

# 객체 면적중심 기반의 색 추적 시스템 개발

지유나\*, 강유경\*\*, 이현\*

\*신문대학교 컴퓨터공학과

\*\*신문대학교 IT교육학부

e-mail: [irish0308@nate.com](mailto:irish0308@nate.com), [aquamint99@naver.com](mailto:aquamint99@naver.com),

[mahyun91@sunmoon.ac.kr](mailto:mahyun91@sunmoon.ac.kr)

## A Development of a Color Tracking System based on the Central Value of the Object

Yu-na Ji\*, Yu-Kyung Kang\*\*, Hyun Lee\*

\*Dept of Computer Science and Engineering, Sun Moon University

\*\*Division of Information Technology Education, Sun Moon University

### 요 약

최근 무인로봇 시장이 커지면서 국방관련, 민간, 상업적으로 무인로봇의 수요가 증가하고 있다. 이는 사람이 직접 접근할 수 없는 재난발생지역, 환경오염지역, 방사능위험지역, 국가 간 분쟁지역 등에 무인항공기나 무인지상로봇을 투입하여 실시간 원격으로 재난피해 정도를 관찰하고 인명구조작업, 생존자 위치파악, 국경지대 순찰 등에 효과적인 시스템으로 활용할 수 있다. 하지만, 제한적인 환경에서 가지고 있는 부품들이 기능을 제대로 동작하지 않을 때, 하나의 센서 또는 부품을 활용한 위치추적, 원격제어 및 자율주행 등이 가능해야 한다. 따라서 본 논문에서는 1개의 이미지 센서를 통해, 객체의 색을 인식하고 추적하는 시스템을 개발하고자 한다. 특히, 기존의 거리감지 센서 기반의 색 객체 추적 알고리즘에 비교하여, 면적중심 값의 변화량에 따라 객체추적의 반응속도를 증가시켜 쉽게 객체를 추적할 수 있는 시스템을 개발하고자 한다.

### 1. 서론

최근 무인로봇 시장이 커지면서 국방관련, 민간, 상업적으로 무인로봇의 수요가 증가하고 있다. 이는 사람이 직접 접근할 수 없는 재난발생지역, 환경오염지역, 방사능위험지역과 이스라엘, 아프가니스탄, 시리아와 같은 국가 간 분쟁지역 등에 무인항공기(UAV)와 무인지상로봇(UGV)을 투입하여 실시간 원격으로 재난피해 정도를 관찰하고 인명구조작업, 생존자 위치파악, 국경지대 순찰 등에 효과적인 시스템으로 활용할 수 있다. 하지만, 제한적인 환경에서 무인항공기나 무인지상로봇이 임무수행 중, 가지고 있는 부품들이 기능을 제대로 동작하지 않는 경우가 발생하고 있다[1-3]. 따라서 하나의 센서 또는 부품을 활용한 위치추적, 원격제어 및 자율주행 등이 가능해야 한다. 특히, 소형, 소중량 및 저용량 배터리를 가지고 있는 무인로봇의 경우, 크기가 크거나 전력소모가 큰 센서는 사용하지 않고, 카메라 기능을 활용한 영상센서나, 음파로 거리측정이 가능한 초음파 센서를 활용해 위치추적 시스템에 사용하고 있다. 예를 들어, 영상센서와 거리감지 센서를 이용해 무인로봇과 물체와의 상대적 거리가 생겼을 때, 그 거리를 0으로 만들고 물체와 거리를 일정한 고도로 유지하는 것을 목표로 하여, 초음파 센서를 이용해 상대적 거리를 수신 받아 객체를 추적하는 방법을 이용하고 있다. 하지만, 거리 감지 센서를 사용한 경우에는 카메라가 물체를 발견하여 추적하는 반응 속도는 빠르나, 물체가 위치한 곳으로

이동하는 반응 속도는 4~5초 정도 지난 후에 움직이는 것을 발견 할 수 있었다[4-6]. 따라서 본 논문에서는 거리감지 센서를 이용하는 대신에 면적중심 값을 계산하여 면적중심 값의 변화량에 따라 객체 추적의 반응속도를 증가시키는 알고리즘을 개발하고자 한다. 또한 객체 면적 중심 기반의 색 추적시스템을 개발하여 반응속도를 증가시키고, 기존 알고리즘 기반의 시스템과 비교분석하고자 한다.

본 논문은 다음과 같이 구성하고자 한다. 2장에서는 시스템 개발을 위한 테스트 기반으로 사용한 무인지상로봇에 대한 설명을 하고 3장에서는 면적 중심 기반의 색 객체 추적 알고리즘을 제안하고자 한다. 4장에서는 거리감지 센서 기반의 알고리즘과 반응속도를 비교분석하고, 마지막 5장에서 결론을 맺는다.

### 2. 시스템 개발에 필요한 무인로봇 테스트베드



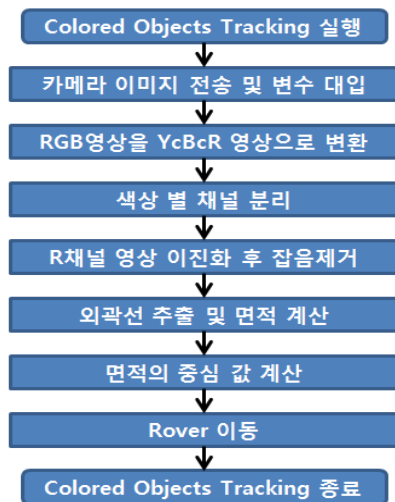
(그림 1) 무인지상로봇 테스트베드로 사용된 Rover

시스템 개발을 위한 테스트 기반으로 사용한 무인지상 로봇은 그림 1과 같이, 현재 민간용으로 판매중인 Rover I-spy tank(이하 Rover라 칭함)을 사용하였다[7]. 몸체가 작고 가벼운 장난감 용도로 출시되었으며, 케도형식의 바퀴를 장착하여 기본적으로 이동, 방향전환, 적외선기능을 탑재한 1개의 카메라가 달려 있다[8]. Rover의 카메라는 320\*240의 해상도를 지원하고 초당 25 프레임으로 영상을 촬영하고, 동영상은 mp4, 정지영상은 JPEG로 저장한다.

현재 이용되는 Rover의 경우 아이폰 App이나 안드로이드 기반에서도 원격제어 조작이 가능하게 구현되어있다. 하지만, 이는 모두 사람의 손을 이용하여 Rover를 움직이게 해야 하고 컨트롤러와 Rover가 15~20m이상 멀어지면 WiFi 신호가 잡히지 않아 원격제어의 어려움이 있다. 이와 같은, Rover의 제한점은 본 논문에서 설정한 제약적인 조건에서의 무인로봇 시스템을 테스트하기에 적합하여, 본 논문의 시스템 테스트베드로 선정하고 실험을 실시하였다. 특히, 본 논문에서는 추가적인 카메라 및 거리센서, 자이로 센서의 설치 없이, 원격제어가 아닌, 자율주행 방식으로 Rover가 빨간색을 인지하여 위치한 곳으로 이동하는 시스템을 개발하고자 하였다.

### 3. 객체 면적 중심의 색 추적 시스템

Rover에 추가적인 센서를 장착하지 않고 색 객체추적을 구현하기 위해서, 1) 빨간색 물체의 면적 크기를 계산하여 전진과 후진을 하도록 하고, 2) 면적의 중심 값을 측정하여 카메라 화면의 중앙값에서 왼쪽 혹은 오른쪽으로 움직이게 하고, 3) Rover가 카메라 화면의 중심 값을 맞추기 위해 회전하는 방식으로 구현하였다. 그림 2는 면적 중심 기반의 색 객체추적 알고리즘을 보여주고 있다.



(그림 2) 면적중심 기반의 색 객체추적 알고리즘

#### 3.1. 카메라 이미지 전송

Rover의 카메라를 통해 촬영된 이미지를 전송하고 전송된 이미지를 IplImage 변수에 대입한다. 이 때, 사용된 소스코드의 예는 그림 3과 같다.

```

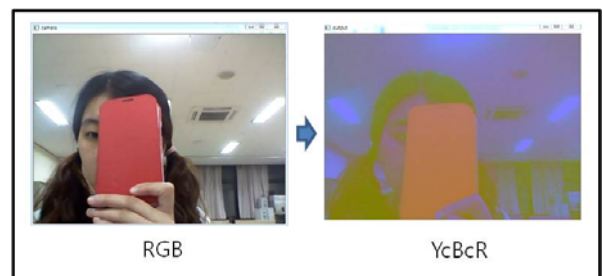
byte [] b = communicator.imageBuffer;
IplImage image = null;

while(true)
{
    b = communicator.GetImageBuffer();
    image = cvDecodeImage(cvMat(1, b.length,
                               CV_8UC1, new BytePointer(b)));
}
    
```

(그림 3) 이미지 전송에 사용된 소스코드의 예

#### 3.2. RGB 영상을 YcBcR 영상으로 변환 및 채널 분리

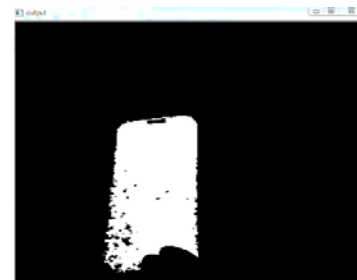
그림 4와 같이, RGB 영상을 YcBcR 영상으로 변환을 한다. 이는 영상처리 자체가 빛에 의한 제약이 많은데, 빛의 영향을 덜 받는 YcBcR영상으로 바꾸어 사용한다면 밝기 성분과 색상성분을 구분하기 쉽고, 색 추출이 잘 되기 때문이다. 변환 후에는 색상별로 채널을 분리한다.



(그림 4) RGB 영상을 YcBcR 영상으로 변환

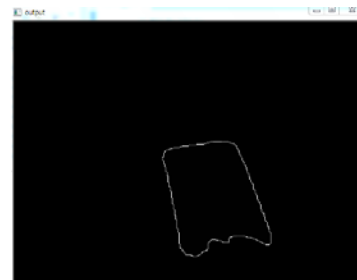
#### 3.3. R채널 영상 이진화 및 잡음제거

R채널 영상을 이진화한 후 색상을 넓게 잡기 위해 옅은 부분과 볼록한 부분의 잡음을 제거한다[9],[10]. 그림 5는 R채널 이진화 및 잡음제거 후의 화면을 보여주고 있다.



(그림 5) R채널 이진화 및 잡음제거

#### 3.4. 외각선 추출 및 면적계산



(그림 6) 외각선 추출 및 면적계산

그림 6과 같이, R채널로 둘러싸인 부분의 외곽선을 구하고 그린 뒤 면적을 계산한다. 이는 거리감지 센서가 장착되어있지 않은 상태라 이동하기 위한 조건으로, 색상의 면적으로 이동조건을 주기 위해 외곽선과 외곽선으로 둘러싸인 면적을 계산하는 방식이다. 그림 7은 소스코드의 예를 보여주고 있다.

```
// 외곽선 구하기
cvFindContours(red,storage,contours,
               Loader.sizeof(CvContour.class),
               CV_RETR_EXTERNAL,
               CV_CHAIN_APPROX_SIMPLE,
               cvPoint(0,0));

// 외곽선 그리기
if(contours != null)
    cvDrawContours(red,contours,cvScalarAll(255),
                  cvScalarAll(128),100,1.8,cvPoint(0,0));

// 외곽선으로 둘러싸인 면적 계산
double size = cvContourArea(contours, CV_WHOLE_SEQ, 0);

(그림 7) 외각선 추출 및 면적계산의 소스코드 예
```

### 3.5. 면적중심 값 계산

그림 8의 소스코드 예와 같이, 면적의 중심 좌표 값을 계산하여 카메라의 중심 좌표 값보다 왼쪽에 있는 경우 왼쪽으로 회전시키고, 오른쪽에 있는 경우 오른쪽으로 회전시켜 Rover를 이동시킨다. 그림9는 Rover 실행에 대한 결과 화면을 보여주고 있다.

```
// 물체의 중심 좌표값 계산
CvMoments moments = new CvMoments();
cvMoments(red, moments, 1);

double moment10 = cvGetSpatialMoment(moments,1,0);
double moment01 = cvGetSpatialMoment(moments,0,1);
double area = cvGetCentralMoment(moments,0,0);

int posX = 0;
int posY = 0;

posX = (int)(moment10/area);
posY = (int)(moment01/area);

// 빨간색 면적의 중심을 표현하기 위한 원
cvCircle(image,cvPoint(posX,posY),5,
         cvScalar(255,0,0,0),2,8,0);
```

(그림 8) 면적중심 값 계산의 소스코드 예






(그림 9) 면적중심 값 계산 실제 적용의 예

## 4. 비교분석

본 논문에서 제안한 면적중심 기반의 색 객체추적 알고리즘의 우수성을 보여주기 위해, 빨간 원의 크기를 인식하는 반응속도를 기본 거리감지 센서를 이용하는 경우와 비교분석하였다. 표 1과 같이, 거리 감지 센서를 이용한 경우에는 카메라 모듈 자체의 반응 속도는 빨랐으나 기기로 정보가 전송되는 시간이 오래 걸려 물체를 발견하고서 이동하기까지의 반응시간은 색의 면적이 작을수록 오래 걸렸다. 그에 비해 본 논문에서 구현한 면적 중심 방식은 거리가 면적에 대한 계산 속도가 빠르기 때문에 동일한 시간에 반응하여 거리 감지 센서를 사용한 경우보다 더 빠르게 이동하는 것을 알 수 있었다.

(표 1) 면적 크기에 따른 반응시간 비교분석

반응시간 면적	거리 감지 센서	면적 중심
	4~6초	1초
	3초	1초
	1초	1초

## 5. 결론

지금까지 본 논문에서는 무인지상로봇(UGV) 테스트베드인 Rover를 이용하여 사용자의 조작 없이도 특정 색을 발견하고 특정 색을 지닌 물체가 위치한 장소로 이동하는 시스템 구현하였다. 특히, 이는 거리감지 센서를 동작시켜 Rover를 이동시키는 방식과 달리 면적중심 값을 계산하여 색 객체추적을 동작하도록 하였다. 또한 기본 거리감지 센서 기반의 알고리즘 반응속도보다 색 면적의 크기에 상관 없이 일괄적인 반응속도를 보여주었다.

계속적인 연구방향으로는 자율주행 기능을 탑재하여 원격제어 범위를 벗어나, 특별한 조작 없이 Rover가 동작하는데 있어, 장애물을 회피하면서 특정 색을 찾아내는 기능을 적용하고자 한다.

## 참고문헌

[1] R. S. Pressman, "Software Engineering A Practitioner's Approach", 6th Edition, McGraw Hill, New York, 2005.  
 [2] K. Celik, S. Chung, M. Clausman, and A. Somani,

- "Monocular vision SLAM for indoor aerial vehicles", IEEE Intl. Conf. on Intelligent Robots and Systems, pp. 1566-1573, 2009.
- [3] L. Matthies, A. Kelly, T. Litwin, and G. Tharp, "Obstacle Detection for Unmanned Ground Vehicles: A Progress Report", Springer, pp. 475-486, 2000.
- [4] Z. Zhou, T. Chen, D. Wu, and C. Yu, "Corridor navigation and obstacle distance estimation for monocular vision mobile robots", JDCTA: Intl. Journal of Digital Content Technology and its Applications, vol. 5, no. 3, pp.192-202, 2011.
- [5] S. Se, D. Lowe, and J. Little, "Vision-based mobile robot localization and mapping using scale invariant features", IEEE Intl. Conf. on Robotics and Automation, pp.2051-2058, 2001.
- [6] H. Zhou, Y. Yuan, and C. Shi, "Object Tracking using SIFT features and mean shift", Computer Vision and Image Understang, vol. 113, no. 3, pp. 345-352, 2009.
- [7] "Rover Open-Source Download Site", the resource available at <http://sourceforge.net/>
- [8] "Open CV Library", the resource available at <http://opencv.org>
- [9] J. Canny, "A computational approach to edge detection", IEEE Transactions on. Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp.679-698, 1986.
- [10] N. Kiryati, Y. Eldar, and A.M. Bruckstein, "A probabilistic Hough transform", Pattern Recognition, vol. 24, no. 4, pp.303-316, 1991.