

## Development of a Visitor Recognition System Using Open APIs for Face Recognition

Kisu Ok<sup>†</sup> · Dongwoo Kwon<sup>\*\*</sup> · Hyeonwoo Kim<sup>\*\*\*</sup> · Donghyeok An<sup>\*\*\*\*</sup> · Hongtaek Ju<sup>\*\*\*\*\*</sup>

### ABSTRACT

Recently, as the interest rate and necessity for security is growing, the demands for a visitor recognition system are being increased. In order to recognize a visitor in visitor recognition systems, the various biometric methods are used. In this paper, we propose a visitor recognition system based on face recognition. The visitor recognition system improves the face recognition performance by integrating several open APIs as a single algorithm and by performing the ensemble of the recognition results. For the performance evaluation, we collected the face data for about five months and measured the performance of the visitor recognition system. As the results of the performance measurement, the visitor recognition system shows a higher face recognition rate than using a single face recognition API, meeting the requirements on performance.

**Keywords :** Open API Integration, Face Recognition, Visitor Recognition System

## 얼굴 인식 Open API를 활용한 출입자 인식 시스템 개발

옥기수<sup>†</sup> · 권동우<sup>\*\*</sup> · 김현우<sup>\*\*\*</sup> · 안동혁<sup>\*\*\*\*</sup> · 주홍택<sup>\*\*\*\*\*</sup>

### 요약

최근 보안에 대한 관심과 필요성이 증가하면서 출입자 인식 시스템의 수요가 증대되고 있다. 출입자 인식 시스템은 출입자를 인식하기 위해서 다양한 생체인식 방법을 사용하고 있다. 본 논문에서는 다양한 특성과 강점을 가진 다수의 얼굴인식 Open API 서비스를 통합하고, 그 인식 결과를 앙상블 함으로써 인식 성능을 개선하는 얼굴인식 기반 출입자 인식 시스템을 제안한다. 또한 다양한 얼굴 인식 Open API 서비스를 앙상블 하는 출입자 인식 시스템의 구조를 제안한다. 성능 측정은 약 5개월 간 수집한 얼굴 데이터를 이용하여 수행하였으며, 측정결과로 본 논문에서 제안하는 출입자 인식 시스템이 단일 얼굴인식 Open API 서비스를 사용했을 때보다 더 높은 얼굴인식률을 보임을 확인하였다.

**키워드 :** Open API 통합, 얼굴 인식, 출입자 인식 시스템

### 1. 서론

예전부터 국가기관, 기업, 주택 등에서 기밀유지 및 안전을 위해 다양한 형태의 보안 시스템을 활용해 왔다. 출입자 관리 시스템은 대표적인 보안 시스템이며, 출입자 인식 시스템이

출입자 관리 시스템의 핵심 요소이다. 초기 출입자 인식 시스템은 인력을 사용하는 수동 시스템이었으나, IT 기술의 발전에 따라 ID 카드를 사용하는 전자동 시스템 형태로 발전하였다. ID 카드 기반의 출입자 인식 시스템은 ID 카드의 위변조에 취약하기 때문에 이를 방지하기 위해서 생체 인식 기반의 출입자 인식 시스템 개발 수요가 높아지고 있다[1].

생체 인식을 위해서 지문, 얼굴, 홍채, 망막, 성문, 필기, 정맥 등의 사용이 가능하지만, 개인 정보가 타인에게 제공된다는 점에서 거부감도 높은 편이다. 언급한 생체 인식 요소 중 얼굴 인식 기반의 출입자 인식 시스템은 비접촉식 방법으로 카메라만 응시하면 되기 때문에 사용자들의 거부감이 적고, 모니터링 등의 관리가 용이한 편이며 카메라의 경제적인 비용이 적다는 장점이 존재한다. 이로 인해, 얼굴 인식 기반의 출입자 인식 시스템의 수요가 증가하고 있으며,

\* 이 논문은 2017년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구이며(2015-0-00274, ICBMS 플랫폼 간 정보 모델 연동 및 서비스 매쉬업을 위한 스마트 중계 기술 개발), 2015년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2015R1D1A1A01059786).

† 준회원: KETI 에너지IT 연구원

\*\* 비회원: 계명대학교 컴퓨터공학과 박사과정

\*\*\* 준회원: 계명대학교 컴퓨터공학과 박사과정

\*\*\*\* 중신회원: 계명대학교 컴퓨터공학부 조교수

\*\*\*\*\* 중신회원: 계명대학교 컴퓨터공학부 교수

Manuscript Received: August 12, 2016

First Revision: November 23, 2016

Accepted: January 24, 2017

\* Corresponding Author: Hongtaek Ju(juht@kmu.ac.kr)

국외에서는 Inttelix [2], TCIT [3], DigiFace [4], 국내에서는 퍼스텍 [5], VS코리아 [6] 등의 업체에서 얼굴 인식을 이용한 출입자 인식 시스템을 공급하고 있는 실정이다.

기존의 얼굴 인식 기반 출입자 인식 시스템은 자체적으로 얼굴 인식 모듈을 개발하여 출입자를 인식하는 방법을 주로 사용하고 있다[5, 6]. 하지만 기존의 방법은 제한된 수의 얼굴 인식 모델을 사용한다는 단점이 존재하기 때문에 다양한 환경에서 일정 수준의 인식률을 유지하기 위한 추가적인 개발 비용이 수반된다.

이를 해결하기 위해서 본 논문에서는 얼굴 인식 Open API 기반의 출입자 인식 시스템을 개발하고자 한다. 최근 들어 통신 기술의 발전과 더불어 오픈소스 소프트웨어의 개발 및 배포가 증가하면서 다양한 종류의 소프트웨어가 Open API 형태로 자유로운 사용이 가능하다. 다양한 얼굴 인식 모델 기반의 Open API들 중 상황에 맞는 얼굴 인식 Open API를 선택해 출입자 인식 시스템을 기존보다 빠른 시일 내에 개발할 수 있고, 이로 인해 개발 비용의 감소가 가능하다. 대표적인 얼굴 인식 Open API 제공 업체로는 Lambda [7], Betaface [8], KAIROS [9], Face++ [10] 등이 있다.

단일 얼굴 인식 Open API를 사용해 출입자 인식 시스템을 개발할 경우 기존과 마찬가지로 특정 환경에서는 일정 수준의 인식률을 유지할 수 있지만, 환경이 변할 경우 인식률이 저하될 수 있다. 이를 해결하기 위해서 본 논문에서는 사진 특성에 대한 인식의 강점들이 서로 다른 얼굴 인식 Open API들을 활용해 얼굴 인식률을 높이는 앙상블 기법을 활용한다. 서로 다른 얼굴 인식 모델을 사용하는 Open API들을 함께 사용함으로써 특정 상황에서 인식률이 저하되는 단점을 방지하면서 전체적인 인식률을 향상시키는 방법이다.

본 논문에서는 얼굴 인식 Open API를 기반으로 출입자 인식 시스템을 개발하고자 한다. 제안하는 출입자 인식 시스템은 크게 네 부분으로 나누어진다. Open API들과 통신하고 결과를 앙상블하는 Open API Adapter, 사용자들에게 사용 환경을 제공하는 User Interface, 사용자로부터 받은 명령을 해석하는 Client Adapter, Client Adapter에서 Open API Adapter로 사용자의 입력 및 인식 결과를 전달하는 Message Router이다.

본 논문에서는 제안하는 출입자 인식 시스템의 실현 가능성과 성능 평가를 위해서 제안하는 출입자 인식 시스템을 구현하고 성능을 측정하였다. 개발된 출입자 인식 시스템은 여러 얼굴 인식 Open API들을 통합하여 마치 하나의 얼굴 인식 Open API처럼 사용할 수 있다. 추가적으로 카메라와 도어락을 출입자 인식 시스템과 연동하여 출입자 관리 시스템으로 확장 개발할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 얼굴 인식을 이용한 출입자 인식 시스템, 인식 결과를 앙상블 하는 방법, 얼굴 인식 Open API에 대한 관련 연구를 설명한다. 3장에서는 얼굴 인식 출입자 인식 시스템의 설계에 대해서 다루고, 4장에서는 출입자 인식 시스템을 구현하고 성능을 평가한다. 마지막으로 5장에서는 결론 및 향후 연구에 대해서 논의한다.

## 2. 관련 연구

### 2.1 얼굴인식 요인

일반적으로 얼굴인식은 인간이 일상생활을 영위하는 중에서도 얼굴을 촬영하고 인식이 이루어지게 된다. 그렇기 때문에 동일한 사람을 인식하더라도 일상의 상황에 따라 인식결과가 변할 수 있다는 것을 의미한다[11]. 인식 결과가 변할 수 있는 상황의 요인들은 무수히 존재하고, 그 상황의 요인이 인식결과에 미치는 영향을 줄이고자 하는 다양한 연구들이 있다.

Pagano 등 [12]은 인식 결과에 영향을 미치는 요인을 random noise, gradual changes, abrupt changes, recurring contexts 네 가지로 정의하였다. 각 요인을 분석하여 인식결과에 미치는 영향을 줄이는 연구를 하였다. Wen 등 [13]은 인식 결과에 영향을 미치는 요인 중 하나를 body weight로 정의하고, body weight의 변화에 따라 얼굴인식의 결과가 변화함을 보였다. Body weight는 small weight changes와 large weight changes로 나뉘고 large weight changes의 경우 최대 26.7%만큼 인식결과에 영향을 끼치는 것을 확인하였다.

이와 같이, 얼굴인식이 출입자 인식 시스템에 적용되면, 출입자 인식 시스템에서도 출입의 상황에 따라 인식 결과가 다양하게 변할 수 있다. 본 논문에서도 변화하는 출입 상황에 따라 얼굴 인식 결과에 영향을 미치는 요인들을 최소화하여 출입자 인식 시스템 성능을 개선하는 방법에 대해서 논의한다.

### 2.2 출입자 인식 시스템

본 장에서는 얼굴 인식을 활용하여 특정한 공간에 출입하는 인원을 감시하는 출입자 인식 시스템에 관한 연구들을 소개한다. Kim 등 [14]은 얼굴 인식 및 소리 위치 추정 기술을 이용한 스마트 도어락 시스템을 제안하였다. 소리 위치 추정 기술은 방문자 얼굴의 대략적인 위치를 찾아 카메라가 방문자의 얼굴 방향으로 촬영할 수 있도록 카메라 촬영 방향 및 각도를 조정하기 위해 사용된다. 스마트 도어락 시스템은 촬영된 얼굴에서 방문자의 신원을 인식하고 그 결과를 거주자에게 전송한다.

Lucas 등 [15]은 주택의 방문자를 인식하여 초인종 역할을 수행하는 스마트 모델을 제안하였다. 얼굴 인식을 통해 방문자를 인식하고 스마트폰으로 알림을 전달한다. 얼굴 인식은 OpenCV 라이브러리를 이용하여 만든 얼굴 인식 알고리즘을 사용하였고, 캠브리지 대학의 The ORL Database of Faces 실험실에서 작성한 얼굴 데이터 셋인 AT&T database of faces에 저장된 얼굴들을 이용하여 실험을 수행하였다. 제안하는 모델 성능은 총 3번 출입자 인식 실험을 통해 입증하였고 성능 측정 결과 최대 90%의 인식률을 얻었다.

권기현 등 [16]은 스마트폰에서 얼굴 인식을 이용하여 출

입자 관리 시스템을 제안했다. 스마트폰에 얼굴 인식 시스템을 구현하였고 얼굴 인식 성능은 PCA (Principle Component Analysis) 방식과 LDA (Linear Discriminant Analysis) 방식을 각각 적용하여 측정하였다. 성능 측정 결과 PCA 방식은 74.5%, LDA 방식은 87.15%의 인식률을 얻었다.

Table 1은 위에서 언급한 관련 연구에서 사용한 얼굴 인식 기술과 성능 측정 결과를 정리한 것이다. 기존의 출입자 인식 시스템에서는 단일 얼굴 인식 기술만 이용하는 출입자 인식 시스템을 제안한다. 하지만 본 논문에서 제안하는 출입자 인식 시스템은 다양한 특성과 강점을 가진 여러 얼굴 인식 Open API를 활용한다는 점에서 차별성이 존재한다.

Table 1. Recognition Performance of Visitor Recognition Systems

Related Work	Face Recognition Algorithm	Recognition Performance
Kim et al. [11]	LBP conversion, AdaBoost	95%
Lucas et al. [12]	OpenCV	90%
Kwon et al. [13]	PCA(Eigenface), LDA(Fisherface)	PCA 74.5%, LDA 87.15%

### 2.3 앙상블 기법

본 논문에서 제안하는 출입자 인식 시스템에서 사용하는 앙상블 기법은 다양한 특성과 강점을 가진 다중 얼굴 인식 Open API들을 사용해 얼굴 인식률을 개선하는 방법이다. 앙상블 기법은 통계 및 기계학습에서 주로 사용되고 있으며, 더 나은 예측 성능을 얻기 위해서 여러 학습 알고리즘을 조합하여 학습하는 방법이다[17]. 대표적인 앙상블 학습법으로는 Random Forest [18], AdaBoost [19]가 있다.

Tang 등 [20]은 다양한 얼굴 인식 알고리즘을 앙상블하여 얼굴 인식 성능을 개선하는 FRCM (Face Recognition Committee Machine)을 제안하였다. FRCM의 기본 개념은 Committee Machine에 기초한다. Committee Machine은 다양한 Experts의 결과를 결합하여 하나의 결과를 얻는 것을 의미하고, static structure, dynamic structure 두 가지 구조로 구분할 수 있다. Static structure는 일반적인 앙상블 기법으로 알려진 방식과 동일한 방법을 사용한다. Static structure 기법은 데이터를 Experts에 입력하고 그 출력된 값을 가공하지 않고 바로 사용한다. Dynamic structure는 입력된 데이터들에 직접적으로 관여하는 방법으로, 입력 데이터에 따라 Experts의 가중치를 다르게 한다. FRCM은 Committee Machine의 static structure를 사용하고, Committee Machine에서 Experts는 PCA, LDA, EGM, SVM, Neurel network 5개의 얼굴 인식 알고리즘을 이용한다. FRCM은 각 알고리즘을 이용하여 얼굴 인식을 실시하고 투표를 통해 가장 유사한 얼굴을 인식해내는 시스템이다. 성능 평가는 캠브리지 대학의 얼굴 데이터 셋인 AT&T database of faces와

UCSD Computer Vision에서 제공하는 얼굴 데이터 셋인 Yale face database를 이용하여 측정하였고, 각각 97.8%, 98%의 얼굴 인식률을 보였다.

제안하는 출입자 인식 시스템은 Committee Machine의 static structure의 구조를 사용한다. 하지만 본 논문에서는 Committee Machine의 Experts를 얼굴인식 알고리즘으로 사용하는 것이 아니라 얼굴 인식 Open API를 대상으로 앙상블 기법을 적용하였고 구현했다는 점에서 기존 연구와 차별된다.

### 2.4 얼굴 인식 Open API

얼굴 인식 Open API를 활용한 출입자 관리 시스템을 설계하고 구현하기 위해서 사용 가능한 얼굴 인식 Open API에 대해서 분석하였다. 본 논문에서 분석한 얼굴 인식 Open API는 BetaFace, Lambda, KAIROS이다.

Fig. 1은 BetaFace, Lambda, KAIROS의 얼굴 인식 Open API들이 공통적으로 사용하는 처리 과정을 나타내고 있다. 얼굴 인식 Open API는 두 가지 과정으로 실행된다. 첫 번째는 얼굴 등록 과정으로 사용자가 업로드 한 사진에서 얼굴을 추출하여 저장소에 저장한다. 저장소에 저장된 얼굴은 얼굴 인식을 위한 데이터 셋이 된다. 두 번째는 얼굴 인식 과정이다. 얼굴 인식 과정은 얼굴을 탐지하고, 얼굴 인식 Open API에 기존에 등록해둔 얼굴들과 비교하여 가장 유사한 얼굴을 찾는다. 얼굴 인식 Open API들은 얼굴들의 유사한 정도를 확률로 평가하고 서비스 사용자에게 얼굴 인식의 확률 값과 확률이 높은 얼굴의 식별자를 반환한다.

각 서비스는 공통적으로 얼굴 등록과 얼굴 인식에 대한 API들을 제공하고 있다. Lambda의 앨범생성, 얼굴등록, 앨범활성화와 Betaface의 사진업로드, 얼굴정보확인, 얼굴등록과 KAIROS의 얼굴 등록 API이다. Lambda의 얼굴 인식과 Betaface의 얼굴 인식, 얼굴 인식결과확인 그리고 KAIROS의 인식은 얼굴 인식 과정에 포함되는 API들이다.

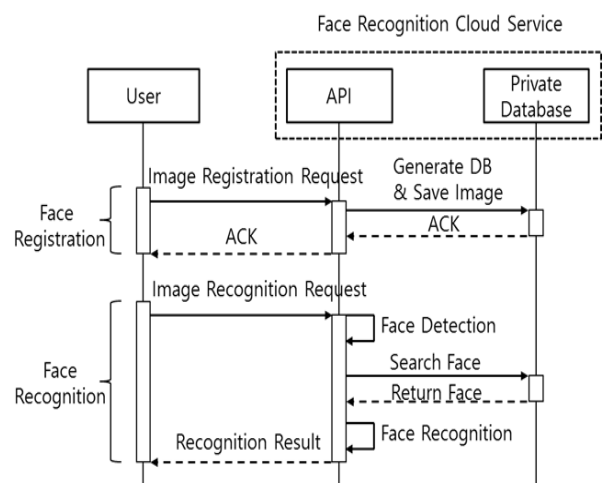


Fig. 1. Service Flow of Face Recognition Open APIs

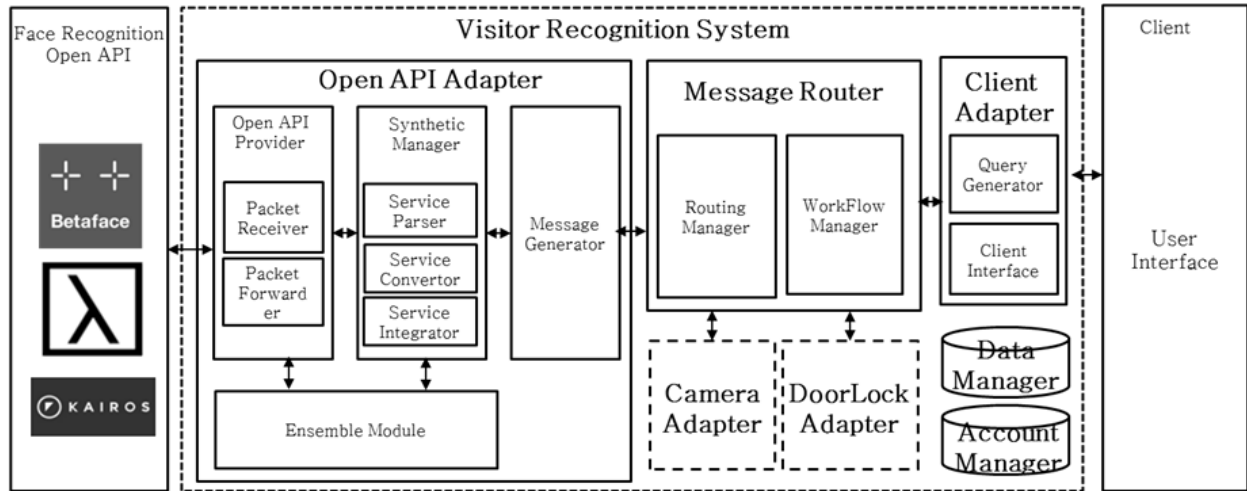


Fig. 2. Architecture of the Proposed Visitor Recognition System

### 3. 출입자 인식 시스템

#### 3.1 출입자 인식 시스템 구조

본 논문에서 제안하는 출입자 인식 시스템은 얼굴 인식을 이용하여 특정 공간에 출입하는 출입자들을 인증 및 감시할 수 있는 시스템이다. Fig. 2는 본 논문에서 제안하는 출입자 인식 시스템의 구조를 나타낸 그림이다. 출입자 인식 시스템은 크게 Client Adapter, Message Router, Open API Adapter와 User Interface로 구성된다.

User Interface는 사용자로부터 명령을 입력받아 Client Adapter 내의 Client Interface에게 전달한다. Client Adapter 내의 Query Generator는 사용자로부터 전달된 명령어를 시스템에 맞게 해석한 후 내부 메시지를 생성한다. 생성된 메시지는 Message Router로 전달되어 메시지의 유효성을 검증한다. 메시지가 유효할 경우에는 Routing Manager를 통해 전달되어야 할 경로를 설정한다. 전달될 경로가 설정된 메시지는 WorkFlow Manager를 통해 해당 Adapter로 전송된다. Message Router는 출입자 인식 시스템에 카메라와 도어락 등이 추가되어 출입자 관리 시스템으로의 확장 시 내부 Adapter 또는 모듈 간의 통신을 담당하기 위해 존재한다.

Open API Adapter는 기존의 Open API들과 통신하고 결과를 이상불하는 역할을 수행하며 네 개의 모듈로 구성되어 있다. 첫 번째 모듈은 Synthetic Manager이다. Synthetic Manager는 Service Parser, Service Convertor, Service Integrator로 구성되어 있다. 얼굴 인식 Open API에서 사용하는 변수는 BetaFace, Lambda, KAIROS에서 각각 다르게 사용되기 때문에 Service Parser에서 변수를 통합하는 역할을 한다. Table 2는 Service Parser에서 합성되는 얼굴 인식 Open API의 통합 변수를 나타낸 것이다.

Service Convertor는 각 얼굴 인식 Open API의 응답으로 반환되는 자원의 표현 형식을 통합한다. 응답 데이터는 XML 또는 JSON과 같은 형태로 표현될 수 있다. Service Integrator

는 얼굴 인식 기본 단계를 통합하는 역할을 담당한다. Table 3은 Service Integrator에서 얼굴 인식의 기본 단계에 따라 합성되는 얼굴 인식 Open API의 인터페이스를 나타낸다.

Open API Adapter의 두 번째 모듈은 Ensemble Module이다. Ensemble Module에서는 얼굴 인식 Open API의 인식 결과를 분석하여 얼굴 인식 기능을 개선하는 역할을 한다. Table 4는 얼굴 인식 Open API의 특징 셋에 대해 나타낸 표이다. Lambda API는 얼굴 인식의 결과 표기방법이 백분율로 표기되고, 등록된 인원 중에 가장 유사한 얼굴을 찾아내기 때문에 비허가자를 판별하기가 어렵다. Betaface는 인식 성능 옵션을 이용하여 얼굴 인식의 성능을 조정할 수 있고, 얼굴 인식 결과가 True/False로 제시되어 비허가자의 인식이 가능하다. KAIROS는 개인 저장소에서 얼굴을 여러 저장소에 중복으로 등록 가능 하여 얼굴을 관리하기가 편리하다. 각 Open API 서비스들의 특징을 고려하여 얼굴 인식 이상불 알고리즘을 구현한다.

Open API Provider는 Lambda, Betaface, KAIROS Open API와 통신을 담당하고 Message Generator는 인식 결과를 User Interface에게 전달하기 위해 내부 메시지를 생성하여 Message Router에게 전달한다.

Table 2. Integration Parameters of Face Recognition Open APIs

Integration Parameter	Face Recognition Open API Parameter
Image	Lambda: files, urls Betaface: image, url KAIROS: url
Image_name	Lambda: entryid Betaface: person_id KAIROS: subject_id
Private_name	Lambda: album Betaface: targets KAIROS: gallery_name



Table 3. Interfaces of Face Recognition Open APIs

Category	Function Name	Parameter Name
Face registration API	Face_Register	image, image_name, private_name
Face recognition API	Face_Recognition	image, private_name
Face recognition test API	Face_OpenAPI_Test	test_image_set, recognition_image_set

Table 4. Feature Set of Face Recognition Open APIs

Open API service	Performance option	Face size option	Personal storage	Result display result	Service method
Lambda	None	None	Unchangeable	Percentage, Percent sum=1	Request-Response
Betaface	Quality control	None	Changeable	Percentage, True/False	Polling
KAIROS	None	8X	Overwriting available	Percentage	Request-Response

3.2 얼굴인식 앙상블 알고리즘

본 장에서는 얼굴 인식률을 향상시키기 위해서 앙상블 기법을 적용한 알고리즘을 설명한다. 앙상블 알고리즘에서는 Open API의 얼굴 인식 결과인 얼굴 인식률을 이용하여 앙상블을 실시한다. Table 5는 출입자 인식 시스템에서 사용하는 앙상블 알고리즘을 표현하고 있다.

얼굴 인식을 할 때 각각의 Open API에서는 얼굴 인식 결과

를 다른 형태로 반환한다. Betaface에서는 얼굴 인식 결과를 참 또는 거짓으로 반환한다. 하지만, Lambda, KAIROS는 얼굴 인식 결과를 참 또는 거짓이 형태가 아닌 확률로 반환한다. 또한 각 Open API는 서로 다른 인식 모델을 사용하기 때문에 동일한 얼굴 사진이라도 서로 다른 결과를 반환할 수 있다. 따라서 각 얼굴 인식 Open API의 인식 결과를 참 또는 거짓으로 판단할 수 있는 기준이 필요하다.

본 논문에서는 이 판단기준을 THR로 정한다. THR는 출입자 인식 시스템을 사용하는 환경에 따라 가변적인 값이다. THR은 출입자 인식 시스템에 실제 사용자들의 얼굴을 등록하고 인식을 실시하여 얻을 수 있다. 이 때 등록자와 비등록자의 인식률이 최대가 되는 값으로 THR이 선택된다. 본 논문에서는 등록자의 인식률을 얻기 위해서 등록자 10명에 대한 50장의 영상을 이용하였고, 비등록자의 인식률을 얻기 위해서 등록되지 않은 인원에 대한 50장의 영상을 사용하였다. 여기서 얼굴 인식률은 정확도를 의미하고, 등록자의 인식률과 비등록자의 인식률이 최대가 될 때의 확률을 THR로 정한다. 측정 결과, Lambda는 30%, Betaface는 60%, KAIROS는 70%의 THR 값을 얻었다. Lambda의 경우에는 비등록자에 대한 얼굴 인식이 불가능하므로 정밀도가 최대가 되는 값을 기준으로 THR를 선택하였다.

Table 5. Ensemble Algorithm of Face Recognition APIs

Algorithm 1. Face Recognition Ensemble Algorithm
1: $L\_ \leftarrow$ Lambda face recognition API
2: $B\_ \leftarrow$ Betaface face recognition API
3: $K\_ \leftarrow$ KAIROS face recognition API
4: $THR \leftarrow$ Standard probability of True
5: $PR \leftarrow$ Face recognition probability result of an image
6: $NR \leftarrow$ Face recognition person_name result
7:
8: IF $L\_PR \geq L\_THR$ AND $B\_PR \geq B\_THR$ AND $K\_PR \geq K\_THR$
9: IF $L\_NR, B\_NR, K\_NR$ two or more equal
10: RETURN NR
11: ELSE
12: RETURN False
13: ELSE
14: IF $L\_PR, B\_PR, K\_PR$ , two higher than THR
15: IF $L\_NR, B\_NR, K\_NR$ three equal
16: RETURN NR
17: ELSE
18: RETURN False
19: ELSE
20: RETURN False

Table 5에서 L은 Lambda 얼굴 인식 API, B는 Betaface 얼굴 인식 API, K는 KAIROS 얼굴 인식 API를 의미한다. THR은 얼굴 인식 결과가 참으로 판단되는 얼굴 인식률을 의미한다. THR 값은 실험을 통해 얻은 값이고, 출입자 인식 시스템의 환경에 따라 변경될 수 있다. PR은 얼굴 인식 결과 가장 유사한 얼굴의 얼굴 인식률을 의미한다. NR은 얼굴 인식 결과 가장 유사한 얼굴의 식별자를 의미한다.

앙상블 알고리즘의 순서는 다음과 같다. 첫 번째는 얼굴 인식 Open API에서 얼굴 인식 결과를 얻고, 그 결과 얼굴 인식률이 각 얼굴 인식 Open API의 THR보다 모두 높은지 확인한다. 모두 높으면 인식된 결과가 동일한 인물인지를



Fig. 3. Strengths in Face Recognition Open APIs

확인하고, 2개 이상의 얼굴 인식 Open API에서 같으면 등록자로 인식한다. 두 번째는 얼굴의 인식 결과 인식률이 각 얼굴 인식 Open API의 THR보다 높은 것이 두 개 이상인지 확인한다. 두 개 이상 높으면 얼굴 인식 결과가 모두 같은 결과를 얻는지 확인하고, 같을 경우에 등록자로 인식한다. 이외의 결과는 모두 비등록자로 판단한다.

Fig. 3은 얼굴 데이터 셋에서 얼굴 인식률이 얼굴 인식 Open API마다 다르게 나왔던 측면 사진, 흔들린 사진, 작은 얼굴에 대한 사진을 나타낸다. 측면 얼굴이 촬영된 사진의 경우에는 Lambda, Betaface에서 높은 인식률을 얻었고, KAIROS에서 낮은 인식률을 얻었다. 흔들린 얼굴이 촬영된 사진의 경우에는 Betaface, KAIROS에서 높은 인식률을 얻었고, Lambda에서 낮은 인식률을 얻었다. 측면, 흔들린, 작은 얼굴은 앙상블 알고리즘에 따라 개선된 얼굴 인식 결과를 얻을 수 있다.

### 3.3 출입자 인식 시스템 실행 과정

본 논문에서 제안하는 출입자 인식 시스템의 시퀀스 다이어그램은 Fig. 4와 같다. 출입자 인식 시스템에서 메시지의 흐름은 얼굴 등록의 메시지 흐름과 얼굴 인식의 메시지 흐름

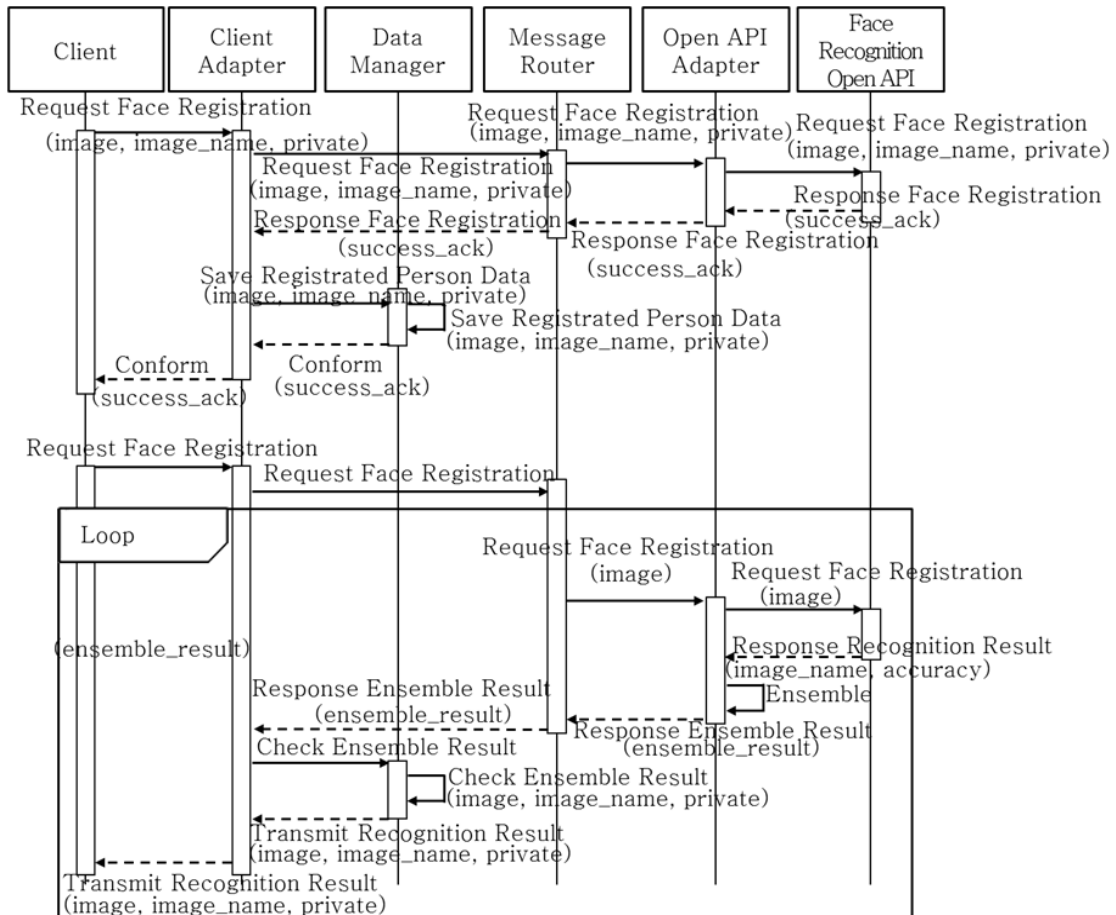


Fig. 4. Sequence Diagram of the Visitor Recognition System

름으로 나뉜다. 얼굴 등록의 메시지 흐름은 사용자가 얼굴 등록 서비스를 출입자 인식 시스템의 Client Adapter에 요청하면서 시작된다. Client Adapter는 얼굴 등록을 위해 클라이언트에게서 전달받은 사람의 이름과 사진을 Message Router에 전달한다. Message Router는 Client Adapter에서 전달받은 얼굴 등록 요청 메시지를 Open API Adapter로 전달한다. Open API Adapter에서는 전달받은 얼굴 등록 메시지에서 등록하려는 사람의 이름과 사진을 Lambda API, Betaface API, KAIROS API를 통해 각각 서비스를 요청한다. Lambda API, Betaface API, KAIROS API는 얼굴 등록 결과를 Open API Adapter에 전송하고, Client Adapter에게 등록 완료 정보가 전달된다. 전송결과를 Data Manager에 저장하고, 클라이언트에게 전달한다.

얼굴 인식의 메시지는 Client Adapter로 얼굴 인식 메시지를 요청하면서 시작된다. 얼굴 인식 요청 메시지는 Message Router로 전달된다. Message Router는 얼굴 인식 요청 메시지와 인식 요청한 사진을 Open API Adapter에 전달한다. Open API Adapter는 얼굴 인식을 위해 각 얼굴 인식 Open API에 사진을 전송한다. 얼굴 인식 Open API는 등록된 얼굴 중에 유사한 얼굴과 얼굴의 유사 정도를 나타내는 인식 확률값을 반환한다. Open API Adapter는 얼굴에 대한 인식 확률값을 이용하여 앙상블을 실시한다. 앙상블 된 인식 결과는 Client Adapter에게 전달되어 User Interface에 결과를 나타낸다. 얼굴 인식의 메시지 흐름은 클라이언트가 정지를 요청하기 전까지 계속 반복된다.

#### 4. 성능 평가

##### 4.1 출입자 인식 시스템의 성능 측정 환경 및 방법

이 절에서는 출입자 인식 시스템의 성능 측정 환경 및 방법에 관해서 설명한다. 먼저 출입자 인식 시스템의 성능을 측정하기 위해서 얼굴데이터 셋을 수집하였다. 출입허가자에 대한 얼굴 데이터 셋은 네트워크 카메라를 출입자 인식 시스템에 연동하고, 2015년 12월 2일부터 2016년 5월 7일까지 약 5개월간 대학 실습실에 출입하는 인원의 얼굴을 촬영하여 수집하였다. 그리고 촬영된 영상에서 얼굴 탐지 이벤트가 발생하여 별도로 저장한 사진 343장을 얼굴인식 Open API 서비스에 적용하였고 이 때 얼굴로 인식이 가능한 사진들을 추출하였다. 여기서 추출된 169장의 사진이 출입허가자 얼굴 데이터 셋으로 사용되었다. 이 데이터 셋에서 출입허가자로 등록된 사람은 10명으로 사전에 실험 사용 목적의 허가를 받고 실험을 수행하였다. 출입 비허가자에 대한 데이터 셋은 인터넷 검색을 통해 쉽게 찾을 수 있는 100장의 임의의 사진을 이용하였다.

첫 번째 성능 측정은 정밀도 측정이며, 다음 Equation (1)과 같이 계산할 수 있다.

$$Precision = \frac{TP}{(TP + FP)} \tag{1}$$

Equation (1)에서, TP(True Positive)는 출입 허가자를 허가자로 판단하는 경우, FP(False Positive)는 출입 허가자를 비허가자로 판단하는 경우를 의미한다. 정밀도는 이미 등록된 사람에 대하여 등록된 사람으로 인식하는 성능이다. 따라서 비등록 된 사람이 매우 적고 대부분 출입자가 등록된 사람인 경우에 유용한 성능이다.

두 번째 성능 측정은 특이도 측정이다. 아래 Equation (2)를 사용하여 계산이 가능하다.

$$Specificity = \frac{TN}{(FN + TN)} \tag{2}$$

Equation (2)에서 FN(Flase Negative)는 출입 비허가자를 허가자로 판단하는 경우, TN(True Negative)는 출입 비허가자를 비허가자로 판단하는 경우를 의미한다. 특이도는 등록되지 않은 사람이 등록되지 않은 사람으로 인식하는 성능이다. 허가되지 않은 사람이 한 명이라도 허용되지 않아야 할 때 중요한 성능이다. 정밀도와 특이도를 측정하는 두 실험을 통해서 얼굴 인식 시스템의 출입 허가자 얼굴 인식률과 출입 비허가자 인식률을 확인할 수 있다.

세 번째 성능 측정은 정확도 측정이다. 정확도는 등록된 사람이 등록된 사람으로 인식하고, 등록되지 않은 사람이 등록되지 않은 사람으로 인식하는 성능이다. 정확도의 측정은 다음 Equation (3)을 사용한다.

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \tag{3}$$

성능측정은 얼굴 인식 Open API의 저장소에 얼굴을 저장하고 출입 허가자에 대한 얼굴 데이터 셋을 통해 얼굴 인식을 수행한다. 얼굴 인식 Open API의 저장소에 저장되는 얼굴은 총 10개이며 얼굴 데이터 셋에서 임의로 50개의 얼굴을 선택하여 10회 얼굴 인식을 하고 결과 값을 확률로 나타낸다.

인식률과 더불어 인식 시간에 대한 실험도 함께 실시한다. 인식 시간은 얼굴인식을 요청하고 그 결과 값을 얻을 때까지의 시간을 의미한다. 인식 시간의 측정은 얼굴 데이터 셋의 모든 사진을 이용하여 측정한다.

##### 4.2 출입자 인식 시스템의 성능 측정 결과

이 절에서는 성능 측정 결과를 논의한다. 이 실험에서는 단일 Open API를 사용할 때와 제안하는 출입자 인식 시스템을 사용할 때의 얼굴 인식률을 비교한다. 본 논문에서 제안한 출입자 인식 시스템의 성능을 비교하기 위해서 출입 허가자 및 출입 비허가자의 얼굴 인식률을 측정한다. 출입

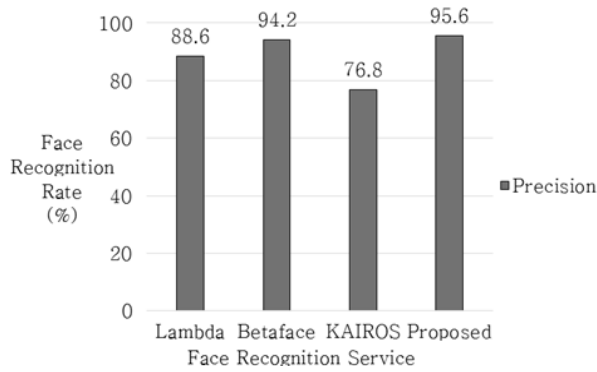


Fig. 5. Precision of Face Recognition Services

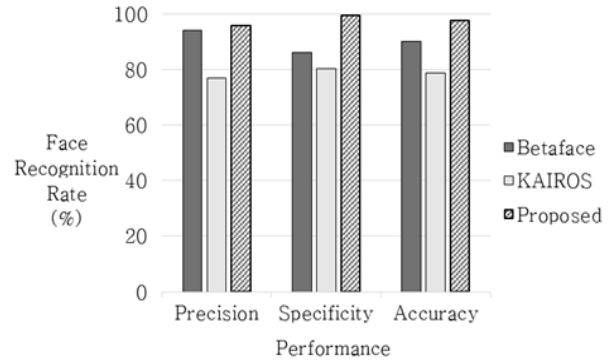


Fig. 7. Performance of Face Recognition Services

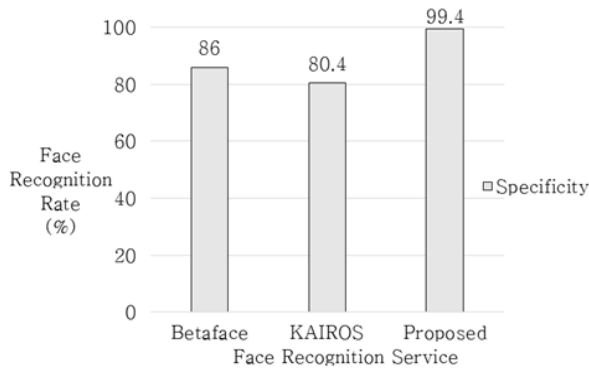


Fig. 6. Specificity of Face Recognition Services

허가자 성능측정은 성능 요구사항의 정밀도를 의미하고, 출입 비허가자 성능 측정은 성능 요구사항의 특이도를 의미한다. 정밀도와 특이도를 이용하여 성능 요구사항의 정확도를 측정한다.

Fig. 5는 얼굴 인식 Open API와 제안된 출입자 인식 시스템의 정밀도를 측정된 그래프이다. 이 그래프는 출입 허가자 10명을 등록하고, 얼굴 데이터 셋에서 등록된 10명에 대한 얼굴 50개를 10회 인식하여 얻은 결과이다. 그림에서 제안된 출입자 인식 시스템의 허가자 인식률이 95.6%로 단일 Open API의 인식률에 비해서 높은 것을 확인할 수 있다. KAIROS는 다른 얼굴 인식 Open API보다 얼굴 인식률이 낮은 것을 확인할 수 있다. 따라서 단독으로 사용하여 출입자 인식 서비스를 구현하였을 때 서비스 성능 요구사항을 만족하지 못할 수도 있다. 본 논문에서 제안하는 얼굴 인식 서비스는 각 Open API 서비스의 얼굴 인식 결과를 앙상블 알고리즘을 이용하여 인식하였기 때문에 단일 얼굴 인식 Open API를 사용했을 때보다 정밀도가 Lambda 대비 7%, Betaface 대비 1.4%, KAIROS 대비 18.8% 향상된 것을 확인할 수 있다. 앙상블 알고리즘에서는 하나의 얼굴 인식 Open API의 인식 결과가 나쁘더라도 나머지 두 개의 Open API에서 좋은 얼굴 인식률을 보이면 등록자로 판단할 수 있기 때문이다.

Fig. 6은 얼굴 인식 Open API와 제안된 출입자 인식 시스템의 특이도를 측정된 그래프이다. 이전 실험과 동일하게 출입 허가자 10명을 등록하고, 얼굴 데이터 셋에서 출입 허가자가 아닌 얼굴 50개를 10회 인식하여 얻은 결과이다. Lambda는 저장소의 얼굴에서 가장 유사한 얼굴을 인식하는 것에 중점을 두기 때문에 비허가자의 얼굴 인식에 적합하지 않아 비허가자 인식 측정 실험에서는 제외하였다. 실험 결과, 특이도는 Betaface 86%, KAIROS 80.4%, 제안된 출입자 인식 시스템 99.4%이다. 얼굴 인식 Open API 서비스의 출입 비허가자 얼굴 인식률은 출입 허가자를 인식하는 것보다 매우 낮은 인식률을 보인다. 얼굴 인식 Open API 서비스가 낮은 특이도를 보이는 이유는 각각의 Open API의 얼굴 인식 알고리즘이 등록된 인원들을 인식하는 것에 중점을 두기 때문에 등록되지 않은 인원들을 인식해 내는 것이 어렵다. 제안된 출입자 인식 시스템에서는 앙상블 알고리즘을 사용하여 인식 오류를 줄여 99.4%의 비허가자 얼굴 인식률을 보인다. 단일 얼굴 인식 Open API를 사용했을 때보다 특이도가 Betaface 대비 13.4%, KAIROS 대비 19% 개선된 것을 확인할 수 있다. 제안한 앙상블 알고리즘은 Betaface와 KAIROS에서 인식한 출입자가 같아야하고, 인식 확률도 THR보다 높아야 하므로 높은 특이도를 얻을 수 있었다.

Fig. 7은 얼굴 인식 Open API와 제안된 출입자 인식 시스템에서 정확도를 측정된 그래프이다. 정확도는 Betaface 90.1%, KAIROS 78.6%, 제안된 출입자 인식 시스템 97.5%였다. 그 외에 얼굴 인식 Open API 서비스별 인식 시간은 Lambda에서 평균 2.32초, KAIROS에서 평균 4.04초, Betaface에서 평균 7.0초가 측정되었다.

실험 결과를 통해, 단일 얼굴 인식 Open API를 사용할 때 보다 앙상블 알고리즘을 사용하는 제안된 출입자 인식 시스템의 얼굴 인식률이 정확도 97.5%, 특이도 99.4%, 정밀도 95.6%로 더 높음을 알 수 있다. 본 논문에서 제안하는 출입자 인식 시스템은 다양한 특성과 장점을 가진 서로 다른 얼굴 인식 모델을 사용하는 Open API 서비스들의 인식 결과를 조합하여 얼굴 인식을 수행하기 때문이다.



## 5. 결론 및 향후 연구

본 논문에서는 다양한 환경에서 적용이 가능한 출입자 인식 시스템의 요구사항을 만족하기 위해 다양한 특성과 강점을 가진 여러 얼굴 인식 Open API를 앙상블하여 사용하는 출입자 인식 시스템을 제안하였다. 출입자 인식 시스템의 실현 가능성을 보이기 위해 제안한 시스템을 구현하고 성능을 측정하였다. 성능 측정 결과에서 허가자의 얼굴 인식률과 비허가자의 얼굴 인식률이 단일 Open API 인식률에 비해서 더 우수함을 보였다. 향후 연구로는 상황에 따라 사용하는 얼굴 인식 Open API마다 가중치를 다르게 설정하는 앙상블 알고리즘을 개발하고 출입자 관리 시스템으로 확장하고자 한다.

## References

[1] R. Jafri and H. R. Arabnia, "A survey of face recognition techniques," *Information Processing Systems*, Vol.5, No.2, pp.41-68, 2009.

[2] Inttelix, Inttelix [Internet], <http://www.inttelix.com>.

[3] TCIT, TCIT [Internet], <http://www.tcit-us.com>.

[4] Digiface, Digiface [Internet], <http://www.digiface.com.br>.

[5] FIRSTEC, FIRSTEC [Internet], <http://www.firstecom.co.kr>.

[6] VS-KOREA, Smart-Face [Internet], <http://www.vs-korea.com>.

[7] Lambda Labs, Lambda Labs [Internet], <https://lambdal.com/face-recognition-api>.

[8] Betaface, Betaface API [Internet], <https://betafaceapi.com>.

[9] Kairos, Kairos [Internet], <https://www.kairos.com>.

[10] Face++, Face++ [Internet], <https://www.faceplusplus.com>.

[11] S. Z. Li and A. K. Jain, *Handbook of face recognition*, 2nd ed. Springer, 2011.

[12] C. Pagano, E. Granger, R. Sabourin, A. Rattani, G. L. Marcialis, and F. Roli, "Efficient adaptive face recognition systems based on capture conditions," in *Proceedings of Computational Intelligence in Biometrics and Identity Management*, pp.60-67, 2014.

[13] L. Wen, G. Guo, and X. Li, "A study on the influence of body weight changes on face recognition," in *Proceedings of IEEE International Joint Conference on Biometrics*, pp. 1-6, 2014.

[14] T. Kim, H. Park, S. H. Hong, and Y. Chung, "Integrated system of face recognition and sound localization for a smart door phone," *IEEE Transactions on Consumer Electronics*, Vol.59, No.3, pp.598-603, 2013.

[15] M. A. H. Lucas, L. A. Luis, E. B. M. Maria, R. Mariano, T. Juliana, and G. Sergio, "Smart doorbell: An ICT solution to enhance inclusion of disabled people," in *Proceedings of ITU Kaleidoscope Trust in the Information Society*, pp.1-7, 2015.

[16] K. H. Kwon and H. B. Lee, "Gate Management System by Face Recognition using Smart Phone," *The Korea Society of Computer and Information*, Vol.16, No.11, pp.9-15, 2011.

[17] G. D. Thomas, "Ensemble Methods in Machine Learning," *Multiple Classifier Systems*, Vol.1857, pp.1-5, 2000.

[18] K. H. Tin, "Random decision forests," in *Proceedings of Document Analysis and Recognition*, pp.278-282, 1995.

[19] G. Ratsch, T. Onoda, and K. R. Muller, "Soft margins for AdaBoost," *Machine Learning*, Vol.42, No.3, pp.287-320, 2001.

[20] H. M. Tang, M. R. Lyu, and I. King, "Face recognition committee machine," in *Proceedings of International Conference on Multimedia and Expo*, Vol.3, pp.425-428, 2003.



### 옥 기 수

e-mail : ok7323@kmu.ac.kr

2015년 계명대학교 컴퓨터공학과(학사)  
2016년 계명대학교 컴퓨터공학과(석사)  
2016년~현재 KETI 에너지IT 연구원  
관심분야: 유무선 네트워크, 시맨틱  
온톨로지, 머신 러닝



### 권 동 우

e-mail : dwkwon@kmu.ac.kr

2010년 계명대학교 컴퓨터공학과(학사)  
2012년 계명대학교 컴퓨터공학과(석사)  
2013년~현재 계명대학교 컴퓨터공학과  
박사과정  
관심분야: 미디어 스트리밍, 인터넷 침입  
예측, 네트워크 관리 및 보안



### 김 현 우

e-mail : hwkim84@kmu.ac.kr

2010년 계명대학교 컴퓨터공학과(학사)  
2012년 계명대학교 컴퓨터공학과(석사)  
2012년~현재 계명대학교 컴퓨터공학과  
박사과정  
관심분야: IoT 관리, 방화벽 정책 추론 및  
관리, 네트워크 관리 및 보안



### 안 동 혁

e-mail : donghyeokan@kmu.ac.kr  
2006년 한동대학교 전산전자공학부(학사)  
2013년 한국과학기술원 전산학과(박사)  
2014년 성균관대학교 컴퓨터공학과  
초빙교수  
2015년 삼성전자 책임연구원

2015년~현 재 계명대학교 컴퓨터공학부 조교수  
관심분야: 유무선 네트워크, IoT, 콘텐츠 중심 네트워크



### 주 흥 택

e-mail : juht@kmu.ac.kr  
1989년 한국과학기술원 전자계산학과(학사)  
1991년 포항공과대학교 컴퓨터공학과(석사)  
1997년 대우통신종합연구소 선임연구원  
2002년 포항공과대학교 컴퓨터공학과(박사)  
2002년~현 재 계명대학교 컴퓨터공학부  
교수

관심분야: 네트워크 및 시스템 관리, IoT 관리, SDN 네트워크  
관리, 인터넷 침입 예측, 네트워크 보안