

시각장애인을 위한 모바일 기반 장애물 탐지 연구

조수형, 김호진, 박상순, 최유준, 이수원
승실대학교 소프트웨어학부

ch5951130@naver.com, ghwls328@gmail.com, parkss9677@gmail.com, cyj7732@gmail.com, swlee@ssu.ac.kr

A Study on Mobile-based Obstacle Detection for Blinds

Su-Hyeong Cho, Ho-Jin Kim, Sang-Sun Park, Yu-Jun Choi, Soowon Lee
School of Software, Soong-Sil University

요약

사물 인식이란 컴퓨터에 입력되는 이미지에서 사용자가 정의한 사물들을 컴퓨터 비전 기술을 이용하여 인식하는 과정으로, 사물 인식을 이용하면 컴퓨터가 카메라를 통하여 입력되는 이미지에서 장애물 등 특정 사물의 인식 결과를 사용자에게 알려줄 수 있다. 본 논문에서는 YOLO 사물 인식 알고리즘을 이용하여 시각장애인에게 전방의 장애물을 인식하여 알려줄 수 있는 기술을 제시한다. 해당 기술은 실용성을 고려하여 모바일 환경에서 이용할 수 있으며, 서버와의 연동을 통해 실시간으로 사용자에게 사물 인식의 결과를 알려줄 수 있다.

1. 서론

인공지능 기술의 발달로 다양한 분야에서 AI 기술이 연구 및 적용되고 있지만 시각장애인들을 위한 연구는 아직 다른 분야에 비해 비중이 낮고 미흡하다. 시각장애인들에게 필요한 것은 눈을 대신하여 전방의 위험한 물체들을 확인하고 알려줄 수 있는 기능이다. 이러한 기능의 역할을 할 수 있는 것이 딥러닝 기반의 사물 인식 알고리즘이다. 사물 인식 알고리즘을 모바일 기반으로 적용하여 시각 장애인이 보행 시 위험한 장애물을 알려줄 수 있다면, 시각 장애인들이 높은 접근성으로 이용할 수 있다는 장점이 있다. 이러한 배경에서 본 연구에서는 스마트폰 등 모바일 기반으로 사물 인식 알고리즘을 이용하여 장애물을 탐지하여 알려주는 시스템을 제시한다.

제안 방법은 먼저 시각장애인이 스마트폰 카메라를 이용하여 촬영한 전방의 이미지를 서버로 송신한다. 서버에서 이미지를 수신하면 YOLO 알고리즘[1]을 이용하여 전방에 있는 사람, 자전거, 전동 킥보드, 차량 4 종류의 장애물을 탐지한다. 이후 각 장애물까지의 거리를 계산하여 피해야 하는 장애물의 우선순위를 정한 후 스마트폰으로 그 결과를 전송한다. 스마트폰에서는 해당 정보를 토대로 사용자에게 소리 및 진동을 이용한 실시간 알람을 보낸다.

2. 관련 연구

영상에서 장애물을 인식하는 연구로는 [2]와 [3] 등이 있다. 두 연구 모두 스마트폰을 기반으로 HOG[4] 방식을 이용하여 구동된다. [2]의 경우에는 사람에 대해서 장애물로 탐지하고 그 사람과의 거리를 측정하며, [3]의 경우에는 현재 사용자가 걷고 있는 도로를 먼저 식별하고 도로 외의 부분을 장애물로 판단하여 어떤 장애물인지 식별한다.

본 연구에서는 장애물 탐지의 정확도를 높이기 위하여 HOG 대신 YOLO 알고리즘을 사용한다. 그러나 YOLO 알고리즘은 상대적으로 많은 연산량을 요구하므로 스마트폰의 부하를 줄이기 위해 서버를 활용한다. [표 1]은 본 연구의 제안 방법과 [2] 와 [3]의 연구를 비교한 것이다.

[표 1] 관련 연구와의 비교

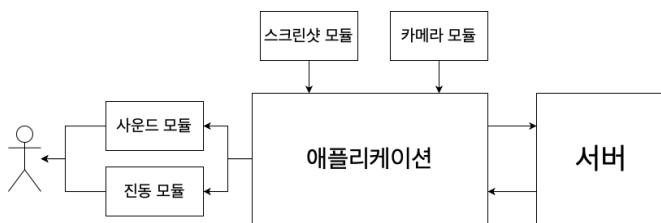
	알고리즘	거리 측정	장애물	구동 방식
[2]	HOG	0	사람	스마트폰
[3]	HOG	X	모든 장애물	스마트폰
제안 방법	YOLO	0	사람, 자전거, 전동 킥보드, 차량	스마트폰 + 서버

이와 같은 연구 외에도 시각장애인을 위해 장애물을 탐지해주는 제품들이 존재한다[5~7]. 이러한 제품은 기본적으로 구매를 필요로 하므로 본 연구에서는 대부분의 사람이 보유하고 있는 스마트폰을 활용하여 접근성을 높인다. [표 2]는 본 연구의 제안 방법과 기존 장애물 탐지 관련 제품을 비교한 것이다.

[표 2] 관련 제품과의 비교

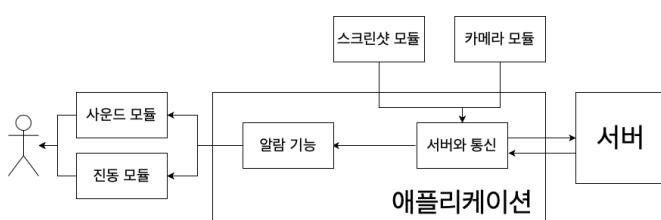
제품명	제품 형태	장애물 범위	네비게이션
BuzzClip [5]	초음파탐지 클립형식	근거리	X
WeWalk [6]	스마트 휴지팡이	근거리	0
Envision Glasses [7]	구글글래스	근장거리	0
제안방법	스마트폰 어플	근장거리	X

3. 제안 방법



[그림 1] 시스템 구조도

본 연구에서 제안하는 시스템의 구조도는 [그림 1]과 같으며 애플리케이션과 서버로 구성되어 있다. 모바일 환경에서 실시간으로 대량의 객체를 탐지하는 것은 기기에 부담을 많이 주며 객체 탐지의 정확성을 높이기 위해서는 많은 연산을 필요로 하므로 본 연구에서는 모바일 기기를 서버와 연결하여 장애물을 탐지하고 우선순위에 따라 알림을 제공한다.

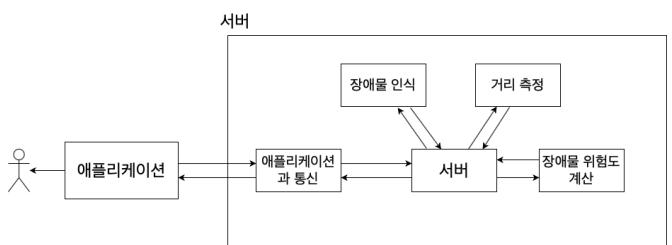


[그림 2] 애플리케이션 구조도

본 연구에서 제안하는 시스템의 애플리케이션 구조도는 [그림 2]와 같다. 애플리케이션에서는 사용자의 단말기의 모듈들을 이용하여 실시간으로 진행 방향의 길거리 영상을 촬영하고 일정한 시간마다 사진 정보

를 base64로 변환한 후 단말기의 정보와 함께 서버로 전송한다.

서버에서 정보를 처리하는 동안 애플리케이션은 실시간으로 계속해서 길거리 영상을 촬영하며, 동시에 서버에서 처리된 정보를 받으면 그것을 토대로 사용자에게 사운드 및 진동으로 알림을 주는 기능을 한다.



[그림 3] 서버 구조도

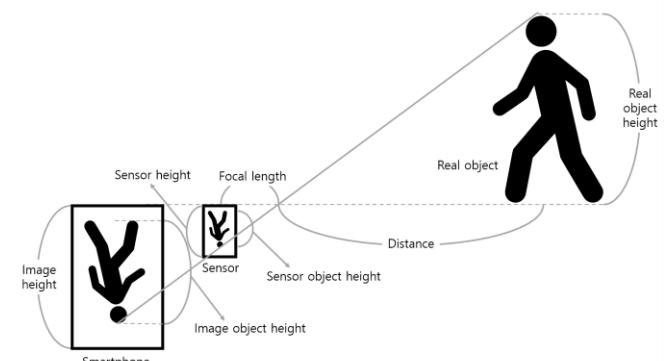
본 연구에서 제안하는 시스템의 서버 구조도는 [그림 3]과 같다. 서버는 base64로 인코딩되어 전달받은 사진을 일정한 크기로 변환한다. 그 후 YOLO를 사용하여 이미지 내의 객체를 탐지하고 사용자에게 위험이 되는 우선순위를 결정한다. 서버는 분석된 위험 정보를 단말기로 전송한다

3.1 객체 탐지

서버에서는 애플리케이션으로부터 받은 base64 정보를 OpenCV를 활용하여 정해진 크기의 이미지로 변환한다. YOLO는 변환된 이미지를 동일한 크기의 구역으로 나누고, 각 구역의 중심을 기준으로 미리 정의된 형태의 경계 박스의 개수를 예측한 후, 각각의 박스가 예측하는 객체와 일치하는지를 계산한다. 이러한 YOLO의 특성을 통해 본 연구에서는 장애물로 지정한 사람, 자전거, 전동 킥보드, 차량 4 종류의 객체를 탐지한다.

3.2 거리 측정

본 연구에서 사용하는 장애물과의 거리 측정 알고리즘은 [그림 4]에 기초한다.



[그림 4] 카메라와 물체와의 관계

[그림 4]는 실제 사물이 스마트폰 카메라의 센서를 지나 화면에 보이기까지의 과정으로 편홀 카메라의 원리를 확장한 것이다. 식 (1)은 센서와 실제 객체와의 관계를 이용하여 거리를 계산하는 식이다.

$$Distance = \frac{Focal_length \times Sensor_object_height}{Sensor_height} \quad (1)$$

여기서 *Sensor_object_height*는 활용할 수 없는 값이기 때문에 스마트폰 화면상의 이미지를 활용하여 사물과의 거리를 식으로 정리하면 식 (2)와 같다.

$$Distance = \frac{Focal_length \times Real_object_height \times Image_height}{Image_Object_height \times Sensor_height} \quad (2)$$

안드로이드에서 제공하는 *CameraCharacteristics* 클래스의 *LENS_INFO_AVAILABLE_FOCAL_LENGTH*와 *SENSOR_INFO_PHYSICAL_SIZE*를 이용하면 초점거리와 센서의 높이를 모두 구할 수 있다. 다만, 문제가 되는 부분은 *Real_object_height*에 해당하는 부분으로 실제 객체의 높이를 알 수 없다는 것이다. 이러한 문제를 해결하기 위해서 장애물 별 평균적인 높이 값을 이용한다. 장애물별 높이의 편차는 비율적으로 크지 않기 때문에 최종적으로 계산된 *Distance* 값에 큰 오차를 주지 않으며, 이를 통해서 큰 오차없이 장애물의 우선순위를 도출할 수 있다.

4. 실험

4.1 개발 환경

본 애플리케이션은 Android 플랫폼에서 Typescript 및 React Native를 사용하여 VsCode로 개발하였으며, 서버는 Typescript와 Python 및 Node.js를 사용하여 개발하였다.

4.2 상황별 수행 결과

본 연구에서는 사람, 자전거, 전동 킥보드, 차량, 총 4개의 객체를 장애물로 설정하고, 각 장애물의 위험도를 사람의 경우 1 단계, 자전거 또는 전동 킥보드의 경우 2 단계, 그리고 차량의 경우 3 단계로 설정하였다. 서버에서는 각 장애물의 종류와 시각장애인과 해당 장애물의 거리를 고려하여 가장 위험도가 높다고 판단되는 상황을 애플리케이션에 전달한다. 본 연구에서 설정한 우선순위 및 알림을 울리는 기준은 [표 3]과 같다. 1 순위의 경우는 3 가지 조건 중 하나만 일치하면 알림을 보낸다. 만약 여러 개의 상황이 동시에 나타났을 때는 더 높은 우선순위를 가지는 알림이 전달된다.

[표 3] 각 장애물 별 알림 방식

우선 순위	알림	사람	자전거/전 동킥보드	차량
1	비프음 1 회	1.5m 이내	3m 이내	4m 이내
2	진동 3 회	-	-	8m 이내
3	진동 2 회	-	8m 이내	-
4	진동 1 회	6m 이내	-	-

[표 4] 각 상황에 따른 알림 결과

상황	인식결과	알림	알림 조건
사람 1 명	Person A / 5.07m	진동 1 회	사람 알림
사람 2 명	Person A / 3.63m Person B / 5.30m	진동 1 회	사람 알림
사람 1 명 자전거 1 대	Person A / 4.72m Bicycle A / 3.98m	진동 2 회	자전거 우선 알림
전동킥보드 1 대	Kickboard A / 7.41m	진동 2 회	전동킥보드 알림
사람 2 명 자전거 1 대	Person A / 1.40m Bicycle A / 5.92m Person B / 7.01m	비프음 1 회	근접 사람 긴급 알림
사람 3 명 차량 1 대	Person A / 2.92m Person B / 2.95m Car A / 4.77m Person C / 6.38m	진동 3 회	차량 우선 알림
차량 1 대 전동킥보드 2 대	Car A / 2.12m Kickboard A / 5.51m Kickboard B / 10.23m	비프음 1 회	근접 차량 긴급 알림

[표 4]는 [표 3]의 장애인별 알림방식에 따라 진행된 장애물 탐지 상황 결과이다. [표 4]에서 첫 번째 상황의 경우, 촬영된 영상을 YOLO로 분석한 결과 사람 1 명만이 장애물로 탐지되었고 해당 장애물과의 거리가 6m 이내이기 때문에 [표 3]의 조건에 따라 진동 1 회로 알림을 주었다. 세 번째 상황의 경우 사람 1 명과 자전거 1 대가 장애물로 탐지되었고 사람과의 거리는 6m 이내, 자전거와의 거리가 8m 이내이다. 이 경우 동시에 두 가지 상황이 나타났으나 자전거가 우선순위가 더 높기 때문에 진동 2 회로 알림을 주었다.

5. 결론 및 향후 연구

본 연구에서는 시각장애인을 위한 모바일 기반 장애물 탐지 애플리케이션을 제안하였다. 제안 애플리케이션은 서버와의 연동을 통해 YOLO 라이브러리를 사용한 객체 탐지 모듈과 핀홀 카메라의 기본 원리를 활용한 거리 측정 모듈로 구성된다. 본 애플리케이션은 모바일 환경을 기반으로 실행되기 때문에 사용자의 휴대성이 간편하고 접근성이 높다는 장점이 있다. 또한, 시각장애인에게 위험에 대해 능동적으로 대처할 수 있는 수단을 제공하여 안전한 보행에 도움이 될 수 있을 것으로 기대된다.

본 기술은 보행 시각장애인의 안전과 직결된 기능인만큼 서버와의 통신 및 보안성을 강화하고 실시간으로 자체적인 학습을 통해 시각장애인에게 안전한 상황으로 유도할 수 있는 기능 연구가 중요하다. 또한 보행자에게 더 안전한 상황을 알려주기 위하여 차도와 인도를 구분하고 보행이 가능한 길을 구분하는 방법에 관한 추가적인 연구가 필요하다.

감사의 글

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW 중심대학사업의 연구결과로 수행되었음
(2018-0-00209)

참고문헌

- [1] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi: You Only Look Once: Unified, Real-Time Object Detection, CVPR 2016, 779-788, 9 May 2016.
- [2] Fernando Garcia, Jesus Urdiales, Juan Carmona, David Martin and Jose Maria Armingol: Mobile based Pedestrian Detection with Accurate Tracking, 2016 IEEE Intelligent Vehicles Symposium (IV), 44-48, 19-22 June 2016.
- [3] Ruxandra Tapu, Bogdan Mocanu, Andrei Bursuc, Titus Zaharia: A SmartPhone-Based Obstacle Detection and Classification System for Assisting Visually Impaired People, 2013 IEEE International Conference on Computer Vision Workshops, 444 – 451, 2-8 Dec. 2013.
- [4] Navneet Dalal, Bill Triggs, Histograms of oriented gradients for human detection, 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 886-893, 20-25 June 2005.
- [5] BuzzClip [Product], Available from:
<https://www.imerciv.com/user-guide/index.shtml>

- [6] WeWalk [Product], Available from:
<https://wewalk.io/en/product/>

- [7] Envision Glasses [Product], Available from:
<https://www.letsenvision.com/envision-glasses>