梯度,因此其混合结果为原图像与目标图像的混合。

确定了梯度域,就可以进行高斯塞德尔迭代。迭代可以采用原图像作为初始值,也可以采用目标图像作为初始值。根据应用的不同,这将产生不同的效果。另外用户还可以控制迭代次数,这也将产生多种多样的效果。

5 实验

文献[2]算法虽已得到了相当不错的融合结果,但相比之下,本文的自由融合方法在保证融合结果的同时,还有一个很大的优点,就是用户可以直接控制前景映射图的走向。由于专利权的问题,无法得到文献[2]的可执行文件,也没有足够的时间来实现文献[2]的算法,因此,本文用从文献[2]及其网站上面获取的图片来进行对比实验。相应的运算结果都是从其论文中获得。另外,还做了泊松图像编辑和直接融合的对比实验。几乎所有的结果都显示出本文算法的强大功能及其灵活性,对于目前的图像融合技术,本文的算法可以得到更好的效果、更快的计算速度以及多样的融合效果。

实验 1 是为了证明自由融合算法可以实现用户引导前景映射图的走向,以便获得灵活的融合效果(如图 3 所示)。由图 3(c)、图 3(f)、图 3(i) 可以看出,根据用户提示信息的不同,提取的前景映射图也不尽相同。图 3(d) 中提取了人与整个摩托车。图 3(g) 中根据用户提示信息把摩托车前轮变成背景层,而只提取人、车身与车后轮。图 3(j) 只提取了人和车身,这可以根据用户的要求以及应用的不同,而做出灵活的改动。图 3(e)、图 3(h)、图 3(k) 为最终效果图,由图 3(e)、图 3(h)、图 3(k) 可以看到,无论用户的提示信息如何改变,本文算法都可以得到令人满意的融合结果。这些结果全部采用目标图像作为初始值,并且迭代 200 次。

实验 2 是为了说明随着初始值与迭代次数的不同,其融合效果也不尽相同(如图 4 所示)。图

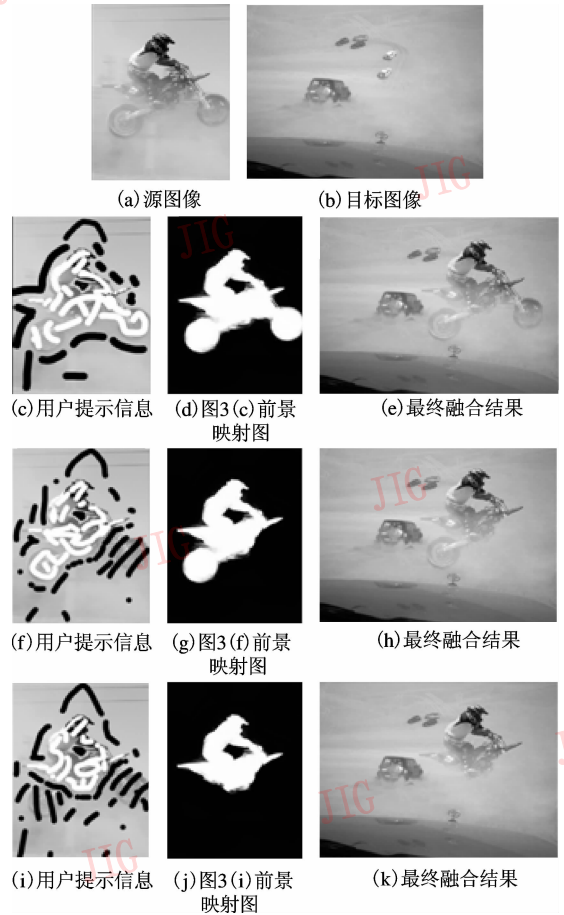


图 3 用户引导信息的作用

Fig. 3 The role of user guided information

4(a) ~ 图 4(d) 分别为源图像、用户信息、前景映射图和目标图像。实验在计算之前先把源图像转化为灰度图,因为这样可以消除色调不一致的影响,以保证梯度域的质量。图 4(e) ~ 图 4(g) 均为融合结果。如果迭代次数足够多,那么使用这两种初始值应该可得到趋于一致的效果。如果想要使融合的色调更接近源图像,则可以使用源图像作为初始值;反之则使用目标图像作为初始值。

实验 3 的目的是为了进行多个算法的对比,其融合结果见图 5。从图中可以看出,直接贴图的结果其边缘出现了瑕疵,用文献[1]的方法融合图像,由于边缘的不准确,因此融合效果不自然。由本文方法的融合结果(图 5(h)、(i))可见,鸭子和它倒影的边缘处,都得到了令人满意的融合结果。图 5(f) 和图 5(h) 的融合均以源图像作为初始值,迭代次数均为 20 次。

图 6 为采用文献[2]中的图像作为源图像与目标图像的融合结果,同时还与泊松编辑法和直接贴

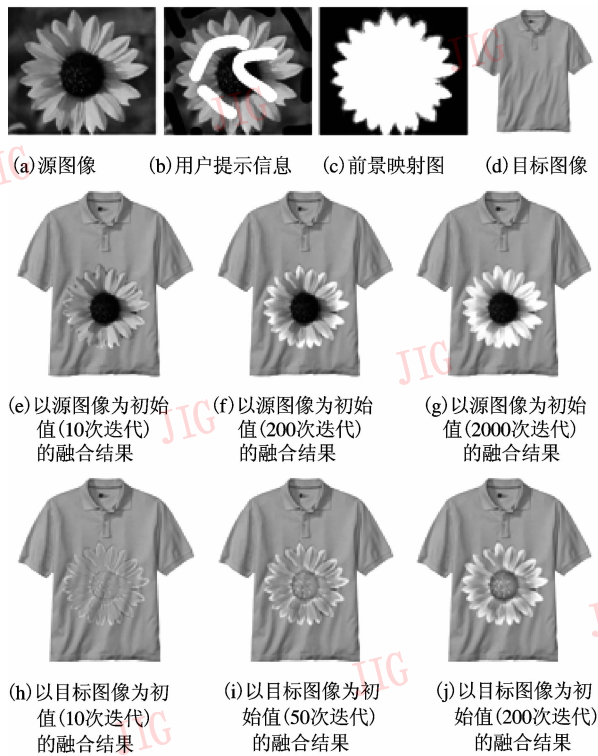


图 4 不同的初始值与迭代次数得到的融合结果

Fig. 4 The result using different initial value and iterating number



图 5 不同算法的融合结果比较

Fig. 5 Comparison between different algorithms

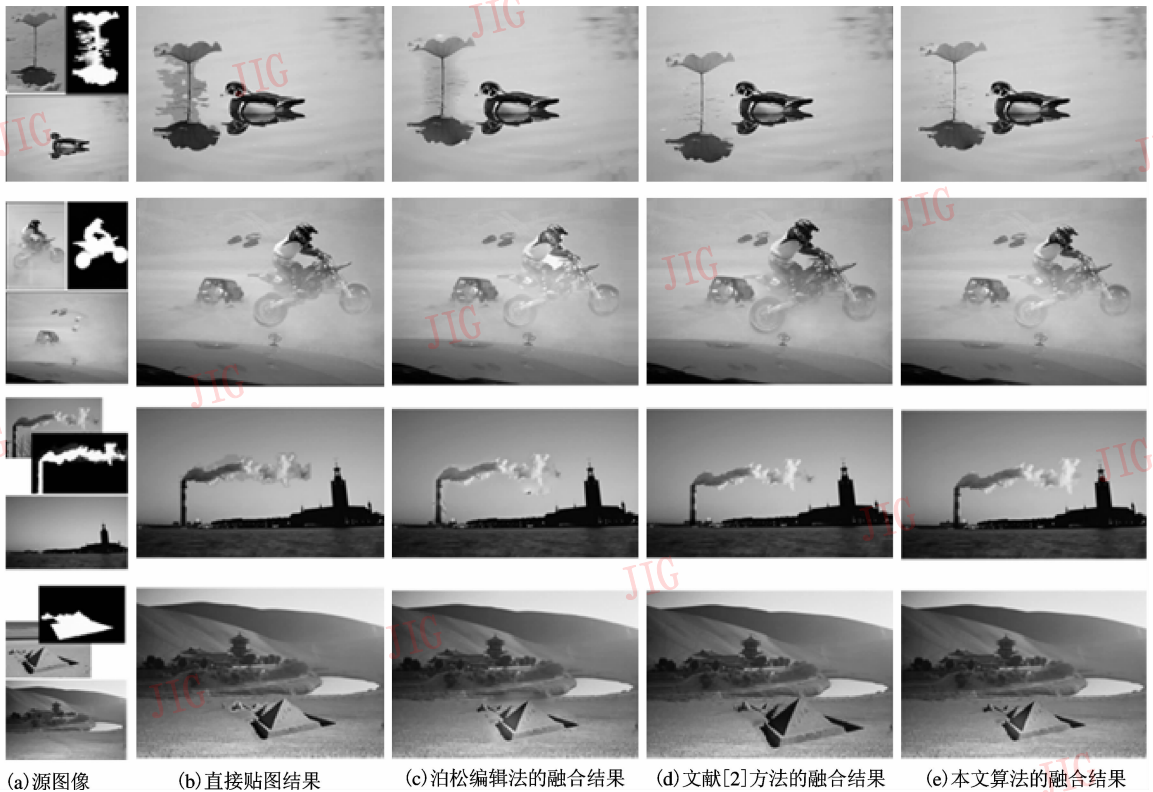


图 6 不同算法融合结果的比较

Fig. 6 Comparison between different algorithms

图法进行了对比实验。由图 6 可以看到,在使用文献[2]中的图片进行测试时,自由融合方法的融合结果毫不逊色,而且实验 1 已经证明本文算法具有更灵活的特点,同时对于色彩和边缘比较复杂的图像,本文算法的适应能力更加强大。对于绝大多数的应用来说,自由融合方法均可取得令人满意的融合效果。现今流行的算法大多只能适用于某一类特定的图片,相比之下,自由融合方法由于放宽了对图片的限制条件,因此具有普遍适用性。

实验 4 是使用前面给出的混合梯度域,其实现的图像混合效果如图 7 所示。除了图 2 中的常规图

像混合之外,本实验还得到了一些相当有趣的结果。这不仅更加扩大了该算法的应用范围,也为用户提供了更多的图像融合解决方案。该图像混合算法可以应用于艺术图像以及创意设计,图 7 为一些有趣的融合结果。其中荷花图可以与图 7(a)、(b)混合,而达芬奇自画像只能与图 7(b)混合,这是由于图 7(a)的大多数像素点的梯度绝对值均比较大的缘故。由于把图 7(a)与图 7(f)混合,其混合梯度域实际上相当于图 7(a),因此图 7(a)与图 7(f)的混合没有任何效果。而由于图 7(b)的梯度比较缓和,所以大多数图像均能够与它进行混合。

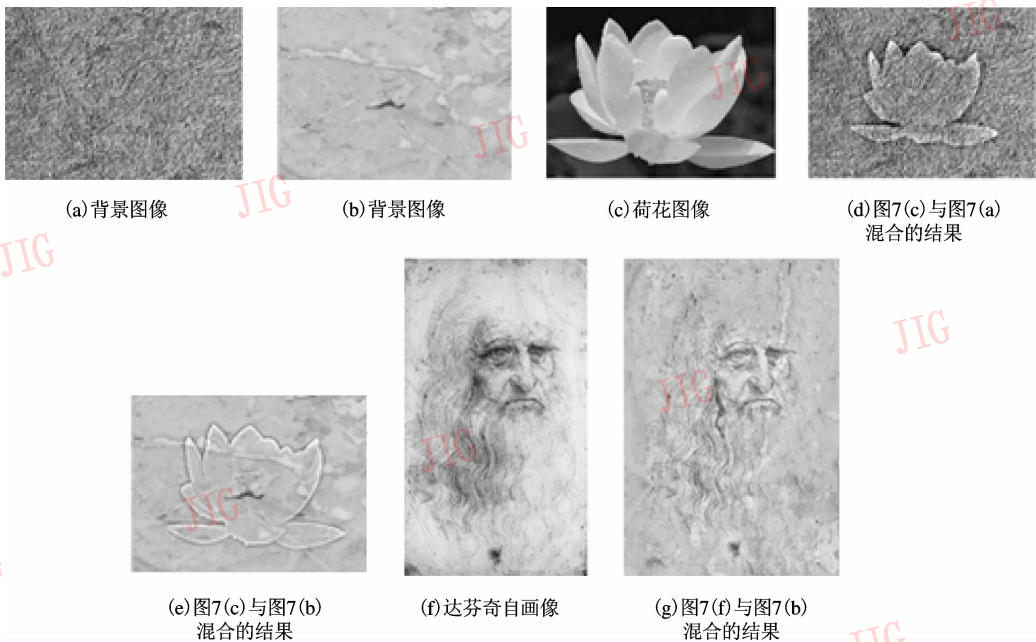


图 7 图像混合

Fig. 7 Image blend

在运算速度上,由于提取边缘时需要解一个稀疏线性方程组 $(L + \lambda D_s) \alpha = \lambda b_s$,其中 L 是一个对称半正定矩阵,一般可采用 LU 的方法来解决它。对于一个 400×300 大小的图像,在 P4 2.8GHz、768MB 的机器上,大约用去几秒钟就可以得到前景映射图。而对于后续的高斯塞德尔迭代,其花费的时间则与用户设置的迭代次数和误差阈值有关。平均来说,对于 200×200 大小的迭代区域,每 100 次迭代大约需要 4s。

6 结 论

本文提出的自由融合方法通过将两种不同的

算法结合成一个整体,给出了一个完整的图像融合解决方案。同时也与最新的图像融合算法做了大量的对比实验,并获得了完美的图像融合效果。实验证明,本文算法不仅可进行高质量的无缝融合,同时可得到最优的计算速度。该算法首先利用抠像技术来提取图像边缘,然后迭代求解泊松方程,从而取得了自然的融合效果。实验证明,由于用户的信息、初始值以及迭代次数的不同,因此可以获得多种多样的融合效果,同时也可实现图像的自然混合。本文算法与最新的无缝融合算法相比,对于前景与背景的颜色变化复杂混乱的图像,自由融合方法仍可得到准确的边缘,并可以实现由用户引导前景映射图的走向;对于图像中间

有洞的情况也不会受到任何影响,这使得该算法具有相当的灵活性;对于更加复杂的图像,只要有足够精确的用户输入信息,则本文的算法便可以得到令人满意的融合效果。在这个算法的基础上,不仅可以进行视频无缝融合的研究,也可以结合现存的技术开发一整套图像融合软件。

参考文献 (References)

- 1 Perez P, Gangnet M, Blake A. Poisson image editing [A]. In: Proceedings of ACM SIGGRAPH[C], San Diego, CA, USA, 2003: 313 ~ 318.
- 2 Jia Jia-ya, Sun Jian, Tang Chi-keung, *et al.* Drag-and-drop pasting [A]. In: Proceedings of ACM SIGGRAPH[C]. New York, USA, 2006: 631 ~ 637.
- 3 Levin A, Lischinski D, Weiss Y. A closed form solution to natural image matting[A]. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition[C]. New York, USA, 2006, 1:61 ~ 68.
- 4 Wang Jue, Cohen F. An iterative optimization approach for unified image segmentation and matting [A]. In: Proceedings of IEEE International Conference on Computer Vision[C]. Beijing, China, 2005,2:936 ~ 943.
- 5 Sun J, Jia J, Tang C. Poisson matting[A]. In: Proceedings of ACM SIGGRAPH[C]. New York, USA, 2004: 315 ~ 321.
- 6 Chuang Y, Curless B, Salesin D, *et al.* A bayesian approach to digital matting [A]. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition[C]. New York, USA, 2001, 2:264 ~ 271.
- 7 Ruzon M, Tomasi C. Alpha estimation in natural images [A]. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition[C], New York, USA, 2000: 18 ~ 25.
- 8 Berman A, Vlahos P, Dadourian P. Comprehensive Method for Removing from an Image the Background Surrounding a Selected Object[P]. US Patent: 6,134,345, 2000.
- 9 Li Y, Sun J, Tang C. Lazy snapping[A]. In: Proceedings of ACM SIGGRAPH[C], New York, USA, 2004: 303 ~ 308.
- 10 Rother C, Kolmogorov V, Blake A. "Grabcut"-interactive foreground extraction using iterated graph cuts[A]. In: Proceedings of ACM SIGGRAPH[C], New York, USA, 2004: 309 ~ 314.
- 11 Shi J, Malik J. Normalized cuts and image segmentation[A]. In: Proceedings of IEEE International Conference Computer Vision and Pattern Recognition[C], San Juan, Puerto Rico, 1997:731 ~ 737.
- 12 Wu Di-guang. Calculus of Variations [M]. Beijing: Higher Education Press, 1987: 65 ~ 80. [吴迪光. 变分法[M]. 北京: 高等教育出版社, 1987: 65 ~ 80.]