

우분투 기반 라즈베리 파이3의 영상 인식 시스템 개발

김규현* · 장종욱*

*동의대학교

Development of Ubuntu-based Raspberry Pi 3 of the image recognition system

Gyu-Hyun Kim* · Jong-Wook Jang*

*Dong-Eui University

E-mail : kim33276@naver.com, jwjang@deu.ac.kr

요 약

최근 IoT를 활용한 웨어러블 기기 및 무인 이동체에 관한 연구가 활발히 진행되고 있다. 그 중 무인 이동체는 IT 기술들의 집약체라고 할 수 있다. 로봇, 자율 주행, 장애물 회피, 데이터 통신, 전력, 영상 처리 등의 기술들이 합쳐진 것을 무인 이동체 또는 무인 로봇이라고 부른다. 무인 이동체의 최종 목표는 수동이 아닌 자율 주행을 하여 목적지까지 안전하고 신속하게 도달하는 것을 목표로 한다.

본 논문에서는 무인 이동체의 기술들 중 하나인 영상 처리 분야를 다루고자 한다. 현재 배터리의 기술로는 무인 이동체가 최대 1시간까지 주행할 수밖에 없다는 것을 감안하여, 전력 소비를 최소한으로 줄이기 위해 소형 컴퓨터인 라즈베리 파이3를 사용하여 영상 인식 시스템을 설계하고자 한다. 제안하고자 하는 시스템은 카메라로부터 받는 영상의 모든 것을 인식하는 시스템을 목표로 한다.

ABSTRACT

Recently, Unmanned vehicle and Wearable Technology using IoT research is being carried out. The unmanned vehicle is the result of IT technology. Robots, autonomous navigation vehicle and obstacle avoidance, data communications, power, and image processing, technology integration of a unmanned vehicle or an unmanned robot. The final goal of the unmanned vehicle manual not autonomous by destination safely and quickly reaching.

This paper managed to cover One of the key skills of unmanned vehicle is to image processing. Currently battery technology of unmanned vehicle can drive for up to 1 hours. Therefore, we use the Raspberry Pi 3 to reduce power consumption to a minimum. Using the Raspberry Pi 3 and to develop an image recognition system. The goal is to propose a system that recognizes all the objects in the image received from the camera.

키워드

Unmanned Vehicle, Unmanned Robot, Raspberry Pi 3, Image Processing, Power Consumption

1. 서 론

최근 IoT를 활용한 웨어러블 기기 및 무인 이동체(unmanned vehicle)에 관한 연구가 활발히 진행되고 있다. 무인 이동체는 자율 주행이 가능한 로봇을 말하는데, 로봇 메카니즘, 영상처리, 네트워크 통신, 인공지능 등의 여러 IT 기술들이 융합되어 자율 주행을 가능하도록 한다. 자율 주행이 가능한 무인 이동체는 친환경 청정에너지에 대한

수요의 확대 및 정보통신 기술의 발달로 인하여 첨단 기능이 추가되어 실내 전시장 또는 박물관의 안내 업무, 자연도로의 실외순찰, 우주탐사 등의 다양한 분야에서 사용되고 있다[1].

본 논문에서는 무인 이동체의 핵심 기술 분야들 중 하나인 영상처리에 대해 연구하고자 한다. 무인 이동체는 드론 또는 자동차, 로봇 등 베이스에 따른 영상처리 기술이 각각 다르게 적용된다. 베이스 시스템에 따라 인식해야 하는 개체의 수

가 다른데, 그 중 자동차를 베이스로 한 자율주행에 필요한 영상 인식의 개체 종류는 크게 5가지이다. 사람, 차량, 차선, 물체, 건물들 전부를 인식이 되어야 자율 주행을 하는데 있어 문제가 발생하지 않는다. 인식 개체 수가 제일 많은 자동차를 대상으로 한 무인 이동체의 연구는 테스트 환경에 따른 제약 때문에 주행 로봇을 이용하여 연구를 진행하였다.

II. 관련연구

객체의 특징점을 이용한 기존 연구는 PC 환경과 같은 고성능 CPU와 대용량의 메모리를 제공하는 환경에서 이루어져 왔다. 다양한 환경에서 지역적인 특징을 이용하여 대응점을 찾기 위해서는 크게 특징점 추출과 기술어(descriptor) 생성의 두 가지 과정이 필요하다. 첫 번째 특징점 추출 과정은 영상에서 환경이 변하더라도 확실하게 찾아질 수 있는 특징의 위치를 찾는 것이다. 특징점은 대응되는 영상에서도 물리적으로 동일한 곳에 위치하여야 하며, 다른 영상에서 해당 특징점이 찾아질 확률 또한 높아야 한다. 두 번째 과정은 첫 번째 과정에서 찾은 특징점을 설명할 기술어를 만드는 과정이다. 기술어는 환경 변화에 불변하는 특징을 가져야 하며, 다른 특징점과 구분되어야 한다. 마지막으로 찾아진 특징점의 표현자를 비교하여 대응점을 찾을 때 속도를 향상시키기 위해 최소한의 데이터를 이용하여 기술어를 만들어야 한다[2].

대표적으로 헤리스 코너 검출에서는 객체가 가지고 있는 코너 정보를 이용하는 방법을 제안하고, 이를 바탕으로 다양한 방법론들이 연구되기 시작했다. Lowe의 스케일 불변 특징 변환, 즉 SIFT(Scale Invariant Feature Transform) 알고리즘[3]의 경우 크기와 회전에 불변하는 특징점을 객체로부터 추출하는 과정을 거쳐 인식하고자 하는 객체의 크기 변경 및 회전등에 상관없이 항상 동일한 위치의 특징점을 인식하도록 연구되었다. 다만 특징점이 객체의 변화에 영향을 덜 받기 위해서는 검출 및 비교에 많은 연산량을 요구하게 되어 실제 PC 환경에서도 실행 속도가 저하되는 현상이 나타난다. 속도 향상과 SIFT의 인식 능력에 유사한 특징 추출 방식인 SURF(Speeded Up Robust Features) 알고리즘[4]의 경우 속도는 SIFT에 비하여 높아졌으나 모바일 환경에서 사용하기에는 여전히 인식 속도의 문제가 발생하고 있으며 이를 개선하기 위한 다양한 방법의 연구가 진행되고 있다. 위의 두 방법론은 객체의 변화에 상관없이 특징점을 추출하고 이를 비교함으로써 다양한 조건(스케일, 사이즈, 노이즈, 빛 등)에 영향을 받지 않고 객체를 안정적으로 인식할 수 있게 된다. 하지만 불변의 객체 특징점 도출을 위해 많은 계산량을 요구하게 되어, 본 연구에서는 시중에 나와 있는 모듈 중 성능이 제일 좋은 라즈베리파이3를 이용하여 실시간 객체 인식에 대

한 연구를 진행하였다. 이에, 위에 언급한 방법론의 단점을 극복하고 객체 인식의 속도를 향상하기 위하여 SF(ShiTomasi Function)[4]을 이용한 객체의 코너 데이터를 검출하여 실시간 인식 시스템을 개발하고자 한다.

III. 시스템 구조 및 설계

본 논문의 목적은 차세대 무인 자동차에 적용될 기술을 연구하고자 함이다. 본 논문에서 언급되는 연구는 라즈베리파이3, Kinect 카메라를 이용하여 연구를 진행하고 있다. 연구하고자 하는 시스템은 무인 이동체와의 연동을 목적으로 하고 있다. 무인 이동체는 이동에 필요한 전력을 외장 배터리로부터 공급 받는 시스템이기 때문에, 전력 낭비를 최대한으로 줄이기 위해 라즈베리파이를 사용하였다.

3.1 영상처리 알고리즘

본 논문에서는 라즈베리파이, Kinect 카메라를 이용한 실시간 영상 인식 시스템을 개발하기 위하여 사람, 물체 인식 등에 사용되는 OpenCV 라이브러리를 사용하였다. 이러한 실시간 영상 인식을 위해 영상 인식의 기본이 되는 특징점 추출하는 방법을 제안한다. Kinect 카메라로부터 입력된 RGB 색상값을 코너 데이터 검출을 위하여 그레이 색상 값으로 변환, SF 알고리즘을 이용하여 객체의 코너 데이터를 검출하는 3단계계를 통해 특징점을 추출한다. 추출된 특징점의 위치 및 거리와 기술어를 계산하여 저장한다. 이러한 객체의 코너 데이터 검출을 위해 다음과 같은 SF 알고리즘의 행렬식을 사용한다.

$$M_h = \begin{vmatrix} \left(\frac{\delta I(x,y)^2}{\delta_x}\right) & \left(\frac{\delta I(x,y)}{\delta_x}\right)\left(\frac{\delta I(x,y)}{\delta_y}\right) \\ \left(\frac{\delta I(x,y)}{\delta_x}\right)\left(\frac{\delta I(x,y)}{\delta_y}\right) & \left(\frac{\delta I(x,y)}{\delta_y}\right) \end{vmatrix}$$

만약 어느 임의의 점에서 행렬 M_h 의 두 개의 고유 값이 크다면, 임의의 방향을 갖는 작은 변화는 그레이 값의 변화로 나타날 것이다. 코너의 x 특징점 같은 경우엔 이러한 변화가 평탄한 영역에 비해 크게 나타난다. 코너 응답 함수는 식(2)로 주어진다.

$$R = \det M_h - k(\text{trace } M_h)^2 (2)$$

구해진 코너 응답 함수 R의 값은 영상의 모든 픽셀에 나타나게 된다. 아주 미세한 변화에 따른 코너 응답 값이 존재하기 때문에 영상 전체에 threshold 을 조정하여 적절한 수치를 설정하게 된다.



그림 1. threshold = 1. threshold 값의 변화에 따른 객체 코너 정보 추출.

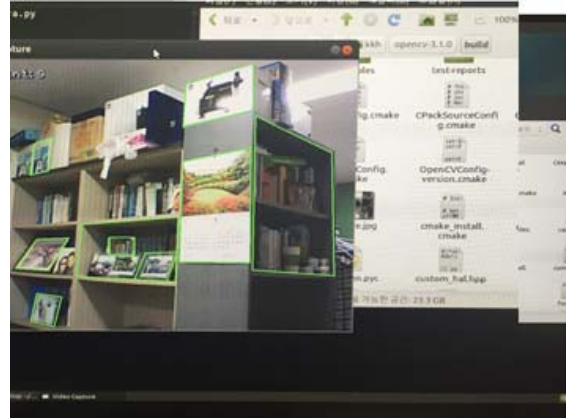


그림 3. threshold = 60. threshold 값의 변화에 따른 객체 코너값 및 영역 추출.

3.2 코너 데이터를 이용한 영역 추출

객체 인식 과정은 검출에 필요한 특징점과 에지 데이터의 선을 검출한다. 코너 데이터 추출은 RGB 색상의 이미지를 그레이스케일 영상으로 변환 후, SF 알고리즘을 이용한다. SF 알고리즘으로부터 추출된 코너 데이터와 주변 픽셀 데이터를 배열에 저장하여 최종 객체의 특징점으로 사용할 예비 작업을 진행한다.

특징점은 객체의 상하좌우 코너, 주변 픽셀 정보 및 픽셀 값의 합, 코너 간의 거리를 가진다. 객체 내부의 사분면 정보에 존재하는 코너 데이터를 주 정보로 하여, 주변 픽셀 정보 및 거리를 비교 대상으로 한다. <그림 2>은 객체 정보 생성흐름을 보여주고 있으며, 이미지 입력 및 코너 데이터 추출, 특징점 후보군 구성, 객체 특징점 도출, 객체 특징점 간의 거리 및 주변 픽셀 정보 4단계로 구성 된다.

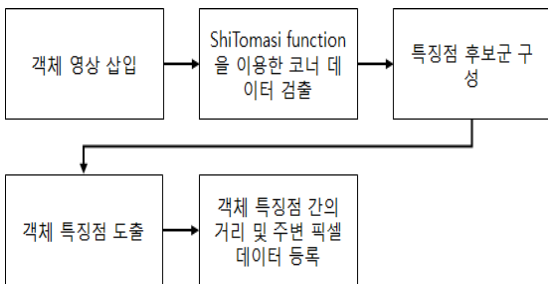


그림 2. 객체 특징점 생성 흐름도

<그림 3>는 threshold 값을 60으로 주어 영역을 인식한 결과 값이다. threshold 수치에 따른 코너 값의 검출 횟수가 차이 나는 것을 알 수 있다.

3.3 실험 내용 및 결과

본 실험은 라즈베리파이3와 Kinect 카메라를 사용하여 실험을 진행하였다. 라즈베리파이3의 환경은 Quad Core 1.2Ghz, 1GB Ram이며, Ubuntu OS를 사용하였다.

실험에 사용된 데이터는 라즈베리파이3에 장착된 Kinect 카메라를 사용하여 획득한 데이터를 이용하였다.

위에서 제시한 환경에서 각각의 알고리즘(SIFT, SURF, FAST)에 대한 성능을 측정하였을 때, 기존의 인식 알고리즘은 속도, 인식률 등의 문제가 존재한다. 그리고 실험에 사용한 보드의 성능 문제로 인한 영향도 있다. 각각의 알고리즘을 사용하여 비교한 결과는 아래 <표.1>과 같다.

표 1. 각각의 알고리즘별 객체 인식 결과

구분	평균 검출 속도 (sec)	thresh old 값	정상 인식	비정상 인식	인식 률
SIFT	4.8	1	50	0	100%
	2.4	60	50	0	100%
SURF	3.9	1	50	0	100%
	1.5	60	50	0	100%
FAST	3.6	1	29	21	58%
	1.2	60	39	11	78%
S.F	2.8	1	48	2	96%
	0.4	60	45	5	90%

IV. 결 론

이 연구는 라즈베리파이3를 사용한 환경에서 객체를 실시간으로 검출하는 것에 목적을 두고 연구하였다. 먼저 객체의 특징점을 찾아내어 객체 인식이 가능한 부분을 파악하고, 이를 모바일 환경에서 적용 가능하도록 실시간 객체 검출 및 인지하는데 중점을 두었다.

향후 과제로는 각각의 개체 별의 특징을 분석하여 개체가 무엇(사람, 차량, 차선, 물체)인지 분석하는 알고리즘에 대해 연구하여 무인 이동체에 필요한 기술을 보완하고자 한다.

감사의 글

이 논문은 2016학년도 동의대학교 연구년 지원과 미래창조과학부 및 정보통신기술진흥센터의 Grand ICT연구센터지원사업의 연구결과로 수행되었음.

(IITP-2016-R71181610050001002)

참고문헌

- [1] 송호범, 조재승, 황일규, "스마트폰으로 제어되는 다중 센서를 내장한 무인 순찰 이동체의 구현", 2013년도 한국정보기술학회 논문지 제11권 제12호, 2013.12, pp 1-11
- [2] 김대환, 박금춘, 김신덕, "모바일 환경 응용을 위한 코너 특징점 기반의 회전 객체 검출", 한국컴퓨터정보학회 학술발표논문집 21권 2호, 213.7, pp23-26
- [3] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," Int'l J. Computer Vision ,Vol. 60, no. 2, pages. 91-110, 2004.
- [4] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," European Conference on Computer Vision, Vol. 3951, pages. 404-417, 2006
- [5] Jianbo Shi and Carlo Tomasi. "Good Features to Track". IEEE Conference on Computer Vision and Pattern Recognition, pages 593-600, 1994.