文章编号:1000-582X(2011)06-132-05

融合智能检测的 DNA 序列预处理新方法

刘 君[1,2],熊忠阳[1],王银辉[1]

(1. 重庆大学 计算机学院,重庆 400044; 2. 重庆广播电视大学 理工学院, 重庆 400052)

摘 要:提出一种融合智能检测的 DNA 序列预处理新方法。该方法不需要预先给出载体序列、剪接位点和克隆适配片段等信息,通过统计分析、随机搜索和构建图操作等方法自动发现并定位垃圾信息。以 Zebrafish DNA 序列为样本进行的预处理实验结果证明该方法能够显著提高 DNA 序列预处理的效率和准确性,在处理超长序列时更稳定、错误率更低。

关键词:DNA 序列;预处理;智能检测;载体;污染清除

中图分类号:TP301 **文献标志码**:A

Novel approach for DNA sequence preprocessing with intelligent detection

LIU Jun[1,2], *XIONG Zhong-yang*[1], *WANG Yin-hui*[1]

(1. College of Computer Science, Chongqing University, Chongqing 400044, P. R. China;
2. College of Science and Technology, Chongqing Radio & TV University, Chongqing 400052, P. R. China)

Abstract: A novel approach for DNA sequence preprocessing by merging intelligent detection is proposed. This approach can automatically find and locate contaminants using statistical analysis methods, random search and graph-theoretic operations, while no extra background information, such as vector sequence, splice site and clone adapter are needed during preprocessing. Experiments on Zebrafish DNA show that the approach can significantly improve the efficiency and accuracy of DNA sequence preprocessing and provide more stable performance than the conventional methods do, particularly in high-throughput DNA sequence preprocessing.

Key words: DNA sequence; preprocessing; intelligent detection; vector; contaminants removal

DNA 测序和处理技术日新月异[1],桑格测序(双脱氧核糖核酸链末端终止法)[2]是生物 DNA 和基因解码的基础并得到了广泛的应用。高通量的桑格序列分析首先要将 DNA 片段克隆到载体并将其转植到 Escherichia coli 中来放大原始的 DNA 片段。在这个过程中短小的载体适配片段会被附加到这些序列的末端来提高克隆效率。因此,通过这个过程的 DNA 序列会包含一些对基因解码无用的垃圾数据,如少量的载体序列、剪接位点和载体适配片段。而且在 DNA 测序和碱基读取的过程中常常会因为自动测序机的误差、DNA 测序曲线的波峰形状、波峰间距、信号强度以及背景噪声等原因产生低质量(不可信)的碱基区域。生物 DNA 本身在克隆的时候也会发生变异而造成基因序列的污染。所有这些对基因解码无关的因素都需要在 DNA 序列数据处理的开始阶段被清除,这个过程被称为 DNA 序列预处理。

近年来国外涌现了许多 DNA 序列预处理工具和软件,如 Lucy[3]、Crossmach[4]和 VecScreen[5]等。这些工具的原理都是把每次读到的碱基序列和克隆

收稿日期:2010-10-20

基金项目:中国博士后科学基金资助项目(20070420711);重庆市科委自然科学基金计划资助项目(2007BB2372)。

作者简介:刘君(1977-),女,重庆大学博士研究生,主要从事智能计算方向研究,(Tel)13883113187;(E-mail)junlucq@163.com。

载体进行对比,将具有高度相似性的碱基序列标记出来。它们都需要以下 3 种特征信息的支持:1)克隆载体序列;2)剪接位点片段;3)克隆适配片段。然而存储在公共数据库中的 DNA 序列通常会去掉这些特征信息,例如存储在 NCBI(national center for biotechnology information)中的 DNA 序列曲线会去掉载体剪接位点信息。因为缺乏这些支持特征信息,现有的 DNA 序列预处理工具在对垃圾碱基信息的过滤和清除过程中效率不高,而且当 DNA 序列长度增加的时候出错概率会显著升高。

针对现存传统 DNA 序列预处理工具的不足,提出一种新的融合智能检测的 DNA 序列预处理方法,它不需要预先给出载体序列、剪接位点和克隆适配片段等信息,通过统计分析、随机搜索和图操作等方法自动发现并定位垃圾信息。此新方法可以作为组件工具供 DNA 序列数据处理管道系统调用。

1　DNA 序列预处理

生物基因的提取和序列分析需要经过若干流水处理步骤,称为 DNA 序列的数据处理管道,如图 1 所示。首先构建文库并通过化学作用使 DNA 链在每一个碱基处终止并染色,通过变性凝胶的电泳和激光照射可以得到终止于不同碱基的 DNA 片段普带。这些普带通过碱基读取程序生成碱基序列文件,进行预处理后再经过片段组装 cDNA 聚类、基因检测等步骤来发现记录遗传信息的基因编码序列。

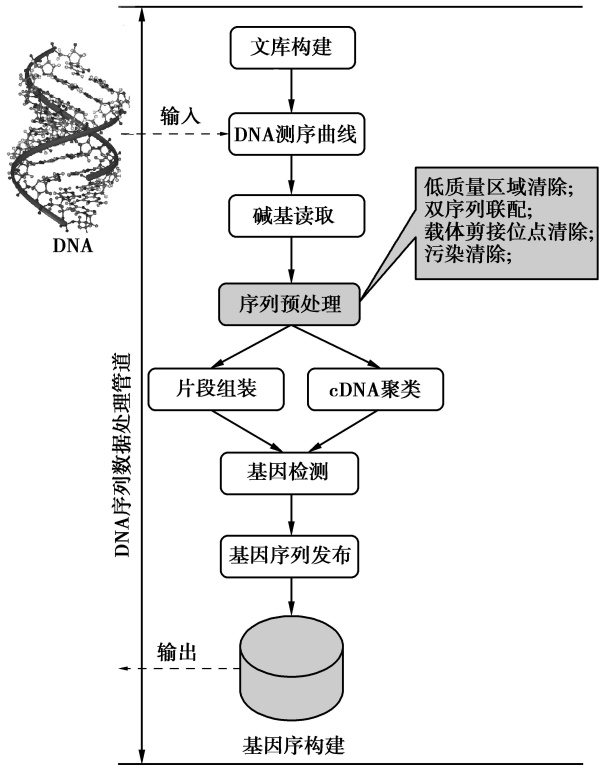


图 1　DNA 序列数据处理管道中的预处理过程

在整个 DNA 序列数据处理管道中,预处理是比较关键的步骤。虽然此步骤并不进行实质性的输出,但是它能将 DNA 序列处理各个阶段产生的对基因序列解码无用的垃圾信息清除,如 DNA 本身克隆过程中产生的变异污染、测序过程中增加的载体序列、剪接位点和适配片段等,以及在碱基读取过程中由于普带波峰形状、间距、信号强度和背景噪声造成的碱基错误。清除这些无用信息会极大地提高后期碱基片段组装、cDNA 聚类和基因检测的准确性和效率。

2　关键技术及实现

2.1　低质量区域清除

碱基读取程序(base-calling program)对 DNA 测序曲线处理后生成碱基序列文件 $B_f=\{b_1,b_2,\cdots,b_n\mid n\in Z^+\}$,其中碱基片段 Seg$[i,j]$定义为区间$\{i,i+1,\cdots,j\}$满足 $1\leqslant i\leqslant j\leqslant n$。碱基片段 Seg$[i,j]$的质量系数定义为:$Q_s(i,j)=\delta^*(q_i+q_{i+1}+\cdots+q_j)$,其中 $Q_f=\{q_1,q_2,\cdots,q_n\mid n\in Z^+\}$为与碱基序列对应的碱基质量跟踪文件,$\delta$ 为质量控制权重系数。DNA 序列预处理第一步"低质量区域清除"要解决的核心问题是:用户给定碱基测序质量控制阈值 Q,通过 1 个线性时间算法找到并标记所有碱基片段 Seg$[i,j]$,满足 $Q_s(i,j)\geqslant Q$。碱基测序质量控制指标 Q 定义为

$$Q=-10\times\log_{10}(\mathrm{PoE}),\tag{1}$$

其中 PoE (probability of error)定义为碱基读取过程中的错误概率,它和 DNA 测序曲线的波峰形状、波峰间距、信号强度以及背景噪声等因素相关[6]。

实现高质量碱基区域标定首先需要定义前缀质量函数:$f_j=Q_s(1,j)$,其中 $j=1,\cdots,n$,令 $f_0=0$。显然,$Q_s(i,j)=f_j-f_{i-1}$,接下来需要找到最长碱基片段满足 $f_j\geqslant f_{i-1}+Q$。

引理 1. 令 $0\leqslant i^*\leqslant j^*\leqslant n$,使得 $f_{j^*}\geqslant f_{i^*}+Q$ 和$(i^*+1,j^*)\geqslant Q$ 为最大,如果 $i^*>0$ 有

$$f_{i^*}<f_0,f_1,\cdots,f_{i^*-1}。\tag{2}$$

如果 $j^*<n$ 有

$$f_{j^*}>f_n,f_{n-1},\cdots,f_{j^*+1}。\tag{3}$$

显然,区域$[i^*+1,j^*]$为满足 $Q_s(i^*+1,j^*)\geqslant Q$ 的最长碱基片段。

算法 1 DNA 序列优质区域标记	
1：	输入：DNA 序列文件 B_f，碱基质量跟踪文件 Q_f；质量控制阈值 Q
2：	输出：最长 DNA 子序列 T 满足碱基片段质量系数高于 Q，否则输出空
3：	$f_0 \leftarrow 0$; FOR $i \leftarrow 1, \cdots n$ DO $f_i \leftarrow f_{i-1} + b_i$
4：	$k \leftarrow 1, l_1 \leftarrow 1$
5：	FOR $i \leftarrow 1, \cdots n$ DO
6：	IF $f_i < f_{lk}$ THEN $k \leftarrow k + 1$, $l_k \leftarrow i$
7：	$m \leftarrow 1$, $r_1 \leftarrow n$
8：	FOR $j \leftarrow n, \cdots 1$ DO
9：	IF $f_j > f_{rm}$ THEN $m \leftarrow m + 1$, $r_m \leftarrow j$
10：	max $\leftarrow 0$, $T \leftarrow$ null
11：	$i \leftarrow 1$, $j \leftarrow m$
12：	WHILE $i \leqslant k$ AND $j \geqslant 1$ DO
13：	WHILE $i \leqslant k$ AND $f_{li} + Q > f_{rj}$ DO $i \leftarrow i+1$ END WHILE
14：	IF $i \leqslant k$ THEN
15：	WHILE $j \geqslant 1$ AND $f_{li} + Q \leqslant f_{rj}$ DO
16：	IF $r_j - l_i >$ max THEN
17：	max$\leftarrow r_j - l_i$, $T \leftarrow [l_i + 1, r_j]$
18：	$j \leftarrow j - 1$ENDIF
19：	END WHILE
20：	END IF
21：	END WHILE
22：	RETURN T

证明：对于式(2)，假设存在 $i < i*$ 使得 $f_i \leqslant f_{i*}$，则 $j* - i > j* - i*$，然而 $f_{j*} \geqslant f_{i*} + Q \geqslant f_i + Q$，这和题设矛盾。式(3)也可以类似证明。

算法 1 给出了对 DNA 序列中最长优质碱基区域进行检测和标记的方法，该方法具有线性时间复杂度 $O(n)$。低质量区域清除需要首先调用算法 1 遍历并标记出所有满足质量控制阈值的片段，然后将未标记的区域清除，符合要求的所有碱基片段被载入数据处理管道中等待下一步处理。

2.2 双序列联配

碱基序列文件经过前述低质量区域清除后，允许用户增加 1 个可选的处理过程，即双碱基序列联配。通过对采用了不同碱基序列读取程序得到的碱基序列文件进行联配可以扩展预处理碱基序列的精度，如基础碱基序列文件由 ABI 377 自动测序机和碱基读取程序 phred[7] 生成，而辅助联配序列文件可以由 ABI 3700 自动测序机和碱基读取程序 TraceTuner[8] 生成。

双序列联配的主要步骤如下：首先将基础碱基序列转换为长度为 16bp 单位的片段簇，并对这些片段簇进行排序和去重，每个片段簇都含有 1 个指向其在原碱基序列文件中位置的索引号(指针)。同时辅助联配序列文件也同样被转换为和基础碱基序列相同的格式。接下来的联配过程要快速扫描辅助碱基序列中和基础碱基序列一致的片段簇，扫描过程将不一致的片段簇的索引号记录到一个被称为脱靶计数器的缓存中。扫描结束后脱靶计数器中的数值被当作 2 个碱基序列在它们最佳联配区域中的相对偏差。利用相对偏差可以定位碱基序列在高质量区域中的中心联配带。中心联配带可以采用经典的深度优先的搜索算法来实现。通过中心联配带的对比可以进一步去掉碱基序列中的低质量或在测序过程中有较大概率出错的碱基，提高了碱基序列处理的精度。

2.3 载体剪接位点和污染清除

在通过自动测序机(如 ABI 3700 等)测序输出 DNA 序列中通常含有克隆载体剪接位点片段，在后期的基因片段侦测过程中需要检测到这些克隆载体剪接位点片段并将其去掉。提出的预处理方法采用了将不确定性和确定性策略整合在一起的智能载体片段检测方法。该方法包含 2 个步骤：一是基于随机搜索的载体序列过滤，二是基于图操作的载体片段检测。

算法 2 载体片段序列过滤	
1：	输入：优质碱基区域序列 S
2：	输出：前 N 个最可能是载体片段的序列
3：	FOR m 次迭代 DO
4：	选择 k 个索引生成散列函数 $h(s)=\{s_{i1}, \cdots s_{ik}\}$
5：	FOR EACH l-mers DO
6：	FOR $j = 1$; $j \leqslant k$; j++ DO
7：	k-mer[j]= l-mer[s_{ij}]
8：	END FOR
9：	散列所有具有相同 k-mer 桶的 l-mers
10：	END FOR
11：	FOR EACH 桶 DO
12：	IF 映射频率大于桶门限值 φTHEN
13：	输出所有桶，l-mers，片段索引到 hashtable
14：	END IF
15：	END FOR
16：	END FOR
17：	FOR EACH 桶 DO
18：	统计(片段索引 1，片段索引 2)对的数目
19：	END FOR
20：	排序并返回前 N 个序列

设 $S=\{s_1, \cdots, s_n\}$ 为 n 个经过前述过程处理生成的优质碱基区域，$M(l, d)$ 为长度为 l 的 d 个可能存在的载体剪接位点片段模板。载体片段检测首先需要采用随机搜索的碱基序列过滤方法来将大量非载体剪接位点片段过滤掉，从而减轻第二步载体片

段检测的计算压力。序列过滤方法分为 2 步:局部敏感散列 LSH(locality sensitive hashing)[9] 和序列评分 SS(sequence scoring)。LSH 的思想是利用简单的散列函数将多维空间中的对象映射到具有高概率汇集近距离对象的桶(Bucket)中,采用 LSH 方法将少量载体片段从大量序列中过滤出来。当 n 个优质碱基区域给定后,定义散列函数 $h(x)$ 通过随机选择 k 个位置将长度为 l 的片段(l-mer)映射到长度为 k 的片段(k-mer)。如果 $k \leqslant l-d$ 并且 k 不是太小,那么在这种情况下载体片段比随机 l-mer 有较大概率被映射到相同的桶中。考虑有 3 个序列 s_1,s_2 和 s_3,其中 s_1 和 s_2 含有载体片段而 s_3 没有。假设这些序列中的 l-mer 已经被 $h(x)$ 映射,那么 $s_1 \& s_2$ 中的 l-mer 比 $s_1 \& s_3$ 和 $s_2 \& s_3$ 有较大的概率被共同映射到 1 个桶中。算法 2 给出了具体的序列过滤方法。

算法 3　载体片段检测	
1:	输入:包含载体剪接位点片段的候选序列 S
2:	输出:载体剪接位点片段
3:	团集合 Q←null,C←null
4:	FOR EACH s_{1i} AND s_{2j} DO
5:	IF $Dis(s_{1i}, s_{2j}) \leqslant 2d$ THEN
6:	插入$\{s_{1i}, s_{2j}\}$到 Q 如果其大小为 2
7:	END IF
8:	END FOR
9:	FOR EACH $m=3\cdots p$ DO // p 为团最多顶点数
10:	FOR EACH Q 中的$(m-1)$团 cli 和顶点 s_{mj} DO
11:	从 Q 中删除 cli
12:	IF $cli=cli\cup\{s_{mj}\}$为大小为 m 的团 THEN
13:	IF cli 是可转换团 THEN
14:	将 cli 转换为中心串并插入 C 中
15:	ELSE 将 cli 插入 Q 中
16:	END IF
17:	END IF
18:	END FOR
19:	END FOR
20:	FOR EACH Q 中的剩余 p 团 cli DO
21:	将 cli 转换为中心串并插入 C 中
22:	END FOR
23:	返回所有通过中心串验证的载体片段

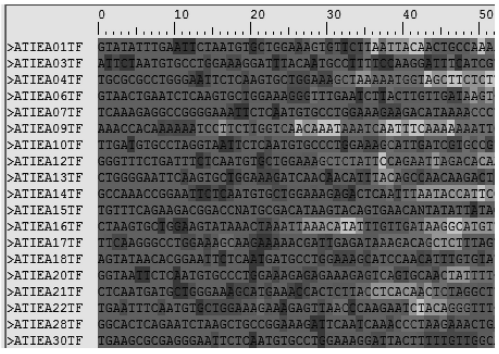
设 $S=\{s_1,\cdots,s_N\}$ 为通过算法 2 得到的 N 个最有可能是载体剪接位点片段的序列,载体片段检测的任务是要检测并发现能转换为中心串(center strings)的片段构成的子集,这些片段即是要清除的目标载体片段。给定 2 个 l-mers x 和 y,Dis(x, y) 定义为 x 和 y 的汉明距离,对于 l-mers 的集合 $S=\{s_1,s_2,\cdots s_n\}$,其中心串 C 定义为任意满足 Dis(C, s_i) $\leqslant \gamma$ 的 l-mers。

载体片段的检测通过构建图 $G(S, l, d)$ 并检测团(目标载体片段)的方式实现。对序列 s_i 中的任意位置 p,建立顶点 $s_{i,p}$ 表示 s_i 中从 p 点开始长度为 l 的串,在顶点 $s_{i,p}$ 和 $s_{j,q}$ 之间建立边,满足 $i \neq j$ 并且两顶点之间的汉明距离小于 $2d$。算法 3 给出了团检测的实现,其中可转换团和中心串的验证机制请参考文献[10-11]。

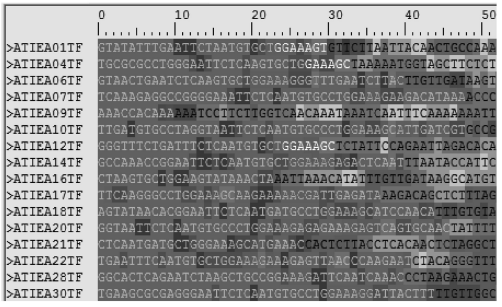
DNA 序列中的污染由很多因素造成,污染导致的变异片段在整个序列中呈现不同的分布。利用提出的智能检测算法可以很方便的检测出基因污染片段并标记去除,用户只需要提供相应的检测模板即可。

3　应用实例

为了验证提出的智能检测算法及中间件的实现,验证数据集样本为以 Zebrafish DNA [12] 的 shotgun 测序(采用 ABI 3700 自动测序机)结果导出的不同的 300 条 DNA 序列,长度约为 1 500 000 bp,并且在其中间随机增加了不同长度(7 到 50 bp)的各种载体剪接位点片段、污染片段等杂质(见图 2)。



(a)序列碱基文件载入



(b)预处理结果

图 2　附带质量跟踪文件的 Zebrafish DNA

图 2(a)给出了 Zebrafish 的 DNA 序列的导入碱基文件,碱基质量跟踪文件采用不同的颜色表示该碱基在读取过程中的质量系数。色调偏暖表示该碱基在测序和曲线读取的过程中出错的概率小,色调偏冷表示该碱基在测序和曲线读取的过程中出错

的概率大。图2(b)是经过全部预处理过程后的结果,连续标识为黑色的碱基序列表示其为高质量的碱基片段,是可能的基因候选区域,连续标识为白色的碱基序列表示其为载体剪接位点片段,连续标识为灰色的碱基序列表示其为在预处理第一步(低质量区域清除)过程中去除的低质量碱基区域,污染碱基序列直接被清除,如ATIEA03TF,ATIEA13TF,ATIEA15TF等。

图3给出了提出的融合智能检测的DNA序列预处理方法与传统预处理方法Lucy[3]在预处理过程中的出错次数比较。在长度较小(1 000 bp到50 000 bp)的DNA序列预处理过程中,提出的智能检测方法并无太明显的优势,但是在长度较大(50 000 bp以上)的DNA序列预处理过程中,研究方法表现出了非常高的准确性,出错概率始终能够保持在0.03%的范围内。而传统的预处理方法的出错概率随着DNA序列的长度增加而急剧上升,并且测序结果较不稳定(蓝色曲线在DNA长度超过60 000 bp后呈现了较大的起伏)。

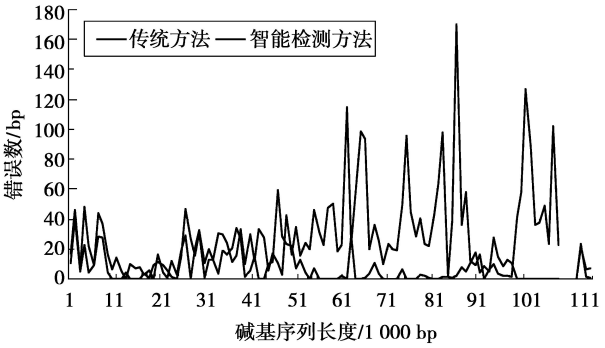


图3 融合智能检测的DNA序列预处理方法与传统预处理方法中的出错次数比较

4 结 论

DNA序列预处理能将对基因序列解码无用的垃圾信息清除。清除这些无用信息会极大地提高后期碱基片段组装、cDNA聚类和基因检测的准确性和效率。而传统的预处理工具清除这些无用信息的时候需要额外的特征信息支持。然而存储在公共数据库中的DNA序列通常会去掉这些特征信息,这给预处理带来了不便,并且影响后续处理的精确性。针对现存传统DNA序列预处理工具的不足,提出了一种新的融合智能检测的DNA序列预处理方法,它不需要预先给出载体序列、剪接位点和克隆适配片段等信息,通过统计分析、随机搜索和图操作等方法自动发现并定位垃圾信息。以Zebrafish DNA序列为样本进行的预处理试验结果证明方法能够显著提高DNA序列预处理的效率和准确性,在超长序列处理的时候更稳定、错误率更低。

参考文献:

[1] MARGULIES M, EGHOLM M, ALTMAN W E, et al. Genome sequencing in microfabricated high-density picolitre reactors [J]. Nature, 2005, 437: 376-380.

[2] SANGER F, NICKLEN S, COULSON A R, et al. DNA sequencing with chain-terminating inhibitors[J]. Proceedings of the National Academy of Sciences, 1977, 74(12): 5463-5467.

[3] CHOU H H, HOLMES M H. DNA sequence quality trimming and vector removal[J]. Bioinformatics, 2001, 17(12): 1093-1104.

[4] LUCY. Software tools are available for vector removal: Crossmatch [EB/OL]. [2009]. http://www.phrap.org/phredphrapconsed.html.

[5] LUCY. Software tools are available for vector removal: VecScreen [EB/OL]. [2009]. http://www.ncbi.nlm.nih.gov/VecScreen.

[6] RICHARDS S, LIU Y, BETTENCOURT B R, et al. Comparative genome sequencing of drosophila pseudoobscura: chromosomal, gene, and cis-element evolution [J]. Genome Research, 2007, 15(1): 1-18.

[7] EWING B, HILLIER L WENDL M C. Base-calling of automated sequencer traces using phred. I. Accuracy assessment [J]. Genome Research, 1998, 8(3): 175-185.

[8] PARACAL T. Capturing the most information from the latest DNA sequencing systems. [EB/OL]. [2009]. http://www.paracel.com/htm/tracetuner.html.

[9] INDYK P, MOTWANI R. Approximate nearest neighbors: towards removing the curse of dimensionality [C]// The Thirtieth Annual ACM Symposium on Theory of Computing. May 23-26, 1998, Dallas, Texas. USA: ACM Press, 1998.

[10] DONG X, SUNG S Y, SUNG W K, et al. Constrained based method for finding motif in DNA sequences[C]// 4th IEEE Symposium on Bioinformatics and Bioengineering (BIBE), 2004. May19-21, 2004, Taichung, Taiwan. [S.l.]: IEEE Computer Society, 2004.

[11] SUNG W K, LEE W H. Fast and accurate probe selection algorithm for large genomes [C]// IEEE Computer Society Bioinformatics Conference (CSB). August 11-14, 2003, Stanford, CA. [S.l.]: IEEE Computer Society, 2003.

[12] REHBEIN H, BOGERD J. Identification of genetically modified zebrafish (Danio rerio) by protein- and DNA-analysis[J]. Journal für Verbraucherschutz und Lebensmittelsicherheit, 2007, 2(2): 122-125.

(编辑 侯 湘)