

基于标识的变位对折压缩算法研究

周华君, 丁爱芬, 吕小俊

(云南大学旅游文化学院 信息学院, 云南 丽江 674199)

摘要: 为提高数据的无损压缩效率, 采用基于标识的变位对折压缩算法对基于动态规划的图像无损压缩算法进行编码改进, 将待压缩数据片段中的较大值数据片段进行对折变换, 减少信息的表示位长度, 增加数据雷同度, 从而减少第1步动态规划方法压缩的数据段长度, 同时增加第2步哈夫曼编码的数据重叠度, 提高编码率。结果表明, 通过动态规划、变位对折、哈夫曼编码3种方法的整合, 提高了数据无损压缩效率。

关键词: 动态规划; 无损压缩; 变位对折; 哈夫曼编码

中图分类号: TP312 **文献标识码:** A **文章编号:** 1674-5639 (2019) 03-0093-06

DOI: 10.14091/j.cnki.kmxyxb.2019.03.020

Research of Identification-based Displacement on Folding Compression Algorithm

ZHOU Huajun, DING Aifen, LV Xiaojun

(College of Information, Tourism and Culture College of Yunnan University, Lijiang, Yunnan, China 674199)

Abstract: To improve the lossless compression efficiency of data, the identification-based displacement folding compression algorithm was used to improve the encoding of the image lossless compression algorithm based on dynamic programming. The larger data segments in the data segments to be compressed is folded to reduce the length of information representation and increase the similarity of data, thus reducing the length of data segments compressed by the first dynamic programming method and also increasing the data overlapping of Huffman Encoding in the second step to improve the coding rate. The results show that the efficiency of lossless data compression is effectively improved by integrating three methods: dynamic programming, displacement folding and Huffman Encoding.

Key words: dynamic programming; lossless data compression; displacement folding; Huffman Encoding

随着计算机应用技术的快速发展, 以及多媒体的日益普及, 数据呈现出爆炸状态。在这种情况下, 如何高效地进行数据传输和存储, 已成为计算机系统需要解决的一个问题。而数据压缩就是解决高效数据传输、存储的通用方法之一, 其在媒体数据传输、大数据存储等方面有着广泛的应用。但是数据压缩算法的好坏直接影响着压缩效率。

一般数据压缩算法的处理技术分为统计编码、预测编码、变换编码、混合编码, 这些编码都有其特殊的压缩条件和最优压缩环境^[1]。其中, 统计编码是基于不同数据特征(单个数据、数据片段、数据偏移)的出现概率进行特定的编

码压缩。预测编码是采用对实际值与预测值的预测误差进行相关性分析的方法, 在特定精度条件下, 通过减少比特编码实现数据压缩。变换编码是通过将数据空间采用空间映射算法, 减少数据量和冗余度, 再对变换后的信号进行编码。而混合编码是使用上述3种编码形式的任意组合进行编码。此外, 数据压缩技术从是否对信息有损角度又可分为有损编码和无损编码^[2-4]。其中, 有损编码的压缩效率一般为几十到几百倍, 但是数据信息存在必然的损失。无损编码的压缩效率在2~5倍^[5], 但数据信息无损失。

从算法时空效率的角度而言, 不同压缩方法的计算机实现存在着不同资源制约。如统计编码的统

收稿日期: 2018-12-25

作者简介: 周华君(1984—), 男, 湖北襄樊人, 讲师, 主要从事算法设计与信息安全研究。

计过程需要对数据进行全面计数, 如果压缩数据量大, 需要消耗大量的内存空间. 而变换编码涉及空间映射, 计算的时间复杂度高.

目前图像数据的压缩基本采用基于某种特征值的有损压缩^[3,6-7], 如小波变换^[8]、KPCA 等方法^[9-10], 但在高精度图像识别上, 有损压缩会产生较大噪声^[11-12].

基于上述原因, 本文提出一种基于标识的变位对折压缩算法, 设计一种可以快速高效连续流式压缩算法的协议, 并利用动态规划算法划分待压缩数据片段^[13], 通过变位对折提高数据重复率, 最后通过哈夫曼统计编码^[2], 实现数据片段的压缩. 由于算法通过在最优化片段上进行数据位的变位对折, 从而提高了哈夫曼编码的编码效率.

1 问题提出

对于 n 个待压缩的图像数据 $\{p_1, p_2, \dots, p_n\}$, 其中 p_i 表示像素点的值. 将其分割成 m 个连续片段 S_1, S_2, \dots, S_m , 其中在第 $i \sim 1$ 个像素片段 S_i 中 ($1 \leq i \leq m$), 有 $l[i]$ 个像素, 且该段中每个像素点都只用 $b[i]$ 位表示. 设 $t[i]$ 表示前 i 个连续段中 p_i 的个数, 即 $t[i] = \sum_{k=1}^{i-1} l[k]$, 于是第 i 个像素段 S_i 的元素位置表示为 $t[i] + 1$ 至 $t[i] + l[i]$, 则第 i 个像素段 S_i 的最大数据所占二进制空间为 $h_i = \lceil \log(\max_{t[i]+1 \leq k \leq t[i]+l[i]} p_k + 1) \rceil$, 因此只需要 3 个 bit 位表示 $b[i]$, 如果限制 $1 \leq l[i] \leq m$, 则只需要 8 个 bit 位表示 $l[i]$, 因此第 i 个像素段所需的存储空间为 $l[i] \times b[i]$ 位. 按此格式存储像素序列 $\{p_1, p_2, \dots, p_n\}$, 共需要 $\sum_{i=1}^m l[i] \times b[i] + 11m$ 位的存储空间. 如何使存储空间最少, 采用动态规划算法解决如下:

设 $l[i], b[i]$ 是 $\{p_1, p_2, \dots, p_n\}$ 的最优分段. 显而易见, $l[1], b[1]$ 是 $\{p_1, p_2, \dots, p_{l[1]}\}$ 的最优分段, 且 $l[i], b[i]$ 是 $\{p_{l[1]+1}, \dots, p_n\}$ 的最优分段, 即图像压缩问题满足最优子结构性^[13].

设 $S[i] (1 \leq i \leq n)$ 是像素序列 $\{p_1, p_2, \dots, p_n\}$ 的最优分段所需的存储位数. 由最优子结构性知:

$$s[i] = \min_{l \leq k \leq \min\{i, 256\}} \{s[i-k] + k \times b_{\max}(i-k+1, i)\} + 11,$$

其中 $b_{\max}(i, j) = \lceil \log(\max_{i \leq k \leq j} p_k + 1) \rceil$.

以此编写的压缩算法所需的计算时间为 $O(n)$ ^[13].

通过上述分析可知, 该算法具有如下特征:

1) 算法适用于数据比较集中的情况; 2) 如果 $b_{\max}(i, j)$ 越小, 算法压缩率越好; 3) 算法重点是去除多余的 0 bit 位, 保留有效的 1 bit 位.

2 算法改进

通过对算法处理结果的特征进行分析, 可知算法处理结果具有以下特点:

1) 算法的处理过程使得压缩结果中数据比特流 1 的数量急剧靠拢;

2) 算法的处理过程使得压缩结果中数据比特流 0 的个数大量减少;

3) $b_{\max}(i, j)$ 大小严重影响数据的压缩效率^[13].

综上, 算法的处理过程导致压缩结果中比特流 1 的数量急剧靠拢, 0 的数量大量减少, 这就造成了结果数据的重叠度显著提高, 为哈夫曼编码提供了良好的适用环境^[14]. 为提高数据的重叠度, 将 $b_{\max}(i, j) \geq 5$ 的片段进行对折变换, 进一步减少一次压缩结果的序列长度, 提高数据的重叠度.

将原有的 3 个 bit 的信息位进行重新定义: 令第 1 个 bit 位代表是否反转, 如果需要反转则置 1, 反之则置 0, 后两个 bit 标志数据所占存储空间位数, 最大值为 3, 代表占用 4 个 bit 的存放空间, 如表 1 所示.

表 1 各比特位标识定义

后两个 bit 值	代表所占空间的 bit 位
0	1
1	2
2	3
3	4

建立 0 到 15 的哈夫曼编码表, 并将 16 至 255 的数据经过对折法进行重新编码, 就可将数据完整集中在 4 个 bit 的位置上, 而信息位则可用小于 4 个 bit 的位数表示, 如表 2 所示.

表2 重排后的编码

实际数据	高4位	低4位	实际数据	高4位	低4位	实际数据	高4位	低4位
16	1	0	17	1	1	18	1	2
19	1	3	20	1	4	21	1	5
128	8	0	129	8	1	130	8	2
250	15	10	251	15	11	252	15	12
253	15	13	254	15	14	255	15	15

从重排后的编码（表2）可以看出，16到255中的任何一个数据的两部分均可以表示0到15范围。如何将数据的编码范围缩小到0~15以内，从统计的角度而言，海量数据在0~15值域内的重复程度非常高。而重复程度越好，采用哈夫曼编码的效率就会越好，同时采用图像的上述编码过程可以将数据中0出现次数降低，1出现次数相对比较集中，如此使得大数出现的概率急剧增加，也使得哈夫曼编码中大数的出现频率提高，两种方法综合使用可以使数据的压缩率显著提升。

为方便压缩算法的使用，需要设计一套合适的协议便于程序解析，设计协议如下：

- 1) 信息位的描述，如表3所示；
- 2) 变换信息位的描述，如表4所示；
- 3) 协议位的描述，如表5所示。

表3 信息位描述

实际压缩数据的个数	每个数据所占位数	实际数据
8 bit	3 bit	N bit
每个分段数据个数不超过255	每个数据不超过255，所占位数不超过8，用3 bit表示	经过去掉前面0后的数据

表4 变化后的信息位描述

实际要压缩数据的个数	是否需要对折	动态规划后要压缩的数据
8 bit	1 bit	3 bit + 实际数据
每个分段数据个数不超过255	如果要压缩的数据比较大，即原信息位的第2描述位≥2，采用对折方式，否则不变	变换前的后两个信息位进行压缩

表5 协议位描述

首部	数据报长度	数据报编码域长度	数据报实际数据起始位
数据报编号	数据粒度	哈夫曼编码表	压缩后数据的序列

其中各部分代表内容如下：

- 1) 首部. 代表压缩算法的信息描述；
- 2) 数据报长度. 代表要变换的数据报的大小，方便以后比较数据的完整性；
- 3) 数据报编码域长度. 代表对数据折半后的哈夫曼编码表的内容所占长度；
- 4) 数据报的实际数据起始位. 代表压缩后序列从总的数据报的何处开始；
- 5) 数据报编号. 方便对数据报进行序列重组；
- 6) 数据粒度. 代表数据压缩时处理单位的量，数据越多，粒度就越大，越方便开辟空间进行压缩，也可以表示成压缩算法的使用次数；
- 7) 哈夫曼编码表. 为压缩后数据的还原作参考，在实际压缩中可将上一次哈夫曼编码表和数据序列作为下一次的待压缩数据；
- 8) 压缩后数据的序列. 经过动态规划、变位对折及哈夫曼编码的实际压缩数据序列。

3 算法的逻辑描述

算法以未压缩的序列流为输入，采用Compress算法进行变位压缩，该算法的逻辑流程如下：

- 1) 输入序列流；
- 2) 采用Compress动态规划算法进行变位压缩；
- 3) 判定是否需要对折，若需要对折，转入4)，否则直接转入5)；
- 4) 对折并加入哈夫曼编码表，然后转入6)；
- 5) 直接加入哈夫曼编码表；
- 6) 执行哈夫曼编码，然后转入1)。

其中Compress算法逻辑流程如下：

- 1) 获取输入序列长度len，构建矩阵S[len]，初始化为0；
- 2) 初始化信息包头header = 11；
- 3) for i = 1； i <= len； i ++；

```

4) bmax = length (p [i]) //获取第 i 的像素
值的二进制位长度;
5) S [i] = S [i-1] + bmax;
6) L [i] = 1; //假设当前片段默认从 1 处
断开;
7) for J=2; j < i&j < len; j ++;
8) if bmax < b [i-j+1] then bmax = b [i-j
+1];
9) if S [i] >= S [i-j+1] + j * bmax then
S [i] = S [i-j] + j * bmax L [i] = j//存在比
当前最优片段划分更优片段划分, 则更新当前最优
片段划分为新片段划分;
10) end for;
11) S [i] += header; //给出当前片段的最
优划分;
12) end for;
13) 按照 L[i] 执行变位压缩.

```

算法通过向后移动的方法逐个判定当前片段的最优片段, 如果发现后一个新划分比前一个划分更优, 则以新划分作为当前最优划分. 依次求解输入序列的全局最优解.

4 算法实验

基于标识的变位对折压缩算法利用动态规划、变位对折和哈夫曼编码进行数据压缩变换, 其中动态规划算法的压缩效率受小数位片段的影响较大, 其压缩结果产生较多的 1 比特位和较少的 0 比特位, 哈夫曼编码受数据重叠度影响较大, 这里分别进行动态规划的移位编码算法实验、不同重叠度的哈夫曼编码实验和不同频度的 1 比特位压缩实验.

4.1 动态规划的移位编码算法实验

以 500 B 数据量为一个测试数据片段, 以 0.02 表示小数据 (小数据 ≤ 16 , 16 占 4 个 bit 位) 出现概率的步长 (0.02 表示小数据出现概率为 0.02), 生成随机序列, 对每个步长片段应用动态规划算法执行 500 次测试, 以 500 次测试的压缩值均值为衡量数据. 对比测试结果如图 1 所示, 其中横轴代表小数出现频率, 纵轴代表压缩后片段实际长度.

由图 1 可见, 小数值数据量越多, 压缩片段表示位越短, 压缩效率越高, 即压缩效率与小数值片

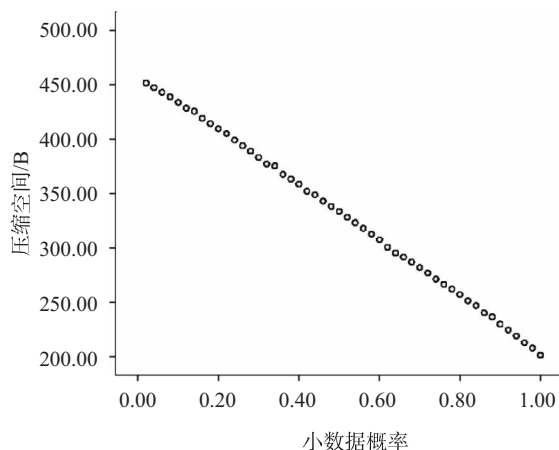


图1 不同概率密度下的小数据片段压缩效率

段概率成反比.

4.2 不同重叠度的哈夫曼编码实验

由于哈夫曼编码的算法效率取决于数据的重叠度, 这里以 256 B ($256 \times 8 = 2048$ bit) 为 1 个压缩片段, 测试不同数据波动范围的哈夫曼编码长度. 如图 2 所示, 横轴代表数据波动范围 (1 个字节的取值范围 0 ~ 256), 纵轴代表压缩片段长度.

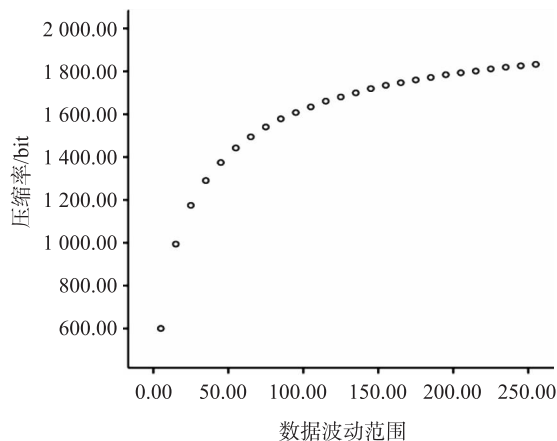


图2 不同数据波动范围下的哈夫曼编码压缩效率

由图 2 可见, 在 256 字节作为输入流的情况下, 数据编码范围越大 (即数据重叠度越低), 哈夫曼编码压缩效率越低 (表示数据所需要的空间越大). 通过曲线回归, 根据调整的 R^2 、标准误差和 p 值^[15], 最终选择对数模型 (表 6 ~ 表 8), 其表达式为:

$$y = 183.606 + 306.053 \times \ln(x), 0 \leq x \leq 256,$$

其中: y 代表压缩率, x 代表数据波动范围.

表6 对数模型容忍度

R	R^2	调整 R^2	标准误差
0.995	0.990	0.989	30.448

表7 对数模型方差检验

指标	平方和	df	均方	F 值	p
回归	2 146 219.020	1	2 146 219.020	2 314.994	0.000
残差	22 250.273	24	927.095		

表8 对数模型系数

指标	未标准化系数		标准化系数	<i>t</i>	<i>p</i>
ln（波动范围）	306.053	6.361	0.995	48.114	0.000
常数	183.606	29.701		6.182	0.000

4.3 不同频度的1 bit 位压缩实验

由于哈夫曼编码的算法效率取决于数据的重叠度, 这里以 256 B ($256 \times 8 = 2\,048$ bit) 为一个压缩片段, bit 位 1 出现频率的波动范围为 1 ~ 2 048, 通过 1 000 次实验取平均值, 得到的结果如图 3 所示, 其中横轴代表 2 048 bit 空间上不同频度的 1 bit 位数 (个), 纵轴代表实际压缩率。

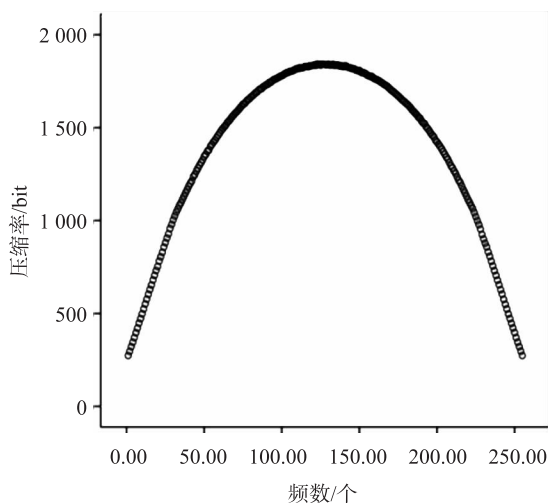


图3 不同频度的1 bit位哈夫曼编码压缩效率

从图 3 可看出, bit 位为 1 的频度在 984 ~ 1 064 区段时, 压缩效率最差, 但重码率较高, 然后随着 1 的减少或者增多压缩效率显著提升, 提升效果取决于移位变换的非均匀化程度。通过曲线回归, 调整的 R^2 和 p 值^[15], 最终选择二次曲线模型 (表 9 ~ 表 11), 其表达式为:

$$y = 311.606 + 3.059x - 0.001x^2, 0 \leq x \leq 2048,$$

其中: y 代表压缩率, x 则代表 1 bit 位的出现频度。

表9 二次曲线模型容忍度

R	R^2	调整 R^2	P
0.997	0.995	0.994	34.647

表10 二次曲线模型方差检验

指标	平方和	df	均方	F 值	p
回归	54 737 408.654	2	27 368 704.327	22 798.917	0.000
残差	302 510.577	252	1 200.439		

表11 二次曲线模型系数表

指标	未标准化系数		标准化系数		t	p
频数	3.059	0.015	3.878	206.812	0.000	
频数 2	-0.001	0.000	-4.004	-213.536	0.000	
常数	311.606	6.560		47.497	0.000	

5 讨论与小结

5.1 讨论

基于标识的变位对折压缩算法结合动态规划算法、标识对折方法和哈夫曼编码对输入流数据进行变化, 通过动态规划算法, 找到最优化分片段, 并移除该片段每个数据的前部分 0 bit 位, 实现数据的变位存储, 同时通过加少数数据片段中的 0 bit 位, 增加了 1 bit 位的概率, 使数据 1 bit 位较紧凑, 也增加了数据的重叠度, 然后通过变位对折进一步增加数据的重叠度。在重叠度较高的环境下, 利用哈夫曼编码, 综合提升了数据的无损压缩效率。

基于标识的变位对折压缩算法中的 Compress 算法的时间复杂度为 $O(n)$, 所需的空间为 $O(n)$, 即压缩时间复杂度和空间复杂度与压缩片段长度成正比。而哈夫曼编码的码率伴随着 Compress 的过程进行统计, 且编码的码元范围在 0 ~ 15, 其时间复杂度为 $O(1)$, 即压缩时间复杂度为常数。综上, 基于标识的变位对折压缩算法的时间复杂度为 $O(n)$, 空间复杂度为 $O(n)$, 压缩效率均与压缩片段的长短成正比。

基于标识的变位对折压缩算法仅通过变位与局部统计的方法就减少了算法的复杂性。算法针对的数据是普通的数据, 具有较好的通用性。此外, 算法充分应用到哈夫曼编码与图形编码的各自优点, 算法的重码率高, 数据位减少快。算法具有无损流

式压缩性,支持数据的顺序传输、使用和存储,所以可以适用于多线程、低内存、序列式的数据传输与使用需求的场合,比如手机、PDA 等,也可以用于网络多媒体的快速传递与播放,以及图书馆的海量电子图书的压缩存放。

5.2 小结

通过对图像压缩算法与哈夫曼编码的分析,提出一种基于标识的变位对折无损压缩算法,该算法结合了两种算法的各自最优环境,并构建适合的策略,实现两种算法的优化组合,解决了实际环境中遇到的数据大量顺序传输、使用和存储问题。

[参考文献]

- [1] 陈昌主. 数据压缩算法研究与设计 [D]. 长沙: 中南大学, 2010.
- [2] 曹雪虹. 信息论与编码 [M]. 北京: 清华大学出版社, 2009.
- [3] 田宝玉, 杨洁, 贺志强. 信息论基础 [M]. 北京: 人民邮电出版社, 2008.
- [4] 吴乐南. 数据压缩 [M]. 2 版. 北京: 电子工业出版社, 2005.
- [5] 姜磊, 黄广君. 自适应的无损数据压缩算法 [J]. 计算机工程, 2008 (1): 102 - 104.

- [6] 孔祥魁. 运动图像序列中关键关节点的跟踪优化仿真 [J]. 计算机仿真, 2016 (2): 423 - 426
- [7] 蔡明, 乔文孝, 鞠晓东, 等. 一种新的数据无损压缩编码方法 [J]. 电子与信息学报, 2014, 36 (4): 1008 - 1012.
- [8] 关雪梅. 基于 Matlab 的小波变换图像压缩算法研究 [J]. 赤峰学院学报 (自然科学版), 2018, 34 (9): 58 - 59.
- [9] 蔡楠, 李萍. 基于 KPCA 的图像压缩方法 [J]. 无线电工程, 2018, 48 (12): 1061 - 1064.
- [10] 王慧, 宋淑蕴. 基于 KPCA 提取特征和 RVM 的图像分类 [J]. 吉林大学学报 (理学版), 2017, 55 (2): 357 - 362.
- [11] 无家安. 数据压缩技术及应用 [M]. 北京: 科学出版社, 2009.
- [12] 王盼盼, 姚旭日, 刘雪峰, 等. 基于行扫描测量的运动目标压缩成像 [J]. 物理学报, 2017, 66 (1): 4201 - 4209.
- [13] 王晓东. 算法设计与分析 [M]. 北京: 清华大学出版社, 2010.
- [14] 方炫苏, 黄樟灿, 陈亚雄. 基于主成分分析和分层树集合划分的 Huffman 算法图像压缩研究 [J]. 浙江大学学报 (理学版), 2018, 45 (1): 55 - 58.
- [15] 谢龙汉. SPSS 统计分析与数据挖掘 [M]. 北京: 电子工业出版社, 2012.

(上接第 92 页)

- [4] 周伟, 蔡光明, 黄鹤慧, 等. 高效液相色谱法测定小叶黑柴胡中槲皮素与异鼠李素的含量 [J]. 中国药房, 2017, 18 (9): 693 - 694.
- [5] 宋海龙, 赵璐, 杨海燕. HPLC 法测定沙枣中槲皮素和异鼠李素的含量 [J]. 新疆医科大学学报, 2015, 38 (12): 1510 - 1516.
- [6] 郭毅新, 唐超, 高文分. HPLC 法测定滇柴胡中柴胡皂苷 d 的含量 [J]. 云南中医中药杂志, 2014, 35 (8): 67 - 68.
- [7] 吕飞, 翁德会, 吴士筠, 等. HPLC 法测定凹叶景天中槲皮素和异鼠李素含量 [J]. 化学与生物工程, 2009, 26 (8): 91 - 94.

- [8] 范刚, 普元柱, 杜娟, 等. HPLC 测定印楝叶中的槲皮素和异鼠李素 [J]. 华西药学杂志, 2010, 25 (3): 367 - 368.
- [9] 杨龙辉, 伍丕娥, 王天志. HPLC 测定沙棘膏中槲皮素和异鼠李素的含量 [J]. 华西药学杂志, 2002, 17 (2): 130 - 131.
- [10] 王利胜, 朱盛华, 许晓峰. HPLC 法测定水芹中槲皮素和异鼠李素 [J]. 中草药, 2004, 35 (9): 1061 - 1062.
- [11] 全国认证认可标准化技术委员会. 合格评定 化学分析方法确认和验证指南: GB/T 27417—2017 [S]. 北京: 中国标准出版社, 2017.