

# ToF 카메라를 이용한 제스처 정보의 추출 및 전송

박원창\* · 류대현\* · 최태완\*

Extraction and Transfer of Gesture Information using ToF Camera

Won-Chang Park\* · Dae-Hyun Ryu\*\* · Tae-Wan Choi\*\*\*

## 요약

최근의 CCTV 카메라는 많은 경우 네트워크 카메라이며, 고화질 영상을 인터넷으로 전송하는 경우 큰 부하가 될 수 있다. 본 연구에서는 특정 환경에서 Kinect와 같은 ToF 카메라를 이용하여 제스처 정보를 추출하고 전송하는 방법을 이용하여 트래픽을 감소시킬 수 있는 방법을 제안하고 그 성능을 평가하였다. 제안된 방식은 ToF 카메라의 성능에 의존하므로 응용 분야에 제약이 있을 수 있지만 가정이나 사무실과 같은 소규모 실내공간의 보안 또는 안전 관리에 효율적으로 활용될 수 있다.

## ABSTRACT

The latest CCTV camera are network camera in many cases. In this case when transmitting high-quality image by internet, it could be a large load on the internet because the amount of image data is very large. In this study, we propose a method which can reduce the video traffic in this case, and evaluate its performance. We used a method for transmitting and extracting a gesture information using ToF camera such as Kinect in certain circumstances. There may be restrictions on the application of the proposed method because it depends on the performance of the ToF camera. However, it can be applied efficiently to the security or safety management of a small interior space such as a home or office.

## 키워드

CCTV, ToF Camera, Kinect, Skeleton, Security  
씨씨티브이, 티오에프 카메라, 키넥트, 스켈레톤, 보안

## 1. 서론

최근의 CCTV 카메라는 많은 경우 네트워크 카메라이며, 고화질 영상을 인터넷으로 전송하는 경우 큰 부하가 될 수 있다. MPEG-4 Visual과 같은 계산량이 많은 영상 압축 알고리즘을 적용해야 하므로 CCTV 카메라의 하드웨어와 소프트웨어의 비용도 증가하게

된다.

또한 인터넷을 통해 영상을 전송하는 경우, 패킷이 해킹되어 영상의 내용이 노출될 수 있다. 이 경우 집안 거주자 유무 등 민감한 정보가 잠재적인 침입자에게 제공될 수 있으며, 영상 속에 있는 사람의 프라이버시(privacy)를 손상당할 수 있다.

본 연구에서는 가정이나 사무실과 같은 소규모 실

\* 한세대 학교 IT학부(xjqnrtdl@naver.com)

\*\* 한세대 학교 IT학부 교수(dhryu@hansei.ac.kr)

\*\*\* 교신저자(corresponding author) : 국립 경남과학기술대학교 메카트로닉스공학과 교수(twchoi@gntech.ac.kr)

접수일자 : 2014. 08. 27

심사(수정)일자 : 2014. 09. 25

게재확정일자 : 2014. 10. 17

내공간과 같은 특정 환경에서 Kinect와 같은 ToF 카메라를 이용하여 제스처(gesture) 정보를 추출 및 전송하는 방법을 이용하여 영상 트래픽을 감소시킬 수 있는 방법을 제안하고 그 성능을 평가하였다.

TOF 기술이란, 센서(Sensor)로부터 나온 적외선 혹은 빛의 신호가 촬영하는 장면 내의 물체에 반사되어 돌아오는 시간을 계산하여 거리를 측정하는 방식이다. 촬영의 매 프레임마다 반사되어 돌아온 신호는 카메라의 CCD 센서 앞에 위치한 셔터에 의해 구분되며, 이렇게 획득된 깊이(depth) 정보는 정규화를 통해 깊이 공간을 생성하고 0부터 255의 깊이 레벨을 가지는 깊이 영상으로 변환된다. 이러한 깊이 영상을 이용하면 배경에서 객체를 분리해내는 컴퓨터 비전 문제의 어려움을 쉽게 해결할 수 있고, 사람이나 동물의 골격(Skeleton) 정보와 그 움직인 정보를 비교적 쉽게 얻을 수 있다.

## II. 관련 연구

일반적으로 장면의 깊이 정보를 얻는 방법은 크게 두 가지로 나눌 수 있는데, 수동 센서 방식과 능동 센서 방식이 그것이다. 수동 센서 방식은 촬영된 영상을 기반으로 하여 깊이 정보를 예측한다. 대표적으로 스테레오 정합(stereo matching)을 예로 들 수 있다. 이 방법은 데이터의 획득이 쉽고, 추가적인 장비가 필요하지 않다는 장점이 있지만, 가려짐 문제나 빛의 영향 등에 의해 고품질의 깊이 영상을 얻기가 힘들다.

능동 센서 방식은 측정 장비를 이용하여 장면의 깊이 정보를 직접 획득한다. 대표적 장비인 Time-of-flight(TOF) 깊이 카메라는 카메라에서 나온 빛이 물체에 도달하였다가 돌아오는 시간을 계산하여 깊이 영상을 출력한다. 이 방법은 실제의 깊이 정보를 실시간으로 측정할 수 있는 장점을 가지고 있지만, 장비의 사용에 의한 제약이나 잡음, 왜곡 문제 등이 따른다[1-2].

TOF 기술에서 획득된 깊이 정보는 정규화를 통해 깊이 공간을 생성하고 0부터 255의 깊이 레벨을 가지는 깊이 영상으로 변환된다. 깊이 카메라는 촬영하고자 하는 장면의 깊이 정보, 즉 카메라로부터 장면 내 물체까지의 거리를 측정하여 영상으로 출력하는 장비

이다. 최초의 깊이 카메라는 이스라엘 3DV Systems에서 개발한 ZCam이다. ZCam은 적외선 센서와 일반적인 카메라가 결합된 형태로 이루어져 있으며, 색상 영상의 촬영과 동시에 TOF 기술을 사용하여 장면의 깊이 영상을 획득한다. ZCam은 날씨 방송, 영화의 특수효과 등을 연출할 때에 촬영하는 영상의 전경과 배경을 쉽게 분리하여 합성하기 위한 크로마키(chroma key) 기술을 위해 개발되었으며, 기존의 크로마키 기술과 달리 장면의 깊이 정보를 이용하여 다중 레이어(layer)를 생성할 수 있다.

ZCam의 개발 이후 TOF 기술을 이용한 많은 깊이 카메라들이 개발되었다. 대표적으로 Swiss Ranger와 PMD Technology가 있으며, 이 밖에도 Fotonic, Canesta, Panasonic 등에서도 깊이 카메라를 출시하였다. ZCam 이후의 깊이 카메라들은 크기가 매우 작아지고 성능도 개선되었으며 ZCam보다 낮은 가격으로 구매할 수 있는 장점이 있다. 그러나 ZCam이 동일 시점에서 같은 해상도의 색상 영상과 깊이 영상을 제공하는 것과는 달리, 이후의 깊이 카메라들은 일반적으로 색상 영상을 촬영할 수 없으며, 색상 영상을 촬영할 수 있도록 설계된 카메라라도 촬영되는 깊이 영상과 색상 영상의 해상도 및 촬영 시점이 다른 특성을 가지고 있다[3-4].

TOF 깊이 카메라는 많은 장점을 가지고 있고, 앞에서 설명한 것처럼 부족한 부분을 보완할 수 있는 좋은 방법들도 많이 제안되었다. 그러나 TOF 깊이 카메라의 한계 또한 분명히 존재한다. 첫 번째는 촬영 거리에 따른 촬영 환경의 제약이다. 대부분의 TOF 깊이 카메라는 사용하는 변조 주파수 영역에 따라서 촬영 거리가 제한이 되어있다. 다시 말해, 촬영 환경이 실내로 제한된다는 것이다. 또한 실내라 할지라도 거리 측정을 위한 신호가 반사되어 돌아오기 위한 반사판 역할을 하는 배경이 카메라에 대해 상대적으로 낮은 위치에 있는 경우, 혹은 카메라가 정면이 아닌 약간 위쪽을 바라보는 경우에는 신호가 돌아올 수 없어서 생기는 현상이 발생하게 된다.

최근 가장 대표적인 3차원 깊이 센싱 기술은 프라임 센서에서 원천 기술을 개발하여 만들어진 3차원 깊이 센서로 마이크로소프트에서 Xbox에 실용화함에 따라 그 파급효과가 거의 전 세계에 이를 정도로 넓게 확장되어 있다. 현재 가장 대중적으로 사용하는 동

작인식 라이브러리는 크게 두 종류로 구분되고 있다. 하나는 프라임센서에서 개발한 OpenNI이고, 다른 하나는 마이크로소프트에서 개발한 Kinect SDK이다.

OpenNI는 프라임센서에서 개발과 동시에 발표되었으며, 최초로 센서의 구입자에게만 오픈되었던 것을 Xbox의 대중화를 타고 모두에게 공개하는 형태로 변형되었다. 크게 마이크로소프트에서 개발한 Kinect SDK와의 차이점을 살펴보면 OpenNI의 경우 컬러 카메라의 입력을 사용하지 않고 골격추출의 경우에도 깊이 영상의 자료를 가지고 제약조건을 가지고 골격구조를 추출하는 방식을 취했으며, 이와 반대로 Kinect SDK의 경우에는 인식의 응용에 있어 컬러 카메라의 입력을 사용하는 경우가 많아, 얼굴인식과 음성인식들을 결합하여 사용하고 있다. 또한, Kinect SDK의 경우 골격구조의 추출에서 미리 학습된 다량의 데이터에서 학습을 방법을 통하여 골격구조를 추출하는 방식을 취하고 있다. Kinect의 대중화에 따라 OpenNI를 Kinect에 연동하여 사용할 수 있는 라이브러리도 발표되어, 개발자들이 자유롭게 Kinect를 사용하여 OpenNI를 이용할 수 있도록 지원하고 있다. 최근 들어 이러한 라이브러리를 이용하여 여러 응용에 적용하고 이를 웹에 발표하는 사례들이 늘고 있다. 가장 대표적인 사이트로는 Kinect Hacks가 있다. 이 사이트는 세계 각지에서 여러 응용 분야에 적용한 실제 사례들이 빠른 시간 안에 전파되고 공유되는데 중요한 역할을 하고 있다. 국내에서도 활발한 연구가 진행 중이며, 여러 콘텐츠에 접목한 기술들을 발표하고 있으며, 계속적으로 교육, 국방, 엔터테인먼트 등의 분야에 확장시켜 가고 있다[5].

### III. 시스템 구성 및 구현

#### 3.1 전체 시스템 구성

시스템의 전체적 구성은 그림 1과 같다. 본 시스템에서는, Kinect[6-7]에서 얻은 DepthStream 영상에서 Skeleton Tracking을 통해 제스처 정보를 추출하여 이를 유무선 인터넷을 통해 전송하고, 리모트 PC나 스마트폰과 같은 모바일 기기에서 Skeleton을 다시 그려준다(그림 2 참조).

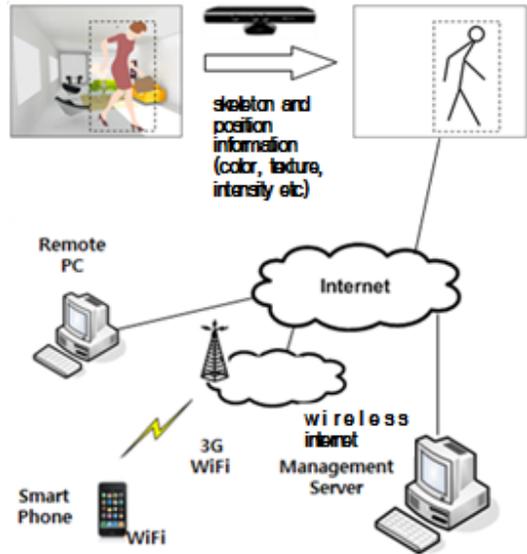


그림 1. 전체 시스템 구성  
Fig. 1 Total system configuration

필요에 따라서 배경 영상이나 객체의 색, 질감이나 밝기 정보를 부가적으로 전송하여 수신 단말기에서 합성하는 것도 가능할 것이다[8-9].

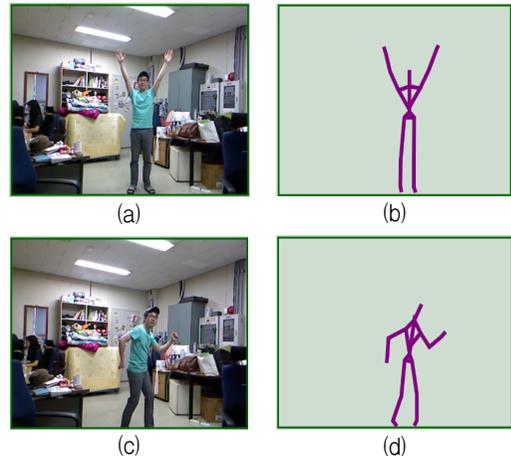


그림 2. 원영상(a), (c)과 골격영상(b), (d)  
Fig. 2 Original image(a), (c) and skeleton((b), (d))

#### 3.2 Kinect 와 NUI의 개발 환경

Kinect를 활용하는 개발 환경에는 기본적으로 Kinect SDK, Window7 OS와 Visual Studio2010이 포함된다.

Kinect SDK for Window는 마이크로소프트에서 발표된 것으로 Kinect를 위한 소프트웨어를 개발하기 위해서는 아래와 같은 3가지 구성요소를 제공한다.

- Kinect를 윈도우에 인식시키기 위한 디바이스 드라이버
- Kinect를 제어하기 위한 API 프레임워크
- 샘플과 문서

Kinect를 이용한 어플리케이션의 개발에는 VC++혹은 C#/VB 등의 개발 언어를 사용하며, NUI Library는 SDK에서 제공되고 있는 인체 구조를 인식하고 제스처를 분석하는 등의 기능을 하는 Kinect용 라이브러리이다. NUI Library는 Kinect로 부터 전송되어온 이미지와 Depth, 그리고 오디오 정보를 취합해서 어플리케이션에서 활용할 수 있는 정보로 전환하는 기능을 가지고 있는 핵심요소이다.

Kinect 하드웨어는 USB 기반의 기술로 PC와 연결되며 커널 모드 드라이버로 작성된 Kinect 드라이버가 하단에 자리 잡고 있다. 그 위에 카메라와 오디오 스택이 있고 그 위에 최상위 추상화 레벨에서 NUI API와 오디오, 비디오 컴퍼넌트(component)가 자리 잡고 있는 형태이다.

### 3.3 DepthStream과 SkeletonStream 영상

Kinect는 Depth 영상이 화면에 비추어지는 객체들에 대한 깊이에 대한 영상으로 DepthStream 영상을 제공한다. 그림 3에 DepthStream 영상과 UserSkeleton Tracking 결과를 표시하였다. DepthStream 데이터는 실시간으로 계속해서 변화하며 UpDate함수에서 DepthStream을 Open한다. 그리고 DepthStream의 출력에서 나오는 데이터를 넣는 공간은 위에 보이는 것처럼 Short형으로 되어 있어야한다. 그 크기는 일일이 계산을 해서 넣을 것이 아니라 그냥 DepthFrame의 크기를 집어넣으면 동적으로 형성된다. Copy-PixelDataTo()는 픽셀로부터 받은 데이터를 컬러프레임으로 복사하는 역할을 수행한다. 2D 객체에 대한 설정은 이미지처리에 대한 이해가 필요한데 프레임의 폭과 가로를 맞추고 bgr규격을 준수하여야 한다. 왜냐하면 센서에 내장되어 있는 칩(chip)이 (0, 0) 픽셀에서부터 읽을 때 각 데이터를 순차적으로 읽기 때문이다.

Update와 Draw는 순차적으로 60 프레임의 설정으로 호출되기 때문에 (한번 업데이트되고 한번 그려주고) 자연스러워 질 수가 있다. 2D 객체를 그리기위한 SpriteBatch를 GameService에서 얻어오기 위해서는 최상위 Game클래스에서 항상 AddGameService해야 한다. DepthStream은 Kinect 센서 자체에서 제공되기 때문에 데이터를 얻어오는 방법이 비교적 쉽다.

본 논문에서 2D DepthStream과 UserSkeleton의 표시상태를 나타낸 이유는 현재 User가 Kinect 센서에 잡혀있는 여부를 확인할 수가 있고, 센서가 적외선으로 깊게 투영을 한다고 해도, Joint 들이 일직선으로 겹쳐버리면 Kinect가 겹쳐진 뼈들에 대해 정확한 추적을 할 수 없기 때문이다. 그래서 해당 Joint들의 트래킹 상태여부를 확인하고 User가 올바른 Joint좌표를 위치시킬 수가 있다.



그림 3. 뎁스 스트림(a) 영상과 골격 추적(b)  
Fig. 3 (a) DepthStream image and (b) UserSkeleton tracking

### 3.4 제스처 정보

본 논문에서 제안하는 제스처 정보의 추출 및 전송의 경우, Kinect에서 읽은 Skeleton 컬렉션을 3d 좌표에서 2d 좌표로 변환한 후 바이트형으로 다시 변환하여 전송하고 전송받는 쪽에서는 역순으로 2d 좌표까지 변환한 후 UI 화면에 디스플레이(display) 한다.

초당 약 30 프레임 정도 발생하는 스켈레톤 관절정보는 총 20개이며 다음과 같다.

- HipCenter, HipLeft, HipRight
- Spine
- ShoulderCenter, ShoulderLeft, ShoulderRight
- Head
- ElbowLeft, ElbowRight, HandLeft, HandRight, WristLeft, WristRight

- KneeLeft, KneeRight, AnkleLeft, AnkleRight, FootLeft, FootRight

그 각각은 각 관절의 절대적인 좌표와 서로 연관관계가 있는 상대적인 좌표로 이루어져 있으며, 이들은 {Microsoft.Kinect.Matrix4} 형식의 매트릭스와 {Microsoft.Kinect.Vector4}의 쿼터니언 (Quaternion) 으로 표현되어 있다. 본 연구에서는 이들 정보를 TCP/IP로 전송하도록 하였으며, 전송시에는 다른 관절정보로부터 계산 가능한 WristLeft, WristRight를 제외한 18가지 관절 정보를 전송하였다. 전송시 필요한 관절 정보는 편의상 다음과 같이 총 5번으로 나누어서 전송하였다.

1. 머리와 몸통그릴 때 9(7+2) 가지 관절 좌표 정보
2. 왼쪽 다리를 그릴 때 5가지 관절 좌표 정보
3. 오른쪽 다리를 그릴 때 5가지 관절 좌표 정보
4. 왼쪽 팔을 그릴 때 4가지 관절 좌표 정보
5. 오른쪽 팔을 그릴 때 4가지 관절 좌표 정보

총 27개의 (x, y)좌표가 필요하며 순서대로 9-5-5-4-4 의 좌표가 필요한데, 헤더 사이즈(좌표 몇개가 전송될지에 관한) 4byte + 7개 관절 정보 (4byte\*2)\*7 = 56byte /// 60byte(헤더+데이터)

헤더 4byte + 2개 관절정보 (4byte\*2)\*2 = 16byte  
/// 20byte(헤더+데이터)

헤더 4byte + 5개 관절정보 (4byte\*2)\*5 = 40byte  
/// 44byte(헤더+데이터)

헤더 4byte + 5개 관절정보 (4byte\*2)\*5 = 40byte  
/// 44byte(헤더+데이터)

헤더 4byte + 4개 관절정보 (4byte\*2)\*4 = 32byte  
/// 36byte(헤더+데이터)

헤더 4byte + 4개 관절정보 (4byte\*2)\*4 = 32byte  
/// 36byte(헤더+데이터)

헤더와 관절 좌표에 관한 데이터의 전송 총 byte는 240byte이며, 1초에 거의 30 프레임이 발생된다.

#### IV. 시험 및 평가

본 논문에서 제안하는 제스처 정보의 추출 및 전송

의 주요한 장점인 전송 데이터 감소를 평가하기 위해 먼저 그림 2의 (a), (c)에 나타난 것과 같이 실내 공간에서 500 프레임 정도의 동영상(640 × 480, 30 프레임/초)을 촬영하였다. 이 영상을 압축하지 않은 경우 약 1.17 Mbyte 정도의 데이터 이지만 FFmpeg으로 압축을 하였을 때는 그림 4와 같은 형태로 데이터가 줄어든다.

압축된 비트스트림에서 I 프레임은 총 3개, P 프레임은 268개, 나머지는 B 프레임으로 구성되도록 하였다. 압축된 비트스트림은 StreamEye를 이용하여 각 프레임의 타입과 데이터 량을 확인하였다. 이 경우 I 프레임은 평균 50 kbyte, P 프레임은 6 kbyte, B 프레임은 1.2 kbyte 정도의 데이터 량이 된다. 물론 움직임이 많은 영상의 경우는 데이터 량이 더 많아질 것이다.

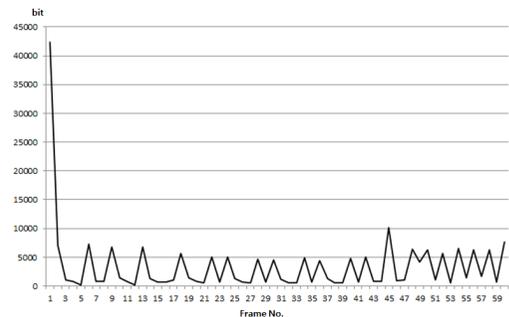


그림 4. H.264로 압축한 경우의 각프레임 별 데이터 량(byte)

Fig. 4 Data rate in the case of frame-by-frame compression with H.264

본 논문에서 제안한 것과 같이 제스처 정보의 추출 및 전송을 한다면 앞 절에서 설명한 것과 같이 각 프레임당 헤더와 관절 좌표에 관한 데이터 240byte 정도만 전송하므로 영상 트래픽을 감소시킬 수 있다.

그러나 실제 상황에서 100% 완벽한 움직임을 트래킹하지 못하여서 동작에 제한이 있으며, Kinect가 Skeleton을 트래킹할 때 서로의 뼈가 겹치는 경우 트래킹이 실패하는 등의 문제도 발생한다. 이러한 경우 2대 이상의 Kinect를 이용한다면 다양한 포즈(pose)를 트래킹하는 것도 가능하며 누워있는 자세에 대한 동작이나 미세한 부분까지의 관절 트래킹이 가능할 것이다.

## V. 결론

깊이 센서로부터 만들어진 공간정보를 가지고 있는 영상은 화면상에 나타나는 사용자를 실제의 공간에서 인지하고 그 사용자의 동작에 대한 인식을 수행하여 응용 프로그램에 적용하기까지의 과정이 기존의 2차원 영상에서 같은 작업을 진행하는 것보다 훨씬 수월하게 적용될 수 있는 장점을 가지고 있다. 그 중 한 가지는 사람에게 주어진 골격의 기본조건을 이용하여 스켈레톤 정보를 추출하기가 용이하다는 것이다.

본 논문에서는 특정 환경에서 Kinect와 같은 ToF 카메라를 이용하여 제스처 정보를 추출 및 전송하는 방법을 이용하여 영상 트래픽을 감소시킬 수 있는 방법을 제안하고 그 성능을 평가하였다. 제안된 방식은 각 프레임당 헤더와 관절 좌표에 관한 데이터 240byte 정도만 전송하므로 영상 트래픽을 크게 감소시킬 수 있음을 확인 하였다. 이러한 방식은 ToF 카메라의 성능에 의존하므로 응용 분야에 제약이 있을 수 있지만 가정이나 사무실과 같은 소규모 실내공간의 보안 또는 안전 관리에 효율적으로 활용될 수 있다.

### 감사의 글

본 논문은 2013년도 경남과학기술대학교 기성회 연구비 지원에 의하여 연구되었음.

## References

- [1] S. An, "Face detection and recognition with SURF for human-robot interaction," *In Proc. IEEE Int. Conf. on Automation and Logistics*, Shenyang, China, Aug. 2009, pp. 1946-1951.
- [2] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int. J. of Computer Vision*, vol. 60, no. 2, 2004, pp. 91-110.
- [3] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Surf : Speeded up robust features," *Computer Vision and Image Understanding(CVIU)*, vol. 110, no. 3, 2008, pp. 346-359.
- [4] J.-H. Oh, Y. Jung, Y. Cho, C. Hahm, H. Sin, and J. Lee, "Hands-up: Motion Recognition using Kinect and a Ceiling to Improve the Convenience of Human Life," *CHI EA '12 Proc. of the 2012 ACM Annual Conf. Extended Abstracts on Human Factors in Computing Systems Extended Abstracts*, Austin, TX, May 2012, pp. 1655-1660.
- [5] Z. Ren, J. Meng, J. Yuan, and Z. Zhang, "Robust hand gesture recognition with kinect sensor," *MM '11 Proc. the 19th ACM Int. Conf. on Multimedia*, Scottsdale, AZ, Nov. 2011, pp. 759-760.
- [6] C.-S. Won, S.-M. Kim, J.-S. Park, B.-W. Yoon, and J.-K. Song, "Hand Shape Recognition Based on Kinect and Analysis of the Performance," *Conf. of the Korean Institute of Electronic Communication Sciences*, vol. 7, no. 2, Yeosu, Korea, Nov. 2013, pp. 144-147.
- [7] S.-M. Kim, J.-K. Song, B.-W. Yoon, and J.-S. Park, "Height Estimation using Kinect in the Indoor," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 9, no. 3, 2014, pp. 343-350.
- [8] J.-S. Park and S.-J. Yi, "Development of Video Data-base and a Video Annotation Tool for Evaluation of Smart CCTV System," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 9, no. 7, 2014. pp. 739-746.
- [9] I.-S. Kim and H. Shin, "A Study on Development of Intelligent CCTV Security System based on BIM," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 6, no. 5, 2011. pp. 789-795.

저자 소개



**박원창(Won-Chang Park)**

2014년 현재 한세대학교 IT 학부 재학중

※ 관심분야 : IoT, M2M, 정보보호, 영상처리



**류대현(Dae-Hyun Ryu)**

1983년 부산대학교 전기기계공학과 졸업(공학사)

1985년 부산대학교 대학원 전자공학과 졸업(공학석사)

1997년 부산대학교 대학원 전자공학과 졸업(공학박사)

1987년~1998년 2월 전자통신연구원 선임연구원

1998년 3월~현재 한세대학교 IT 학부 교수

※ 관심분야 : IoT, M2M, 정보보호, 영상처리



**최태완(Tae-Wan Choi)**

1983년 2월 부산대학교 대학원 전자공학과 졸업(공학석사)

1996년 2월 부산대학교 대학원 전자공학과 졸업(공학박사)

1984년 12월~1991년 2월 (주)LG전자 디지털어플라이언스 연구소 선임연구원

1997년 3월~현재 국립 경남과학기술대학교 메카트로닉스공학과 교수

※ 관심분야 : 신호처리, 정보통신, 영상처리, Computer Vision