

# 嵌入式构件的非功能属性<sup>①</sup>

匡 扬, 杨贯中, 朱贤斌

(湖南大学 软件学院, 长沙 410082)

**摘 要:** 构件已成为嵌入式领域的重点研究对象之一, 一个高质量的嵌入式构件不仅包含构件的功能属性, 同时还包含构件的非功能属性(Non-Functional Attribute, 简称 NFA), 但目前缺乏对嵌入式构件非功能属性的研究. 主要研究嵌入式构件的非功能属性, 并对非功能属性用架构设计语言(Architecture Design Language, 简称 ADL) 进行描述; 阐述了嵌入式构件开发过程中与非功能属性的应对关系, 以利于保证嵌入式构件的质量. 最后通过实例来予以验证.

**关键词:** 嵌入式构件; 构件质量; 非功能属性; ADL; 构件开发过程

## Non-Functional Attributes of Embedded Components

KUANG Yang, YANG Guan-Zhong, ZHU Xian-Bin

(College of Software, Hunan University, Changsha 410082, China)

**Abstract:** Component has become a key research object in the field of embedded, a high quality embedded component includes not only its functional attributes, but also its non-functional attributes(abbreviation ADL). However, it lacks the research on the non-functional attributes of embedded components. This paper mainly studies the non-functional attributes of embedded component, and describes them by using Architecture Design Language (abbreviation ADL) and elaborates the correspondence relations between the embedded components development process and the non-functional attributes, so as to guarantee the quality of the embedded components. At last, this article gives an example to verify it.

**Key words:** embedded components; component quality; NFA; ADL; component development process

## 1 引言

在嵌入式领域中基于构件的软件开发的研究越来越广泛, 目前, 针对嵌入式平台的多样性、环境依赖性、实时性、内存消耗等嵌入特性<sup>[1]</sup>, 对于性能、可靠性、可重用性等非功能属性的研究在很早的时候就开始了. 早期主要研究嵌入式软件质量属性的评价标准和质量模型, 在 20 世纪 90 年代慢慢开始提出将非功能属性进行系统地研究, 其研究内容包括非功能属性的描述、非功能属性设计、非功能属性需求等等<sup>[2]</sup>.

在嵌入式构件领域, 主要对构件进行规范化描述和功能性的形式化证明<sup>[1-3]</sup>, 但是并没有涉及到非功能性需求的描述. 目前高可信软件概念已经提出, 用户对嵌入式构件非功能属性的关注也越来越高, 那么

非功能属性必然要涉及到构件的整个生命周期.

本文重点是对嵌入式构件开发过程中非功能属性的研究, 提出嵌入式构件的非功能属性模型, 并对非功能属性 ADL 语言进行描述, 提出了非功能属性指标定义及度量公式, 阐明在嵌入式构件开发与非功能属性之间的一种对应关系, 来保障构件的非功能属性.

## 2 嵌入式构件的非功能属性

国内外研究机构专门推出了一系列嵌入式构件模型, 但这些模型缺乏对非功能属性的研究. 在“面向领域的可信嵌入式软件开发(DTESD)”项目中我们提出了 DTESD 构件模型.

<sup>①</sup> 基金项目:湖南省产学研结合重大科技成果转化项目(2010XK6024)

收稿时间:2012-03-14;收到修改稿时间:2012-04-23

### 2.1 嵌入式构件模型中的属性包

DTESD 构件模型包括接口、实现体、内部结构、属性包以及约束。其中属性包是用于全方位描述构件性质和特征，是一组属性的集合。属性包又包括物理属性包、元数据属性包、可信属性包。

物理属性主要针对硬件构件的特性描述；构件元数据属性包主要是构件基本信息的描述；可信属性包包括功能属性和非功能属性，功能属性描述了构件必须提供的服务或者在特定的环境下所做出的反应，非功能属性描述构件的一些约束<sup>[3]</sup>。本文重点研究可信属性包中的 NFA 属性。图 1 描述了构件模型中关于属性包的 UML 图。

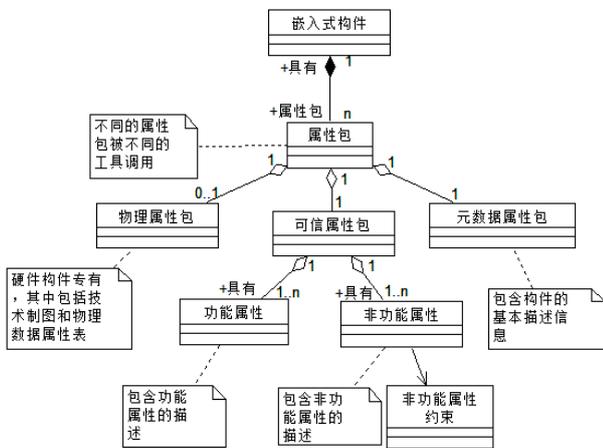


图 1 构件模型中属性包的 UML 图

### 2.2 非功能属性层次模型

关于非功能属性的定义比较经典的描述是：非功能属性是对构件功能属性、构件整体性质或构件开发过程的某种约束，也常常被称为质量属性<sup>[4]</sup>。

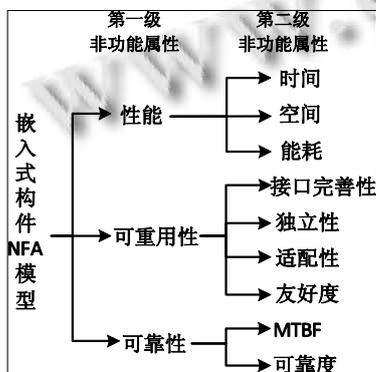


图 2 嵌入式构件非功能属性模型

一般嵌入式软件的非功能属性包括性能、可靠性、可维

护性、可重用性、安全性、实时性和配置性等，对于嵌入式软件构件而言，其特点是：代码规模小，与硬件结合紧密，环境依赖性、时间和空间有限性<sup>[4,5]</sup>。不同的嵌入式领域对 NFA 要求的重点也是不相同的，本文就单个构件而言，将嵌入式构件非功能属性模型分为两层，第一层为第一级非功能属性，由性能、可靠性和可重用性构成；第二层为第二级非功能属性，是把第一级非功能属性分解为可以度量的二级属性。如图 2 所示。

#### 2.2.1 非功能属性一级指标定义及度量

嵌入式构件的第一级非功能属性由性能、可重用性、可靠性组成，用 NFA1—NFA3 表示。

##### 1) 性能(NFA1)

性能是一种非功能属性，是指构件在完成其功能时展示出来的及时性。性能的度量公式如(1)所示：

$$NFA_1 = NFA_{11} \times W_{11} + NFA_{12} \times W_{12} + NFA_{13} \times W_{13} \quad (1)$$

其中，NFA<sub>1</sub> 表示性能的度量值，NFA<sub>11</sub>、NFA<sub>12</sub>、NFA<sub>13</sub> 分别表示二级指标时间、空间、能耗的度量值，

$$\sum_{i=1}^3 NFA_{1i} = 1, NFA_{1i} \in (0,1). W_{11}, W_{12}, W_{13} \text{ 分别表示与指标相对应的权重, } \sum_{i=1}^3 W_{1i} = 1, W_{1i} \in (0,1).$$

##### 2) 可重用性(NFA<sub>2</sub>)

可重用性是指一个构件可以多次重复使用的特性。可重用性的度量公式如(2)所示：

$$NFA_2 = NFA_{21} \times W_{21} + NFA_{22} \times W_{22} + NFA_{23} \times W_{23} + NFA_{24} \times W_{24} \quad (2)$$

其中，NFA<sub>2</sub> 表示可重用性的度量值，NFA<sub>21</sub>、NFA<sub>22</sub>、NFA<sub>23</sub>、NFA<sub>24</sub> 分别表示二级指标接口完善性、独立性、适配性和友好度的度量值， $\sum_{i=1}^4 NFA_{2i} = 1, NFA_{2i} \in (0,1). W_{21}, W_{22}, W_{23}, W_{24}$  分别表示与指标相对应的权重， $\sum_{i=1}^4 W_{2i} = 1, W_{2i} \in (0,1).$

##### 3) 可靠性(NFA<sub>3</sub>)

可靠性是在规定的一段时间和条件下，嵌入式软件构件维持其性能水平的能力<sup>[5]</sup>。度量指标主要有可靠度和平均故障间隔时间 MTBF(Mean Time Between Failure)。可靠性的度量公式如(3)所示：

$$NFA_3 = NFA_{31} \times W_{31} + NFA_{32} \times W_{32} \quad (3)$$

其中，NFA<sub>3</sub> 表示可靠性的度量值，NFA<sub>31</sub>、NFA<sub>32</sub> 分别表示二级指标可靠度和 MTBF 的度量值，

$$\sum_{i=1}^2 NFA_{3i} = 1, NFA_{3i} \in (0,1). W_{31}, W_{32} \text{ 分别表示与}$$

二级指标相对应的权重,  $\sum_{i=1}^2 W_{3i} = 1, W_{3i} \in (0,1)$ .

### 2.2.2 非功能属性二级指标定义及度量

非功能属性二级指标包括时间、空间、能耗、接口完善性、独立性、适配性、友好度、可靠度和 MTBF.

#### 1) 时间(NFA<sub>11</sub>)

时间是指在规约条件下, 构件执行周期或者算法执行的时间. 时间的度量公式如(4)所示:

$$NFA_{11} = 1 - |(X-Y)/Y| \quad (4)$$

其中, NFA<sub>11</sub> 表示时间的度量值, 且 NFA<sub>11</sub> ∈ (0,1), |(X-Y)/Y| ≥ 1 时均取值为 1. X 表示构件执行的实际时间, Y 表示规定构件理想的执行时间. 构件执行时间越接近理想值, NFA<sub>11</sub> 越好.

#### 2) 空间(NFA<sub>12</sub>)

空间是指构件执行过程中开辟的内存空间. 空间的度量公式如(5)所示:

$$NFA_{12} = 1 - |(X-Y)/Y| \quad (5)$$

其中, NFA<sub>12</sub> 表示空间的度量值, NFA<sub>12</sub> ∈ (0,1), |(X-Y)/Y| ≥ 1 时均取值为 1. X 表示构件在执行过程中实际开辟的内存空间, Y 表示总内存空间. 实际开辟的内存空间越少 NFA<sub>12</sub> 越好.

#### 3) 能耗(NFA<sub>13</sub>)

能耗是指构件在执行时使用适当数量和类别的资源的能力. 能耗的度量公式如(6)所示:

$$NFA_{13} = X/Y \quad (6)$$

其中, NFA<sub>13</sub> 表示能耗的度量值, 且 NFA<sub>13</sub> ∈ (0,1), X 表示构件在执行过程中实际所用的资源, Y 表示总资源. 所需要使用的资源数目越少, NFA<sub>13</sub> 越好.

#### 4) 接口完善性(NFA<sub>21</sub>)

接口完善性是指嵌入式构件所提供的功能符合规格说明的能力. 度量公式如(7)所示:

$$NFA_{21} = (X/Y + M/N)/2 \quad (7)$$

其中, NFA<sub>21</sub> 表示接口完善性的度量值, 且 NFA<sub>21</sub> ∈ (0,1), X 表示被用户正确理解的功能数, Y 表示构件的功能接口总数; M 表示构件在测试中检测到有功能缺陷的功能数, N 表示功能接口总数. NFA<sub>21</sub> 越大, 功能接口完善性越好.

#### 5) 独立性(NFA<sub>22</sub>)

嵌入式构件独立性是指构件在功能上具备完善的不依赖外部特殊环境的能力<sup>[6]</sup>. 独立性度量公式如(8)所示:

$$NFA_{22} = 1 - X/Y \quad (8)$$

其中, NFA<sub>22</sub> ∈ (0,1), NFA<sub>22</sub> 表示独立性的度量值, X 表示构件执行中依赖目标环境的功能数, Y 表示被检测的构件功能总数. NFA<sub>22</sub> 越大, 独立性越好.

#### 6) 适配性(NFA<sub>23</sub>)

适配性主要指构件与所属领域环境和构件间组装的难易程度, NFA<sub>23</sub> 表示适配性的度量值. 该度量值通过构件使用者来给定, NFA<sub>23</sub> ∈ (0,1). NFA<sub>23</sub> 值越大, 表示适配性越好.

#### 7) 友好度(NFA<sub>24</sub>)

友好度是指用户使用构件的友好程度, 用户在使用该构件时所需要获取的额外相关领域知识越, 友好度就越好<sup>[6]</sup>. NFA<sub>24</sub> 表示友好度的度量值. 该度量值通过构件使用者来给定, NFA<sub>24</sub> ∈ (0,1). NFA<sub>24</sub> 值越大, 表示友好度越好.

#### 8) 可靠度(NFA<sub>31</sub>)

可靠度是指构件在规定的条件下、规定的时间内完成预定的功能的概率<sup>[6]</sup>. 可靠度的度量公式如(9)所示:

$$NFA_{31}(t) = P(T > t) \quad (9)$$

其中, NFA<sub>31</sub> 表示可靠度的度量值, 且 NFA<sub>31</sub>(t) ∈ (0,1), t 为规约中规定的时间, T 为构件的寿命, 可靠度就是 T > t 的概率.

#### 9) MTBF(NFA<sub>32</sub>)

平均故障间隔时间(MTBF)是指两次相邻故障时间间隔的平均值. MTBF 是度量可靠性的重要指标之一. 度量公式如(10)所示:

$$NFA_{32} = (T_1 - T_2)/(n+1) \quad (10)$$

其中: NFA<sub>32</sub> 表示 MTBF 的度量值, 且 NFA<sub>32</sub> ∈ (0,1), T<sub>1</sub>、T<sub>2</sub> 表示构件从 T<sub>1</sub> 时刻到 T<sub>2</sub> 时刻, n 表示失效次数.

### 2.3 嵌入式构件非功能属性描述

对于非功能属性的描述在很长时间内都是用自然语言描述的, 这种描述方法具有一定的局限性, 后来有学者研究如何用一种规范的语言来描述, 例如 Process Language<sup>[7]</sup>; 也有用 UML 类图的方法描述<sup>[8]</sup>. 本文使用 ADL 语言方法来描述非功能属性, 即在建立构件模型时还要提供对模型中非功能属性的描述. 非功能属性的描述包括两个部分, 第一部分是 NFA 声明, 第二部分是 NFA 之间的约束关系.

定义 1. NFA 声明(NFA Declaration)主要是描述嵌

入式构件非功能属性的值和标度类型, NFA 声明是一个三元组, 具体描述如下所示:

NFA Declaration=[PROPERTY-ID, PROPERTY-VALUE,TYPE-NAME]

PROPERTY-ID 表示 NFA 的标识. PROPERTY-ID 是为了区分构件非功能属性以及子属性.

PROPERTY-VALUE 是该 NFA 声明中的 NFA 的值,它可以是一个逻辑表达式,也可以是一个确切值.

TYPE-NAME 是 PROPERTY-VALUE 中所涉及数值的类型.

定义 2. NFA 约束(NFA Constraint)是用来描述 NFA 值的允许范围以及 NFA 之间存在的关系. 其描述如下所示:

NFA Constraint = [PROPERTY-NAME,VALUE-RANGE, CONSTRAINT-RELATION]

PROPERTY-NAME 与 NFA 声明中一样为非功能属性名, 是 NFA 的标识.

VALUE-RANGE 为 NFA 值的允许范围, 根据构件所要达到的预期值, 给出该构件 NFA 的一个约束范围, PROPERTY-VALUE 的值必须在 VALUE-RANGE 范围内, 用来保障嵌入式构件开发和构件组装过程中的非功能属性满足预期要求. 其形式有四种, 分别为: VALUE-RANGE=[V1,V2]、VALUE-RANGE=(V1,V2)、VALUE-RANGE=(V1,V2、 VALUE-RANGE=[V1,V2]. [V1,V2]表示  $V1 \leq \text{PROPERTY-VALUE} \leq V2$ , (V1,V2)表示  $V1 < \text{PROPERTY-VALUE} < V2$ , (V1,V2]表示  $V1 < \text{PROPERTY-VALUE} \leq V2$ , [V1,V2)表示  $V1 \leq \text{PROPERTY-VALUE} < V2$ .

CONSTRAINT-RELATION 表示 NFA 之间存在的约束关系, 这种关系有表达两种形式, 即关系表达式和 NULL 值. 如果一个 NFA 的约束关系存在, 并通过一个关系表达式来体现, 表示这个 NFA 的值需要另外一个 NFA 来间接度量; 如果一个 NFA 的约束关系用 NULL 表示, 表示这个 NFA 的值是可以直接度量的, 与其它 NFA 不存在一定的约束关系.

### 3 嵌入式构件开发过程中的非功能属性

在基于构件的嵌入式软件开发中, 为保障构件质量在构件开发过程中建立一个与 NFA 相对应的关系.

嵌入式构件开发包括构件分析、构件设计、构件实现、构件测试和构件可信评估五个阶段, 嵌入式构

件非功能需求的描述、非功能规约设计、非功能实现、非功能测试验证以及非功能属性评估应该贯穿在构件开发的流程中. 如图 3 所示.

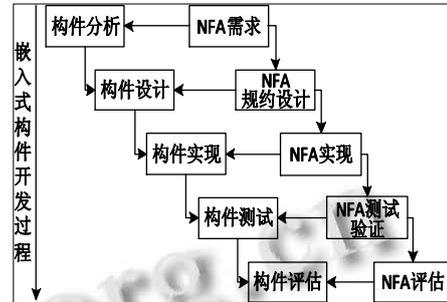


图 3 嵌入式构件开发过程

#### 1) 构件分析阶段

嵌入式构件开发的第一个步骤是对构件进行需求分析, 通常构件需求分析包括功能需求分析和非功能需求分析. 非功能需求一般比较抽象, 所以在该阶段要准确分析构件的非功能属性及其所包含的子属性.

#### 2) 构件设计阶段

构件设计阶段主要是针对构件需求和非功能需求, 将构件的每个功能设计与非功能属性融合在一起考虑, 给出构件模型和 NFA 规约, 构件模型和 NFA 规约的描述采用 ADL 语言来描述.

#### 3) 构件实现阶段

构件实现阶段主要是依据构件模型和 NFA 规约, 将构件的每个功能与非功能属性进行有效的集成, 最终实现构件.

#### 4) 构件测试阶段

非功能属性值的获取途径有多个, 例如测试、专家或者经验丰富的开发人员进行评定、用户调查等, NFA 的度量值很大程度需要通过测试来验证. 在非功能需求和非功能属性规约确定后, 就应该合理的安排测试计划并进行测试, 最终获得二级指标度量值.

#### 5) 构件评估阶段

在测试阶段获取度量值后, 采用层次分析方法对度量值进行综合评判, 最终得到构件 NFA 度量值.

### 4 应用实例

本节以“面向领域的可信嵌入式软件开发环境”项目为背景, 以一个 SPI 驱动构件为研究对象, 对本文所提出的嵌入式构件非功能属性的度量进行说明.

首先对 SPI 驱动构件进行需求分析, 得到嵌入式构件需求模型, 包括功能需求和非功能需求, 非功能需求就是针对构件的特点确定该构件的非功能属性以及各个属性之间的联系。

然后根据非功能需求建立非功能属性层次模型, 见图 2. 确定非功能属性层次模型后, 根据需求模型建立 SPI 驱动构件模型, 见图 4 所示。

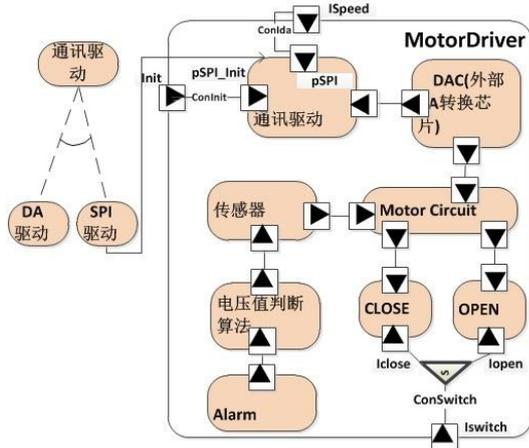


图 4 SPI 驱动构件模型

非功能属性权重的确定由专家组来确定, 针对呼吸机的 SPI 驱动构件的需求和特点, 非功能属性权值为:  $W_1=0.375$ ,  $W_2=0.375$ ,  $W_3=0.250$ ;  $W_{11}=0.378$ ,  $W_{12}=0.288$ ,  $W_{13}=0.334$ ,  $W_{21}=0.359$ ,  $W_{22}=0.291$ ,  $W_{23}=0.216$ ,  $W_{24}=0.134$ ,  $W_{31}=0.528$ ,  $W_{32}=0.472$ 。

通过对 SPI 驱动构件进行测试、专家评定和用户调查, 获得每个二级度量指标的度量值, 并通过公式(1)~公式(10)算得各级非功能属性值, 统计表见表 2。

表 2 非功能属性值

二级指标	度量值	一级指标	度量值
NFA <sub>11</sub>	1.000	NFA <sub>1</sub>	0.921
NFA <sub>12</sub>	0.891		
NFA <sub>13</sub>	0.857		
NFA <sub>21</sub>	1.000	NFA <sub>2</sub>	0.956
NFA <sub>22</sub>	1.000		
NFA <sub>23</sub>	0.886		
NFA <sub>24</sub>	0.859		
NFA <sub>31</sub>	0.950	NFA <sub>3</sub>	0.924
NFA <sub>32</sub>	0.894		

由于篇幅有限, 仅对 SPI 驱动构件的一级 NFA 进

行描述。

PropertiespSPI\_Init

{

NFA {

SPI-NFA Constraint=[Performance, [0.850, 1.000],

$NFA_1=NFA_{11} \times W_{11}+NFA_{12} \times W_{12}+NFA_{13} \times W_{13}$ ;

SPI-NFA Constraint=[ Reusability,[0.855, 1.000],

$NFA_2=NFA_{21} \times W_{21}+NFA_{22} \times W_{22}+NFA_{23} \times W_{23}+NFA_{24} \times W_{24}$ ;

SPI-NFA Constraint=[ Reliability,[0.800, 1.000],

$NFA_3=NFA_{31} \times W_{31}+NFA_{32} \times W_{32}$ ;

SPI-NFA Declaration=[Performance,0.921, TYPE-Performance];

SPI-NFA Declaration=[ Reusability,0.956,

TYPE-Reusability];

SPI-NFA Declaration=[ Reliability,0.924,

TYPE-Reliability];

}

}

性能 NFA 值的允许范围为[0.850,1.000], 0.921 在 NFA 允许范围之内. 可重用性 NFA 值的允许范围为 [0.855,1.000], 度量值为 0.956, NFA 值在允许范围内. 可靠性的 NFA 值的允许范围为[0.800,1.000], 度量值为 0.924, 在 NFA 允许范围内. 因此, SPI 驱动构件的 NFA 是满足非功能需求的。

## 5 结语

本文主要对嵌入式构件的非功能属性进行了研究, 将非功能属性考虑到构件模型中, 提出一种基于构件的非功能属性模型, 在构件开发的各阶段对非功能属性进行度量, 使得构件的非功能属性得到保障, 从而提高嵌入式构件的质量. 下一步将对非功能属性度量方法进行重点研究。

## 参考文献

1 Bunse C, Gross HG. Unifying Hardware and Software

(下转第 164 页)

中欧式距离满足,  $\bar{d}_1 < \bar{d}_4 < \bar{d}_2 < \bar{d}_3$ , 这表明了第一类实验图片最为相似, 其次是第四类实验图片, 最不相似的为第三类图片, 这一规律与实验前依据人类的视觉经验判断期望的实验结果完全符合。

### 3.2 基于灰度共生矩阵和 Tamura 纹理特征提取方法的实验结果分析

① 灰度共生矩阵中的能量, 熵, 能较好的反映出图片间的相似性和差异性, 而惯性矩和局部平稳性则表现的不稳定。灰度共生矩阵充分利用了空间信息, 对图像的对比度, 纹理的走向等描述准确度较好。

② Tamura 纹理特征提取出的特征向量使用欧式距离来衡量相似性结果满足  $\bar{d}_1 < \bar{d}_2 < \bar{d}_4 < \bar{d}_3$ , 通过统计大量的实验数据, 发现对比度的数值大小基本上决定了欧氏距离值的大小, 粗糙度和方向度对相似性度量结果影响很小, 使得对上述两个因子的计算变得毫无意义。实际在使用此种纹理特征提取方法时, 应该采取加权求和的方式来对纹理相似性特征进行计算。

### 3.3 三种纹理特征提取方法试验数据对比分析

表 1 展示了本文采取的三种纹理特征提取实验的实验结果, 说明了依据藤茎类图像的频域, 基于图像傅里叶频谱图径角变换的纹理特征提取方法, 相比灰度共生矩阵方法和 Tamura 方法这两种常见的纹理特征提取方法, 更加符合视觉感知经验判断期望的结果, 能更好的用来定义和描述藤茎类植物图像的纹理特征和准确的进行相似性度量。

## 4 结论

基于图像傅里叶频谱图径角变换的纹理特征提取方法, 相比灰度共生矩阵方法和 Tamura 纹理特征提取方

法能更好的用来定义和描述图像的纹理特征和准确的进行相似性度量研究, 从而可以推广应用在图像检索研究中。

表 1 三种纹理提取方法实验结果对比

实验类别	GLCM度量	Tamura度量	纹理谱度量
第一类实验	$\bar{d}_1 = 0.1751$	$\bar{d}_1 = 2.27803$	$\bar{d}_1 = 10.6131$
第二类实验	$\bar{d}_2 = 0.5278$	$\bar{d}_2 = 2.2824$	$\bar{d}_2 = 22.1907$
第三类实验	$\bar{d}_3 = 1.29092$	$\bar{d}_3 = 17.2431$	$\bar{d}_3 = 39.4755$
第四类实验	$\bar{d}_4 = 0.08397$	$\bar{d}_4 = 6.07442$	$\bar{d}_4 = 19.3124$
实验结果	$\bar{d}_4 < \bar{d}_1 < \bar{d}_2 < \bar{d}_3$	$\bar{d}_1 < \bar{d}_2 < \bar{d}_4 < \bar{d}_3$	$\bar{d}_1 < \bar{d}_4 < \bar{d}_2 < \bar{d}_3$

### 参考文献

- 徐贵力, 毛罕平. 利用傅里叶变换提取图像纹理特征新方法. 光电工程, 2004, 31(11): 55-58.
- 毛秉毅. 旋转不变傅里叶纹理特征提取. 计算机工程与应用, 2007, 43(10): 99-100.
- Amiri MD. Affine Invariant Iris Identification Using Angular and Radial Partitioning IEEE Trans. Signal processing and Information Technology (ISSPIT), 2009 IEEE Int. Symposium on, 2009, 14(17): 574-578.
- 杨波, 徐光佑. 纹理相似性度量研究及基于纹理特征的图像检索. 自动化学报, 2004, 30(6): 991-998.
- Abdesselam A. Texture Image Retrieval Using Fourier Transform, International Conference On Communication, Computer Power (ICCCP'09), 2009, 15(18): 343-348.
- 黄锡滋. 软件可靠性、安全性与质量保证. 北京: 电子工业出版社, 2002.
- 张建明, 孙伟. 基于重庆软件资源库的构件可复用性度量研究. 中国科技论文在线, 2008, 3(2).
- Rosa NS, Cunha PRF, Justo GRR. Process NFA: A Language for Describing Non-Functional Properties. Proc. of the 35th Hawaii International Conference on System Sciences. 2002.
- 张岩, 梅宏. UML 类图中面向非功能属性的描述和检验. 软件学报, 2009, 20(6): 1457-1469.

(上接第 93 页)

- Components for Embedded System Development. Architecting Systems, LNCS3938, 2006, 120-136.
- 杨放春, 龙湘明. 软件非功能属性研究. 北京邮电大学学报, 2004, 27(3).
  - 王志刚, 王民北, 骆雷飞. 一个嵌入式软件构件的 NFA 量化度量模型. 计算机工程, 2006, 32(13).
  - Lauesen S. COTS tenders and integration requirements. Proc of 12th IEEE International Requirements Engineering Conf., 2004: 66-75.