

## Depth 정보를 이용한 CamShift 추적 알고리즘의 성능 개선

주성욱<sup>1</sup> · 최한고<sup>2</sup>

상신브레이크<sup>1</sup>, 금오공과대학교 전자공학부<sup>2</sup>

### Performance Improvement of Camshift Tracking Algorithm Using Depth Information

Seong-UK Joo<sup>1</sup> · Han-Go Choi<sup>2</sup>

Sangsin Brake<sup>1</sup>, Department of Electronic Engineering, Kumoh National Institute of Technology<sup>2\*</sup>

**요약** 본 연구에서는 이동 물체의 색상이 배경 내 색상과 동일하거나 유사한 색상이 존재하는 경우 컬러기반에서 효과적으로 이동 물체의 추적 방법을 다루고 있다. 대표적인 컬러 기반 추적방법인 CamShift 알고리즘은 배경 영상에 이동물체의 색상이 존재하는 경우 불안정한 추적을 보여주고 있다. 이러한 단점을 극복하기 위해 본 논문에서는 물체의 Depth 정보를 병합한 CamShift 알고리즘을 제안하고 있다. Depth 정보 영상의 모든 픽셀의 거리정보를 측정하는 Kinect 장치로부터 구할 수 있다. 실험결과 이동물체의 거리정보를 병합시킨 제안된 추적 방법은 기존 CamShift 알고리즘의 불안정한 추적기능을 보완하였고, CamShift 알고리즘만 사용한 경우와 비교해 볼 때 추적성능을 향상시켰다.

• 주제어 : 물체 추적, Camshift 알고리즘, Depth 정보, Kinect 장치

**Abstract** This study deals with a color-based tracking method of a moving object effectively in case that the color of the moving object is same as or similar to that of background. The CamShift algorithm, which is the representative color-based tracking method, shows unstable tracking when the color of moving objects exists in the background. In order to overcome the drawback, this paper proposes the CamShift algorithm merged with depth information of the object. Depth information can be obtained from Kinect device which measures the distance information of all pixels in an image. Experimental result shows that the proposed tracking method, the Camshift merged with depth information of the tracking object, makes up for the unstable tracking of the existing CamShift algorithm and also shows improved tracking performance in comparison with only CamShift algorithm.

• Key Words : Object Tacking, Camshift Algorithm, Depth Information, Kinect Device

Received 11 October 2017, Revised 01 November 2017, Accepted 02 December 2017

\* Corresponding Author Han Go Choi, Department of Electronic Engineering, Kumoh National Institute of Technology, 61, Daehak-ro, Gumi, Gyeongbuk, Korea. E-mail: hgchoi@kumoh.ac.kr

## I. 서론

컴퓨터 비전과 로봇 응용 시스템에서 카메라를 통해 획득된 영상 정보를 이용하여 물체를 추적하는 연구가 다양하게 진행되고 있다. 이동물체 추적에 관한 연구는 주요 산업, 의료, 군사 분야에 널리 활용되고 있고, 특히 최근에는 로봇틱스 및 위성관련 연구에서 중요한 부분을 차지하고 있다[1]. 이러한 사실을 반영하듯 컴퓨터 비전 관련 분야에서는 물체를 추적하기 위해 다양한 알고리즘이 개발되어 왔다. 매 프레임마다 영상의 차이를 통해 물체를 추적하는 방법부터 추적물체의 형태를 이용하거나 물체의 색상을 이용하여 추적하는 등 여러 알고리즘이 개발되고 있다[2,3].

기존의 이동물체 추적 방법들은 대부분 컬러 영상을 사용하는데 컬러 기반의 이동물체 추적에서 가장 많이 사용되는 방법이 CamShift 알고리즘이다[4~6]. 그러나 CamShift 알고리즘은 이동물체와 동일한 색상을 갖고면서 이동물체 보다 큰 영상영역이 배경영상에 존재할 경우 이동물체가 이러한 배경영상을 지나칠 경우 기존 CamShift 알고리즘은 이동물체를 추적하다가 동일한 색상을 갖는 배경영상 영역에 머물게 되어 더 이상을 이동물체를 추적할 수 없는 근본적인 한계를 갖고 있다. 이를 해결하기 위해서는 컬러 정보 이외의 부가적인 정보를 활용한 방법이 유용한 해결책이 될 수 있는데, 대표적인 정보가 깊이 (Depth) 정보이다[7].

기본적으로 깊이 정보를 구하는 방법은 다수의 카메라 영상이나 단일 카메라의 여러 시점에서 영상을 획득한 후 각 픽셀의 시차 값을 계산하는 방식으로 이루어지는데, 이는 상당한 계산 시간을 요구하므로 물체의 추적과 같은 실시간 응용 분야에서는 적합하지 않다.

최근에는 Kinect의 등장으로 컬러 영상과 함께 깊이 영상도 실시간으로 획득하는 것이 가능해졌다[8]. 마이크로소프트사의 Depth 카메라인 Kinect는 2차원 정보를 획득하는 RGB 카메라와 함께 적외선 센서를 이용하여 측정 대상에 대해 3차원의 정보를 인식할 수 있는 카메라로, 이를 이용하여 영상 정보의 실시간 획득이 가능하게 되었다. 실시간 3차원 정보의 획득은 컴퓨터 비전 및 인간과 컴퓨터 상호작용, 3

차원 영상 재구성과 동작 인식 및 얼굴 인식 등 공학을 비롯한 의학 및 산업 전반에서 활용도가 높으며 이에 따라 다양한 연구가 진행되고 있다. 또한 Kinect를 이용하여 사용자의 움직임을 감지하여 특정 동작을 인지하고 마우스와 키보드를 이용한 제어가 아닌 모션을 이용한 제어가 가능해졌다[9].

본 연구에서는 기존 CamShift 추적 알고리즘의 한계를 극복하기 위해 Kinect의 Depth 정보를 병합한 CamShift 알고리즘 기반의 이동물체 추적을 제안함으로써 색상기반의 물체 추적 알고리즘의 문제점을 개선하고 좀 더 안정된 물체 추적을 구현하고자 한다. 제안된 병합 추적 방법을 구현하여 이동물체 추적 실험을 수행한 결과 기존 CamShift 알고리즘의 불안정한 추적기능을 보완하였으며, 깊이 정보를 사용하지 않은 CamShift 알고리즘과 비교해 볼 때 제안된 방법이 추적성능을 향상시킬 수 있음을 확인하였다.

## II. 제안된 이동 객체 추적 알고리즘

### 2.1 CamShift 알고리즘

객체를 추적하는 방법 중에 컬러 정보를 기반으로 추적하는 알고리즘인 CamShift 알고리즘은 원하는 영역의 컬러정보를 추출하고, 계속되는 영상에서 색상을 비교하여 물체의 중심, 크기, 방향을 함께 추적하는 알고리즘이다.

RGB색상에서 HSV컬러 모델로 변환된 영상에서 CamShift 알고리즘으로 추적을 위해 초기 탐색 윈도우의 크기와 위치를 선택하면, 먼저 객체의 위치와 크기를 참고하여 CamShift 알고리즘이 계산될 영역을 정하고 Hue(색) 영역 기반의 히스토그램을 구한다. 이때 Hue 값은 0~180의 범위를 갖는다. Hue 영역 기반의 히스토그램을 통해 컬러 확률분포가 구해지면 탐색 윈도우에서 객체의 중심을 찾고, 그 중심에 탐색 윈도우를 재위치 시키고 영역을 찾는다. 영역의 컬러 확률분포 스코어가 컬러 히스토그램 확률분포 스코어에 수렴할 경우에는 새로이 탐색 윈도우를 설정하고 수렴하지 않을 경우 탐색 윈도우에서 객체의 중심을 찾고 스코어가 수렴하는 영역을 찾는 일련의 과정들을 수렴할 때까지 반복하게 된다. 그림 1은

CamShift 알고리즘의 이동물체 추적 흐름도이다.

그림 2는 CamShift에 의한 물체 추적 결과이다. 그림 2(a)는 입력 영상인 RGB영상을 HSV 컬러 모델로 변환한 영상의 Hue 채널 영상이고, 그림 2(b)는 HSV 컬러영상에서 추적하려는 컬러의 범위를 지정한 Mask영상이다. 추적하고자 하는 물체인 빨간색 물체가 255인 영역에 포함됨을 알 수 있다. 그림 2(c)는 Hue 채널 영상과 히스토그램을 사용하여 역투영한 영상인 BackProject는 Mask와 AND 연산한 결과로 추적하려는 물체의 Hue 채널 색상과 히스토그램이 같은 부분은 255, 그렇지 않은 부분은 0으로 표시되었다. 그림 2(d)는 추적된 물체 위치를 원과 사각형으로 표시하였다[10].

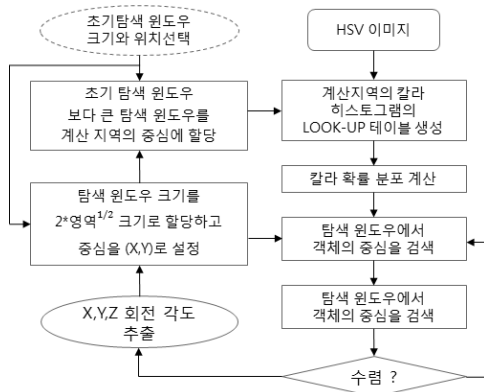


그림 1. CamShift 알고리즘 흐름도  
Fig. 1. Flow chart of Camshift algorithm

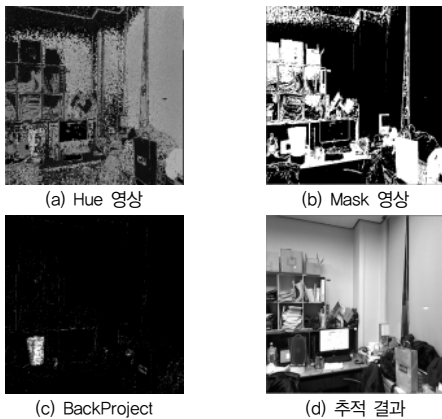


그림 2. Camshift 물체 추적 결과  
Fig. 2. Result of Camshift object-tracking

## 2.2 CamShift 알고리즘의 문제점

CamShift 알고리즘의 경우 이동물체의 히스토그램을 추출하여 색상기반으로 물체의 밀집도를 분석한다. 영상의 배경영역에서 이동물체의 히스토그램과 유사한 영상영역이 없는 경우 이동물체에 대해 안정된 추적을 수행한다. 그러나 배경 영상영역에 이동물체의 색상과 유사한 색상이 존재할 경우 이동물체가 동일 색상의 배경영역을 겹쳐서 통과하게 되면 추적이 이동물체가 아니라 배경영역에 멈추게 되는 경우가 발생한다.

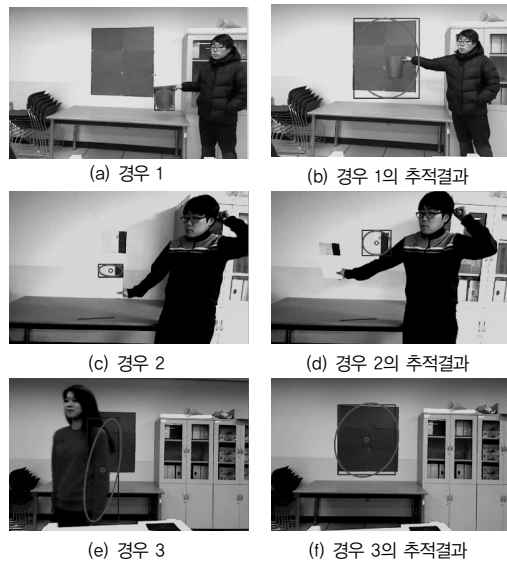


그림 3. CamShift 추적의 문제점  
Fig. 3. Drawback of Camshift tracking

그림 3은 CamShift 알고리즘 추적의 문제점을 3 경우에 대해 보여주고 있다. 그림 3 (a),(c)는 이동물체가 동일한 색상의 배경영역으로 부근으로 이동하는 경우로서 이동물체(그림에서 붉은 원으로 표시된 부분)가 정상적으로 추적하고 있는 모습을 보여주고 있다. 그림 (c)는 이동물체의 일부영역이 배경영역에 중첩된 경우로서 추적이 이동물체와 배경영역을 동시에 포함하고 있다. 그림 3 (b),(d),(f)는 (a),(c),(e)의 이동물체가 동일 색상의 배경영역 위로 통과할 때의 추적 결과로서 배경 영상영역이 이동물체의 영역보다 클 경우 추적이 배경 영상영역에 고정되므로 이동물체 추적이 더 이상 수행되지 않음을 확인할 수

있다. 그림 3과 같이 CamShift 알고리즘에서 선정된 이동물체 추적을 한번 놓치게 될 경우 더 이상 이동물체를 추적하기가 어렵다. 이를 해결하기 위해 본 연구에서는 이동물체와 배경물체와의 거리 정보를 각각 이용하여 동일 색상을 갖는 배경영역으로 인해 이동물체 추적을 수행할 수 없는 CamShift 알고리즘을 추적 성능을 개선하였다.

그림 4는 Camshift 추적에서의 개선 내용을 그림으로 설명하고 있다. 현재 프레임에서 추적 대상의 깊이 정보와 색상 정보를 모두 가지고 있다면 다음 프레임에서 동일 색상의 배경 혹은 장애물 영상이 있다 하더라도 추적 대상을 놓쳐버릴 확률은 매우 적다.

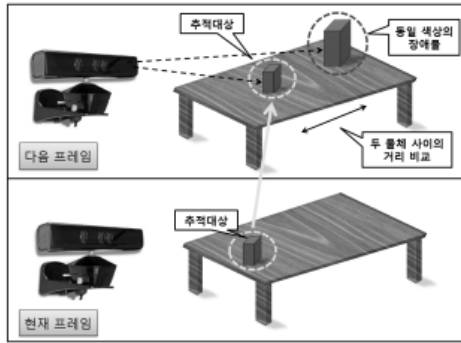


그림 4. 제안된 알고리즘 개념도

Fig. 4. Conceptual diagram of proposed algorithm

### 2.3 Depth 이미지 구현

거리 정보를 구하기 위해 Kinect 장치를 사용하였다. Kinect의 IR Emitter에서 송출된 적외선은 물체에 반사되어 IR Depth Sensor가 인식하고 그 정보를 이용한다. 한 픽셀을 표현하는데 필요한 Depth는 최소 16-bit 이고 이 중 상위 13-bit은 해당 픽셀의 Depth Value, 하위 3-bit은 Player Index 값이다. 따라서 하위 3-bit을 Shift 함으로써 Depth Value를 구할 수 있다. 그림 5는 Depth 값을 시각적으로 표현하기 위해서 아래의 수식을 사용하여 그림으로 표현하였다.  $M(i)$ 는 Depth 값이고 4.095[m]은 Kinect Xbox의 최대 인식거리이며  $X(i)$ 는 0~255의 Gray Scale 값이다.  $X(i)$  값을 RGB 각 픽셀에 적용시켜 Gray Scale 영상의 Depth 이미지를 구현하였다. 거리가 멀수록 Graylevel 값이 크다.



그림 5. Depth 표현 영상

Fig. 5. Image of depth representation

$$\sum_{i=0}^{320 \times 240} X(i) = 255 - (255 \times M(i) \div 4095) \quad (1)$$

$$R_i = G_i = B_i = X_i$$

그림 6은 Gray Scale 히스토그램과 최대 빈도의 히스토그램을 보여주고 있다. 초기 탐색 윈도우의 위치와 크기를 지정할 때 대상 물체의 Color 히스토그램은 그 영역에서 추출된 값이 고정되어 그 색상과 유사한 물체를 추적하는 것이다. 그러나 Depth 정보는 물체의 움직임에 따라 Depth 값이 달라지기 때문에 Gray Scale 히스토그램 또한 고정된 값이 아니라서 매 프레임마다 Gray Scale 히스토그램을 갱신한다. 그림 6의 왼쪽 사진이 히스토그램의 갱신을 보여주고 있다. 그림에서 히스토그램 형태는 유사하지만 차이를 확인할 수 있으며 이러한 차이들이 노이즈가 되어 Gray Scale 히스토그램 정보를 영상추적에 바로 사용하기에는 어려움이 있다. 따라서 이러한 노이즈 성분을 제거하여야 하는데 Gray Scale 히스토그램 정보 중 최대 빈도수의 성분만 추출하고 나머지 빈도수가 낮은 성분들은 제거시켜 새로운 Gray Scale 히스토그램을 선정하였다. 그림 6의 오른쪽 쪽 그림에서 제거된 히스토그램을 보여주고 있다.

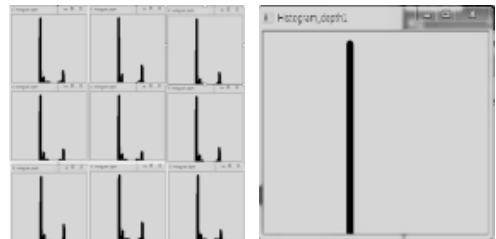


그림 6. Depth 히스토그램

Fig. 6. Depth histogram

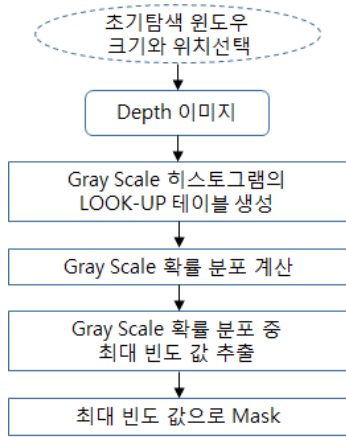


그림 7. Depth 노이즈 제거 흐름도

Fig. 7. Flow chart of depth noise cancellation

그림 7은 Depth 정보를 추출하여 가변되는 Gray Scale 히스토그램에서 노이즈 성분을 제거하여 최대 빈도수 값만 이용하기 위한 방법의 흐름도이다. 초기의 탐색 윈도우의 크기와 위치를 선택한 이후 Depth 이미지를 추출하고 추출한 Depth 이미지의 히스토그램을 바탕으로 확률 분포를 계산한다. 확률 분포의 최대 빈도 값을 추출하여 해당 값으로 Mask 작업을 수행한다.

### 2.4 CamShift 알고리즘과 Depth 정보 병합

CamShift 알고리즘 자체가 컬러 기반 추적 방법이기에 때문에 영상 내에 추적 물체와 유사한 색상 정보를 가진 물체 또는 배경이 존재한다면 정상적인 추적기능을 수행할 수 없음을 확인하였다. 이러한 CamShift 알고리즘의 단점을 보완하기 위하여 2.3에서 기술된 Depth 기반의 Gray Scale 히스토그램과 기존의 CamShift 알고리즘을 병합하여 주변 또는 배경 내에 추적영상과 비슷한 컬러 영역이 존재하더라도 이동물체를 안정적으로 추적할 수 있다.

그림 8은 Backproject 병합 결과를 보여주고 있다. 그림 8(a)는 입력 영상인 RGB 영상에서 추적된 물체의 위치를 사각형(혹은 사각형 내 원)으로 표시하였다. 그림 8(b)는 Color 영역의 Backproject 영상이다. 추적하려는 물체인 빨간색 물체와 배경에서 빨간색 영역 부분이 모두 255 Gray Scale 값을 갖는 영역이라는 것을 알 수 있다. 이러한 경우 추적물체가

유사색상에 가까이 갔을 때 추적범위가 넓어지거나 추적대상이 바뀔 수 있다.

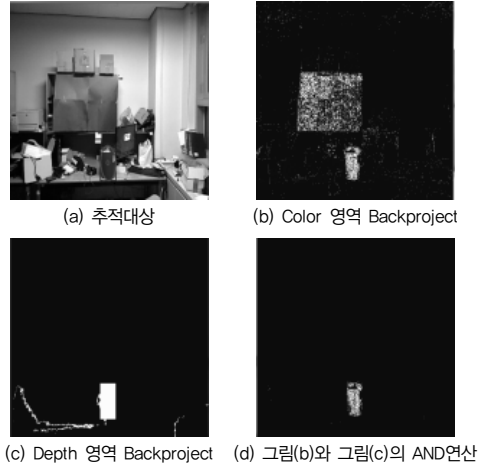


그림 8. 제안된 병합 알고리즘 결과

Fig. 8. Result of proposed merged algorithm

그림 8(c)는 Depth 영역의 Backproject 영상으로서 Gray Scale 히스토그램에서 최대빈도수의 값을 추출하여 Backproject를 추출한 영상이다. 그림 8(d)는 Color 영역의 Backproject 영상과 Depth 영역의 Backproject 영상을 AND 연산을 수행한 영상이다. AND 연산을 통하여 그림 8(b)에서의 유사색상과 그림 8(c)의 유사거리를 제외한 오로지 추적 대상만의 Backproject를 추출할 수 있다. 그림 8(d)의 Backproject 영상으로 추적을 수행하면 유사색상 또는 유사거리에서도 강한 추적 기능을 수행할 수 있음을 확인할 수 있다.

## III. 실험 및 결과

본 연구에서 제안한 방법은 Intel(R) Core i7-3770 3.4GHz 사양의 개인용 컴퓨터에 Visual Studio 2010을 사용하여 구현하였다. 카메라는 Microsoft사의 Xbox360 Kinect v1을 사용하였다. Kinect RGB 카메라는 CMOS 카메라로 640×480의 해상도를 지원하며, Kinect Depth 카메라는 CMOS로 320×240의 해상도를 지원한다.

본 연구에서는 영상 크기를 640×480 해상도로 실

험하였으며, 영상 획득 및 디스플레이, 객체 검출, 그리고 기본적인 추적 알고리즘은 Intel 사의 영상처리 오픈소스인 OpenCV 라이브러리를 사용하였다. OpenCV 라이브러리는 PC 환경에 최적화된 코드로서 PC 기반에서 영상처리 및 추적을 하는데 있어 적합하며, CamShift 알고리즘의 경우 OpenCV에서 함수로 제공하고 있다.

기존 CamShift 알고리즘은 초기 히스토그램 컬러 정보만을 이용하여 추적하므로 그림 3과 같이 불안정한 추적을 보였다. 제안된 방법은 기존 CamShift에 Depth 정보를 병합한 알고리즘으로서 개선된 CamShift 추적 알고리즘의 검증에 위해 3가지 경우에 대해 실험을 수행하여 추적 성능을 확인하였다. 첫 번째 실험은 추적 물체의 색상 영역이 배경 대상 영역보다 작은 경우이며, 두 번째 실험은 추적 영역이 배경의 대상영역보다 큰 경우, 그리고 세 번째 실험은 추적 영상이 단일 색상이 아닌 여러 개로 구성된 경우에 대한 실험을 각각 수행하였다.

### 3.1 추적 영역이 배경의 대상영역보다 작은 경우

그림 9는 붉은 색상의 사각형 이동물체(원으로 표시)가 배경영상의 중앙에 이동물체보다 큰 동일한 색상의 사각형이 존재하는 경우에 개선된 CamShift 알고리즘의 영상 추적 결과를 보여주고 있다.

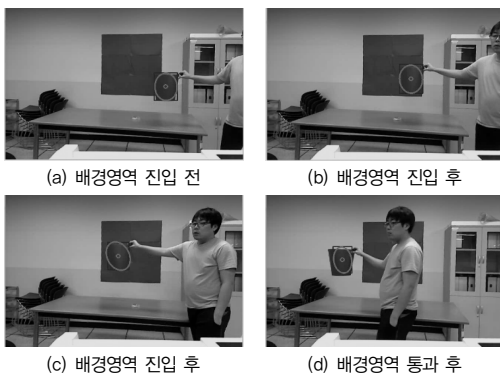


그림 9. 개선된 CamShift 알고리즘의 실험 1

Fig. 9. 1st experiment of improved CamShift algorithm

그림 (a)는 이동물체가 배경 대상영역에 진입하기 직전의 모습이며, 그림 (b)와 (c)는 이동물체가 배경 영역에 진입한 후의 모습이며, 그림 (d)는 이동물체

가 배경영역을 통과한 후의 영상을 각각 보여주고 있다. 그림에서와 같이 동일 색상의 배경영상에 관계 없이 이동물체를 안정적으로 추적함을 알 수 있다. 이러한 추적결과는 그림 3(a),(b) 경우 1에서 이동물체의 추적이 배경영역을 벗어나지 못하는 문제점을 해결함을 확인할 수 있다.

### 3.2 추적 영역이 배경의 대상영역보다 큰 경우

그림 10는 그림 9과는 달리 붉은 색상의 이동 물체 영역이 동일 색상을 갖는 배경영역보다 다소 큰 경우에 대한 추적 결과를 보여주고 있다. 추적대상을 사람으로 설정하였으며 배경영역을 가로 질러 지나갈 때 개선된 CamShift 알고리즘은 안정적인 추적을 수행함을 보여주고 있다.

그림 9에서는 물체를 이동 대상으로 실험하였지만 사람을 추적대상으로 할 때는 물체를 대상으로 할 때 보다 움직임에 따라 추적 범위가 달라지기도 하지만 그림에서와 같이 유사 배경을 가로질러 지나가도 추적 대상이 변하지 않고 초기에 설정한 추적 대상을 계속 유지하여 따라감을 알 수 있다. 이러한 추적결과는 그림 3(e),(f) 경우 3에서 사람을 계속 추적하지 못하고 배경영역에서 벗어나지 못하는 문제점을 해결함을 확인할 수 있다.

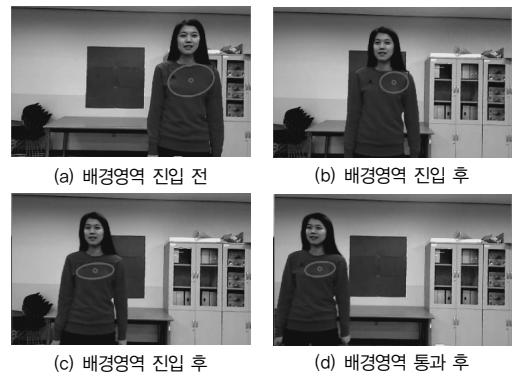


그림 10. 개선된 CamShift 알고리즘의 실험 2

Fig. 10. 2nd experiment of improved CamShift algorithm

### 3.3 추적 영역이 다양한 색상인 경우

이전의 실험은 추적 대상이 단일 색상인 경우에 대한 실험인 반면 그림 11에서는 추적 대상이 다양한 색상을 갖는 물체에 대한 실험을 수행하였다. 그

림에서와 같이 3개의 서로 다른 색상을 갖는 이동물체가 유사한 색상이 배경영상에 존재할 때 개선된 CamShift 알고리즘의 성능을 검증하였다.

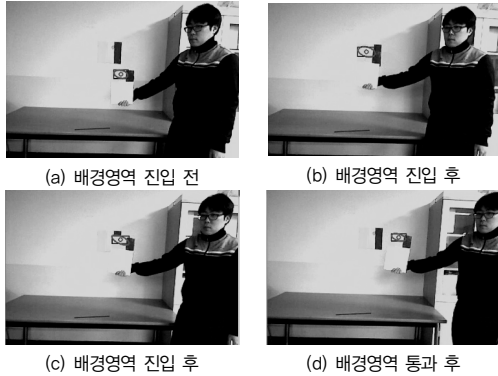


그림 11. 개선된 CamShift 알고리즘의 실험 3

Fig. 11. 3rd experiment of improved CamShift algorithm

위 실험과 마찬가지로 다양한 색상의 이동물체 추적에서도 안정된 모습을 보여주고 있다. 즉, 유사 배경을 가로질러 지나가도 추적 대상이 변하지 않고 초기에 설정한 추적 대상을 계속 유지하여 지속적으로 따라감을 알 수 있다. 이러한 추적결과는 그림 3(c),(d) 경우 2에서 불안정한 추적을 극복할 수 있음을 확인할 수 있다.

#### IV. 결론

본 연구는 기존 CamShift 추적 알고리즘에 Depth 정보를 적용시켜 이동물체의 추적 성능을 향상시켰다. 기존의 CamShift 알고리즘은 이동물체의 색상과 유사한 색상이 배경영역에 존재할 때 이동물체가 동일 색상의 배경 주변으로 이동하거나 중첩될 때 추적 영역이 이동물체가 아니라 배경영역까지 포함하거나 배경영역에서 벗어나지 못하는 문제점을 갖고 있다. 이를 보완하기 위해 기존 CamShift 알고리즘에 Depth 정보를 적용하였다. Depth 이미지의 Gray Scale 히스토그램에서 최대 빈도수를 추출 후 추적 대상에 마스크 함으로서 유사 색상의 배경에 대하여 이동물체 영역을 구분할 수 있게 하여 안정된 추적이 가능하게 되었다. 따라서 기존 CamShift 알고리

즘과의 추적성능 비교를 통해 이동물체를 추적할 때 불안정한 환경에 대해 개선됨으로써 더 안정된 추적이 수행됨을 실험으로 증명하였다. 기존 영상처리 알고리즘들은 영상 정보만을 이용하는데 반해 Kinect의 Depth정보를 통하여 Kinect 활용 방안을 이동물체 추적으로 확대시켰다.

본 연구는 Kinect Depth 정보를 이용한 고정 카메라에 대한 영상추적이지만 세밀한 객체 검출 및 동적 영상추적을 위한 선행 연구에 도움이 될 것이라 생각된다. 향후에는 제안된 방법을 사용하여 다중 물체 추적과 이동 감시 카메라에 적용시켜 물체의 추적 범위를 확대시키는 연구가 필요하다.

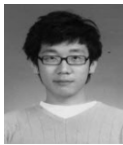
#### REFERENCES

- [1] Y.J. Kim and L.B.Lee, "Moving Object Detection using Single Active Camera", The Korean Institute of Information Scientists and Engineers, Vol 33, No. 2(B), Korea, October 2006.
- [2] J.H Kim and H.G. Choi, "A Moving Object Tracking using Color and OpticalFlow Information", Institute of Signal Processing and Systems, vol. 15, no. 4, pp. 112-118, Korea, October 2014.
- [3] W.J. Lee, Y.S.Kim, S.L. Lee and H.G. Choi, "Real-time detection and tracking of moving objects based on DSP" Institute of Signal Processing and Systems, vol. 11, no. 4, pp. 263-269, Korea, October 2010.
- [4] 김경국, "CAMSHIFT 알고리즘을 이용한 모바일 로봇의 객체추적시스템", 공운대학교 대학원, 석사 학위 논문 2009.
- [5] 김주현, "컬러 및 광류 정보를 이용한 이동물체 추적", 금오공과대학교 대학원, 석사 학위 논문 2013.
- [6] 김영균, "GP 기반 컬러 모델 생성과 개선된 CAMSHIFT 알고리즘을 통한 이동 물체 추적에 관한 연구", 서강대학교 대학원, 석사 학위 논문, 2012.
- [7] 박찬수, "깊이 카메라를 이용한 인체의 부피측정", 연세대학교 대학원, 석사 학위 논문, 2014.
- [8] H.H. Park, J.Y.Choi, J.I. Park and K.S. Moon, "A Study on Hand Region Detection for Kinect-Based Hand Shape Recognition", Journal of

Broadcast Engineers, Vol 18. No. 3, Korea, May 2013.

- [9] C.S. Kim, "Human body volume measurement by using depth camera", Yonsei University Graduate School Master's Thesis, February 2014.
- [10] D.G. Kim, "OpenCV computer vision programming", Kame, 2012.

### 저자소개



**주성욱(Seong-Uk Joo)**

2013년 금오공과대학교 전자공학부(학사)  
2015년 금오공과대학교 전자공학과(석사)  
2015년~현재 상신브레이크(주)  
※관심분야: 디지털신호처리, 영상처리



**최한고(Han-Go Choi)**

1979년 경북대학교 전자공학과(학사)  
1988년 플로리다대학교 전기전자공학과(석사)  
1992년 플로리다대학교 전기전자공학과(박사)  
1979~1986년 국방과학연구소 연구원  
2001년 MIT Research Scientist  
1993년~현재 금오공과대학교 전자공학부 교수  
※관심분야: 신호및영상처리, 생체신호처리, DSP 응용