

基于孤立点发现的彩色图像人脸边缘提取算法^{*}

邵峰晶 孙仁诚 郭振波

(青岛大学信息工程学院 青岛 266071)

摘要 本文针对彩色人脸边缘提取问题,分析了运用孤立点分析方法进行彩色图像人脸边缘提取的可行性,给出并实现了基于孤立点分析的彩色人脸边缘提取算法——RGB-OEA。在此基础上,通过实例应用证明了算法的有效性。

关键词 孤立点,边缘提取,数据挖掘

An Algorithm of Edge Extraction of Color Face Based on Outlier Detection

SHAO Feng-Jing SUN Ren-Cheng GUO Zhen-Bo

(College of Information Engineering, Qingdao University, Qingdao 266071)

Abstract This paper focuses on the problem of edge extraction of color face images, analyzes the validity of edge extraction by outlier detection algorithm, and also proposes an edge extraction algorithm based on outlier detection method. The validity of the new algorithm has been verified by some instances.

Keywords Outlier, Edge extraction, Data mining

1 引言

人脸图像识别是模式识别、计算机视觉和人工智能领域的一个热点。人脸的边缘提取是人脸图像识别的基础和前提。传统的人脸边缘提取方法主要是利用边缘检测算子进行边缘的提取,然后根据人脸轮廓特点去除杂散的冗余边缘并进行边缘的修补^[1,2]。

孤立点发现是数据挖掘^[3,4]中重要的研究内容,其研究目的是发现数据集中的小模式,也就是数据集中被认为与其它数据不相似或不一致的数据对象。而人脸部边缘的像素值与其它像素值有着明显的差别,由此,可以将边缘的像素作为孤立点通过孤立点发现算法实现人脸边缘检测。

本文提出了一种基于孤立点分析方法的彩色图像人脸边缘的提取算法,给出了算法的详细描述和实现。最后,通过实例应用证明了算法的有效性。

2 基于距离的孤立点定义

基于单元的孤立点发现算法^[5]是基于距离的孤立点发现算法中的一种。下面给出基于距离的孤立点的定义。

定义 1 S 为集合 T 中一子集,对于 T 中某一元素 p 和给定常数 $D(D>0)$ 都有 $\|p-q\|>D$,其中 q 为 S 中任意元素,则称 S 为 T 中 p 的超 D 集。记为 S_p 。

定义 2 集合 T 中某一元素 p ,对于给定的正整数 M ,若 T 中存在 p 的一个超 D 集 S_p ,且 S_p 的基数大于 $\|T\|-M$,则称 p 为 T 中关于超 D 集 S_p 和 M 的孤立点。记为 $DB(p, M, S_p)$ 。

3 基于单元的孤立点算法

3.1 RGB 空间中单元格的划分和像素的分配

E. K 和 R. Ng^[5]给出了基于单元的孤立点算法的思路:在 E. K 和 R. Ng 思路的基础上,给出了彩色空间中的基于单元的孤立点算法的详细描述。

设 T 为 RGB 空间中像素的集合,记 C_{ijk} 为第 i, j, k 位置的单元格, T 中对象,即像素为 (r_t, g_t, b_t) 。称 $l=D/(2\sqrt{3})$ 为 RGB 空间中单元格的边长。

对于给定的数据集,给出了以下函数对 RGB 空间进行单元格划分:

$$\begin{cases} A = \left\lceil \frac{\max_{1 \leq t \leq n} r_t - \min_{1 \leq t \leq n} r_t}{l} \right\rceil \\ B = \left\lceil \frac{\max_{1 \leq t \leq n} g_t - \min_{1 \leq t \leq n} g_t}{l} \right\rceil \\ C = \left\lceil \frac{\max_{1 \leq t \leq n} b_t - \min_{1 \leq t \leq n} b_t}{l} \right\rceil \end{cases} \quad (1)$$

其中, $t \in Z^+$, n 为数据集中对象的数目。

给出了以下函数用来将像素值 (r_t, g_t, b_t) 分配到相应的单元格中:

$$\begin{cases} i = \left\lceil \frac{r_t - \min_{1 \leq t \leq n} r_t}{l} \right\rceil \\ j = \left\lceil \frac{g_t - \min_{1 \leq t \leq n} g_t}{l} \right\rceil \\ k = \left\lceil \frac{b_t - \min_{1 \leq t \leq n} b_t}{l} \right\rceil \end{cases} \quad (2)$$

其中, $t \in Z^+$, n 为数据集中对象的数目,其中 $\lceil a \rceil$ 表示不小于 a 的最小整数, Z^+ 表示正整数。

定义 3 若单元格 C_{ijk} 满足

$$(i-2) * (A-i-2) \leq 0 \quad \text{或}$$

$$(j-2) * (B-j-2) \leq 0 \quad \text{或}$$

^{*} 山东省科技攻关重点计划(12090101);科技部中小企业创业基金(00c26223710313);科技部推广应用计划(991144004);山东省高新技术产业发展专项(鲁计高技(2000)13330)。邵峰晶 博士,教授,博士生导师,主要研究领域为智能信息处理技术、嵌入式技术。孙仁诚 硕士,讲师,研究领域为数据仓库与数据挖掘技术。郭振波 学士,高级实验师,研究领域为智能信息处理技术。

$$(k-2) * (C-k-2) \leq 0$$

$i, j \in Z^+, 0 < i < A, 0 < j < B, 0 < k < C$, 称 C_{ijk} 为边界单元格。

运用数据集维度的最大、最小值进行单元格划分,既可保证将数据集中每一点都能分配到相应的单元格中,也保证了单元格的划分数目最少,从而减少了时间复杂度和空间复杂度。

3.2 RGB 空间单元格的结构和性质

为了定义和分析 RGB 空间单元格的结构和性质,首先给出 C_{ijk} 的第一层邻居和第二层邻居的概念,及其它们的相关性质。

设经单元格的划分和像素的分配后,像素集 T 被分配到相应的单元格中。

定义 4 对于单元格 C_{ijk} , 称下列集合(n 为维度 t 上单元格数目)

$$\{C_{uvw} \mid u=f(i), v=f(j), w=f(k), C_{uvw} \neq C_{ijk}\}$$

$$\text{其中 } f(t) = \begin{cases} t, t=1 \text{ 或 } t=n \\ t \pm 1, 1 < t < n \end{cases} \quad t \in Z^+$$

中每一单元格为 C_{ijk} 的第一层邻居,该集合称为 C_{ijk} 的第一邻居层,记为 $L_1(C_{ijk})$ 。

由定义 4 可以看出非边界单元格有 26 个第一层邻居,而边界单元格有 7 或 11 个。

由单元格边长的定义及定义 4 可得如下定理。

定理 1 同一单元格中的像素之间的最大距离至多为 $D/2$ 。

定理 2 若 C_{uvw} 是 C_{ijk} 的第一层邻居,则 C_{uvw} 中任一像素 P 与 C_{ijk} 中任一像素 Q 之间的距离至多为 D 。

定义 5 对于单元格 C_{ijk} , 称下列集合

$$\{C_{uvw} \mid u=f(i), v=f(j), w=f(k), C_{uvw} \notin L_1(C_{ijk}), C_{uvw} \neq C_{ijk}\},$$

中每一单元格为 C_{ijk} 的第二层邻居,该集合称为 C_{ijk} 的第二邻居层,记为 $L_2(C_{ijk})$ 。

其中,

$$f(t) = \begin{cases} 1, t \leq 3 \\ n, t > n-3, t \in Z^+ \\ t \pm 3, 3 < t < n-3 \end{cases}$$

n 为维度 t 上单元格数目

由定义可得,非边界单元格第二邻居层含有两层单元格,单元格的总数目为 $7^3 - 3^3 = 316$ 。

根据定义 5 可推出定理 3。

定理 3 若 $C_{uvw} \neq C_{ijk}$ 既不是 C_{ijk} 的第一层邻居,也不是 C_{ijk} 的第二层邻居,则 C_{uvw} 中任一像素 P 与 C_{ijk} 中任一对象 Q 之间的距离定大于 D 。

上述定理的具体证明见文[5]。基于上述定理给出了定理 4。

定理 4 (1)若单元格中对象数目大于 M ,则该单元格及其第一邻居层中不存在孤立点。

(2)若单元格及其第一邻居层中对象数目大于 M ,则该单元格中不存在孤立点。

(3)若单元格及其第一、二邻居层对象的数目小于 M ,则单元格中所有对象都为孤立点。

可由定义 2 和定理 1、2 直接得 4(1);由定理 2 得 4(2);4(3)可由定理 3 推出。

3.3 RGB 空间中阈值 M 的边界处理

在以上的分析中,阈值 M 被作为常量处理,即 M 的取值不根据单元格所在位置而动态调整,由此易造成边界处的像素值误判。下面将从定理 4(2)、4(3)两方面来分析:

由单元格划分方法和定义 4 可知,边界单元格 C_{ijk} 的 $L_1(C_{ijk})$ 中单元格数目少于非边界的数目,从而在像素值同等分布的前提下, $C_{ijk} \cup L_1(C_{ijk})$ 中对象数目将比非边界单元格及其 $L_1(C_{ijk})$ 中的少。由定理 4(2)可得,边界单元格中的像素值因此将更有可能被误判为孤立点。同理,对于定理 4(3),边界单元格的 $C_{ijk} \cup L_1(C_{ijk}) \cup L_2(C_{ijk})$ 中对象数目将更可能小于 M ,从而也就增加了误判的机率。为此需对阈值 M 进行动态调整。

由上述分析可见,对 M 进行调整的有效方法就是根据单元格所在位置,也就是根据单元格 C_{ijk} 的 $L_1(C_{ijk})$ 和 $L_2(C_{ijk})$ 单元格的数目来调整。基于此,本文针对定理 4(2)、4(3)的不同情况,提出了两个动态调整函数,从而解决了算法中阈值 M 的边界处理问题。

针对定理 4(2)的 M 调整函数为

$$M_{c1} = \lceil \frac{\xi(i)+1 * (\xi(j)+1) * (\xi(k)+1) * M_1}{3^3} \rceil \quad (3)$$

$$\text{其中, } \xi(t) = \begin{cases} t, t=1 \\ n-t, t=n \\ 2, 1 < t < n \end{cases} \quad t \in Z^+$$

n 为维度 t 上单元格数目。

该函数只处理单元格的第一邻居层,称为一层边界阈值调整函数。

针对定理 4(2)的 M 调整函数为

$$M_{c2} = \lceil \frac{\xi(i)+3 * (\xi(j)+3) * (\xi(k)+3) * M_1}{7^3} \rceil \quad (4)$$

$$\text{其中: } \xi(t) = \begin{cases} t, t \leq 3 \\ n-t, t \geq n-3 \\ 4, 3 < t < n-3 \end{cases} \quad t \in Z^+$$

n 为维度 t 上单元格数目,该函数能对单元格的第一、第二邻居层进行处理,称为二层边界阈值调整函数。

3.4 基于单元的孤立点挖掘算法

在上述定义和定理的基础上,给出了基于单元的彩色空间孤立点发现的算法 RGB-OEA:

[RGB-OEA 算法]

Step1 对 RGB 色彩空间进行单元格划分。

Step2 将每个单元格 C_{ijk} 的对象数目 $Count_{0ijk}$ 赋初值为 0。

Step3 对于数据集中的每一个对象 P ,将 P 分配到相应的单元格 C_{ijk} 内,在 C_{ijk} 内存储 P ,再将 C_{ijk} 的 $Count_{0ijk}$ 加 1。

Step4 判断每个单元格 C_{ijk} 的 $Count_{0ijk}$ 是否大于 M ,若成立,将 C_{ijk} 置为 red。

Step5 对于每个为 red 的单元格 C_{ijk} ,若它的第一层邻居不是 red,将它们置为 pink。

Step6 对于每一个既不是 red 也不是 pink 的单元格 C_{ijk} :

a. 计算 C_{ijk} 及其 $L_1(C_{ijk})$ 中所有对象的数目 $Count_{1ijk}$ 的值

b. 如果 $Count_{1ijk} > M_{c1}$,将 C_{ijk} 置为 pink

c. 否则,1. 计算 C_{ijk} 及其 $L_1(C_{ijk})$ 、 $L_2(C_{ijk})$ 中所有对象的数目 $Count_{2ijk}$ 的值

2. 如果 $Count_{2ijk} \leq M_{c2}$,将 C_{ijk} 中的所有对象标记为孤立

点

3. 否则,对于 C_{ijk} 中的每一个对象 P ;

I. 将 $Count_{2,ijk}$ 的值赋给 $Count_p$

II. 对于 $L_2(C_{ijk})$ 中的每一个对象 Q , 计算 PQ 之间的距离 $dist$, 如果 $dist \leq D$, 则 $Count_p$ 加 1, 若 $Count_p > M_{L_2}$, P 不是孤立点, 执行 6(c)(3); 否则, 标记 P 是孤立点。

在算法中, 从 step1 到 step6(c)(2) 都是以单元格为基本单位来判定孤立点, 而不是以像素为单位, 这将大大减少像素间的运算, 从而减少算法的时间复杂度, 提高了边缘检测的效率; step6(c)(3) 是以对象为单位来判断该对象是否为孤立点, 在计算过程中只需计算对象与其所在单元的第二邻居层中对象间的距离, 而不需计算第二邻居层以外的其它对象, 从而也减少了计算时间。

4 实验结果

为了测试算法的有效性, 选择了一幅彩色头像照片进行实验。照片扫描后存储为 BMP 格式的图像文件, 图像的色彩是 24 位, 大小为 150×200 总供包含 30000 个像素。下面对实验结果进行说明。

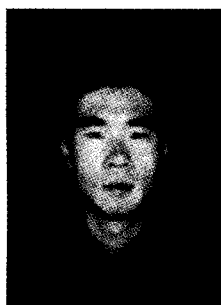


图 1



图 2

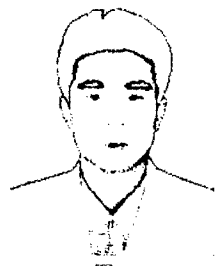


图 3

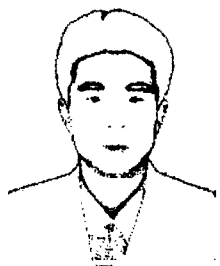


图 4

图 1 是原图像, 图 2、3、4 是当输入的 $D=50, M=70, 100$ 、

150 时, 所得的结果图。由图像可以清楚地看到, 随着 M 的变大, 头像的边缘变得越来越清晰, 此时, 我们得到的孤立点数分别为 734、1266、1703 个像素。因而, 可以通过调节 M 的值, 来得到不同的孤立点数目, 从而也就得到了不同清晰度的边缘图像。同样我们也可以通过调整 D 的值来取得边缘不同清晰度的图片, 下面是取 $M=12, D$ 分别为 23.9、20、17.9, 与此相应的单元格个数分别为 50653、54834、125000, 发现的孤立点分别为 734、1271、1734。在 M 不变的情况下, 随着单元格数目的增加, 所得到的孤立点的数目也相应的增加, 从而使提取的图像边缘也越来越清晰。下图是提取的结果。



图 5

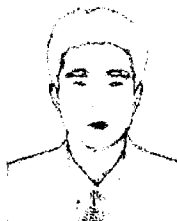


图 6



图 7

对比两种调整方式不难发现, 在所获得的孤立点数目大致相同的条件下, 固定 D 只调整 M 所提取的图像边缘要比固定 M 只调整 D 所提取的清楚, 但是, 这种方式(固定 D , 调整 M)对人面部特征的提取却不够理想, 比较一下, 这两种方式对人嘴的提取, 不难看出, 后一种方式(固定 M 调整 D)能够得到较好的结果, 因而, 在提取的过程中, 可以根据需求的不同, 选择合适的提取方式。

结论 对于人脸的边缘提取问题, 基于单元的孤立点分析算法是一种高效、准确的方法, 它不仅能够准确地提取彩色图像人脸边缘, 而且可以通过调整 D, M 的值来获得不同清晰度的边界, 从而增加了算法的灵活性。试验结果证明了 RGB-OEA 算法的有效性。

参考文献

- 1 Duda R O, Hart P E. Pattern Classification and Scene Analysis. Wiley, New York, 1973
- 2 Kirsch R. Computer Determination of the Constituent structure of Biological Images. Computer and Biomedical Research, 1971, 4 (3): 315~328
- 3 Petrovskiy M I. Outlier Detection Algorithms in Data Mining Systems. Programming and Computer Software, 2003, 29(4): 228~237
- 4 Han Jiawei, Kamber M. Data Mining Concepts and Techniques, 2001
- 5 Knorr E, Ng R, Tucakof V. Distance-Based Outliers: Algorithms and Applications. Very Large Data Bases (VLDB) Journal, 2000
- 6 don; UCL Press, 1996, 196~202
- 7 Lauritzen S L, Spiegelhalter D J. Local computations with probabilities on graphical structures and their application to expert systems [J]. Roy. statist. Soc. Ser B, 1988, 50(11): 17~24
- 8 Lepar V, Shenoy P P. A Comparison of Lauritzen -Spiegelhalter, Hugin and Shenoy -Shafer Architectures for Computing Marginals of Probability Distributions [M]. In G. Cooper and S. Moral, editors, UAI, 1998. 328~337
- 9 Dagum P, Luby M. An optimal approximation algorithm for Bayesian inference [J]. Artificial intelligence, 1997, 93: 1~27
- 10 Shafer G R, Shenoy P P. Local computation in hypertrees [R]. Technical Report 201, School of Business, Uni. Kansas, 1988
- 11 Murphy K. Active learning of causal Bayes net structure [R]. Technical report, Comp. Sci. Div., UC Berkeley, 2001
- 12 Ng A Y, Jordan M I. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes [C]. In NIPS-14, 2002. 245~250
- 13 Spirtes P, Glymour C, Scheines R. Causation, Prediction and Search [M]. MIT Press, 2000. 78~85

(上接第 168 页)

复杂度降低为线性复杂度; 同时在较小样本条件下与似然权重算法的推理精度对比实验表明, PPJT 算法可以保证更高的不确定推理精度。对于解决如何在一般复杂信念网络中应用概率推理实现可信的统计决策问题, PPJT 算法提供了一种高效可行的解决思路和方法。下一步迫切需要研究的工作内容是如何提高更小样本条件下的推理精度。因为对于诸如风险分析这类问题常常无法提供这样大的样本先验, 而且共轭先验条件也本非总能得到满足。随着针对信念网络的不确定推理方法和不确定学习方法的发展, 针对这类问题的研究必将会受到越来越广泛的关注^[8]。

参考文献

- 1 Jensen F V. An Introduction to Bayesian Networks [M]. Lon-