

数字图像研究分析

王向阳 陕西国际商贸学院 陕西咸阳

【文章摘要】

本文以图像为研究对象,详细论述了图像分割的研究方法和主要应用领域。对现有的部分图像分割算法进行分析和研究。

【关键词】

Microsoft; Visual; C++6.0; 阈值分割

1 图像分割概述

早期的图像分割方法可以分为两大类。一类是边界方法,这种方法假设图像分割结果的某个子区域在原来图像中一定会有边缘存在;一类是区域方法,这种方法假设图像分割结果的某个子区域一定会有相同的性质,而不同区域的像素则没有共同的性质。这两种方法都有优点和缺点,有的学者考虑把两者结合起来进行研究。现在,随着计算机处理能力的提高,很多方法不断涌现,如基于彩色分量分割、纹理图像分割。所使用的数学工具和分析手段也是不断的扩展,从时域信号到频域信号处理,小波变换等等。

图像分割是将整个图像区域分割成若干个互不交叠的非空子区域的过程,每个子区域的内部是连通的,同一区域内部具有相同或相似的特性,这里的特性可以是灰度、颜色、纹理等。对于灰度图像来说,区域内部的像素一般具有灰度相似性,而在区域边界上一般具有灰度不连续性。图像分割的数学描述如下:

令集合 R 代表整个图像区域,对 R 的图像分割可以看作是将 R 分成 N 个满足以下条件的非空子集 $R_1, R_2, \dots, R_N$:

$$(1) \bigcup_{i=1}^N R_i = R$$

$$(2) \text{对 } i=1, \dots, N, P(R_i) = \text{TRUE}$$

$$(3) \text{对 } \forall i, j, i \neq j, \text{有对 } R_i \cap R_j = \emptyset$$

$$(4) \text{对 } \forall i, j, i \neq j, P(R_i \cup R_j) = \text{FALSE}$$

$$(5) \text{对 } i=1, \dots, N, R_i \text{ 是连通的区域}$$

$P(R_i) = \text{TRUE}$ 指出在分割结果中,每个子区域的像素有着相同的特性。 $P(R_i \cup R_j) = \text{FALSE}$ 表示在分割结果中,不同的子区域具有不同的特性,它们没有公共的特性。 $\bigcup_{i=1}^N R_i = R$ 表示分割的所有子区域的并集

就是原来的图像,这一点非常重要,因为这一点是保证图像中每个像素都被处理的充分条件。

2 图像分割的研究意义

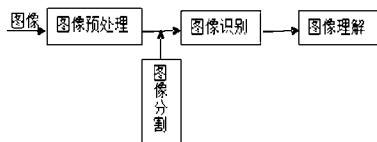


图 1-1 图像分割在整个图像处理过程中的作用

图像识别的基础是图像分割,其作用是把反映物体真实情况的战局不同区域的、具

有不同特性的目标区分开来,并形成数字特征。图像分割质量的好坏直接影响到后续图像处理的效果,甚至决定其成败,因此图像分割的作用是至关重要的。

图像分割是图像分析和理解的第一步,在如下的科学研究和工程技术领域有着广泛的应用。

(1) 工业图像处理:矿藏分析、探伤分析、无接触式检测等。

(2) 军事图像处理:军事目标检测和定位、地形配准、无人驾驶飞机、军事导航系统、地形侦察等。

(3) 生物医学图像处理:计算机断层图像(CT Computer Tomograph)、X光透视、核磁共振图像(MRI)、B超体内病变检测、各种细胞自动计数、病毒细胞的自动检测和识别、生物图片分析等。

(4) 图像传输:数字电视、高清晰度电视(HDTV)、多媒体信息处理、多媒体编码、传输、可视电话、会议电视等。

(5) 文本图像分析处理和识别:文字识别、版面分析和理解等。

(6) 身份鉴定:指纹识别、虹膜识别等。

(7) 机器人视觉:水下机器人、自动化生产线、无人驾驶汽车。

3 图像分割方法概述

分割的问题的困难在于图像数据的模糊和噪声的干扰。到目前为止,还没有一种或者几种完善的分割方法,可以按照人们的意愿准确地分割任何一种图像。实际图像中景物情况各异,具体问题具体分析,需要根据实际情况选择适合的方法。分割结果的好坏或者正确与否,目前还没有一个统一评价判断准则,分割的好坏必须从分割的效果和实际的应用场景来判断。

早期的图像研究中,图像分割的方法主要可以分成两大类。一类是边界方法,这种方法的假设是图像分割结果的某个子区域在原来图像中一定会有边缘存在;一类是区域方法,这种方法的假设是图像分割结果的某个子区域一定会有相同的性质,而不同区域的像素则没有共同的性质。

4 数字图像基本知识

4.1 BMP 位图文件解析

BMP 文件由文件头、位图信息头、颜色信息和图形数据四部分组成。

4.1.1 BMP 文件头信息

BMP 文件头数据结构含有 BMP 文件的类型、文件大小和位图起始位置等信息。其结构定义如下:

```
typedef struct
tagBITMAPFILEHEADER{
    WORD    bfType;        // 位图文件的类型,必须为 BM
    DWORD    bfSize;       // 位图文件的大小,以字节为单位
    WORD    bfReserved1;   // 位图文件保留字,必须为 0
    WORD    bfReserved2;   // 位图文件保留字,必须为 0
}
```

DWORD bfOffBits; // 位图数据的起始位置,以相对于位图文件头的偏移量表示,以字节为单位

} BITMAPFILEHEADER;

4.1.2 位图信息头

BMP 位图信息头数据用于说明位图的尺寸等信息。其结构定义如下:

```
typedef struct
tagBITMAPINFOHEADER{
    DWORD    biSize;       // 本结构所占字节数
```

LONG biWidth; // 位图的宽度,以像素为单位

LONG biHeight; // 位图的高度,以像素为单位

WORD biPlanes; // 目标设备的级别,必须为 1

WORD biBitCount; // 每个像素所需的位数,必须是 1(双色)、4(16 色)、8(256 色)或 24(真彩色)之一

4.1.3 数据读取和颜色分离

BMP 文件有个重要特性,那就是对于数据区域而言,每行的数据它必须凑满 4 字节,如果没有满,则用冗余的数据来补齐。这个特性直接影响到我们读取位图数据的方法,因为在我们看来 x, y 的数据应该在 $y * \text{width} + x$ 这样的位置上,但是因为会有冗余信息,那么必须将 width 用 $\text{width} + \text{该行的冗余量}$ 来处理,而由于位图文件有不同的位数,所以这样的计算也不尽相同。

对于 8 位的位图:

```
int pitch;
```

```
if(width%4!=0)
```

```
pitch=width;
```

```
else
```

```
pitch=width+4-width%4;
```

$\text{index} = \text{buffer} * \text{pitch} + x$; // 因为 8 位位图的数据区域存放的是调色板索引值,所以只需读取这个 index

颜色分离:

```
UCHAR r=quad[index].rgbRed;
```

```
UCHAR g=quad[index].rgbGreen;
```

```
UCHAR b=quad[index].rgbBlue;
```

4.2 数字图像的表达

模拟图像采样和量化的结果是一个实数矩阵。假定一幅模拟图像经过采样得到的数字图像有 M 行 N 列,坐标值变成了离散值。为了方便起见,用整数值来表示离散坐标。这样原点坐标为 $(0,0)$,第一行紧接原点的像素坐标值为 $(0,1)$,这样数字图像可用矩阵表示为:

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

矩阵中的每一个元素代表数字图像的一个像素。

5 总结

在图像为研究对象,论述了图像分割的研究方法和主要应用领域,并得出数据读取和颜色分离结果。

【参考文献】

- [1] 姚敏,等编著.《数字图像处理》机械工业出版社
- [2] 杨枝灵,王开,等,编著《Visual C++ 数字图像获取、处理及实践应用》,人民邮电出版社