

<https://doi.org/10.7236/JIIBC.2023.23.4.99>
JIIBC 2023-4-16

CNN 기술을 사용한 지능형 CCTV 개발

Development of Intelligent CCTV System Using CNN Technology

김도은**, 공희진**, 우지후**, 이재문*, 황기태**, 정인환**

Do-Eun Kim**, Hee-Jin Kong**, Ji-Hu Woo**,
Jae-Moon Lee*, Kitae Hwang**, Inhwan Jung**

요약 이 논문은 IOT 기기인 라즈베리파이와 인공지능 기술을 이용하여 지능형 CCTV를 설계하고 시제품을 실험적으로 개발하였다. CCTV 화면에 있는 사람 수를 탐지하기 위하여 Object Detection 기술을 사용하였으며, 긴급 상황 탐지를 위하여 OpenPose에서 제공하는 Action Detection 기술을 사용하였다. 제안된 시스템은 CCTV, 서버 그리고 클라이언트 구조이고 CCTV는 라즈베리파이와 USB 카메라를 사용하였고, 서버는 리눅스를 사용하였고, 클라이언트는 아이폰을 사용하여 구현하였다. 각 서브 시스템간에 통신은 MQTT 프로토콜을 사용하여 구현하였다. 프로토타입으로 개발된 시스템은 초당 2.7프레임의 이미지를 전송할 수 있었고 초당 0.2 프레임에 대하여 이미지로부터 긴급 상황을 탐지할 수 있었다.

Abstract In this paper, an intelligent CCTV was designed and experimentally developed by using an IOT device, Raspberry Pi, and artificial intelligence technology. Object Detection technology was used to detect the number of people on the CCTV screen, and Action Detection technology provided by OpenPose was used to detect emergency situations. The proposed system has a structure of CCTV, server and client. CCTV uses Raspberry Pi and USB camera, server uses Linux, and client uses iPhone. Communication between each subsystem was implemented using the MQTT protocol. The system developed as a prototype could transmit images at 2.7 frames per second and detect emergencies from images at 0.2 frames per second.

Key Words : CCTV, Raspberrypi, Object Detection, Action Detection, Emergency

1. 서론

최근 1인 가구의 증가, 고립된 생활환경, 빈부의 격차 심화 등으로 개인의 안전과 관련된 범죄가 증가하고 있다. 이를 해결하기 위한 다양한 방안 중 하나가 사회 곳

곳에 CCTV를 설치하는 것이다. 그러나 기존의 CCTV의 기능은 사전 예방 차원이라기 보다는 사후 증거 확보를 위해 CCTV를 활용해 영상을 저장하는 수준이다.

최근 몇 년 동안 인공 지능은 놀라운 발전을 이루었다. 오늘날 인공지능은 전 세계적으로 다양한 분야에서

*정희원, 한성대학교 컴퓨터공학부(교신저자)

**정희원, 한성대학교 컴퓨터공학부

접수일자 2023년 7월 4일, 수정완료 2023년 7월 29일

게재확정일자 2023년 8월 4일

Received: 4 July, 2023 / Revised: 29 July, 2023 /

Accepted: 4 August, 2023

Corresponding Author: jmlee@hansung.ac.kr

Dept. of Computer Engineering, Hansung University, Korea

활용되고 있다. 특히 시각 데이터를 위한 인공지능 기술은 점점 더 많은 영역에서 인간의 개입을 대체하고 있다. 예를 들어, 다양한 이미지에 대한 이미지 분류, 이미지 내 물체 탐지^[1, 2], 영상 속 개인의 행동 탐지^[3, 4] 등 무수한 영역에서 적용되어 발전하고 있다.

본 논문은 이러한 인공지능 기술을 CCTV에 적용하는 것을 목표로 한다. 개발될 지능형 CCTV는 범죄 현장을 신속하게 감지뿐만 아니라 능동적 범죄 예방에도 활용될 수 있다. 그림 1은 개발하고자 하는 지능형 CCTV 시스템의 개략도를 보여준다. 라즈베리파이에 탑재된 카메라에서 전송되는 영상을 인공지능 기술로 사용하여 위급 상황을 감지해 자동으로 경찰에 알린다. 또한, 이동하려는 주변의 CCTV 영상을 모바일 폰으로 볼 수 있으며 특정 위치에 있는 사람의 수를 확인할 수도 있다. 부가적으로 ChatGPT 기술^[5]을 사용하여 주변 시설에 대한 정보를 받을 수 있다.

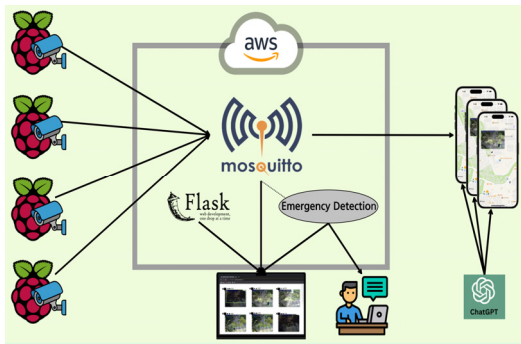


그림 1. 지능형 CCTV의 개략도
Fig. 1. Schematic diagram of intelligent CCTV

2장에서는 본 논문과 관련된 기술들에 대하여 소개하며, 3장에서는 제안된 시스템의 설계 및 구현에 대하여 설명한다. 4장에서는 개발된 시스템에 대하여 성능을 논하며, 연구 결과를 요약하며 향후 연구에 대하여 논한다.

II. 관련 연구

1. IoT 기술

사물 인터넷(IoT)은 인터넷을 통해 다른 장치 및 시스템과 데이터를 연결하고 교환할 목적으로 센서, 소프트웨어 및 기타 기술이 내장된 물리적 개체(사물)의 네트워크를 말한다. 이러한 장치는 일반 가정용품에서 정교한 산업 도구에 이르기까지 다양하다. IoT의 대표적

인 기기가 라즈베리파이이다. 라즈베리파이는 다양한 센서들을 직접 연결하여 데이터를 수집할 수 있으며, 또는 간접적으로 주변 센서들의 그룹과 통신하면서 데이터를 수집하고 서버로 데이터를 전송할 수도 있다. 또한 라즈베리파이는 무선 네트워크 장치가 있어 TCP/IP 통신을 할 수 있다.

IoT의 핵심 기능이 주변 센서 및 서버와 통신하는 기능이다. MQTT(Message Queuing Telemetry Transport)^[6]는 비교적 처리 능력이 열악한 CPU나 작은 메모리 등 제한된 자원을 가진 클라이언트들이 대역폭이 낮은 통신 환경에서도 쉽게 데이터를 주고받을 수 있도록 고안되었다. 특별히 통신을 위해 작성할 프로그램 코드 역시 짧고 간단하기 때문에 IoT 환경에 매우 적합하며, 오늘날에는 표준 IoT 프로토콜로 사용되고 있다. MQTT는 텍스트를 포함하여 이미지, 동영상 등 다양한 정보들을 전송할 수 있으며 한번에 전송할 수 있는 메시지의 크기는 최대 256MB까지 가능하다.

2. 인공지능 기술

딥러닝 중심으로 발전되고 있는 AI 기술은 눈부시게 발전하고 있다^[1, 2, 3, 4, 5, 7]. 딥러닝의 개념은 전혀 새로운 것이 아니다. 20여 년 전부터 활발하게 연구하였으나, 과도한 계산을 요구하기에 실용적으로 활용하지 못하였다. 최근 컴퓨터 능력의 발전으로 딥러닝의 개념이 실제 환경에 적용되고 있다.

CNN(Convolutional Neural Network)^[1, 2]은 컴퓨터 비전에서 일반적으로 사용되는 일종의 딥러닝 신경망 아키텍처이다. 컴퓨터 비전은 컴퓨터가 이미지 또는 시각적 데이터를 이해하고 해석할 수 있도록 하는 인공지능 분야이다. 이 기술에 기반하여 가장 활발히 사용되는 분야가 이미지 분류(Image Classification)^[1], 객체 탐지(Object Detection)^[2]이다. 두 기술은 매우 유사한데 전자는 주어진 이미지가 어떤 카테고리에 속하는지 분류해 내는 것이고 후자는 이미지 내에 존재하는 객체를 인지하고 이미지 내 어디에 있는지를 찾아내는 것이다. 후자를 사용하는 경우 이미지 내에 사람이 몇 명인지 쉽게 알 수 있다. 최근 CNN을 확대 적용하여 사람의 움직임 인지하는 기술이 개발되고 있다. 카네기 멜론 대학에서는 2D 이미지에서 사람의 인체, 손, 얼굴, 발, 키 포인트를 동시에 감지하는 실시간 다중 사람 시스템을 개발하였다^[3, 4]. 이를 이용할 경우 키 포인트의 움직임을 분석하여 사람의 행동을 분류하는 응용 소프트웨어를 개발할 수 있다.

III. 지능형 CCTV 시스템 설계 및 구현

1. 지능형 CCTV의 핵심 기능

본 논문에서 정의하는 지능형 CCTV는 사람이 직접 하여야 하는 일들을 인공지능 기술을 사용하여 최소화 할 수 있는 것을 말한다. 이러한 관점에서 본 논문에서 제안하는 지능형 CCTV는 다음과 같은 기능을 갖는다.

- 인공지능 기술을 이용한 긴급 상황 인식 기능

지능형 CCTV의 핵심 기능은 긴급 상황을 탐지하는 것이다. 기존 CCTV의 경우 사람이 직접 감지하거나 영상의 흔들림 등을 인식하는 경우가 대부분이다. 본 논문에서 설계한 시스템은 인공지능 기술인 동작 감지 기술을 이용하여 사람이 넘어지는 경우를 자동으로 감지하여 경찰서 등 관련 부서에 알린다.

- 인공지능 기술을 이용한 히트맵 기능

히트맵은 객체탐지 인공지능 기술을 활용해 현재 CCTV 영상 속 사람이 몇 명인지 지도에 표시해주는 기능이다. 휴대전화를 이용하여 사용자가 이동하고자 하는 위치에 사람이 밀집된 정도를 표시하여 사용자가 안전한 경로를 선택할 수 있도록 도와주는 기능이다.

- ChatGPT를 이용한 도움 기능

최근 ChatGPT의 활용은 사회 여러 분야에 적용되고 있다. 이 지능형 CCTV에서는 ChatGPT 기능을 이용하여 낯선 거리에서 가까운 편의점이나 버스 정류장을 검색할 수 있는 기능이다.

통신 방식을 사용한다. CCTV 시스템은 라즈베리파이 3 과 USB 카메라로 구성된다. 여기서 주요 모듈은 카메라 모듈과 인체 감지 모듈이다. 서버는 리눅스 서버이다. 서버는 '긴급 상황 탐지' 모듈, '긴급 상황 알림' 모듈, CCTV 현장별 인원수를 나타내는 '히트맵 관리' 모듈 그리고 전체를 모니터링 하는 '대시 보드' 모듈로 구성된다. 클라이언트 시스템은 CCTV 사이트를 영상으로 볼 수 있는 'CCTV 현장 보기' 모듈, ChatGPT를 이용하는 'ChatGPT 쿼리' 모듈로 구성된다.

3. 프로토타입 구현

본 논문에서는 지능형 CCTV의 기능들이 동작 가능 한지를 보이기 위하여 프로토타입을 구현하였다. CCTV 는 라즈베리파이 3을 사용하여 라즈비안 운영체제로 구성하였고, 서버는 AWS에서 제공하는 우분투 20.4 리눅스를 사용하였다. 클라이언트는 iOS가 실행되는 아이폰을 사용하였다.

모든 프로그래밍은 파이선3을 사용하여 구현하였고, MQTT는 Eclipse 재단에서 개발한 Mosquitto^[8]를 사용하였는데, 소스가 공개되어 있고 Linux, MacOS, Windows 등 다양한 운영체제에서 실행된다. 또한, Eclipse 재단에서 제공하는 응용프로그램 개발을 위한 paho.mqtt 라이브러리를 사용하여 개발하였다.

가. CCTV 구현

CCTV는 '카메라' 모듈과 '사람 수 탐지 모듈'로 구성된다. '카메라' 모듈은 라즈베리파이 3에 부착된 USB 카메라로부터 영상을 캡처하는 모듈이다. 이 모듈은 OpenCV 라이브러리에서 제공하는 VideoCapture 객체를 이용하여 구현하였다. VideoCapture 객체는 주기적으로 카메라로부터 이미지를 캡처하도록 한다. 파이선으로 구현하였을 때 640*480 크기의 이미지에 대하여 초당 10프레임 속도로 이미지를 캡처할 수 있었다.

'사람 수 탐지' 모듈은 인공지능 기술을 사용하여 구현하였다. 텐서플로우는 객체탐지에 대한 학습 모델을 제공한다. 이 학습 모델은 이미지에 나타나는 객체를 탐지하여 (객체 이름, 위치, 크기)의 정보를 응용 프로그램에게 알려준다. CCTV에서는 VideoCapture 객체로부터 주기적으로 이미지를 캡처하고 이 학습 모델에 적용하여 찾아지는 (객체 이름, 위치, 크기)들에 대하여 객체 이름이 'person'인 경우에 대해서만 물체를 표시하도록 함으로써 '사람 수 탐지'를 하도록 구현하였다. 실제로 구현하여 성능을 측정해보니 라즈베리파이 3에서 초당

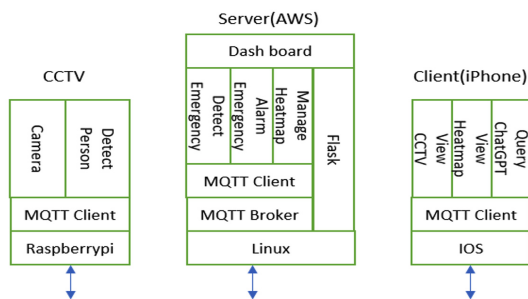


그림 2. 지능형 CCTV 시스템 구조
 Fig. 2. Architecture of Intelligent CCTV System

2. 지능형 CCTV 시스템 구조

지능형 CCTV는 그림 2와 같이 CCTV, 서버, 클라이언트 시스템으로 구성된다. 각 시스템 간의 통신은 MQTT^[6]

2.7 프레임 정도의 속도로 이미지 내 객체를 탐지하였다.

CCTV는 캡처된 이미지와 찾아진 사람의 정보를 MQTT를 통하여 MQTT Broker에 주기적으로 전송하도록 구현하였다.

나. 서버 구현

서버는 AWS(t3-small)에서 제공하는 우분투 20.4 리눅스를 사용하였다. 서버는 ‘긴급 상황 탐지’, ‘긴급 상황 알림’, 사람수에 대한 ‘히트맵 관리’ 기능 및 ‘대시보드’ 기능이 있다. 또한, 서버에는 가장 기본적으로 MQTT Broker에 실행되고 있는데 이는 Mosquitto를 사용하였다. 이 MQTT Broker는 다수의 CCTV로부터 전송되어오는 이미지와 사람탐지 정보를 대시 보드와 필요하여 요청하는 클라이언트로 중계하는 역할이다.

```
def detectEmergency(keyPoints, criteria):
    # 머리부분 키 포인트 정의
    partsOfHead = ["Head", "RShoulder", "LShoulder": 5]
    # 다리부분 키 포인트들 정의
    partsOfLeg = ["RAnkle", "LAnkle"]
    yOfHead = [] # 배열 초기화
    yOfLeg = [] # 배열 초기화

    for point in keyPoints:
        if point in partsOfHead:
            yOfHead.append(point.y)
        if point in partsOfLeg:
            yOfLeg.append(point.y)
    # end of for

    if(differenceOfAvg(yOfHead, yOfLeg)(criteria):
        return 1 # 긴급 상황
    else:
        return 0 # 일반 상황
```

그림 3. 긴급 상황 탐지에 대한 코드
Fig. 3. Code for Emergency Detection

‘긴급 상황 탐지’ 모듈은 OpenPose를 사용하여 구현하였다. 이것은 인체에 대하여 16개의 키 포인트와 14개의 연결라인을 정의하여 인체의 움직임을 탐지하는 학습 모델이다. 본 논문에서는 머리 부분 키 포인트들의 위치 평균과 다리 부분 키 포인트들의 위치 평균을 구한 후 이들의 y 값이 비슷한 경우 사람이 쓰러진 경우로 인식하여 ‘긴급 상황 탐지’로 구현하였다. 그림 3은 긴급 상황 탐지에 대한 축약된 코드이다. 먼저 매개 변수 keyPoints는 OpenPose 학습 모델에 의하여 찾아진 이미지에서 키 포인트들의 위치를 포함하고 있으면 criteria는 사람의 쓰러짐에 대한 기준값이다. differenceOfAvg() 함수는 두 배열에서의 값을 평균하여 그 값들의 차에 대한 절대값을 리턴하는 함수이다.

‘긴급 상황 알림’ 모듈은 긴급 상황이 발생하는 경우 경찰서 등 관련 부서에 알리는 기능으로써 다양한 방법이 있을 수 있다. 본 논문에서는 프로토타입의 개발이므로 간단히 카카오톡을 이용하여 문자를 전송하는 것으로 구현하였다.

‘히트 맵’ 모듈은 다수의 CCTV로부터 주기적으로 현재 지나다니는 사람 수를 받아서 통계 처리를 하고 클라이언트로 부터 요청이 있는 경우 사람 수에 대한 데이터를 전송한다. 이의 통신 방식도 MQTT를 사용하였다.

‘대시 보드’는 전체 지능형 CCTV 시스템을 모니터링 하는 모듈이다. 이 모듈은 웹으로 구현함으로써 인터넷이 가능한 아무 곳에서도 모니터링할 수 있도록 구현하였다. 이를 위하여 웹서버로는 Flask^[9]를 사용하였고 Mosquitto와의 통신을 위하여 웹 소켓을 사용하였다. 그림 4는 대시 보드의 스크린 샷으로 현재 4대의 CCTV로부터 영상을 받아와서 보여주고 있고, 한 지역에서 긴급 상황이 발생하였음을 보여주고 있다. 이 경우 카카오톡으로도 긴급 상황이 발생하였음을 알린다.



그림 4. 대시 보드의 스크린 샷
Fig. 4. Screenshot of Dashboard

다. 클라이언트 구현

클라이언트는 개인이 지능형 CCTV와 모바일 폰으로 연결하여 여러 가지 서비스를 받는 기능이다. 클라이언트는 크게 ‘CCTV 현장 보기’ 기능과 ‘ChatGPT 쿼리’ 기능이 있다. 클라이언트의 모든 기능은 오직 아이폰의 iOS에서 swift 언어로 구현하였다.

‘CCTV 현장 보기’ 모듈은 기본적으로 지도를 기반으로 하는데 이것은 iOS에서 제공하는 MKMapView를 이용하여 구현하였다. 지도 위에는 카메라가 설치된 지점에 카메라 이미지를 보임으로써 사용자는 카메라의 위치를 파악할 수 있도록 하였다. 이것은 MKAnnotation 클래스를 상속받아서 구현하였다. 또한 지도상에는 항

상 현재의 위치가 표시되는데 LocationManager를 이용하여 위치 정보를 획득하였다. 특히, 카메라 이미지를 클릭하면 CCTV 현장 영상이 출력되는데 이것은 CCTV에서 MQTT Broker에 주기적으로 보내는 영상을 MQTT Broker가 이미지를 요청한 클라이언트에 증계해 주기 때문이다. 이를 구현하기 위해서 Cocoapods에서 제공하는 CocoaMQTT5 라이브러리를 사용하여 구현하였다. 또한 MQTT Broker로부터 CCTV 현장에 '사람 수'를 수신받아 사람 수에 기초하여 카메라 이미지의 색상을 파란색에서 빨간색으로 변화하도록 하여 히트맵을 표시할 수 있도록 하였다.

그림 5는 구현된 'CCTV 현장 보기' 모듈에 대한 스크린샷이다. 그림 5에는 현재 4대의 CCTV가 있는데 2대는 하늘색 카메라 이미지로 보이고, 2대는 현재 카메라에서 촬영되고 있는 현장 영상이 보여주고 있다. 즉, 2대에 대해서는 카메라 이미지를 클릭하여 MQTT Broker로부터 영상을 수신하여 실시간으로 보여주는 것이다. 한번 더 클릭하면 영상 수신을 멈추고 카메라 이미지로 돌아간다. 이 'CCTV 현장 보기' 기능을 이용하여 사용자는 본인의 현재 위치에서 가까운 CCTV 현장을 관찰하여 가장 안전한 길을 택하여 목적지로 이동할 수 있을 것이다.

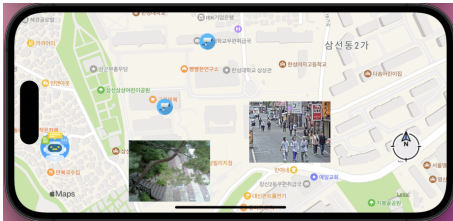


그림 5. 'CCTV 현장 보기'의 스크린샷
 Fig. 5. Screenshot of 'View CCTV Scene'

'ChatGPT 쿼리' 모듈은 사용자가 현재 위치에서 필요한 일들을 ChatGPT의 도움을 받는 것이다. 본 논문에서는 지능형 CCTV와 관련하여 '가장 가까운 편의점', '가장 가까운 버스 정류장' 및 '가장 가까운 카페' 찾기로 제한하였으나 화장실 찾기 등 무수히 많은 응용이 있을 것이다. 그림 6은 ChatGPT에 쿼리하는 간단한 코드이다. apiEndPoint는 웹서비스를 받을 수 있는 URL이고 parameters는 ChatGPT에 있는 다양한 서버를 model을 선택하고 쿼리를 저장하는 것이다. 적절히 task 객체를 생성하고 .task.resume() 함수를 호출하면 그 결과 '후행 클로저'라고 표시된 함수가 콜백으로 호

출된다. 여기서 데이터를 추출하여 채팅 폼으로 출력하도록 구현하였다. 그림 7은 구현된 ChatGPT의 채팅 창이다.

```
let apiEndpoint =
    "https://api.openai.com/v1/chat/completions"
let apiKey = "~~~~~"
func queryChatGpt(query: String) {
    var request = URLRequest(url: URL(string: apiEndpoint))
    request.httpMethod = "POST"

    request.addValue("Bearer \(apiKey)", forHTTPHeaderField:
    "Authorization")
    request.addValue("application/json", forHTTPHeaderField:
    "Content-Type")

    let parameters = [
        "model": "gpt-3.5-turbo",
        "messages": [[role: "user", "content": query]]
    ] as [String: Any]

    let httpBody = try?
    JSONSerialization.data(withJSONObject: parameters,
    options: [])
    request.httpBody = httpBody

    let task = URLSession.shared.dataTask(with: request)
    { (data, response, error) in // 후행 클로저
        // 이것은 task.resume()에 의하여 실행되며
        // JSON 데이터인 data 내용을 분석하여 채팅 창에 출력
    }
    task.resume()
}
```

그림 6. ChatGPT 쿼리 코드
 Fig. 6. Query code for ChatGPT



그림 7. ChatGPT와 채팅 창
 Fig. 7. Chatting Windows of ChatGPT

IV. 성능 논의 및 결론

제안된 시스템에서 '긴급 상황 탐지' 모듈은 신속성과 정확성이 매우 중요하다. 이를 위해서는 카메라에서 부터 촬영되는 이미지가 신속히 '긴급 상황 탐지' 모듈로 전송되어야 한다. 현재 라즈베리파이 3로 구성된 시스템에서 초당 약 2.7 프레임이 전송된다. 이것의 가장

큰 원인은 CCTV에서 사람의 손을 파악하기 위하여 Object Detection 알고리즘이 수행되기 때문이다. 이를 개선하기 위해서는 라즈베리파이의 성능개선이 불가피하다. 다행히 라즈베리파이 4가 판매 중에 있다. 또 다른 '긴급 상황 탐지'의 보틀넥은 OpenPose의 사용이다. 현재 AWS t3.small(vCPU 2개, Ram 2GB)에서 제공하는 유료 리눅스를 사용하는 경우 초당 0.5프레임에 대하여 키 포인트들을 추출하였다. 아직도 실시간으로 키 포인트들을 추출하기에는 너무나 느린 속도이다. 이를 해결하는 가장 쉬운 방법은 서버의 성능을 개선하는 것이다. 다른 해결책으로 전송되어오는 매 프레임마다 키 포인트들을 추출할 필요는 없다. 예를 들어 현재 초당 2.7프레임이 도착하고 하나의 프레임에 대하여 키 포인트들의 추출이 약 2초 소요되므로 약 5.4 프레임마다 하나의 프레임에 대하여 키 포인트들을 추출하여 긴급 상황을 탐지한다면 프레임의 지연없이 긴급 상황을 탐지 할 수 있다. 긴급 상황이라는 관점에서 2초도 긴 시간이라고 할 수 있지만, CCTV의 영상으로부터 2초만에 사람의 쓰러짐을 탐지한다는 관점에서는 허용 가능한 시간이라고 판단한다.

최근 널리 사용되고 있는 IOT 기기인 라즈베리파이와 인공지능 기술을 이용하여 지능형 CCTV를 설계하고 시제품을 실험적으로 개발하였다. 프로토타입 시스템은 설계에 따라 상대적으로 잘 작동하는 것으로 나타났다. 하지만 Object Detection, OpenPose 즉 인공 지능 모듈을 실행으로 전체적인 시스템 속도가 느려지게 개선책이 필요하다.

References

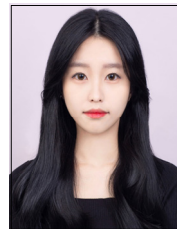
- [1] Khoi, Tran Quang, Nguyen Anh Quang, and Ngo Khanh Hieu. "Object detection for drones on Raspberry Pi potentials and challenges", IOP Conference Series: Materials Science and Engineering. Vol. 1109. No. 1. IOP Publishing, 2021.
- [2] Jae-Moon Lee, Kitae Hwang, Inhwan Jung, Development of a Hole Cup Recognition Model on Golf Green Using Object Detection Technology, The Journal of The Institute of Internet, Broadcasting and Communication (IIBC), Vol. 23, No. 1, pp. 15-21, 2023.
DOI : <http://doi.org/10.7236/IIBC.2023.23.1.15>
- [3] Qiao, Sen, Yilin Wang, and Jian Li. "Real-time human gesture grading based on OpenPose." 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics

(CISP-BMEI). IEEE, 2017.

- [4] Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y. (2017). Realtime multi-person 2d pose estimation using part affinity fields. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7291-7299).
- [5] Fitria, Tira Nur. "Artificial intelligence (AI) technology in OpenAI ChatGPT application: A review of ChatGPT in writing English essay." ELT Forum: Journal of English Language Teaching. Vol. 12. No. 1. 2023.
- [6] Bender, Melvin, et al. "Open-source mqtt evaluation." 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC). IEEE, 2021.
- [7] Choi, Nak-Hun, Oh, Jong-Seok, Ahn, Jong-Rok and Kim, Key-Sun, "A Development of Defeat Prediction Model Using Machine Learning in Polyurethane Foaming Process for Automotive Seat", Journal of the Korea Academia-Industrial cooperation Society, Vol. 22, No. 6, pp. 36-42, 2021.
DOI : <http://doi.org/10.5762/KAIS.2021.22.6.36>
- [8] da Silva, Ana C. Franco, et al. "Opentosca for iot: automating the deployment of iot applications based on the mosquito message broker." Proceedings of the 6th International Conference on the Internet of Things. 2016.
- [9] Mufid, Mohammad Robihul, et al. "Design an mvc model using python for flask framework development." 2019 International Electronics Symposium (IES). IEEE, 2019.

저 자 소 개

김 도 은(비회원)



- 한성대학교 컴퓨터공학부 재학
- 관심분야 : 정보보안, iOS, 인공지능

공 희 진(비회원)



- 한성대학교 컴퓨터공학부 재학
- 관심분야 : 인공지능, 모바일 시스템

우 지 후(비회원)



- 한성대학교 컴퓨터공학부 재학
- 관심분야 : 인공지능, 모바일 시스템

이 재 문(정회원)



- 한양대학교 전자공학과(학사)
- 한국과학기술원 전기및전자공학과(석사)
- 한국과학기술원 전기및전자공학과 (박사)
- 경력: 한국통신 연구개발단
- 관심분야 : 기계학습, 게임프로그래밍, 감성컴퓨팅

황 기 태(정회원)



- 서울대학교 컴퓨터공학과 학사
- 서울대학교 컴퓨터공학과 석사
- 서울대학교 컴퓨터공학과 박사
- 경력
University of California, Irvine
방문교수
University of Florida 방문 교수
- 주관심분야 : 모바일 시스템

정 인 환(정회원)



- KAIST 정보및통신공학과 박사
- 삼성전자 시스템사업부 수석연구원
- 한성대학교 컴퓨터공학과 교수
- 주관심분야 : 망관리, 멀티미디어신, IoT

※ 본 연구는 한성대학교 교내 학술 연구비를 지원받았음