

# 생체인식 소프트웨어의 품질 평가를 위한 XSL 스타일 편집기 설계

윤영미\*, 양해술\*, 신석규\*\*

\*호서대학교 벤처전문대학원

\*\*한국정보통신기술협회 IT시험연구소

e-mail:770mi@hanmail.net, hsyang@office.hoseo.ac.kr, skshin@tta.go.kr

## Design of a XSL Style Edit for Quality Evaluation of Biometric Software

Young-Mi Yoon\*, Hae-Sool Yang\*, Seok-Kyoo Shin\*\*

\*Graduate School of Venture, Hoseo University

\*\*Telecommunication Technology Association

### 요 약

최근 IT분야의 보안 분야와 함께 생체인식 분야는 급성장하였으며 국내외에서 생체인식 소프트웨어에 대한 시험·인증 수요가 증대되고 있으나 생체인식 소프트웨어에 대한 품질을 평가할 수 있는 방법에 대한 연구는 미흡한 실정이다. 생체인식 소프트웨어의 품질시험과 평가를 위한 기준으로 평가모듈(Evaluation Module)이 도출되어야 하며, 평가모듈을 효율적으로 공유하거나 교환하기 위해 XML를 이용할 수 있다. 또한, 생체인식 소프트웨어의 인증과정과 관련된 문서들은 다양한 스타일로 구성되어 있으며 평가대상에 따라 관련문서들의 스타일도 변경되어야 한다. 본 논문에서 XML 기반 평가모듈에 XSL(eXtensible Stylesheet Language)을 이용하여 여러 스타일 시트로 적용할 수 있도록 XSL 스타일 편집기를 설계하였다.

### 1. 서론

IBG(International Biometric Group)의 조사에 따르면 생체인식 시장 규모는 매년 30%정도 성장하고 있으며[3], 국내에서도 매년 시장규모 및 범위가 확대되고 있는 추세다. 생체인식 소프트웨어는 지문, 얼굴, 홍채 등의 생체정보를 이용하는 분야로 구분될 수 있다. 현재는 지문인식 분야의 S/W가 50% 이상을 차지하고 있지만 점차적으로 홍채인식, 음성인식 분야로 그 범위가 확대되고 있으며 정보보안의 개념에서 중요한 역할을 담당할 것으로 보인다[4]. 그러나 생체인식 소프트웨어에 대한 품질을 평가할 수 있는 방법에 대한 연구는 미흡한 실정이다.

생체인식 소프트웨어는 다양한 생체인식 분야의 사용자 요구를 수용할 수 있어야 하며, 생체인식 소프트웨어의 품질향상 및 신뢰성 높은 소프트웨어 개발을 유도하도록 해야 한다.

따라서, 다양한 생체인식 분야의 소프트웨어를 시험·평가하기 위한 평가모듈을 도출해야 하며, 도출

된 평가모듈은 XML을 활용하여 효율적으로 공유하거나 교환할 필요성이 있다[5]. 또한, 생체인식 소프트웨어의 시험·인증과 관련된 문서들은 평가모듈의 내용변경시 관련 결과 문서들도 재작성해야 하는 문제점이 있다. XSL[8]은 XML 문서에 대한 템플릿 규칙을 이용하여 문서의 여러 요소에 대해 하나의 스타일 규칙을 적용할 수 있으며, 스타일 편집이 체계적이고 일관된 형태로 XML 데이터에 적용될 수 있다.

본 논문에서는 XML 기반으로 개발된 생체인식 소프트웨어의 평가모듈에 XSL을 이용하여 여러 스타일 시트로 적용할 수 있도록 XSL 스타일 편집기를 설계하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구들을 살펴보고 3장에서는 XML 기반 생체인식 소프트웨어의 평가모듈에 대해 설명한다. 4장에서는 XSL 스타일 편집기의 설계를 설명하고, 마지막으로 5장에서는 결론을 내린다.

2. 관련연구

2.1 국외 생체인식 시험·인증 현황

- NBTC(National Biometric Test Center, 미국): 1997년 Biometric Consortium이 설립하여 상업적 BioMetrics 제품의 성능평가를 수행하고 있다.
- BC(Biometric Consortium, 미국): 미정부의 생체인식 연구개발의 구심점으로 성능시험 요구사항 연구 및 생체인식시험 관련활동을 수행하고 있다.
- NIST(National Institute of Standards and Technology, 미국): BC에 참여하고 있으며, 얼굴인식시험프로그램에 대한 시험을 수행하고 있다.
- SNL(Sandia National Laboratories, 미국): 음성인식, 망막인식, 지문인식, 패턴인식, 서명인식 제품들에 대한 성능평가를 수행하고 있다.
- AfB(Association for Biometrics, 영국), BWG(Biometrics Working Group, 영국): 생체인식 시험 및 평가를 하는 정부기관들로 공동으로 생체인식 장치의 시험과 결과보고에 관한 가이드라인 연구를 수행하고 있다.
- BioIS(독일): 생체인식 S/W 시험실을 구축하여 생체인식 시스템의 성능평가를 수행하고 있다.
- TeleTrust(독일): 생체인식 S/W의 상호호환성 평가기준을 개발하였고, 일반적인 평가방법에 대한 연구를 수행하고 있다.
- Biotest: 생체인식S/W를 비교분석하는 표준안개발 및 시험베드 구축을위한 프로젝트를 진행중이다.
- 히다찌(일본):통산성(MITI)의 지원으로 생체인식 S/W 성능측정표준개발 및 시험을수행하고 있다.
- BASEL 프로젝트(이스라엘): 이스라엘 정부가 수행하는 과제로서 접근제어 시스템에 사용되는 손 및 손가락영상 획득장치의 시험을 수행하고 있다.

2.2 국내 생체인식 시험·인증 현황

국내의 경우 TTA의 S/W 시험인증팀에서 BioAPI 표준적합성 시험방법 및 절차의 개발과 함께 국내 생체인식 기술표준화를 주도하고 있으며, 이와 연계하여 시험·인증을 추진하고 있다. 또한, 생체인식포럼은 생체인식 분야의 국내 표준제정활동과 표준적합성 및 상호운용성 시험기술을 연구 중에 있어 국내에서도 생체인식 분야에 대한 시험·인증 기술이 개발되고 있다.

3. XML 기반 생체인식 소프트웨어의 평가모듈

최근 특정 응용 분야에 사용될 수 있는 문서의 내용(content)에 의미를 부여하여 사용자간에 자료를

보다 효율적으로 공유하고 교환할 수 있는 구조적 문서로서 XML(eXtensible Markup Language)[6]이 각광을 받고 있다.

생체인식 소프트웨어의 평가모듈은 XML을 이용하므로 XML 구문에 대한 유효성을 검사하기 위해 스키마[7]로 검증(validating) 과정을 거치게 된다[5]. <표 1>은 ISO/IEC 9126[1]를 기반으로 생체인식 소프트웨어의 품질특성과 부특성에 대한 요소를 정리한 것이고, <표 2>는 ISO/IEC 12119[2]를 기반으로 평가모듈 체계에 대한 스키마를 정리한 것이다.

<표 1> 품질특성의 XML 태그

요소	구분	
<EM>	기능	ISO/IEC 9126의 품질특성과 부특성의 정의와 선정된 평가항목, 에트리뷰트를 관리
	자식요소	<Characteristics>+ <Sub-Characteristics>+ <MetricSpec>+
<Characteristics>	기능	6가지 품질특성을 분류하여 관리
	자식요소	<Functionality>+, <Reliability>+ <Usability>+, <Efficiency>+ <Maintainability>+, <Portability>+
<Sub-Characteristics>	기능	각 품질특성의 부특성을 분류하여 관리
	자식요소	<Suitability>+, <Accuracy>+, <Interoperability>+, <Security>+ <Compliance>*, <Maturity>+ <FaultTolerance>+, <Recoverability>+, <Undersatandbility>+, <Learnability>+ <Operability>+, <Attractiveness>+ <TimeBehavior>+, <ResourceUtilization>+, <Analyzability>+, <Changeability>+, <Stability>+, <Testability>+, <Adaptability>+, <Installability>+, <CoExistence>+, <Replaceability>+

+: 반드시 필요한 요소, \*: 반복 가능하며 선택적인 요소

<표 2> 평가항목의 XML 태그

요소	구분	
<MetricSpec>	기능	메트릭명과 선정된 평가항목들을 관리
	자식요소	<Outline>+, <Application>+, <Reference>+, <Metric>+, <Process>+, <AnalysisReport>+
<Outline>	기능	메트릭의 개념, 측정 목적, 메트릭의 범주, 용어 설명, 비교를 관리
	자식요소	<Concept>+, <Purpose>+, <Category>+, <Term>+, <Remark>+
<Outline>	기능	메트릭의 개념, 측정 목적, 메트릭의 범주, 용어 설명, 비교를 관리
	자식요소	<Concept>+, <Purpose>+, <Category>+, <Term>+, <Remark>+
<Application>	기능	적용 대상 및 필요 자원, 기법, 적용시 고려사항을 관리
	자식요소	<ObjectResource>+, <Method>+, <Consideration>+
<Reference>	기능	참조 문서를 관리
	자식요소	<Document>+
<Metric>	기능	측정 항목, 측정 방법, 계산식을 관리
	자식요소	<Items>*, <Method>*, <Calculation>+
<Process>	기능	상세절차를 관리
	자식요소	<Resource>+, <Detail>+
<AnalysisReport>	기능	측정치, 매핑, 측정 결과의 해석, 보고사항을 관리
	자식요소	<Value>+, <Analysis>+, <Report>+

4. XSL 스타일 편집기의 설계

4.1 생체인식 소프트웨어의 시험 양식의 기록

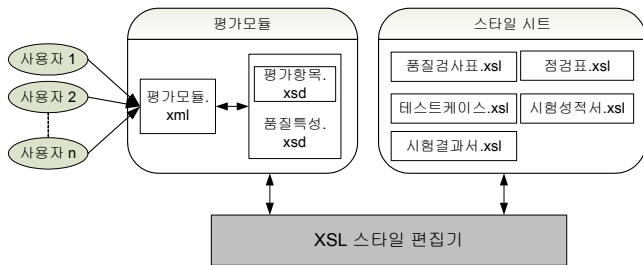
생체인식 소프트웨어의 품질 평가를 수행하는 과정에서 기록되는 양식으로는 <표 3>와 같은 항목들로 설정하였다.

<표 3> 시험 양식

시험양식	개념	기록항목
품질 검사표	품질시험 과정에서 활용할 수 있는 자료로서 품질시험원에 의해 필요한 최소 필요사항을 포함하여 테이블의 형태로 구성	메트릭명, 측정항목, 계산식, 결과영역, 결과값
점검표	메트릭의 측정항목에 관련된 점검사항표	기능 점검표, 상호운용성 점검표, 표준준수 확인 점검표 등
테스트 케이스	명세대로 동작하는가에 대한 점검	시험사례명, 시험방법, 입력자료, 예상결과, 시험결과 등
시험 결과서	시험결과에 대한 집계 결과의 기록	품질특성, 부특성, 메트릭, 측정값
시험 성적서	문제점 기록서 및 시험결과서를 바탕으로 하여 발견된 문제점에 대해 품질특성의 관점에서 전반적인 문제점을 제시	시험대상, 품질특성, 시험측정 결과

4.2 XSL 스타일 편집기의 구조

본 연구에서 제안하는 XSL 스타일 편집기는 여러 평가자가 생체인식 소프트웨어에 대한 품질평가를 수행하는 과정에서 관련된 여러 문서들을 체계적이고 일관된 형태로 적용할 수 있게 지원하는 편집도구로서 그 구조는 (그림 1)과 같다. (그림 1)은 <표 1>과 <표 2>의 태그 셋을 이용하여 평가모듈의 유효성을 검사하고, 유효한 문서이면 도출하고자 하는 스타일 시트를 적용한 것이다.



(그림 1) XSL 스타일 편집기 구조

XML과 XSL을 도입하여 생체인식 시스템을 구축하였을 경우 시험·인증과정에 관련된 스타일을 일관되게 유지관리할 수 있으며, 필요에 따라 여러 스타일에 적용할 수 있다. 또한, 데이터와 스타일을 별도로 관리할 수 있고, 스타일 적용을 여러 요소에 반복적용 할 수 있다.

4.3 XSL 스타일 편집기의 기능

XSL는 XML 문서를 다른 문서로 변환해줄 수 있

는 언어로서 문서를 변환할 경우 DOM이나 SAX를 이용하는 것보다 사용이 용이하다. XSL 프로세서가 XML 문서와 XSL 스타일 시트를 받아들여 출력결과를 얻는 과정은 변환과 포매팅의 단계로 나눌 수 있다. 첫 번째 단계는 XSLT 트리 변환으로 XML 문서에 대한 트리를 출력을 위한 트리(포매팅 객체 트리)로 변환한다. 두 번째 단계는 XSL-FO 포맷터가 중간 객체 트리를 이용하여 XSL-FO 요소로 지정된 스타일 결과대로 화면, 음성, 또는 기타 매체에 해당하는 출력할 수 있는데, 현재 많이 사용되는 예가 HTML로의 변환이다.

생체인식 소프트웨어의 평가자는 XML 기반으로 설계된 여러 메트릭에 ‘품질검사표’ 스타일 시트를 반복적으로 적용할 수 있으며, 특정 스타일을 적용할 경우 <xsl:for-each> 또는 <xsl:template> 태그에 의해 XML의 여러 부분에 다양하게 적용할 수 있다. 또한, XSL 스타일 편집기에서는 반복 구조를 중첩될 수 있으며, XSL 스타일을 효율적으로 공유하거나 교환할 수 있기 때문에 XML 데이터의 구조에 따른 XSL 스타일의 변경 적용이 용이하다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<EM xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
NamespaceSchemaLocation="quality.xsd" Characteristics="Functionali
ty" Sub-Characteristics="Accuracy">
...
<MetricSpec name="기능구현 정확성 정보 제공">
  <Outline>
    <Concept>'기능 구현 정확성 정보 제공'이란 생체인식 시스템의 각
    기능과 그 기능에 대한 명세가 정확히 일치하는가를 의미한다.
    </Concept>
    <Purpose>이 평가모듈은 제품설명서 및 사용자문서에 ...</Purpose>
    <Category>기능성-기능구현 정확성 정보 제공</Category>
    <Term>해당 사항 없음</Term>
    <Remark>사용자 문서에 생체인식 시스템의 각 기능에 ...</Remark>
  </Outline>
  <Application>
    <ObjectResource>적용대상: 사용자 문서, 생체인식 시스템
    </ObjectResource>
    <Method>해당 사항 없음</Method>
    <Consideration>해당 사항 없음</Consideration>
  </Application>
  <Reference>
    <Document>ISO/IEC12119 3.2.2, ISO/IEC9126 8.1.2</Document>
  </Reference>
  <Metric>
    <Items>생체인식 시스템에서 제공하는 전체 기능 수(A)</Items>
    <Method>A: 생체인식 시스템에서 제공하는 전체 기능 수
    ...</Method>
    <Calculation>기능 구현 정확성 정보 제공(AIP)= B/A</Calculation>
  </Metric>
  <Process>
    <Resource>기능점검표</Resource>
    <Detail>1) 기능항목을 식별한다. 미리 작성된 기능점검표가 ...
    </Detail>
  </Process>
  <AnalysisReport>
    <Value>0<!-- 기능 구현 정확성 정보 제공(AIP)&lt;!--=1 </Value>
    <Analysis>-측정값은 1에 가까울수록 바람직하다.
    ...
  </AnalysisReport>
  <Report>1) 기능 구현 정확성 정보 제공: AIP ... </Report>
</AnalysisReport>
</MetricSpec>
</EM>
```

(그림 1) 기능 구현 정확성 정보 제공.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/
Transform" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
/>
<xsl:template match="/">
<html>
<head />
<body>
<xsl:for-each select="EM">
...
<table align="left" border="1" page-break-after="auto"
page-break-before="auto">
<tbody>
<tr>
<td align="center" width="168">Metric Name</td>
...
<xsl:for-each select="MetricSpec">
<xsl:for-each select="Outline">
<xsl:for-each select="Remark">
<xsl:apply-templates />
</xsl:for-each>
</xsl:for-each>
</xsl:for-each>
...
<tr>
<td align="center" height="1" rowspan="2" width="168">Measurement Item</td>
...
<xsl:for-each select="MetricSpec">
...
<li>
<xsl:apply-templates/>
</li>
...
<td align="center" width="168">Calculation Equation</td>
...
<td align="center" width="168">Result Region</td>
...
<td align="center" width="107">Result Value</td>
...
<td align="center" width="168">Problem</td>
...
<xsl:template match="AnalysisReport">
<xsl:apply-templates />
</xsl:template>
</xsl:stylesheet>

```

(그림 2) 품질검사표.xslt

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/
Transform" xmlns:fo="http://www.w3.org/1999/XSL/Format" xmlns:xsi="
http://www.w3.org/2001/XMLSchema-instance">
<xsl:variable name="fo:layout-master-set">
<fo:layout-master-set>
<fo:simple-page-master master-name="default-page" page-height="
11in" page-width="8.5in" margin-left="0.6in" margin-right="0.6in">
<fo:region-body margin-top="0.79in" margin-bottom="0.79in" />
</fo:simple-page-master>
</fo:layout-master-set>
</xsl:variable>
<xsl:template match="/">
<fo:root>
<xsl:copy-of select="$fo:layout-master-set" />
<fo:page-sequence master-reference="default-page" initial-page-n
umber="1" format="1">
<fo:flow flow-name="xsl-region-body">
<fo:block>
<xsl:for-each select="EM">
<fo:table text-align="left" width="100%" space-before.optimum
="1pt" space-after.optimum="2pt">
<fo:table-column column-width="168pt" />
<fo:table-column column-width="349pt" />
<fo:table-column column-width="107pt" />
<fo:table-column column-width="126pt" />
<fo:table-column />
<fo:table-body>
<fo:table-row>
<fo:table-cell text-align="center" width="168pt" padding-start="3
pt" padding-end="3pt" padding-before="3pt" padding-after="3pt" displa
y-align="center" border-style="solid" border-width="1pt" border-color
="black">
<fo:block>Metric Name</fo:block>
</fo:table-cell>
<fo:table-cell number-columns-spanned="4" number-rows-spann
ed="3" text-align="left" width="594pt" padding-start="3pt" padding-en
d="3pt" padding-before="3pt" padding-after="3pt" display-align="cente
r" border-style="solid" border-width="1pt" border-color="black">
<fo:block>
<xsl:for-each select="MetricSpec">
<xsl:for-each select="Outline">
<xsl:for-each select="Remark">
<xsl:apply-templates />
</xsl:for-each>
...

```

(그림 3) 품질검사표.xml

(그림 2)는 '기능 구현 정확성 정보 제공' 매트릭의 이고 (그림 3)은 '품질검사표' 스타일 시트이고, (그림 4)는 '품질검사표' 스타일 시트를 HTML 문서로 변환하기 위해 XSL-FO를 적용한 예이다.

생체인식 소프트웨어의 품질평가를 수행하는 과정에서 XSL 기반 스타일 편집기는 평가모듈에 대한 데이터의 종류와 개수에 따라 결과문서는 다른 형태로 적용할 수가 있으며, 시험·인증과 관련된 문서들을 고루 지원하지 못하는 문제점을 해결할 수 있다.

## 5. 결론

본 논문에서는 다양한 생체인식 분야의 소프트웨어를 시험·평가하기 위한 XML 기반 평가모듈에 템플릿 규칙을 이용하여 문서의 여러 요소에 대해 하나의 규칙을 적용할 수 있고, 스타일 편집이 체계적이고 일관된 형태를 적용할 수 있도록 XSL 스타일 편집기를 설계하였다. 따라서, 생체인식 소프트웨어의 평가모듈에 대한 구조변경시 스타일 적용이 유연하고 확장가능하기 때문에 작업진행의 효율성을 보장할 수 있을 것이다.

향후 연구과제는 본 논문에서 제안된 XSL 스타일 편집기를 실제 생체인식 시스템에 활용될 수 있는 도구로서 개발할 예정이다.

## 참고 문헌

- [1] ISO/IEC 9126, "Information Technology-Software Quality Characteristics and Metrics-Part 1, 2, 3", 1997
- [2] ISO/IEC 12119, "Information Technology-Software Package-Quality Requirements and Testing", 1994
- [3] <http://www.biometricgroup.com/>
- [4] 2004년도 정보통신표준화백서, 한국정보통신기술협회, 2004.
- [5] 윤영미, 황석형, 양해술, "생체인식 소프트웨어의 평가모듈을 위한 XML 스키마 설계", 한국정보처리학회 춘계학술대회, pp. 377-380, 2004. 11.
- [6] <http://www.w3.org/TR/2004/REC-xml-20040204>
- [7] <http://www.w3.org/TR/2001/REC-xmlschema-0-20010502/>
- [8] <http://www.w3.org/TR/2001/REC-xsl-20011015/>