

使用高效的 c 均值聚类算法的 图象阈值化方法

刘健庄 涂予青

(西安电子科技大学, 西安 710071)

摘要 Otsu (1979) 的阈值化法被认为是一种良好的图象分割方法。本文提出一个适合图象分割的高效的 c 均值聚类算法。它与 Otsu 法有完全相同的分割结果, 但计算时间约减少了一个数量级。

关键词 图象分割; 阈值比; c 均值聚类算法

1. 引言

在图象处理和计算机视觉中, 阈值化是图象分割中广泛使用的一种有效的工具。在最近有关阈值化方法及其性能比较的综述文章^[1,2]中, 基于类别可分性准则的 Otsu 法^[3]被认为是一种性能良好的自动阈值选择方法, 并且其时间消耗相对其它阈值化方法是比较少的。本文针对图象的特点, 提出了 c 均值聚类算法在阈值化中的快速实现方法, 并且证明了该算法与 Otsu 法等价, 但计算速度约提高了一个数量级。

2. Otsu 阈值化方法

这里首先简要地介绍 Otsu 阈值化方法^[3]。设一个图象的像素点数为 N , 它有 L 个灰度级 $(0, 1, \dots, L-1)$, 灰度级为 i 的像素点数为 n_i , 那么 $N = \sum_{i=0}^{L-1} n_i$ 。设灰度级

i 发生的概率为 $p_i = n_i/N$, 则有 $\sum_{i=0}^{L-1} p_i = 1$ 。设阈值 t 将图象直方图分成两类 C_0 和 C_1 (物体和背景), 即 C_0 和 C_1 分别对应具有灰度级 $\{0, 1, \dots, t\}$ 和 $\{t+1, t+2, \dots, L-1\}$ 的像素。设 $\sigma_w^2(t)$, $\sigma_b^2(t)$ 和 σ_t^2 分别表示直方图中阈值为 t 时的类内方差、类间方差和总方差, 那么最优阈值可以通过求下面的某个准则函数的最大值得到, 即

$$\lambda(t) = \sigma_b^2(t)/\sigma_w^2(t), \quad k = \sigma_t^2/\sigma_w^2(t), \quad \eta = \sigma_b^2(t)/\sigma_t^2 \quad (1)$$

此时得到的 t 被认为对 C_0 和 C_1 进行了最合理的分割。这里有

$$\sigma_b^2(t) = \omega_0(t)[\mu_0(t) - \mu_T]^2 + \omega_1(t)[\mu_1(t) - \mu_T]^2 = \frac{[\mu_T \omega_0(t) - \mu(t)]^2}{\omega_0(t)[1 - \omega_0(t)]}$$

$$\sigma_w^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t)$$

$$\mu_T = \sum_{i=0}^{L-1} i p_i, \mu(t) = \sum_{i=0}^t i p_i, \mu_0(t) = \mu(t) / \omega_0(t)$$

$$\mu_1(t) = \frac{\mu_T - \mu(t)}{1 - \omega_0(t)}, \sigma_0^2(t) = \sum_{i=0}^t \frac{[i - \mu_0(t)]^2 p_i}{\omega_0(t)}, \sigma_1^2(t) = \sum_{i=t+1}^{L-1} \frac{[i - \mu_1(t)]^2 p_i}{\omega_1(t)}$$

其中 $\omega_0(t)$ 和 $\omega_1(t)$ 表示 C_0 和 C_1 发生的概率, $\mu_0(t)$, $\mu_1(t)$ 和 μ_T 分别表示 C_0 , C_1 和原图的均值, $\sigma_0^2(t)$ 和 $\sigma_1^2(t)$ 表示 C_0 和 C_1 的方差, $\mu(t)$ 表示直方图上从 0 到 t 灰度级的一阶累积矩。上述各式的详细推导可参看文献[3]。

(1)式的三个准则函数是等价的^[3], 由于 σ_T^2 与 t 无关, $\sigma_B^2(t)$ 的计算最简单, 因此最佳阈值 t^* 可以通过求 $\sigma_B^2(t)$ 的最大值得到, 即

$$\sigma_B^2(t^*) = \max_{0 \leq t < L-1} \{\sigma_B^2(t)\} \quad (2)$$

文献[1—3]列举了许多例子证实, Otsu 法是一种有效的阈值化方法, 并且由于 $\sigma_B^2(t)$ 的计算只包含零阶累积矩 $\omega_0(t)$ 和一阶累积矩 $\mu(t)$, 因此 Otsu 法只需要相对少的计算时间。

3. c 均值聚类算法^[4]及其在图象分割中的快速实现

设 $X = \{x_1, x_2, \dots, x_n\} \subset R^p$ 是一有限数据集, R^p 是 p 维欧氏空间; V_{cn} 是所有 $c \times n$ 阶矩阵集合, c 是满足 $2 \leq c < n$ 的整数; 那么 X 的 c 类划分空间为

$$M_c = \{U \in V_{cn} | u_{ik} \in \{0, 1\}, \forall i, k; \sum_{i=1}^c u_{ik} = 1, \forall k; 0 < \sum_{k=1}^n u_{ik} < n, \forall i\} \quad (3)$$

这里 u_{ik} 称为隶属度, $u_{ik} = 1$ (或 0), 表示第 k 个数据 x_k 属于 (或不属于) 第 i 类。设 $V = \{v_1, v_2, \dots, v_c\}$ 表示各类的聚类中心, 其中 $v_i \in R^p$, $1 \leq i \leq c$ 。定义一个均方误差和准则函数 J_c 为

$$J_c(U, V) = \sum_{i=1}^c \sum_{k=1}^n u_{ik} d_{ik}^2 \quad (4)$$

其中 $d_{ik} = \|x_k - v_i\|$ 表示 x_k 与 v_i 的欧氏距离。通过最小化准则函数 $J_c(U, V)$, 可以得到数据集 X 的最佳 c 类划分。该划分可以通过如下的迭代优化算法实现。

(1) 取定 c 值, $2 \leq c < n$, 初始化 $U^{(0)} \in M_c$ 。依次取迭代步数 $b = 0, 1, 2, \dots$ 。

(2) 利用 $U^{(b)}$ 和下式计算 c 个均值 $\{v_i^{(b)}\}$:

$$v_i = \left(\sum_{k=1}^n u_{ik} x_k \right) / \left(\sum_{k=1}^n u_{ik} \right), \forall i \quad (5)$$

(3) 按如下方式调整 $U^{(b)}$ 为 $U^{(b+1)}$:

$$u_{ik}^{(b+1)} = \begin{cases} 1, & d_{ik}^{(b)} = \min_{1 \leq i \leq c} \{d_{ik}^{(b)}\}, \\ 0, & \text{其它}, \end{cases} \quad \forall i, k \quad (6)$$

直接将上述算法应用于图象分割, 由于 $n = N$, 计算量是比较大的。考虑到图象的灰度级数 L 通常满足 $L \ll N$, 若聚类过程针对灰度直方图进行, 便可以大大提高算法速度。此时由于 $x_k, v_i \in R'$, 以下将它们记为 x_k 和 v_i 。 $x_k \in \{0, 1, \dots, L-1\}$, 若 $x_k = l$, 所有值为 l 的 x_k ($k \in \{1, 2, \dots, N\}$) 的点数为 n_l , 那么对二值分割, (4)式的准则函数为

$$J_c = \sum_{i=1}^2 \sum_{k=1}^N u_{ik}(x_k - v_i)^2 = \sum_{i=1}^2 \sum_{l=0}^{L-1} u_{il}n_l(l - v_i)^2 \quad (7)$$

n_l 即为图象灰度直方图上灰度值为 l 的像素点数, u_{il} 此时表示灰度为 l 的像素对第 i 类的隶属度. 原算法中的(5)式和(6)式此时为

$$v_i = \left(\sum_{l=0}^{L-1} u_{il}n_l l \right) / \left(\sum_{l=0}^{L-1} u_{il}n_l \right), \quad \forall i \quad (8)$$

$$u_{il}^{(b+1)} = \begin{cases} 1, & [d_{il}^{(b)}]^2 < [d_{2l}^{(b)}]^2, \\ 0, & \text{其它}, \end{cases} \quad u_{2l}^{(b+1)} = 1 - u_{il}^{(b+1)}, \quad \forall l \quad (9)$$

利用(8)和(9)式实现的算法与采用(5)和(6)式实现的算法产生完全相同的分割结果, 但由于 $L \ll N$, 计算速度大大提高. 在(9)式中采用 $d_{il}^{(b)}$ 来进行距离比较, 可以省去费时的开方运算.

4. Otsu 法与 c 均值聚类算法的等效性证明

在 Otsu 法中, 由于 $\sigma_w^2 = \sigma_0^2(t) + \sigma_1^2(t)$, σ_w^2 与 t 无关, 因此搜索 $\sigma_w^2(t)$ 的最小值同样可以得到相同的最佳阈值 $t^*[3]$. 以下有

$$\begin{aligned} \sigma_w^2(t) &= \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t) \\ &= \left(\sum_{i=0}^t p_i \right) \left[\sum_{i=0}^t (i - \mu_0(t))^2 p_i / \omega_0(t) \right] \\ &\quad + \left(\sum_{i=t+1}^{L-1} p_i \right) \left[\sum_{i=t+1}^{L-1} (i - \mu_1(t))^2 p_i / \omega_1(t) \right] \\ &= \frac{1}{N} \left(\sum_{i=0}^t n_i \right) \left[\sum_{i=0}^t \left(i - \sum_{j=0}^i j n_j / \sum_{j=0}^i n_j \right)^2 \left(n_i / \sum_{j=0}^i n_j \right) \right] \\ &\quad + \frac{1}{N} \left(\sum_{i=t+1}^{L-1} n_i \right) \left[\sum_{i=t+1}^{L-1} \left(i - \sum_{j=t+1}^{L-1} j n_j / \sum_{j=t+1}^{L-1} n_j \right)^2 \left(n_i / \sum_{j=t+1}^{L-1} n_j \right) \right] \\ &= \frac{1}{N} \sum_{i=0}^t \left(i - \sum_{j=0}^i j n_j / \sum_{j=0}^i n_j \right)^2 n_i \\ &\quad + \frac{1}{N} \sum_{i=t+1}^{L-1} \left(i - \sum_{j=t+1}^{L-1} j n_j / \sum_{j=t+1}^{L-1} n_j \right)^2 n_i \end{aligned}$$

上式中, $\left(\sum_{j=0}^i j n_j / \sum_{j=0}^i n_j \right)$ 和 $\left(\sum_{j=t+1}^{L-1} j n_j / \sum_{j=t+1}^{L-1} n_j \right)$ 分别表示 C_0 类和 C_1 类的类中心值, 用 v'_0 和 v'_1 表示. 另外 i 和 n_i 分别表示灰度值和对应灰度值 i 的像素点数. 为了便于与 c 均值算法比较, 将它们表示为 l 和 n_l . 此时有

$$N\sigma_w^2(t) = \sum_{l=0}^t n_l(l - v'_0)^2 + \sum_{l=t+1}^{L-1} n_l(l - v'_1)^2 = \sum_{i=0}^1 \sum_{l=0}^{L-1} u'_{il}n_l(l - v'_i)^2 \quad (10)$$

这里

$$u'_{0l} = \begin{cases} 1, & \text{当 } l = 0, 1, \dots, t, \\ 0, & \text{当 } l = t+1, t+2, \dots, L-1, \end{cases} \quad u'_{1l} = 1 - u'_{0l} \quad (11)$$

$$v'_i = \left(\sum_{l=0}^{L-1} u'_{il}n_l l \right) / \left(\sum_{l=0}^{L-1} u'_{il}n_l \right), \quad i = 0, 1 \quad (12)$$

比较(7)式和(10)式、(8)式和(12)式,可以看出,尽管 Otsu 法与 c 均值聚类算法的出发点不尽相同,但前者的一个等价的准则函数 $\sigma_w^2(t)$ 实质上也是基于最小均方误差准则,因此当 $\sigma_w^2(t)$ 和 J_c 均取最小值时,由此产生的最佳分类是完全相同的。

5. 实验结果和结语

实验采用了 10 幅 256×256 大小,灰度范围为 0—255 的真实图象进行分割,本文的算法与 Otsu 算法均产生完全相同的分割结果。图 1(a) 和 (b) 是其中的两幅图象, (c) 和 (d) 是相应的分割结果。可以看出,分割效果是令人满意的。算法均在时钟 8MHz 的 IBM-PC/AT 微机实现。10 幅图象的实验结果表明, Otsu 法的计算时间平均为每幅 25s, 本文的算法平均每幅只用 2s。

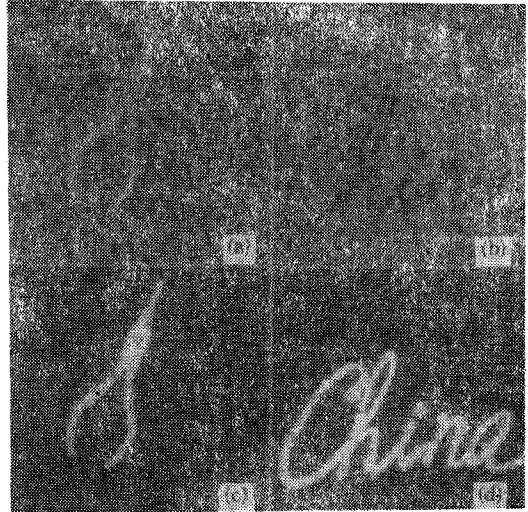


图 1 实验中采用的两幅图象及分割结果

本文提出了一种适合图象分割的高效的 c 均值聚类算法,它与 Otsu 法有着完全相同的良好分割性能,但本文的算法计算速度约提高了一个数量级,因此具有更大的实用价值。

参 考 文 献

- [1] P. K. Sahoo et al., *Computer Vision, Graphics, and Image Processing*, 41(1988)2, 233—260.
- [2] S. U. Lee et al., *Computer Vision, Graphics, and Image Processing*, 52(1990)2, 171—190.
- [3] N. Otsu, *IEEE Trans. on SMC*, SMC-9(1979)1, 62—66.
- [4] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, New York: Plenum, (1981).

THRESHOLDING OF IMAGES USING AN EFFICIENT c -MEAN CLUSTERING ALGORITHM

Liu Jianzhuang Tu Yuqing

(Xidian University, Xi'an 710071)

Abstract Otsu's method (1979) is considered as a good thresholding method for image segmentation. In this paper, an efficient c -mean clustering algorithm which is suitable for image segmentation is proposed. It yields the same results as Otsu's method, but its computational time is about one order of magnitude less than that of Otsu's method.

Key words Image segmentation; Threshold; c -means clustering algorithm