

조명 제어를 위한 스마트 상황 센서

이창준¹, 박영채¹, 김혜지¹, 정은기¹, 황광일¹

¹인천대학교 임베디드시스템공학과

taeraciq@inu.ac.kr, inu201701735@inu.ac.kr, khj7248@inu.ac.kr,

eunki96@inu.ac.kr, hkwangil@inu.ac.kr

Smart Moment Sensor for Light Control

Chang-Jun Lee¹, Young-Chae Park¹, Eun-Gi Jeong¹, Hae-Ji Kim¹, Kwang-Il

Hwang¹

¹Dept. of Embedded System Engineering, Incheon National University

요약

단순한 기능만을 갖고 있던 조명 시장에 전원과 밝기, 색상 외에도 다양한 기능들이 추가된 스마트 전구 제품들이 출시되어 사용자에게 편리함과 새로운 경험을 제공해주고 있지만, 여전히 사용자의 수동 제어를 필요로 하고 있는 상황이다. 스마트 상황 센서 LightMe는 이에 AI 기술을 접목해 사용자의 조작을 최소한으로 하면서도 쉽게 접근할 수 있는 환경을 제공한다.

1. 서론

실내 조명의 조도와 색도의 미세하고 유동적인 변화를 사용하면 인간의 정서적, 인지적 반응을 이끌어 낼 수 있을 뿐만 아니라 시간의 흐름을 인지하는 수준에도 영향을 미칠 수 있다.[1] 스마트 전구의 기술 발전에 따라 더욱 조도와 색도의 정교한 조작이 가능해지면서 실내 조명의 이러한 이점을 더욱 잘 활용할 수 있게 되었다.

스마트 전구는 단순히 전력 공급의 유무에 따라 on/off 되던 기존의 전구에서 한 단계 발전하여 스마트 디바이스와 통신하여 제어되는 전구를 말한다. 현재 시장에 출시된 스마트 전구로는 음성 명령에 반응하는 제품이나 설정된 자동화/루틴을 통해 사용자의 생활에 밀접하게 작동하는 제품들이 있다.

하지만 해당 제품들 역시 사용자의 수동 조작과 단순히 트리거-액션 방식에 의존하고 있다. LightMe는 현재 시장의 스마트 전구에 AI 기술의 이점을 적용해 지정된 공간의 상황을 인식하고 그에 적합한 동작을 할 수 있도록 스마트 전구에 정보를 제공한다.

2. 요구사항

일반 사용자에게 더욱 편리한 사용자 인터페이스를 제공하기 위해서는 현재의 모든 스마트홈 플랫폼이 취하고 있는 자동화 기능의 구조를 개선해야 한

다. 현재 스마트홈 플랫폼들은 “만약 ~한다면 ~하고, 이후에 ~한다.”의 조건문 형태를 취하고 있다.

사용자에게 노출되는 외부적으로는 이러한 논리적 과정을 되도록 숨겨야 한다. 사용자가 “사물”, “사람”, “환경” 등의 핵심적이고 물리적인 몇 가지 요소에 집중할 수 있도록 유도한다면 일반적인 사용자에게도 쉽고 이해하기 쉽게 접근할 수 있을 것이다.

또한 현재 스마트홈은 그다지 똑똑하지 못하고 활용성도 높지 않다. 자동화 기능을 활용한다고 하더라도 사용자가 직접 일일이 조건과 동작을 지시해야 하고, 지속적인 관심과 관리가 필요하다. 조그마한 동작들을 자동화로 추가하다 보면 계속해서 개수가 늘어나 관리에 어려움이 생기기도 한다. 공간의 상황을 추정하여 스마트 기기들을 알아서 동작시킬 수 있다면 스마트홈을 조금 더 똑똑하게 발전시킬 수 있을 것이다.

3. 동작 예시

상황 인식 센서는 지정된 공간의 물품들을 기반으로 상황을 추정하기 위해 사용되는 객체 인식 기술과 상황의 중심이 되는 사용자의 자세를 인식하기 위한 동작 인식 기술을 결합하여 사용한다.

- 동작 인식

mediapipe의 동작 인식 기능을 활용하여 각 신

체 노드 간의 y좌표값 차이, 세 노드를 잇는 두 직선의 각도를 이용하여 누운 자세, 서 있는 자세, 앉아있는 자세의 세 가지 동작을 인식하도록 구현했다. [2]

```
#calculate angles
def calculate_angle(a, b, c):
    a = np.array(a)
    b = np.array(b)
    c = np.array(c)

    radians = np.arctan2(c[1]-b[1], c[0]-b[0]) - np.arctan2(a[1]-b[1], a[0]-b[0])
    angle = np.abs(radians*180.0/np.pi)

    if angle > 180.0:
        angle = 360-angle

    return angle
```

(그림 1) 두 직선의 각도를 구하는 코드

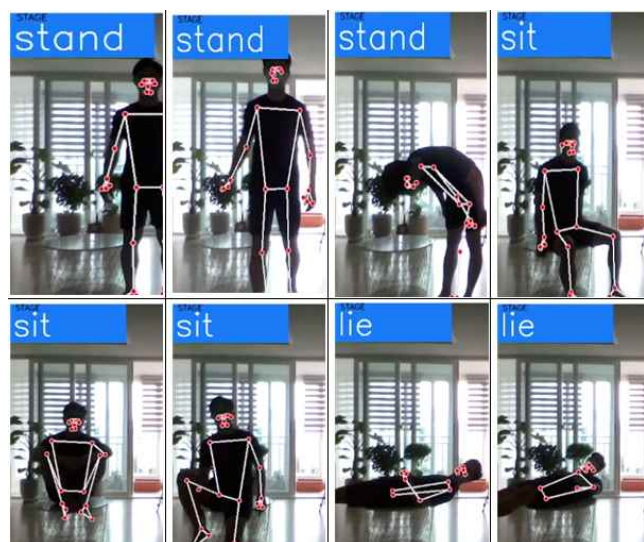
```
#calculate Y diff
def calculate_Y_diff_abs(a, b):
    a = np.array(a)[1]
    b = np.array(b)[1]
    return abs(a-b)

def calculate_Y_diff(a, b):
    a = np.array(a)[1]
    b = np.array(b)[1]
    return (a-b)
```

(그림 2) y좌표값의 차이를 구하는 코드

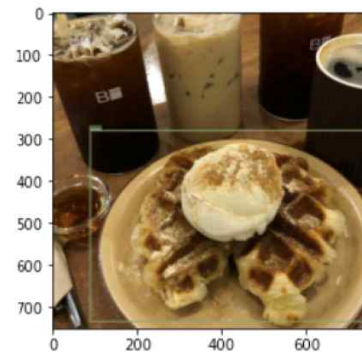
- 객체 인식

YOLOv4 객체 인식 기능을 활용하여 공간 안의 객체(클래스)들을 인식하고 이름, 클래스 이름일 확률, 좌표값, 비율 좌표값의 결과를 얻어냈다.



(그림 3) 동작인식 예시

와플
0.9261455535888672
89 280 752 733
0.1186666666666667 0.37333333333333335 1 0.9773333333333334



['와플 [89, 280, 663, 453]']

(그림 4) 객체인식 예시

- 상황판단 알고리즘

상황 센서는 일정한 시간 간격을 두고 지속적으로 설치된 공간의 사진을 촬영한다. 먼저 촬영된 물체들을 인식하여 속성값을 부여하거나 조정한다. 각 물체는 이름, 가중치, 존재 여부, 지속 카운터, 마지막 업데이트 시간의 다섯 가지 핵심 속성을 갖는다.

```
1 from datetime import datetime
2
3 class Object:
4     obj_name: str
5     moment_weight: dict[str, float]
6     is_presence: bool
7     counter_flag: int
8     last_update: datetime
```

(그림 5) 물체 객체의 기본 속성 구조

가중치는 사용자가 사전에 설정한 각 상황에 물체가 어느 정도의 영향을 주는지를 수치화한 값이다. 이 가중치는 현재 시각, 사용자와의 좌표 거리, 장면 내 존속 시간 등에 의해 지속해서 변경된다.

지속 카운터는 가중치에 필요한 수치이다. 지속해서 촬영된 공간의 상태에서 연속적으로 발견될 경우 카운터를 증가시킨다. 오랜 기간 공간에 존속한 만큼 상황에 일으키는 변화가 적기 때문에 가중치를 낮추는 역할을 한다.

- 허브

허브는 상황 센서와 스마트 전구들, 추가적인

스마트 디바이스들 중간에서 정보 처리와 연산을 담당하는 중계자이다. 허브는 지속해서 연결된 디바이스들의 상태를 받아오며 필요한 이벤트를 발생시키고, 각기 다른 네트워크 프로토콜(ex: Zigbee, Wi-Fi)의 기기들을 하나의 생태계로 통합시킬 수 있도록 한다. 전반적인 기술은 오픈소스 프로젝트 “Home-Assistant”[3]를 참고하였다.

- 애플리케이션

애플리케이션은 사용자들에게 노출되는 인터페이스의 역할을 한다. 허브가 처리한 정보들을 받아 기기들의 상태와 추정된 현재 상황 정보를 사용자가 관찰할 수 있는 형태로 제공한다. 또한 수동 조작이 필요할 경우에도 사용자는 어플리케이션을 통해 제어권을 가진다.

상황 정보에 대한 자세한 사항을 조회할 수 있도록 현재 상황을 추정한 근거(물체, 자세)와 그 수치를 제공하고, 해당 물체의 형태를 확인하기 위한 사진 정보도 제공한다.

4. 결론

상황 인식 센서는 스마트홈을 직접 경험하며 느낀 불편함을 해소해보고자 시작하였다. 사용자가 자동화를 일일이 설정해주는 것보다 공간의 정보를 제공하는 것으로 시스템이 알아서 상황을 인식하여 공간에 변화를 주면 좋겠다는 생각으로 진행하였다.

새로운 객체를 스스로 학습한다거나 반복되는 같은 수치에 대하여 하나의 새로운 상황을 스스로 만드는 거의 완전한 자율형 자동화의 구현은 어려웠지만, 현재 사용할 수 있는 기술들을 접목하여 사용자가 현재의 스마트홈에서 겪는 어려움을 해소해보고자 고민하며 개발해보았다.

프로젝트를 진행하면서 현재 스마트홈의 어려움에 어떠한 것들이 있는지 조사해볼 수 있었고, 이를 해결하기 위해서는 어떤 방안들이 있을지 고민해볼 수 있는 시간이 되었다.

참고문헌

- [1] 석현정/김곡미, LED 조명 색도에 따른 시간 인지에 대한 연구, 감성과학, Vol. 13, No. 1, pp.69-78, March 2010
- [2] MediaPipe Pose Table of Contents
<https://google.github.io/mediapipe/solutions/pose>
- [3] home-assistant (2022) core [Source Code].
<https://github.com/home-assistant/core>

Acknowledgement

※ 본 프로젝트는 과학기술정보통신부 정보통신창의인재양성사업의 지원을 통해 수행한 ICT멘토링 프로젝트 결과물입니다.