

# 一种图像背景更换方法的实现

## An Implementation of Background Segmentation Algorithm

吴 黎 (温州职业技术学院 计算机系 温州 325035)

**摘 要:** 针对相对复杂图像的背景更换问题,本文先运用模糊 C 均值聚类算法(FCM)对图像进行模糊分割。再根据模糊分类后的图像,本文设计了一种图像目标提取方法。实验表明,这种方法能还原模糊分类后的图像目标,并使背景部分更换成其他颜色,从而实现图像目标的提取。最后,设计了一种新的数据融合法将两种不同的合成算法结合起来,最终使目标图像与其他背景图像有机的合成在一起,从而实现图像的背景更换。

**关键词:** 图像分割 模糊聚类 图像目标提取 图像合成

图像背景更换是指在图像中运用图像分割技术来提取热点目标对象,然后应用图像合成算法将背景更换成其他背景。它广泛应用于影视作品和多媒体设计等。常见的图像分割方法,如阈值分割方法有相当大的局限性<sup>[4]</sup>,要求图像的背景主要是单色,难以适用于相对复杂的背景。与阈值分割方法相比,利用模糊聚类方法在对彩色图像进行分割时能够把 3 个彩色分量 RGB 作为一个整体进行考虑,分割效果较好。模糊 C 均值聚类(FCM)是主要的模糊聚类方法,其基本思想是通过迭代来优化用于表示图像像素点与 C 类中心的相似度的目标函数,获取极大值,从而得到最优聚类。为此,本文将模糊聚类方法运用于图像分割,先利用分层聚类<sup>[3]</sup>把图像数据分成一定数量的色彩相近的子集,再利用子集中心点和分布密度对图像进行模糊聚类,由于聚类样本数量显著减少,故能保留图像的有用信息。根据模糊分类后的图像,本文还提出了一种目标提取方法。该方法能快速还原模糊分类后的图像目标,并使背景部分更换成其他颜色,从而实现图像目标的提取。最后,本文设计了一种新的数据融合法将两种不同的合成算法结合起来,最终使目标图像与其他背景图像有机的合成在一起,从而实现图像的背景更换。综上所述,图像背景更换方法的总体步骤可以如下图 1 所示:

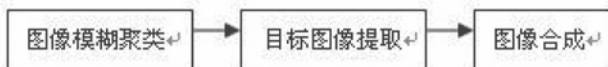


图 1 图像背景更换方法的总体步骤图

## 2 模糊 C 均值聚类方法的设计

将数据集  $C_x = \{x_1, x_2, \dots, x_n\} \in R^m$  平均分为  $t$  个数据点数为  $b$  的子集 ( $n = bt$ ), 记为  $C_x^l, l = 1, 2, \dots, t$ 。数据点  $x_i$  的密度函数定义为该点邻域内的数据点个数  $N_i$  的邻域则定义以为中心,  $r$  为半径的超球体。先

定义函数  $u(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$ , 则  $x_i$  邻域内的数据点密度为:

$$D_i^l(x_i') = \sum_{j=1}^b u(r - \|x_i' - x_j'\|) \quad (1)$$

如果数据点  $x_{cl}^l \in C_x^l$  具有最大的分布密度, 即

$$D_{cl}^l = \max_i D_i^l, \text{ 那么子集 } C_x^l \text{ 的第 } l \text{ 个聚类中心为:}$$

$$cen_{cl}^l = \frac{\sum_{x_j' \in C_{cl}^l} x_j' * D_j^l(x_j')}{\sum_{x_j' \in C_{cl}^l} D_j^l(x_j')} \quad (2)$$

其中,  $C_{cl}^l$  为  $x_{cl}^l$  邻域内的数据点集。找出第 1 个聚类中

心后,接着在集合 $\{C_x^l - C_{cl}^l\}$ 里查找具有最大密度函数值的数据点 $x_{c2}^l$ ,并将其邻域内的数据点的质心作为第 2 个聚类中心 $cen_{c2}^l$ 。重复同样的过程,直至

$$C_x^l - \sum_k C_{ck}^l = \emptyset \quad k \text{ 为获得的中心个数。}$$

令 $l=1,2,\dots,t$ 重复上述过程,可获得第一层聚类的中心点集和对应的分布密度,把这些聚类的中心重新组合成一个新的数据集,对这个数据集再进行第二层聚类运算,密度函数由式(5)改为式(7)

$$D_i = \sum_{j=1}^{n_c} u(r - \|x_i - x_j\|) * D^{(1)}(x_j) \quad (3)$$

其中 $n_c, D^{(1)}(x)$ 分别为第一层获得的中心个数和中心密度。如此,可得聚类中心 $C_{cj}, j=1,2,\dots,n_p$ ,为第二层获得的中心个数。

这样,通过分层聚类把样本大小为 $n$ 的数据集根据一定的相似性准则分成 $np$ 个子集,即 $S = \bigcup_{i=1}^{n_p} S_i$ ,中

心点构成的数据集记为 $C_x^c = \{x_1^c, x_2^c, \dots, x_{n_p}^c\}$ ,每个中心 $x_i^c$ 对应的数据密度即子集 $S_i$ 大小记为 $D^c(x_i^c)$ 。

FCM 算法是一种迭代优化的运算方法,其需要反复计算 $u_{ik}$ 和 $v_i$ 。当样本数量 $n$ 很大时,该计算极为耗时。利用分层聚类,可以根据一定的相似性准则,先把整个彩色图像点集 $S$ 分为 $n_p$ 个子集 $S_k (k=1,2,\dots,n_p)$ ,再在这些子集上进行模糊聚类,将会极大地提高聚类速度<sup>[7]</sup>。

由于每个子集 $S_k$ 内像素点的色彩比较接近,到中心点 $v_i$ 的距离可以近似地用 $S_k$ 的中心到 $v_i$ 的距离来表示,即:

$$d_{il}^2(x_l, v_i) \approx d_{ik}^2(x_k^c, v_i) = \|x_k^c - v_i\|_A^2 \quad (4)$$

模糊矩阵 $U$ 的大小由原来的 $n \times c$ 变为 $n_p \times c$ ,隶属度的计算公式为

$$u_{ik} = \left[ \sum_{j=1}^c \left( \frac{d_{ik}(x_k, v_i)}{d_{jk}(x_k, v_j)} \right)^{2/(m-1)} \right]^{-1} \quad k=1,2,\dots,n \quad (5)$$

但对于距离的计算,则用式(4)。c个聚类中心的计算公式为:

$$v_i = \frac{\sum_{k=1}^{n_p} (u_{ik})^m x_k^c D^c(x_k^c)}{\sum_{k=1}^{n_p} (u_{ik})^m D^c(x_k^c)} \quad i=1,2,\dots,c \quad (6)$$

目标函数式的计算为:

$$J_m(U, V) = \sum_{k=1}^{n_p} \sum_{i=1}^c (u_{ik})^m d_{ik}^2(x_k^c, v_i) D^c(x_k^c) \quad (7)$$

如果 $C_x^l, l=1,2,\dots,t(8)$ ,则 $x_k$ 属于第 $i$ 类。

实际应用中,可以对式(1)、式(3)的 $r$ 值进行适当调整以获得合适的 $n_p$ 值。设 $r_{\min}$ 取为数据点各分量上的标准偏差的最小值<sup>[2]</sup>, $r_{\min}/2$ 和 $r_{\min}/10$ 分别是式(1)、式(3)中 $r$ 比较理想的取值。由于 $n_p$ 远小于 $n$ ,利用式(4)、式(5)、式(6)和式(7)进行模糊聚类,计算速度将会得到很大提高。

### 3 模糊 C 均值聚类方法的实现

分层聚类算法具体实现步骤可以分为两个部分,如下所示:

#### (1) 第一层聚类运算

1. 将数据集 $C_k$ 均分为 $t$ 个数据点数为 $b$ 的子集( $n = bt$ ),记为 $C_x^l, l=1,2,\dots,t$

2. 初始化 $l=1$

3. 按以下步骤求各子集的聚类中心

(a) 根据式(1)计算邻域内的数据点密度

(b) 设 $k=1$

(c) 在集合 $\left\{ C_x^l - \sum_{h=2}^k C_{c(h-1)}^l \right\}$ 里查找具有最大密

度函数值 $D_{ck}^l$ 的数据点,满足

$$D_{ck}^l = \max_i D_i^l, x_{ck}^l \in \left\{ C_x^l - \sum_{h=2}^k C_{c(h-1)}^l \right\}$$

其中 $D_{ck}^l$ 为邻域 $x_{ck}^l$ 内的数据点集

(d) 根据式(2)求子集 $C_{ck}^l$ 的第 $k$ 个聚类中心

(e) 如果 $C_x^l - \sum_{h=1}^k C_{ch}^l \neq \emptyset$ ,转步骤3(c), $k=k+1$ ;否则,转步骤4

4. 如果 $l=t$ ,结束运算,否则,转步骤3, $l=l+1$

(2) 第二层聚类运算(与第一层聚类运算的步骤3相似)

1. 把经过第一层聚类运算得到的聚类中心重新组

合成一个新的数据集, 设  $C_x$  为这些聚类中心  $x_i (i=1, 2, \dots, n_c)$  的集合

2. 根据式 (3) 计算  $X_i$  的密度函数值  $D_i$

3. 设  $k=1$

4. 在集合  $\left\{C_x - \sum_{h=2}^k C_{c(h-1)}\right\}$  里查找具有最大密度函数值  $D_{ck}$  的数据点  $X_{ck}$ , 满足  $D_{ck} = \max_i D_i, x_i \in \left\{C_x - \sum_{h=2}^k C_{c(h-1)}\right\}$ 。然后, 根据式 (2) 求子集  $C_x$  的聚类中心  $cen_k$  (第  $k$  个聚类中心)

5. 如果  $C_x - \sum_{h=1}^k C_{ch} \neq \emptyset$ , 转步骤 4,  $k=k+1$ ; 否则 结束运算, 得到各聚类中心  $cen_j, j=1, 2, \dots, n_p$

$n_p$  为第二层获得的聚类中心个数。

综上所述, 利用模糊 C 均值聚类算法 (FCM) 进行彩色图像分割的方法如下所示:

1) 首先利用分层聚类算法对图像数据进行聚类, 获得  $n_p$  个子集;

2) 对中心集  $C_x^c$  按照数据密度的值从大到小重新排序, 并对  $c$  赋值;

3) 以  $C_x^c$  中前  $c$  个元素初始化中心点集  $V$ ;

4) 利用式 (4)、式 (5)、式 (6) 和式 (7) 进行模糊聚类;

5) 选取合适的矩阵范数 (如欧几里德距离) 比较  $J_m^{(k)}$  和  $J_m^{(k+1)}$ , 若  $\|J_m^{(k+1)} - J_m^{(k)}\| < \varepsilon$ , 则停止迭代, 转到第 6 步; 否则  $k=k+1$  转到第 4 步;

6) 利用  $V$  重新计算  $U$  并根据式 (8) 分割图像。

## 4 图像目标提取

根据模糊 C 均值聚类算法, 将整个图像点集分类为有限的  $C$  个子集, 任一像素点必须完全属于某一子集, 这就为下一步的图像目标提取作好准备。图像目标提取可以分为以下两个步骤。

(1) 在 FCM 模糊分类后的照片中, 用矩形工具选择目标样本集。设置一个标志数组  $flag$  ( $pic.width$ ) ( $pic.height$ ), 当  $flag=0$  时, 表示背景, 当  $flag=1$  时, 表示目标, 初始值默认为全 0。同时设置目标样本数组  $ObjectPixel(3, C)$ , 3 表示 RGB 三个值,  $C$  表示划分

的类数。遍历该矩形选中的每一像素, 将该像素点的标志设为 1, 并将该像素值与目标样本数组做比较, 看是否已经存在, 如果已存在, 便不作保存, 否则, 将该像素值保存于目标样本数组中。

(2) 遍历原图像中的每一像素, 将其像素值与目标样本相比较。如果相同, 则置该像素的  $flag$  为 1, 将其像素值设为最初的像素值, 还原目标图像; 否则  $flag$  为 0, 并将该像素值设为其他背景色, 更换背景图像。

## 5 图像合成算法的设计实现

本文采用层结构和数据融合<sup>[5]</sup>两种图像合成方法, 在模块实现时需要设计适当的算法协调它们之间的转换, 同时需要设置目标图像透明度参数和边缘模糊度参数以实现图像软合成。具体算法设计如下所示:

(1) 打开背景图像和目标图像, 设  $sb, sf$  分别表示背景图像和目标图像 (目标图像, 其背景已经设为黑色 (0 0 0), 当然也可以是其他颜色), 同时设置目标图像透明度参数  $tr$  (0% - 100%) 和边缘模糊度参数  $bm$  (0 像素 - 100 像素), 其中  $tr=0$ , 表示目标图像不透明, 若  $tr=100$ , 则表示目标图像完全透明 (只见背景, 不见目标),  $bm=0$ , 表示目标图像和背景图像之间的边缘是硬合成的, 若  $bm$  的值不为 0, 则表示目标图像和背景图像之间的边缘软合成的程度。

(2) 遍历  $sf$  的每一个像素 ( $x, y$ )

{ 设置边缘模糊长度  $r$ , 并初始化为  $bm$ ;

如果  $r > x$ , 则  $r = x$ ; 如果  $r > y$ , 则  $r = y$ ;

如果  $r > sf.Width - x$ , 则  $r = sf.Width - x$ ; 如果  $r > sf.Height - y$ , 则  $r = sf.Height$ ;

(注: 从目标图像的四周进行模糊化。)

设置背景图像相对与目标图像的一个像素点  $GetSourcePixel(x + Posx, y + Posy, sb)$ , 其中 ( $Posx, Posy$ ) 表示目标图像在背景图像的相对位置, 同时设置  $c1$  为目标图像的一个像素点,  $GetSourcePixel(x, y, sf)$ ; 设置  $c2$  为结果像素点, 保持结果值;

如果  $c1$  为黑色像素, 则  $c2 = c$ ;

否则, 先进行目标图像的透明化处理 ( $c1 = c * tr + c1 * (100 - tr)$ );

然后进行图像的边缘模糊化处理 ( $c2 = (c * (bm - r) + r * c1) / bm$ );

(注:离边缘越近,与背景越相似,相似的大小用  $bm$  来控制。)

保存结果像素  $c2$  ;}

## 6 实验分析

为了说明图像背景更换方法的适用效果,本文分别从图像目标提取和图像合成两个方面对其进行

分析。

### 6.1 图像目标提取实验分析

现以皮鞋照片作为例子,先对原始图像进行 FCM 运算,如图 2(b)所示,再对模糊分类后的图像进行图像目标提取,将背景换成黑色,如图 2(c)所示。

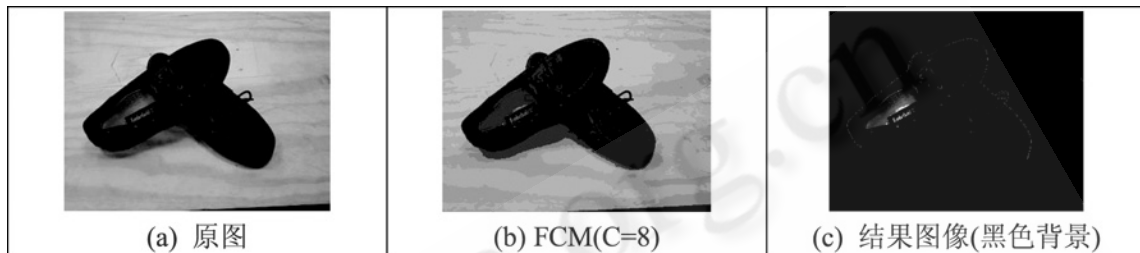


图 2 图像目标提取结果图(图像分辨率为  $236 * 329$ )

如实验结果所示,FCM 能将原图中目标和背景划分为不同的类,从而将两者准确的区分开来。

### 6.3 图像合成实验分析

为了说明该图像合成算法在图像背景更换中的适用效果,本文分别将图像分割后的目标图像和其他背景图像进行硬合成和软合成,对其效果进行比较分析。当目标图像透明度参数  $tr$  和边缘模糊

度参数  $bm$  均为 0 时,显示的合成图像如图 3(c)所示,目标部分完全覆盖相对应的背景部分,两者存在明显的边缘;调整  $tr$  和  $bm$  参数,就可以根据需要得到合成图像,如图 3(d)所示。实验结果表明该图像合成算法能够根据用户需要有效地将目标与其他背景(如图 3(b))合成在一起,从而最终完成图像的背景更换。

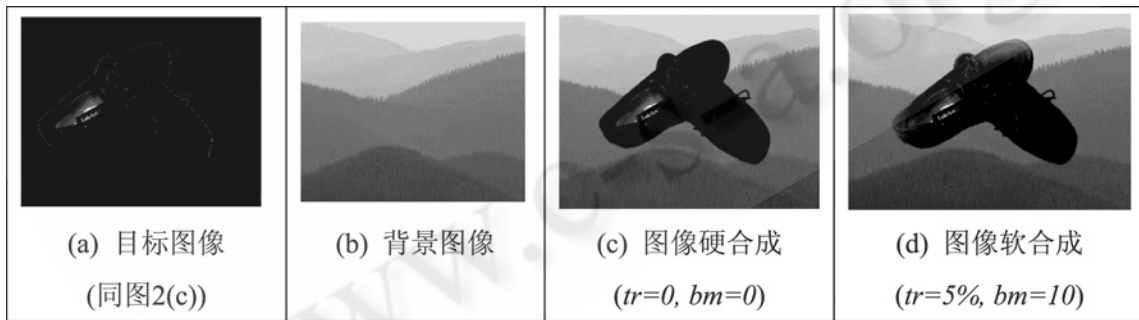


图 3 图像合成效果图

## 参考文献

- 1 丁震,胡钟山,扬静宇,等.一种基于模糊聚类的快速二值化方法.计算机学报,增刊,1998:331-334.
- 2 Jain A K, Murty M N, Flynn P J. Data clustering: A review. ACM Computer Survey, 1999, 31(3): 264-323.

- 3 Tao C W. Unsupervised fuzzy clustering with multi-center clusters. Fuzzy Sets and Systems, 2002, 128(3): 305-322.
- 4 刘恒.虚拟背景下自助照相中成像技术研究.系统仿真学报, 2003, 15(3).
- 5 冈萨雷斯.数字图像处理(第二版).电子工业出版社, 2003.