

BioAPI기반 시스템 모듈을 검증하기 위한 적합성시험 방법 연구

이 유 영* · 권 영 빈**

요 약

생체인식기술을 응용한 제품이 다양해지면서 상호운용성의 문제가 제기되어 국제표준화가 진행 중인 BioAPI(Biometric Application Programming Interface)를 기반으로 한 생체인식 제품의 개발이 증대되고 있다. 그러나 BioAPI를 적용하여 개발된 제품에 대한 성능 측정 및 표준규격의 적합성 여부에 대한 신뢰성 있는 평가 도구가 마련되어 있지 않은 상태이다. 본 논문은 생체인식시스템이 BioAPI 규격의 요구사항 및 기준에 얼마나 만족하여 개발하였는가를 검증하기 위한 적합성시험 방법을 연구하였다. 제안된 적합성 시험기술의 첫 번째는 BioAPI명세서가 제공하는 각 함수들을 제대로 구현하였는가에 대한 확인과 사용여부를 평가하는 것이다. 이것은 응용에서 API(Application Provider Interface)함수를 호출하면 프레임워크를 통해 BSP(Biometric Service Provider)를 실행하는 것으로 이때 파라미터와 선행함수의 선택이 필요하다. 두 번째는 BioAPI의 해당 테스트케이스를 분석하여 모듈관리, 핸들기능, 검증기능에 대한 시나리오를 평가하는 것이다. 실험은 BioAPI 컨소시엄에서 제공하는 샘플프로그램과 상용 지문검증시스템의 BSP를 사용하여 제안하는 적합성 평가 방법에 대한 실험을 수행하였다. 이에 따라 BioAPI를 기반으로 한 BSP들이 요구사항에 적합하게 개발되었는지를 판단할 수 있었다.

A Study on Conformance Testing Method to Verify the BioAPI Based System Module

Yoo-Young Lee* · Young-Bin Kwon**

ABSTRACT

Recently the biometric recognition technology is intensively studied and the standardization of the technology has been highly demanded for its commercialization. Currently many biometric recognition products are being developed based on the BioAPI(Biometric Application Programming Interface) specification. However, the reliable testing tools (or scenarios) to evaluate performance and conformance of the products are not shown yet. In this paper, a conformance testing method is presented, which verifies a biometric recognition system to meet the requirements of the BioAPI standard. Two different testing procedures are used in the proposed method. The first procedure evaluates that each functions offered in the BioAPI specification are correctly implemented and that the functions are actually used in the system. Through the procedure, a BSP(Biometric Service Provider) system is executed on the framework of the BioAPI functions. It requires selection of parameters and precedent functions that should be executed first. The second procedure evaluates the abilities of module management, handling operations and verification process by the analysis of the test cases. It tests the correctness of the system operation when a testing scenario is given. The proposed testing method is applied on a fingerprint verification BSP using the sample BSP provided by the BioAPI consortium. The experimental results shows the benefits of the proposed testing method.

키워드 : BioAPI, 생체인식(Biometric Recognition), 적합성시험(Conformance Test), 지문검증시스템(Fingerprint Verification System)

1. 서 론

생체 인식은 신체적 특성이나 행동 양식을 이용하여 신원을 확인하는 방법으로서, 증명서의 소지나 분실의 위험과 압호를 기억해야하는 등의 불편함이 거의 없다. 이런 장점 때문에 생체 인식에 대한 연구가 활발히 진행되어 왔으며 최근에는 이것을 이용한 제품들이 선보이고 있고 그 이용도

증가하고 있는 추세이다. 생체 인식은 일반적으로 생체의 특징점을 얻기 위해 필요한 데이터를 얻는 하드웨어 장치와 추출된 특징점을 인식하는 소프트웨어 그리고 시스템의 응용 프로그램에 의해 운용된다. 시장이 확대되고 다양한 형태의 제품이 선보임에 따라 각 단계별 제품 간에 호환성 문제가 제기되고 있는데, 지금까지는 생체 인식용 개발 장비에서 제공되는 소프트웨어 개발 환경 SDK(Software Development Kit)에 맞추어 응용 프로그램을 작성해야하는 형태의 개발환경이 대부분이어서 개발 환경이 다른 생체 인식 제품 간에는 호환성이 거의 없는 실정이다.

* 이 논문은 2004년도 중앙대학교 교비연구비 지원에 의해 연구되었습니다.

† 준 회원 : 중앙대학교 대학원 컴퓨터공학과

** 종신회원 : 중앙대학교 컴퓨터공학과 교수
논문접수 : 2004년 7월 9일, 심사완료 : 2004년 11월 2일

이러한 상호운용성 문제가 현안으로 대두되면서 시스템을 통합하여 표준화하기 위한 많은 연구가 이루어지고 있다. 그 대표적인 예로 미국에서는 생체인식 응용프로그램 인터페이스(Biometric Application Programming Interface : BioAPI) 컨소시엄을 구성하여 "BioAPI 명세서 V1.1"을 제공하고 있다[1]. BioAPI 명세서의 목적은 생체인식기술에서 사용하는 알고리즘, 자료구조 등의 각 형태들을 하나로 융합하고 응용 개발자와 통합자, 사용자를 위한 장치의 독립성과 상호운용성에 대한 표준 생체인식시스템 응용인터페이스를 개발하는 것이다[2]. 이러한 이유로 최근 산업체에서는 BioAPI에 기반한 생체인식시스템을 개발 중에 있으며 제공된 제품이 BioAPI 규격에 맞도록 제대로 구현되었는가에 대한 준용 여부를 검증하기 위해서는 적합성시험 도구가 필요할 것이다. 적합성시험이란 개발된 제품이 표준규격의 요구사항에 만족하여 구현하였는가의 충실함에 대한 평가를 말한다[3]. 즉, 생체인식시스템에 대하여 BioAPI 표준규격을 정확히 적용하여 개발하였는지 인증하여 준다면 제품의 중복개발 및 표준규격의 준용성 여부를 검사하기 위하여 생성되는 시간과 비용의 절감이 가능하다. 또한 제품의 여러 오류요소를 발견하고 이를 개선한다면 제품의 질적인 향상과 사용자에게 제품에 대한 신뢰성을 줄 수 있을 것이다. 뿐만 아니라 국제기준에 맞는 제품개발을 유도하여 제품의 국제경쟁력 향상 및 국내 제품의 해외 수출에도 큰 영향을 줄 수 있으며 생체인식시스템의 이용활성화 및 개발활성화를 통한 시장 확대가 가능하다. 그러나 이러한 장점에도 불구하고 현재 국·내외적으로 BioAPI를 적용하여 개발된 제품에 대한 성능 측정 및 표준규격의 적합성 여부에 대한 평가 방법과 도구가 마련되어 있지 않은 상태이다.

본 논문에서는 개발된 생체인식 서비스 공급자(Biometric Service Provider : BSP)인 생체인식시스템이 BioAPI 표준규격의 각 모듈에 해당하는 사항 및 기준에 얼마나 만족하여 개발하였는가를 검증하기 위한 적합성시험 방법을 도출하고 모델을 제안한다. 일반적으로 생산된 제품에 적용되는 적합성시험 도구는 시험기술, 시험절차, 시험결과문서로 구성된다[4]. 시험기술이란 표준명세서에서 요구한 항목과 기능에 대하여 각 제품이 이 표준을 준용하는가를 검사할 수 있는 요구사항분석 및 적합성시험 방법을 도출하는 것이다. 현재 BioAPI의 기능은 크게 모듈관리(Module Management), 핸들기능(Handle Operation), 검증(Verification), 식별(Identification), 분산모듈, 데이터베이스로 나뉘어 제공되고 있다. 그러나 국·내외적으로 BioAPI를 기반으로 하는 제품이 대부분 개발 중에 있으며 현재 시장에 있는 제품의 경우 BioAPI 기능의 일부만을 지원하고 있어 이 제공되는 기능을 중심으로 연구한다. 따라서 본 연구에서 시험기술은 두 가지에 초점을 두어 방법을 도출하였다. 첫 번째는 BioAPI 명세서가 제공하는 각 함수들을 제대로 구현하였는가에 대한 확인과 그 사용여부를 평가하는 것이고 두 번째는 BioAPI의 해당 테스트케이스를 분석하여 모듈관리, 핸들기능, 검증기능에

대한 시나리오를 평가하는 것이다. 전자의 적합성평가는 응용 공급자 인터페이스(Application Provider Interface : API) 함수를 호출하고 프레임워크를 통해 BSP를 실행하는 것으로 이때 파라미터와 선행함수에 대한 선택이 중요하다. 후자의 경우 BioAPI명세서에서 분석한 결과로 도출된 시나리오에 대하여 그 동작 기능이 올바르게 수행되는지에 대한 평가를 말한다. 시험절차란 시험을 시작하기 이전에 구성하며 구현된 모듈들을 기술적인 처리순서에 잘 따르는지에 대하여 실질적으로 시험하는 것이다. 본 연구에서 시험절차를 위해 BioAPI 컨소시엄에서 제공하는 샘플프로그램과 A사의 지문검증시스템의 2개의 BSP를 사용하여 제안된 적합성 평가 방법을 적용했다. 기본함수와 시나리오에 대한 적합성 시험은 실제 구현을 통하여 기본함수 테스트 실행 중에 보여주는 중간결과와 모든 기본함수의 테스트가 종료된 후에 보여주는 최종 종합 결과, 그리고 마지막으로 각 테스트케이스를 검증한 시나리오의 결과로 분류하여 도출하고 이를 분석하여 적합성 여부를 판단한다. 시험결과문서는 인증기관을 두어 소프트웨어 개발자나 최종 사용자들에게 제공하고 표준규격에 대한 오류 또는 모호한 부분이 발견된다면 정정하기 위하여 논의하여야 할 것이다.

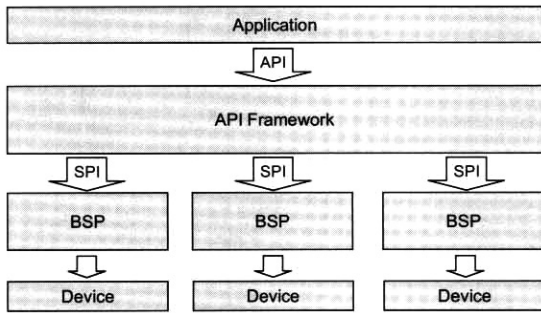
본 논문의 구성을 살펴보면 다음과 같다. 2장은 BioAPI의 정의에 대한 간단한 개요를 설명하고 기존 표준명세서에 대한 적합성시험 연구에 대하여 분석한다. 3장은 본 논문의 BioAPI에 대한 표준 적합성 평가 방법을 제안하고 설명한다. 4장은 제안된 모델을 설계하고 구현하며 마지막으로 5장에서는 실험을 통하여 결과를 도출하고 분석하고자 한다.

2. 관련 연구

생체인식기술을 하나의 새로운 인증 방법으로 산업에 광범위하게 적용하기 위해서는 여전히 해결되어야 하는 많은 문제들이 남아 있다. 즉, 표준화에 대한 체계마련과 지속적인 연구가 부족하여 국내 생체인식 산업의 성장을 저해하는 가장 큰 요인으로 지적되고 있으며, 유관 기술 및 응용분야와 활발한 협력이 이루어지지 못하고 있는 실정이다. 더욱이 불특정 다수가 아닌 전체 국민이 생체인식을 이용하기 위해서는 표준화를 통해 상호운용성을 제공하는 시스템의 설계, 데이터의 저장 및 관리, 보안과 사용하는 하드웨어의 표준규격이 필요하다. 생체인식기술의 공통적인 기능들을 표준규격으로 구성한다면 개발자나 사용자들에게 개발시간 단축과 비용 절감의 이점이 제공될 것이다[5]. 또한 표준규격을 적용한 제품들은 개발된 상품의 질과 좀 더 선택적이고 경쟁력 있는 시장, 제품의 가격, 특정 업체에 의존하지 않는 균형적인 발전을 가져올 수 있다. 마지막으로 대중적인 관심이 높아질 것이고 그 이용도 보편화 되어 정보보안 등이 구현된 제품의 제조와 활용이 활성화 될 것이다. 이러한 목적으로 미국의 BioAPI 컨소시엄에서는 BioAPI명세서를 개발하였다.

2.1 BioAPI(Biometric Application Programming Interface)

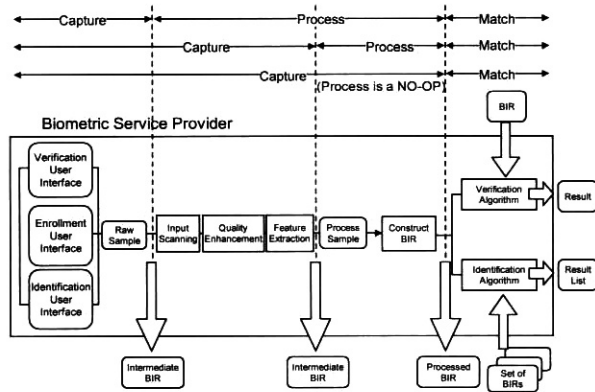
생체인식기술 전 분야에 적용 가능한 응용프로그램 인터페이스를 제공하기 위해 BioAPI 컨소시엄이 구성되었으며 사용자그룹과 개발자 그룹으로 구성된 표준화 관련 민간단체로 현재 “BioAPI명세서 버전1.1”이 개발되어 있다.



* 출처 : Catherine Tilton, The BioAPI Specification, INCITS MI Meeting, 2002.

(그림 1) BioAPI 구조

이 표준규격에 목적은 생체인식기술의 각 형태들을 하나로 융합하고 응용개발자와 통합자, 사용자를 위한 장치의 독립성과 플랫폼을 가져오기 위한 표준 생체인식시스템 응용프로그램 인터페이스를 개발하는 것이다[2]. BioAPI의 애플리케이션은 API를 통하여 프레임워크와 인터페이스하며 BSP는 서비스 공급자 인터페이스(Service Provider Interface : SPI)를 통하여 프레임워크와 인터페이스를 수행한다. 생체인식시스템은 BSP함수에서 등록(Enroll) 및 검증(Verify), 식별(Identify) 기능을 구현하고 각각의 BSP는 해당하는 장치들을 가진다[1].



*출처 : BioAPI Specification Ver1.1, 2001[2]

(그림 2) 가능한 구현 단계 방법

위 그림과 같이 모든 유형의 생체인식기술의 기본적인 모델은 동일하다. 첫 번째로 사용자의 초기 등록 시 템플릿을 구성한다. 이 작업은 센서를 사용하여 몇 개의 샘플을 수집하고 그 샘플로부터 특징을 추출한다. 그 결과를 템플릿으로 구성하는 과정으로 등록(Enrollment)이라 하며 이 템플릿

은 응용프로그램에 의해 저장된다. 이때 기존의 암호인증방식에서 암호가 저장되어 사용할 수 있다. (그림 2)는 검증과 인식의 다양한 단계들을 BSP라는 박스 안에서 볼 수 있다. 구현 단계는 크게 영상획득(Capture), 처리(Process), 정합(Match)로 구분할 수 있으나 BSP에 따라 달라질 수 있다.

2.2 적합성 시험 개요

적합성이란 개발된 제품이 표준 또는 표준 규격의 요구사항에 만족하여 구현하였는가의 충실함에 대한 평가로 정의할 수 있다[3]. 즉, 표준 명세서의 모든 사항에 대하여 그 사용 형식이 잘못 되었거나, 생략되었을 때에는 생략됨을 알리는 메시지를 제공하여야 한다. 적합성 시험 기술은 표준 명세서, 시험 도구(시험기술, 시험절차, 시험결과 문서), 시험을 관찰할 수 있는 조직으로 구성되어야 한다[5].

본 연구에서의 '표준 명세서는 “BioAPI명세서 버전1.1”의 기술문서를 사용한다. 일반적으로 시험 도구의 시험 기술은 표준 명세서에서 요구한 항목과 기능에 대하여 각 제품이 이 표준에 준용한지를 검사할 수 있는 요구사항 분석 및 적합성 시험 방법을 도출한다. 본 연구의 시험 기술에 대한 내용은 3장에서 설명하였다. 시험 절차는 시험을 시작하기 이전에 구성하며 구현된 모듈들을 기술적인 처리순서에 잘 따르는지에 대하여 실질적으로 시험한다. 본 연구의 시험 절차에 대한 내용은 4장에서 설명하였다. 마지막으로 시험 결과 문서는 소프트웨어 개발자나 최종 사용자들에게 제공된다. 이때, 표준 규격에 대한 예러 또는 모호한 부분이 발견된다면 이들을 정정하기 위하여 표준 규격을 관리하는 기관과 논의하여야 할 것이다.

2.3 기존의 표준적합성시험 연구에 대한 분석

BioAPI에 대한 표준적합성 시험에 기존 연구사례는 아직 존재하지 않는다. 그러나 생체인식기술 분야가 아닌 다른 표준명세서에 따른 적합성시험 사례를 조사하고 이에 대한 분석을 통하여 생체인식기술의 표준인 BioAPI에 적합성 시험에 적용하고자 한다.

SyncML(Synchronization Markup Language) 표준은 이종시스템 간의 공통적 정보표현 언어의 제정과 그 정보의 교환 절차를 규정한 것이다[6]. SyncML에 대한 적합성을 검증받기 위해서는 우선 SyncML그룹에서 제시하는 기본 12가지 Two-Way을 분석한다. Two-Way란 클라이언트와 서버 간에 상호메시지를 주고받을 수 있다는 시나리오이다. 이 시나리오의 클라이언트와 서버간의 메시지 교환을 확인하였고 이를 통해서 12단계의 테스트케이스를 통과해 적합성을 검증하였다. ebXML(Electronic business XML) 표준이란 전자상거래 메시지교환을 위한 범용적인 통신 프로토콜 방식을 정의하며 신뢰성 있고 안전한 전자상거래 문서전달을 지원할 수 있는 신뢰 전송서비스와 메시지 봉투 구조를 정의한다. 적합성 시험은 ebXML 메시지 서비스 2.0을 사용하였으며 요구사항을 추출하고 각 항목을 평가하기 위한 테스트

케이스를 정의하였다[7]. 즉, XML 형태의 파일로 저장되어 테스트 케이스 저장소에 함께 저장되면, 테스트 드라이버는 이 파일의 정보를 이용하여 테스트를 실행한다. LOM(Learning Object Metadata)은 IEEE-SA에 따라 2002년에 표준으로 승인되었다. IEEE LOM 표준의 승인과 네트워크 학습 기술의 진보로 다수의 데이터베이스 생성자, 콘텐츠 개발자와 학습관리시스템 벤더는 LOM 모델을 사용하여 관련된 학습 리소스를 기술하게 되었다[8]. 이때 메타데이터(Metadata)가 LOM 모델에 적합한지를 판단하기 위해 메타데이터를 입력 받으면 그것을 분석하고 스키마를 확인 후 필수요소들을 검증하는 방식으로 적합성시험을 연구하였다.

이와 같이 표준명세에 대한 몇 가지 적합성시험 연구에 대하여 살펴보았다. 관련연구를 분석한 결과 적합성을 검증하기 위한 정확한 요구사항 추출 및 테스트케이스에 대한 분석이 필요함을 알 수 있었다. 그러나 적합성 시험을 위한 요구사항과 테스트케이스는 각 표준에 따라 각기 다른 방법론의 도출이 필요하다. 따라서 본 연구의 생체인식기술의 BioAPI 표준 적합성시험의 경우 이에 맞는 요구사항과 테스트케이스에 해당하는 시나리오를 분석하여 적합성 시험방법을 제안하였다.

3. BioAPI 표준적합성 시험방법의 제안

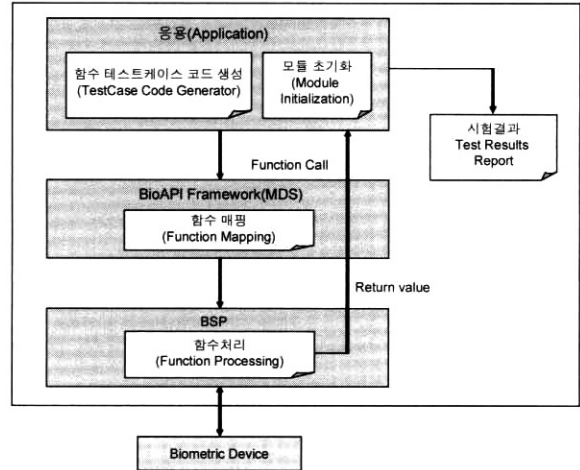
본 연구의 적합성 시험은 두 가지에 초점을 두어 개발하였다. 첫 번째는 표준명세서의 BSP의 함수별로 그 사용여부를 평가하는 것이고 두 번째는 표준규격의 시나리오를 분석하여 이에 따른 BSP의 모듈관리, 핸들, 검증, 식별, 클라이언트/서버, 데이터베이스 기능에 대한 테스트케이스를 평가하는 것이다.

3.1 BioAPI 함수별의 적합성 시험

BSP의 표준 적합성 시험을 위해서는 같은 이름의 API 함수를 호출하여야 한다. API 함수들은 내부적으로 해당 BSP 함수에 대한 정보를 로드할 때 모듈 디렉토리 서비스(Module Directory Services : MDS) 모듈로부터 호출한다. 그때 SPI와 연결되어지고 프레임워크에서 에러검사 및 파라미터 검사를 수행한 후 실행되게 된다. API함수로 BSP를 접근하기 위해서 우선 프레임워크가 연동할 수 있도록 기본적인 버전 정보와 초기화 작업이 필요하다. 그리고 현재 설치된 BSP 제대로 메모리에 로드되고 프레임워크에 부착되어 있는지 확인하고 API함수를 사용하여 BSP 함수를 실행 가능하도록 설정한다. 다음은 함수별 적합성 시험을 위한 전체적인 절차과정을 보여준다.

우선 각 해당 함수에 대한 적합성 시험을 위한 테스트케이스를 생성하고 미리 설치된 BSP의 모듈을 검사하여 메모리에 로드되어 있는지 확인한다. 각 함수들의 검사시 해당하는 파라미터 지정과 이전에 선행되어야 할 함수에 따라 여러 개의 함수가 연속적으로 수행되어야 할 경우가 있기 때

문에 많은 실행과정이 필요하다. GUI에서 테스트할 함수를 호출하면 프레임워크에서 함수매핑하게 되고 해당 SPI함수를 통하여 BSP 함수가 실행된다. BSP가 실행되는 각각의 함수에 대한 리턴값이 애플리케이션에 전달되고 모든 테스트 수행과정이 끝나면 결과를 화면에 출력한다.



(그림 3) 함수별 적합성시험 절차 과정

3.2 BioAPI 시나리오를 통한 BSP 모듈 적합성 시험

적합성 시험 방법은 각 BSP가 BioAPI기반으로 함수를 구현하였는지의 여부를 체크하는 것이 우선적으로 수행되어야 할 것이다. 그러나 적합성 시험은 기본함수에 대한 테스트는 물론 BioAPI기반의 동작 기능이 올바르게 수행되는지에 대한 기능테스트가 필요하다. BioAPI는 시나리오를 크게 6가지로 나누어 볼 수 있다. 첫 번째 모듈관리는 BSP를 로드하고 해제하며, 두 번째 핸들관리는 BIR에 대한 핸들을 얻거나 BIR의 헤더정보를 가져온다. 세 번째 검증기능은 영상을 등록하고 검증을 위한 영상을 획득하고 처리하여 기존 영상과 1:1로 정합하는 것이다. 네 번째 식별기능은 사용자 입력에 대하여 모든 저장된 템플릿 집합을 하나씩 정합하는 것으로 템플릿 집합 중에서 사용자와 가장 유사한 데이터를 지정하는 방법이다. 다섯 번째 클라이언트/서버 기능은 분산 환경에서 데이터베이스 저장지 좀 더 안전한 환경에서 실행할 수도 있도록 한다. 데이터베이스기능은 선택적으로 사용할 수 있으며 템플릿에 대한 DB 기능을 제공하여 준다. 그러나 국·내외적으로 BioAPI 해당하는 기능을 모두 구현한 제품은 개발되지 않아 적합성 시험에 어려움이 있다. 즉, 적합성 시험을 하기 위해서는 각 해당 함수에 적당한 파라미터를 넘겨주고 선행함수에 대한 설정을 해주므로 그 과정을 수행할 수 있다. 하지만 이런 함수와 기능자체를 지원하지 않은 BSP를 대상으로 적합성 검사를 수행할 경우에는 필요한 설정들을 할 수가 없게 된다.

따라서 본 연구는 현재 제공되는 BSP의 기능인 모듈관리, 핸들관리, 검증기능을 중심으로 하여 BioAPI 적합성 시험을 수행하고자 한다. 본 절에서는 이 세 가지 기능의 각 시나리

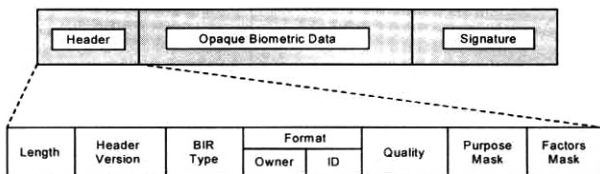
오에 대한 적합성 시험 방법을 분석하고 설계하였다.

3.2.1 모듈관리(Module Management) 시나리오

모듈관리 기능의 모듈로드는 사용하고자 하는 BSP 모듈이 선택되었을 경우 해당 BSP를 위한 메모리공간을 할당하고 로드하여 실행 준비상태로 만든다. 모듈부착은 표준과 맞지 않는 BSP의 각각의 함수명을 표준에 맞도록 연결하는 작업을 하여 사용자가 표준 API 함수를 호출했을 경우 원하는 함수가 실행되도록 한다. 모듈탈착은 모듈 핸들에 의해 인식된 BSP의 모듈을 응용으로부터 분리한다. 모듈언로드는 모듈로드에서 수행하였던 BSP 모듈의 초기화 작업을 제거하고 모든 이벤트에 대하여 해제한다. 따라서 BioAPI_ModuleAttach 함수는 BioAPI_ModuleLoad 함수 호출 후에 호출하며 BioAPI_ModuleUnload 함수는 BioAPI_ModuleDetach 함수 수행 후에 호출한다. 모듈핸들을 가져올 수 없다면 적합성 테스트를 할 수 없으므로 TEST_UNKNOWN 메시지를 출력한다.

3.2.2 핸들기능(Handle Operation) 시나리오

BioSPI_GetBIRFromHandle 함수는 생체인식 식별 레코드(Biometric Identification Record : BIR)에 대한 핸들을 얻거나 BioSPI_GetHeaderFromHandle 함수는 모듈의 핸들로부터 BIR의 헤더정보를 가져온다. BioSPI_FreeBIRHandle 함수는 BIR에 대한 핸들을 해제한다. 이렇게 BIR의 헤더구조와 핸들정보를 얻기 위해선 위 3가지 핸들관련 함수를 사용하여야 한다. 다음은 BIR의 개념과 구조에 대한 간단한 설명이다. BIR은 등록/검증/인식 등을 수행하기 위하여 데이터와 관련 정보를 담고 있는 헤더 부분으로 이루어져 있다. 일반적으로 연속적으로 저장된 자료는 템플릿 작업을 수행하는 등록과정을 통하여 BIR을 생성한다. BIR의 헤더 구조는 (그림 4)과 같이 정의되어 있다.

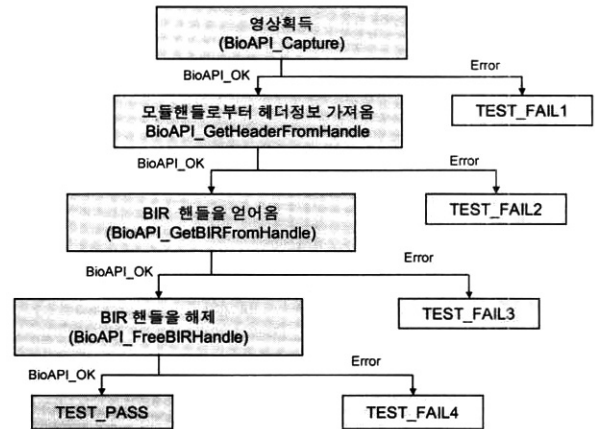


출처 : BioAPI Specification Ver1.1, 2001[2]

(그림 4) BIR의 구조

유형(Type)은 미가공(Raw), 중간처리(Intermediate), 처리된(Processed) 데이터 등의 형식으로 구성되어 있으며 형식(Format)은 업체와 그룹(Owner)에 대한 아이디를 등록한다. 목적(Purpose)은 영상등록의 목적을 구분하며 요소(Factors)는 14개의 생체인식 종류를 구분하여 준다. 서명(Signature)는 선택적으로 사용할 수 있으며 템플릿을 생성할 때 불투명 생체 데이터(Opaque Biometric Data)에 더하여 부호화(Signed)/암호화(Encrypted)와 같이 암호키를 사용하는 페이로드(Payload)의 형식으로 구성할 수 있다. BIR 구조의 적합성 시험을 하기 위해서는 다음 BioAPI_GetHeaderFromHandle 함수

의 테스트가 필요하며 이 함수가 정상적으로 수행되었다면 BIR 헤더구조가 정상적으로 확인되었음을 의미한다.



(그림 5) 핸들기능 시나리오 설계

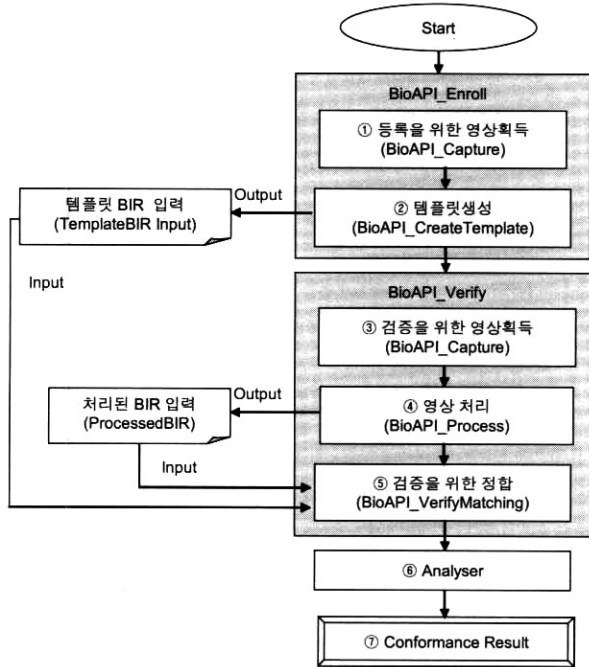
BIR 생성을 위해서는 우선 영상을 획득하여야 하기 때문에 핸들기능을 테스트하기 위해 반드시 BioAPI_Capture 함수를 먼저 수행해야 한다. BioAPI_FreeBIRHandle, BioAPI_GetBIRFromHandle, BioAPI_GetHeaderFromHandle 함수 중 어느 하나라도 NULL이거나 모듈핸들을 가져올 수 없다면 핸들기능의 적합성 시험을 할 수 없기 때문에 TEST_UNKNOWN 메시지를 출력한다. 핸들기능 시나리오는 (그림 5)과 같으며 BioAPI_GetHeaderFromHandle 함수를 수행 후에 BIR 구조를 체크하여 BioAPI_OK 메시지가 리턴되면 BioAPI_GetBIRFromHandle 함수를 수행하여 해당하는 BIR를 가져온다. 이때도 마찬가지로 함수 수행 후에 BioAPI_OK 메시지가 리턴되면 BioAPI_FreeBIRHandle 함수를 호출하여 가져온 BIR를 해제한다.

3.2.3 검증(Verification-1 : 1)과정 시나리오

생체인식 기술에서 검증을 하기 위해서는 우선 생체 영상을 획득하여 등록하고 검증을 위한 영상을 다시 획득한 후 기존에 등록된 영상과 비교하여 정합의 여부를 판단하는 일련의 과정이 필요하다. 지문검증시스템의 경우 지문취득과 지문품질예측, 지문품질향상, 지문특징추출, 지문검증과 같은 절차과정을 통하여 개인인증을 수행한다. 하지만 지문품질예측이나 품질향상, 지문특징추출과 같은 경우에 다른 생체인식기술과 그 방법이 다를 수 있다. 예를 들어 지문의 특징추출과 얼굴의 특징추출 방법은 많은 차이가 있다.

때문에 BioAPI는 다양한 생체인식기술의 공통된 최소 기능만을 표준화하였다. 즉, 검증과정은 BioAPI_Enroll 함수 호출 후 BioAPI_Verify 함수가 호출되는지의 대한 테스트가 필요하다. 등록은 BioAPI_Capture 함수를 사용하여 등록을 위한 영상을 획득하고 이 영상을 BioAPI_CreateTemplate 함수를 사용하여 템플릿 BIR로 생성하여 저장한다. 검증은 BioAPI_Capture 함수를 사용하여 검증을 위한 영상을 획득

한 후 이 영상을 BioAPI_Process 함수를 사용하여 영상처리를 수행한다. 처리된 영상은 처리된 BIR로 생성된다. 그리고 BioAPI_VerifyMatch 함수를 사용하여 템플릿 BIR과 처리된 BIR를 비교하여 정합한 후 그 결과를 리턴한다.



(그림 6) 생체인식시스템 검증시나리오 절차과정

4. 제안된 BioAPI 적합성시험을 위한 설계 및 구현

4.1 적합성 시험항목 및 절차 도출

적합성 시험에 필요한 기본적인 사항과 함께 GUI의 각 메뉴로부터 그룹 및 단일 선택이 가능하도록 제작한다. 일반적으로 개발자들은 상징적인 모델을 검사하고 분석하기 위하여 표준 형식으로 정의된 내용 기준으로 하여 접근을 시도한다[9-11]. 함수들은 개별적으로 작동 가능한 것과 그룹으로 작동해야 하는 것으로 나누어 가능한 한 최저레벨에서 테스트가 가능하도록 구성시킨다[12]. BioAPI 모듈 검사 중 서비스 제공자의 결과물은 BioAPI 프레임워크 기반의 애플리케이션으로 만들어져 있다고 가정하며 모든 정보는 프레임워크를 통하여 가져온다. 시험 절차는 아래와 같은 순서로 이루어지며, 사용되어지고 있는 함수의 정확한 적용 및 해당 함수의 정확한 운영에 대한 시험을 동시에 수행한다.

- 설치된 BSP 모듈검사 및 리스트 작성
- 선택된 BSP 모듈의 메모리 로드 점검
- BSP와 프레임워크와의 연결 상태 점검
- 선택된 BSP 모듈의 메모리 해제상황 점검
- BSP와 프레임워크와의 연결해제 점검
- 테스트를 하기 위한 함수목록과 시나리오를 GUI로부터

입력

- 선택된 함수들에 대한 적절한 파라미터 지정
- 지정된 값에 의한 테스트 수행
- 테스트 결과를 출력

외부로 보이는 선택항목 이외에도 적합성 시험에 필요한 기본적인 사항은 자동적으로 수행되며, GUI의 각 메뉴로부터 그룹 및 단일 선택이 가능하도록 설계하였다. 또한 함수들을 한 번에 테스트하기 위한 자동 테스트도 가능하다. 함수들 중 몇몇의 함수는 이전에 선행되어야 할 함수가 존재하고, 경우에 따라서는 한 번에 여러 개의 함수가 연속적으로 수행되어야만 실행이 가능할 때도 있다. 이러한 함수의 경우 자동적으로 필요한 선행함수들이 실행되어지며, 선행함수 실행이 불가능 할 경우 해당 선택된 함수 역시 실행이 불가능하다.

4.2 적합성 시험의 기본 설계

BioAPI 표준을 따르는 애플리케이션 프로그램의 작동은 반드시 프레임워크의 MDS 모듈을 거쳐야 한다. 또한, 모든 데이터와 메모리 관리하는 레지스트리(Registry)의 관련된 내용은 이 MDS 모듈에서 처리한다. MDS 모듈의 기본적인 기능은 아래와 같이 4가지 항목들로 나눌 수 있다.

4.2.1 BSP 함수들의 등록/해제 관리

BSP가 설치될 경우 해당되는 BSP의 기본정보를 레지스트리에 등록하고 리스트를 관리한다. 만약 BSP가 제거될 경우 해당되는 정보도 동시에 제거한다.

4.2.2 레지스트리 개체 관리

BSP의 등록과 데이터 구조에 대한 레지스트리 부분에 대한 전반적인 관리를 수행한다. 레지스트리 개체는 장치가 부착(Attach)이나 탈착(Detach)될 때 생성/삭제된다. 이것은 해당 정보의 저장, 호출, 삭제, 검색 등이 포함된다.

4.2.3 각종 데이터 스키마 관리

BSP 정보를 포함한 각종 데이터들에 대해서 저장, 분류 등의 전반적인 관리 작업을 수행한다.

4.2.4 기타 BSP 정보들의 관리

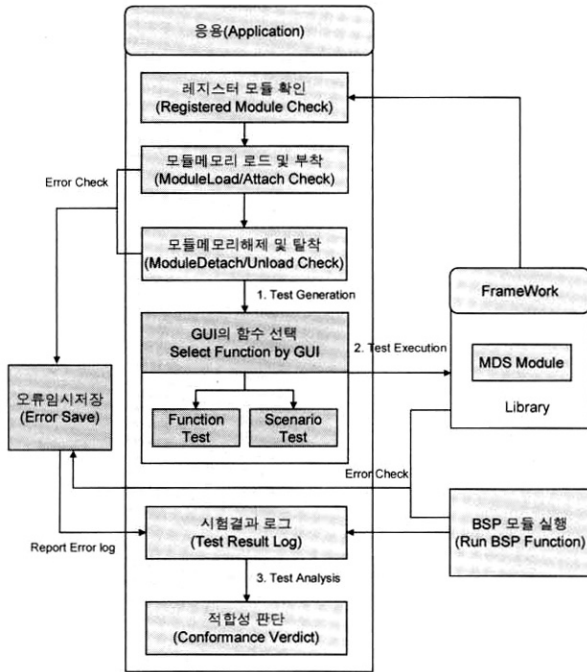
레지스트리의 BSP에 관련된 항목들이 구성되어 있으며 이 항목들에 대한 관리를 수행한다.

실제 BSP 함수를 받아 사용하는 애플리케이션 프로그램은 프레임워크에서 제공하는 함수를 통해 다음과 같은 일을 수행한다.

- ① 기본적인 버전정보 및 초기화
- ② MDS를 통해 설치된 BSP의 확인
- ③ MDS를 통해 설치된 BSP를 로드
- ④ API 함수를 통해 BSP함수를 실행

4.3 적합성 시험기술의 수행 절차

적합성 검사를 수행하는 프로그램은 다음 (그림 7)과 같은 절차를 따르며, 각 절차는 아래와 같이 설명된다.



(그림 7) BioAPI 표준적합성 시험 도구 설계

4.3.1 등록 모듈 검사

MDS로부터 정보를 받아 미리 설치된 BSP 모듈에 대한 정보를 얻어 온 후 설치된 모든 BSP 모듈에 대한 리스트를 작성하고 이 리스트를 화면에 출력하여 테스트할 BSP 모듈을 선택 가능하도록 구성한다.

4.3.2 모듈 로드 검사

사용하고자 하는 BSP 모듈이 선택되었을 경우 해당 BSP를 위한 메모리 공간을 할당하고 로드 하여 실행 준비상태로 만든다. 만약 이것이 실패하였을 경우 BSP전체는 실행 불가능하게 되므로 프로그램은 종료되어야 한다.

4.3.3 모듈 부착상태 검사

BSP 모듈들은 기본적인 함수들에 대해 자체적으로 독립적인 정의를 하여 실제 DLL 외부로 가져온 내용들은 표준과 맞지 않게 되어있다. 모듈부착은 이러한 사항에 대해 각각의 함수명을 표준에 맞추도록 연결하는 작업을 하여 사용자가 표준함수를 호출했을 경우 원하는 함수가 실행되도록 한다.

4.3.4 모듈 해제 검사

한번 로드 된 BSP 모듈은 그 사용이 끝났을 경우 사용된 메모리를 반환하여야 한다. 만약 메모리 반환이 제대로 이루어지지 않을 경우 여러 개의 BSP를 로드하고 해제하는 프로그램에서 심각한 메모리 낭비 현상이 발생할 수 있다.

4.3.5 모듈 탈착상태 검사

모듈부착에서 연결된 일련의 표준함수들은 다음 BSP를 위하여 그 연결을 해제하여야 한다. 만약 이것이 수행되지 않을 경우 실제 사용되는 BSP의 함수명이 이전에 사용되었던 BSP함수명과 틀리므로 전체기능이 작동하지 않게 된다.

4.3.6 GUI의 함수 선택

각각의 함수들은 하나씩 선택하여 시험이 가능하며 기본 함수테스트는 모든 함수에 대한 적합성 시험이 자동으로 수행하도록 구현하였다. 시나리오테스트도 기능별로 선택하여 테스트 할 수 있으며 이 또한 모든 테스트케이스를 자동으로 수행 할 수 있도록 구성하였다.

4.3.7 BSP 함수 실행

GUI로부터 선택된 개별, 전체 함수들에 대해 지정된 일련의 절차에 의해 테스트를 수행한다.

4.3.8 오류 저장

테스트 된 각각의 결과는 오류 저장소에 저장된다.

4.3.9 시험 결과 로그

임시저장소에 저장된 에러결과와 테스트 결과를 모든 테스트가 종료될 경우 별도의 창에 출력하게 된다.

4.3.10 적합성 판단

테스트 결과 메시지를 분석하여 BioAPI 표준적합성에 준용하는지를 판단하고 개선사항을 도출한다.

프레임워크의 라이브러리를 이용한 테스트에 있어서 최초 모듈의 메모리로드 및 부착으로부터 시작하여 각 함수의 적절한 파라미터를 설정하고 API 레벨에서 함수를 호출한 후 해당 함수의 리턴값을 이용하여 결과를 판단한다. 에러들 중 BioAPI_ERRCODE_INCOMPATIBLE_VERSION에 대해서는 최초 프레임워크와의 접속 시에 체크하게 되어 이 에러가 나타날 경우 적합성 시험은 수행될 수 없다.

본 적합성 시험도구는 BioAPI 컨소시엄에서 제공하는 BioAPI 프레임워크가 설치되어야 있어야 한다. 개발된 도구의 OS는 Microsoft Win95/98/2000/XP에서 모두 실행이 가능하며 CPU는 펜티엄 90MHz 이상, HDD는 최소 5MB이상에서 수행이 가능하다.

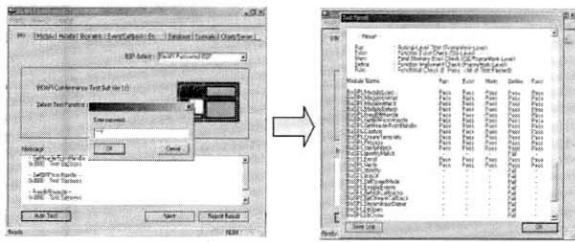
5. 실험결과 및 분석

적합성 검증을 받기 위해서는 우선 BioAPI에서 제공하는 기본함수와 시나리오의 테스트케이스를 모두 만족하여야 한다. 본 연구는 BioAPI 컨소시엄에서 제공하는 샘플시스템과 지문검증시스템의 2개의 BSP를 대상으로 실험하였다. 이것은 적합성 시험이 단순히 1종에 국한된 것이 아니라 범용성을 지닐 수 있음을 제시해 주는 중요한 결과가 된다. 각 BSP가 제공하는 BioAPI함수는 다음 <표 1>과 같다.

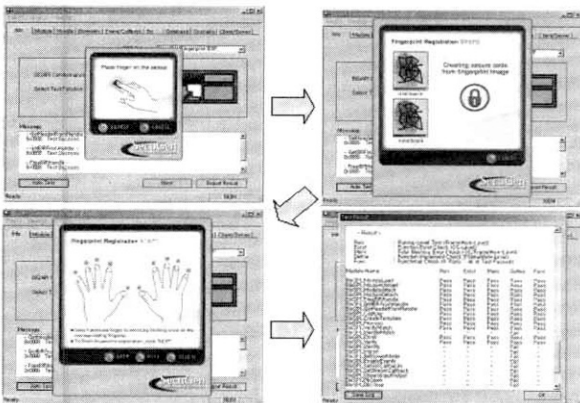
〈표 1〉 3개의 BSP가 지원하는 BioAPI 제공함수

구분	BSP 함수	Password	지문검증 시스템
모듈 관리	BioSPL_ModuleLoad	Yes	Yes
	BioSPL_ModuleAttach	Yes	Yes
	BioSPL_ModuleUnload	Yes	Yes
	BioSPL_ModuleDetach	Yes	Yes
핸들링	BioSPL_FreeBIRHandle	Yes	Yes
	BioSPL_GetBIRFromHandle	Yes	Yes
	BioSPL_GetHeaderFromHandle	Yes	Yes
생체 인식 기능	BioSPL_Capture	Yes	Yes
	BioSPL_CreateTemplate	Yes	Yes
	BioSPL_Process	Yes	Yes
	BioSPL_Enroll	Yes	Yes
	BioSPL_Verify	Yes	Yes
	BioSPL_VerifyMatch	Yes	Yes

다음은 각 BSP의 적합성 시험의 실제 실행과정을 보여준다. (그림 8)은 BSP Select에서 Password를 선택하고 AutoTest를 누르거나 해당 함수를 선택하여 Next를 누르면 BSP가 실행되며 Password의 입력을 받게 된다. 각 함수가 수행되어질 때 마다 중간에 메시지를 확인할 수 있으며 모든 테스트가 종료되면 오른쪽 그림과 같이 종합결과를 보여준다. 이와 마찬가지로 (그림 9)는 지문검증시스템인 Fingerprint BSP를 선택하고 지문영상을 입력 받아 등록하는 검증의 실행과정을 보여준다.



(그림 8) Password BSP 실험 과정



(그림 9) 지문인식시스템의 실험 과정

〈표 2〉 지문검증시스템의 함수별 적합성시험 중간결과

Service Provider Info : SecuGen Fingerprint BSP BSP Name : sgbiobsp.dll Vendor : SecuGen Corporation
- GetHeaderFromHandle - 0x0000 Test Success - GetBIRFromHandle - 0x0000 Test Success - FreeBIRHandle - 0x0000 Test Success - Capture - 0x0000 Test Success - CreateTemplate - 0x0000 Test Success - Process - BIR data type is not intermediate. 0x0000 Test Success - VerifyMatch - 0x0000 Test Success - Enroll - 0x0000 Test Success - Verify - 0x0000 Test Success

〈표 3〉 지문검증시스템의 함수별 최종 종합결과

- Result -					
Run : Runing-Level Test (FrameWork-Level)					
Mem : Fatal Memory Error Check (OS/FrameWork-Level)					
Define : Function Implement Check (BSP/FrameWork-Level)					
Func : Functional Check (If 'Pass' : All of Test Passed)					
Module Name	Run	Exist	Mem	Define	Func
BioSPL_ModuleLoad	Pass	Pass	Pass	Pass	Pass
BioSPL_ModuleAttach	Pass	Pass	Pass	Pass	Pass
BioSPL_ModuleUnload	Pass	Pass	Pass	Pass	Pass
BioSPL_ModuleDetach	Pass	Pass	Pass	Pass	Pass
BioSPL_FreeBIRHandle	Pass	Pass	Pass	Pass	Pass
BioSPL_GetBIRFromHandle	Pass	Pass	Pass	Pass	Pass
BioSPL_GetHeaderFromHandle	Pass	Pass	Pass	Pass	Pass
BioSPL_Capture	Pass	Pass	Pass	Pass	Pass
BioSPL_CreateTemplate	Pass	Pass	Pass	Pass	Pass
BioSPL_Process	Pass	Pass	Pass	Pass	Pass
BioSPL_VerifyMatch	Pass	Pass	Pass	Pass	Pass
BioSPL_Enroll	Pass	Pass	Pass	Pass	Pass
BioSPL_Verify	Pass	Pass	Pass	Pass	Pass

각각의 테스트 결과는 테스트 중에 메시지를 보여주는 중간 결과 과정으로 함수가 존재하고 정상적으로 수행하면 Test_Success의 메시지가 리턴된다. 그러나 함수가 존재하지 않거나 정상적으로 수행되지 않을 경우 "Function Does not implemented"나 "Function Run Failed" 메시지가 리턴된다. 전자의 메시지는 함수가 지원되지 않을 때 발생하며 후자는 해당 함수가 표준규격에 준용하지 않거나 해당 함수

의 실행 모듈이 실행되지 않는 경우에 발생한다. <표 3>은 <표 2>의 중간결과의 내용을 로그에 저장해 두었다가 최종적으로 종합하여 보여주는 결과이다. 항목 중에 Run은 프레임워크 레벨에서 실행이 정상으로 수행되었는지를 체크하고 Exist는 BSP제품에 해당 함수에 대한 존재의 여부를 체크하며 Mem는 메모리에 오류가 없었는가를 체크한다. Define는 BSP에 함수가 정의되어 있는가를 체크하며 마지막 Func은 이 네 가지 항목이 모두 Pass일 경우에만 Pass 메시지를 리턴하고 그렇지 않으면 Fail이 리턴된다.

<표 2>, <표 3>은 (그림 9)의 실행과정을 통하여 얻어진 지문검증시스템의 실험결과이다.

다음은 얼굴검증시스템의 모듈, 핸들, 검증기능에 대한 시나리오 적합성시험을 수행한 결과이다.

<표 4> 지문검증시스템의 시나리오 적합성시험 결과

Operations Test Start
<pre> ** Module Management Start** ----- - ModuleLoadAttach Operation- 1. ModuleLoad Test_Success 2. ModuleAttach Test_Success * Result : Test_Success - ModuleUnloadDetach Operation- 1. ModuleDetach Test_Success 2. ModuleUnload Test_Success * Result : Test_Success ** Handle Operation Start(Test BIR) ** ----- 1. Capture TEST_Success 2. GetHeaderFromHandle Test_Success 3. GetBIRFromHandle Test_Success 4. FreeBIRHandle Test_Success *Result : Test_Success ** Verification Operation Start ** ----- * Enroll_Verify ----- 1. Enroll Test_Success 2. Verify Test_Success * Result : Test_Success * Verification ----- 1. Capture for CreateTemplate Test Success 2. CreateTemplate Test Success 3. Capture for Process Test_Success (BIR data type is not intermediate) 4. Process Test_Success 5. VerifyMatch Test Success * Result : Test_Success </pre>

표와 같이 각 업체에서 제공한 BSP 제공함수의 적합성시

험을 수행하였을 때 샘플시스템과 지문검증시스템의 BSP가 <표 1>의 함수와 결과가 일치하였다. 또한, BioAPI을 기반을 둔 모듈관리, 핸들기능, 검증기능이 시험 항목 및 시나리오의 테스트케이스에 실험결과 Test_Success가 모두 제공되어 BioAPI 표준의 요구사항에 만족함을 확인 할 수 있었다.

6. 결론 및 향후계획

컨소시엄에서 제공하는 프레임워크를 기반으로 생체인식 기술을 개발하고, 제품 개발 초기에 적합성 시험을 사용하여 표준규격의 준용여부를 검사한다면, 개발시간과 비용에서의 절감 효과뿐만 아니라 표준 규격을 준용한 제품이라는 신뢰성을 주게 된다. 또한, 적합성 시험을 통하여 오류요소를 발견하고 이를 극복한다면 개발된 제품은 질적인 향상을 가져올 수 있게 된다.

따라서 본 연구에서는 BioAPI 기반으로 개발된 생체인식 시스템이 BioAPI 규격에 준용하여 개발하였는지에 대한 적합성 시험 방법을 연구하고 개선하였다. 적합성 시험은 각 단계별로 두 가지에 초점을 두어 개발하였다. 첫 번째는 표준명세서의 기본함수를 체크하여 그 사용여부를 평가하는 것이고 두 번째는 BSP 지원하는 기능을 기반으로 해당 테스트케이스를 분석하여 모듈관리, 핸들기능, 검증기능 대한 시나리오를 평가하는 것이다.

실험은 BioAPI 모듈(등록, 검증)에 관한 함수 및 시나리오를 확인하기 위해 컨소시엄에서 제공한 샘플프로그램과 지문검증시스템 BSP의 제품을 사용하였다. 각 구현결과는 기본함수 테스트 실행 중에 보여주는 중간결과와 모든 기본함수의 테스트가 종료된 후에 보여주는 최종 종합 결과, 그리고 마지막으로 각 테스트케이스를 검증한 시나리오의 결과로 분류하여 도출하였다. 이 결과에 따라 각 BSP 모두 BioAPI에 적합한 함수와 시나리오를 사용하여 개발되었음을 검증할 수 있었다. 또한, 두 가지 종류의 BSP의 테스트로 제안된 BioAPI의 적합성 시험도구의 범용성을 증명하였다. 따라서 향후 관련 제품의 BioAPI 적합성 검증 시에도 유용하게 사용될 수 있을 것으로 기대된다.

그러나 위 결과는 시장에 출시된 BioAPI BSP에 대하여 좀 더 다양한 제품에 대한 적합성 시험이 필요할 것이다. 또한, 국·내외적으로 BioAPI을 기반으로 하는 BSP의 식별기능, 분산모듈, 데이터베이스 함수 등은 현재 개발 중에 있다. 각 기능이 제공되는 제품이 향후 개발된다면 관련 함수 및 시나리오에 대한 적합성 시험의 추가 구현이 요구될 것이므로 이에 대한 도구의 확장이 필요하다고 판단된다. 마지막으로 향후 분산 환경의 BSP일 경우에는 데이터의 암호화도 중요한 이슈가 될 것이다. 그러므로 페이로드로서 DES(Data Encryption Standard)를 사용하였을 경우에 각 필수함수 및 시나리오 구현방법에 대한 연구도 필요하다.

참 고 문 헌

[1] L. C. Jain, U. Halici, I. Hayashi, S. B. Lee, S. Tsutsui, "Intelligent Biometric Techniques in Fingerprint and Face Recognition," *CRC Press LLC*, pp.3-21, 1999.

[2] BioAPI Consortium, *BioAPI Specification Version 1.1*, Mar., 2001, <http://www.bioapi.org>.

[3] <http://www.itl.nist.gov/div898/pubs/ar/ar1999/node8.html>.

[4] Martha Gray, Alan Goldfine, Lynne Rosenthal, Lisa Carnahan, "Conformance Testing, National Institute of Standards and Technology," 2000, <http://www.oasis-open.org/cover/conform20000112.html>.

[5] Sharath Pankanti, Ruud M. Bolle and Anil Jain, "Biometric : The Future of Identification," *IEEE Computer*, pp.46-49, Feb., 2000.

[6] 이종필, 최 훈, 윤대균, "SyncML 적합성 상호 연동성 시험연구", *한국정보과학회 추계학술회 논문집*, pp.724-726, 2001.

[7] "ebXML Messaging (2.0) Conformance Test Suite Version 0.3," OASIS eb XML Implementation, Interoperability and Conformance Technical Committee, July, 2002.

[8] Xiang, Shi, Guo Xin Xiang, Yuanchun Shi, "A Conformance Test Suite of Localized LOM Model," 2002.

[9] Paul Ammann and Paul E. Black, *Abstracting Formal Specifications to Generate Software Tests via Model Checking*, Proceedings of the 18th Digital Avionics Systems Conference (DASC), St. Louis, Missouri, IEEE, Vol.2, Section 10.A.6, pp.1-10, October, 1999. Extended version is NIST-IR 6405.

[10] Paul E. Black, *Modeling and Marshaling : Making Tests from Model Checker Counter examples*, 19th Digital Avionics Systems Conference (DASC), Philadelphia, Pennsylvania, IEEE, Section 1.B.3, pp.1-6, October, 2000.

[11] Paul E. Black, George Becker and Neil V. Murray, *Formal Verification of a Merge-Sort Program with Static Semantics*, Proceedings of The 9th International Conference on Computing and Information (ICCI '98), Winnipeg, Manitoba, Canada, edited by Kamal Karlapalem, Amin Y. Noaman, and Ken Barker, pp.271-277, June, 1998.

[12] <http://www.itl.nist.gov/div898/pubs/ar/ar1998/node10.html>.

[13] Chatherine J. Tilton, "An Emerging Biometric API Industry Standard", *IEEE Computer*, pp.130-132, Feb., 2000.

[14] Wilson, John and Tilton, Catherine, *BioAPI Architecture*,

The BioAPI Users and Developers Seminar, 6 April, 2000, Pentagon City, VA and at CardTech/SecureTech, 7, May, 2000, Miami, FL, <http://www.itl.nist.gov/div895/isis/bio-api/index.htm>.

[15] Catherine J. Tilton, BioAPI, "A Prescription for Biometric Interoperability In an Open System Environment," *SAF-LINK*, 2001.

[16] "CDSA/CSSM Authentication : Human Recognition Service (HRS) API Ver 2.0," June, 2001, <http://www.intel.com/ial/security> and <http://www.opengroup.org/publications/catalog/c909.htm>.

[17] "Human Authentication-Application Program Interface (HA-API) Ver 2.0," April, 1998, <http://www.biometrics.org/html/standards.html>.

[18] I/O Software Inc., "Biometric API(BAPI) Device Module Interface Specification(BDMI) Ver 1.1," June, 1998, www.iosoftware.com.

[19] NIST Publication, *A Common Biometric Exchange File Format (CBEFF) For Biometric Interoperability*, NISTIR 6529, Jan., 2001, <http://www.nist.gov/cbeff>.



이 유 영

e-mail : yylee@cviab.cau.ac.kr
 2002년 건양대학교 정보전산학과(공학사)
 2004년 중앙대학교 컴퓨터공학과(공학석사)
 2004년~현재 중앙대학교 컴퓨터공학과 박사과정
 관심분야 : 생체인식, 패턴인식, 정보표준화



권 영 빈

e-mail : ybkwon@cau.ac.kr
 1978년 아주대학교 전자공학과 전교수석 (공학사)
 1981년 한국과학기술원(공학석사)
 1986년 프랑스 파리 ENST(공학박사)
 1986년~현재 중앙대학교 공과대학 컴퓨터 공학과 교수

1999년~현재 컴퓨터연구정보센터 소장(한국과학재단 지정)
 2002년~현재 국제표준화기구 ISO/JTC1/SC37 Convenor
 2003년~현재 중앙대학교 정보대학원장
 2004년~현재 한국정보처리학회 부회장
 관심분야 : 생체인식, 문서인식, 패턴인식, 정보표준화