

CAMShift를 이용한 PTZ 카메라 추적

정회원 장 일 식*, 안 태 기**, 박 광 영***, 종신회원 박 구 만****°

PTZ Camera Tracking Using CAMShift

Il-sik Chang*, Tae-ki An**, Kwang-young Park *** *Regular Members*,
Goo-man Park****° *Lifelong Member*

요 약

본 논문에서는 PTZ(Pan, Tilt, Zoom) 카메라를 이용한 위치 추적을 제안하였다. 추적해야 할 객체가 정해지면 카메라가 움직이는 동안에도 추적이 가능한 CAMShift 추적 알고리즘을 이용하여 객체를 실시간 추적한다. CAMShift 추적 알고리즘은 객체의 크기를 계산하므로 Zoom을 연동한 추적이 가능하다. Pan, Tilt의 위치는 구좌 표계를 이용하여 계산하였다. 이렇게 구해진 PTZ 위치는 PTZ 프로토콜을 이용하여 RS-485 통신으로 객체의 위치를 화면의 중심에 놓이게 함으로써 적합한 추적을 하게 된다. 본 논문에서 제안하는 알고리즘은 다양한 환경에서 추적에 우수한 성능을 확인 할 수 있었다.

Key Words : CAMShift Algorithm, PTZ Camera, PTZ protocol, RS-485

ABSTRACT

In this paper we proposed an object tracking system using PTZ camera. Once the target object is detected, the CAMshift tracking algorithm focuses it in realtime mode as the camera is moving accordingly. Since the CAMShift algorithm takes into account the object size, zoom related tracking is possible. We used the spherical coordinate to gain pan and tilt position. The position information is used to set the center of target object in the middle of the image by using the PTZ protocol and RS-485 interface. Our system showed excellent experimental results in various environments.

I. 서 론

감시 카메라의 종류를 3가지로 나눠 보면 고정 카메라, PTZ 카메라, 전 방위 카메라로 나눌 수 있다. 본 논문에서는 PTZ 카메라를 사용하였다. 고정 카메라를 사용할 경우 감시 감독의 범위가 넓어지게 되면 많은 수의 카메라를 사용해야 하기 때문에 비용이나 효율성 측면에서 단점이 있다. 이러한 점을 고려하였을 때 상하좌우의 회전이 가능한 PTZ 카메라를 채용함으로써 카메라 주변의 모든 영역 및 원거리, 근거리에 대한 감시감독을 가능하도록 하여 고정 카메라가

가지는 단점을 해결할 수 있다. Funahasahi et al.^[7]은 PTZ 카메라를 이용한 계층적 얼굴 추적을 제안하였다. 그러나 추가적으로 고정 카메라가 사용되어야 한다. 고정 카메라는 PTZ 카메라가 이동 중 얼굴의 위치를 갱신 해줌으로써 추적을 도와준다. Amnuaykanyanasin et al.^[8]은 두 개의 상호 협력적인 PTZ 카메라를 이용하여 얼굴을 추적하는 방식을 제안하였다. 하나는 고정으로 다른 하나는 Pan, Tilt를 수행한다. Ajmal Mian^[6]은 실시간 단일 PTZ 카메라를 이용하여 얼굴을 추적하지만 2차원 정보를 이용한 PTZ 추적을 한다. 대부분의 PTZ 추적 논문은 구좌표계를 사용하

※ 본 연구는 국토해양부 도시철도표준화 2단계 연구개발사업의 연구비지원(07도시철도표준화A01)에 의해 수행되었습니다.

* (주)하이트론씨스템즈 지능형시스템팀(ischang@hitron.co.kr), ** 한국철도기술연구원 도시철도표준화연구단,

*** (주)하이트론씨스템즈 채널사업팀, **** 서울산업대학교 매체공학과(gmpark@snut.ac.kr), (°: 교신저자)

논문번호 : KICS2009-12-610, 접수일자 : 2009년 12월 8일, 최종논문접수일자 : 2010년 2월 26일

지 않고 2차원 정보 혹은 높이 정보를 가지고 객체를 추적하게 된다. 본 논문에서는 구좌표계를 이용하여 실시간 추적이 가능한 PTZ 추적 시스템을 제안 한다. 객체 추적 방법으로 CAMShift^[3,5] 방법을 사용한다. CAMShift 방법은 추적 대상이 가지는 색 성분의 색 모델을 만들고 이와 가장 유사한 색을 가지는 영역을 검색하여 추적 대상의 중심위치로 수렴해 가는 방법이다. 색 정보를 이용하기 때문에 배경이 고정되지 않아도 추적이 가능하므로 유동적인 PTZ 카메라에서 실시간 추적이 가능하고, MEANShift^[4] 알고리즘의 문제인 객체의 크기가 변해도 추적이 가능하다. CAMShift는 객체의 크기 및 각도가 계산 되므로 Zoom을 연동한 추적이 가능하다. 화면의 움직여야 하는 좌표는 구좌표계를 이용하여 Pan, Tilt의 각도를 구할 수 있다. 이렇게 구해진 PTZ 값의 제어는 PTZ 카메라의 통신 프로토콜을 사용하여 Pan, Tilt, Zoom을 제어 한다. 본 논문은 총 7장으로 구성된다. 2장에서는 시스템 구성에 관한 소개, 3장에서는 CAMShift를 이용한 객체 추적에 대한 설명, 4장에서는 Pan, Tilt 위치 측정에 대해 제안방식을 설명하고, 5장에서는 Zoom 위치 제어를 설명 하였다. 6장에서는 실험 결과를 분석하고, 마지막으로 7장에서는 결론을 내렸다.

II. 시스템 구성

본 논문에서 구성하는 시스템은 그림 1과 같다.

시스템 구성을 크게 나누면 영상 추적을 위한 PC, PTZ 카메라, 통신 방식을 맞춰주는 RS-232 to RS-485 변환기, 실제 영상을 획득하고 위치를 맞추는 PTZ 카메라로 나눌 수 있다. PTZ 카메라 제어는 키보드를 사용하고 RS-485 방식의 통신 제어를 한다. RS-485 통신이란 디퍼런셜 전송 방식으로 동작하여 전기적 간섭에 강하며 약 1.2Km의 거리까지 통신이 가능한 반이중(Half Duplex) 방식의 통신 방법이다. 그림 2는 PC에서의 추적 시스템의 전체 순서도를 나타낸다. PTZ 위치 제어는 RS-485 통신을 사용하고 실

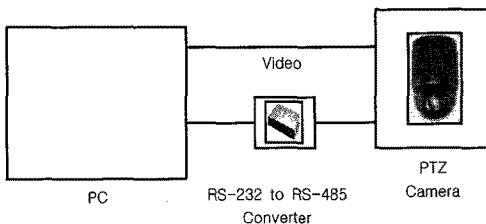


그림 1. 시스템 구성

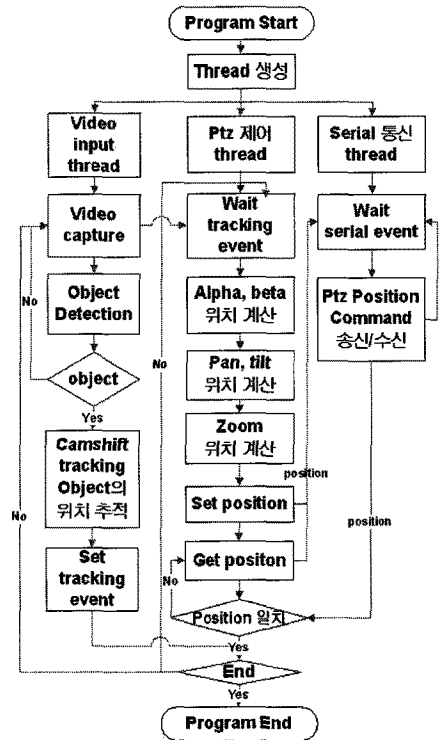


그림 2. 프로그램 순서도

제 PTZ 카메라가 해당 위치에 도달하기까지 시간이 많이 소모된다. 이러한 문제를 해결하기 위하여 3개의 스레드(Thread)를 두었다. 첫 번째 비디오 입력 및 CAMShift 알고리즘을 처리하는 스레드, 두 번째 PTZ 위치 제어하는 스레드, 마지막으로 시리얼 통신을 담당하는 스레드를 두어 각기 독립적이면서 이벤트 방식으로 처리함으로써 실시간 추적이 가능하게 하였다.

III. CAMShift를 이용한 객체 추적

CAMShift는 Continuously Adaptive Mean Shift Algorithm의 약자로 얼굴 추출을 위해 처음 사용되었다. CAMShift의 기반이 되는 MEANShift 알고리즘^[1,2]은 고정된 윈도우 크기를 가지고 색 정보에 기반하여 현재 프레임에서의 추적대상의 중심 위치로 수렴해 가는 방법이다. CAMShift는 기본적으로 Mean shift 알고리즘을 사용하면서 목표의 크기 및 각도를 같이 계산한다. PTZ 카메라로 추적할 때 객체가 가까이 오면 커지고 멀리가면 작아지므로 크기의 변화까지 추적할 수 있는 CAMShift가 적합하다. CAMShift는 다음과 같은 단계[3][5]로 나뉘어 적용될 수 있다.

- 1) 탐색 윈도우의 초기 위치 및 크기를 정한다.
- 2) 탐색 윈도우 내의 명암 분포 중심을 구한다.
- 3) Mean Shift 알고리즘을 반복한다.
- 4) 이어지는 프레임에서 탐색 윈도우의 위치와 크기를 3)에서 얻어진 명암 분포 중심으로 바꾼다.
- 5) 2)에서 4) 과정을 반복한다.

객체 추적 후 영상 내 목표물의 크기 및 각도는 탐색 윈도우 내 명암분포의 1차, 2차 모멘트를 계산하여 구할 수 있다.

$$\begin{aligned}
 M_{20} &= \sum_x \sum_y x^2 I(x,y) \\
 M_{02} &= \sum_x \sum_y y^2 I(x,y) \\
 M_{11} &= \sum_x \sum_y xy I(x,y)
 \end{aligned} \tag{1}$$

$I(x,y)$ 는 화소의 값을 나타내고, x,y 는 탐색 윈도우의 범위를 나타낸다. 계산의 편의를 위해 중간 변수들 a, b 및 c 를 식(2)와 같이 정의한다. 여기서 x_c, y_c 는 탐색 윈도우의 중심을 나타낸다.

$$\begin{aligned}
 a &= \frac{M_{20}}{M_{00}} - x_c^2 \\
 b &= 2 \left[\frac{M_{11}}{M_{00}} - x_c y_c \right] \\
 c &= \frac{M_{02}}{M_{00}} - y_c^2
 \end{aligned} \tag{2}$$

탐색 윈도우 내에 검출된 명암 분포의 가로(l), 세로(w) 크기 및 각도(θ)는 다음과 같이 계산된다.

$$\begin{aligned}
 l &= \sqrt{\frac{(a+c) + \sqrt{b^2 + (a-c)^2}}{2}} \\
 w &= \sqrt{\frac{(a+c) - \sqrt{b^2 + (a-c)^2}}{2}} \\
 \theta &= \frac{1}{2} \tan^{-1} \left(\frac{b}{a-c} \right)
 \end{aligned} \tag{3}$$

식(3)의 l, w 를 이용하여 면적을 계산하면 객체의 크기가 계산 되므로 Zoom의 위치 제어가 가능하게 된다.

IV. Pan Tilt 위치 제어

카메라의 좌표는 그림 3과 같이 X축, Y축, Z축으

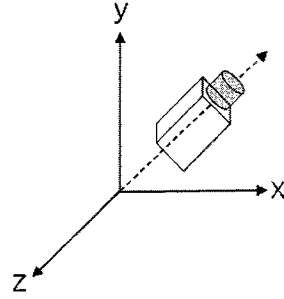


그림 3. 카메라 좌표

로 표시되고 카메라는 -Z축을 향한다고 가정한다. 화면상의 이동해야하는 위치가 (u_p, v_p) 일 경우 Zoom의 위치에 따른 전체 화면에 대한 화각을 알고 있으면 (u_p, v_p) 에서의 식(4)처럼 α, β 를 구할 수 있다. 그림 4는 화면에서의 카메라 이미지를 나타낸다.

$$\alpha = \frac{\theta_w}{W} \cdot u_p, \quad \beta = \frac{\phi_H}{H} \cdot v_p \tag{4}$$

처음 초기 좌표를 -Z축으로 $(0,0,-1)$ 로 설정한 후에 초기 좌표를 카메라 좌표로 변환하면 그림5에 나타난 바와 같이 되고, 식(5)와 같이 표현할 수 있다. $R(Y_u, \alpha)$ 는 Y축을 기준으로 α 만큼의 회전을 의미하고 $R(X_u, \beta)$ 는 X축을 기준으로 β 만큼의 회전을 의미한다.

$$\begin{aligned}
 P_c &= [00-11] \cdot R(Y_u, \alpha) \cdot R(X_u, \beta) \\
 &= [00-11] \begin{bmatrix} \cos\alpha 0 -\sin\alpha 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \cos\beta & \sin\beta & 0 \\ \sin\alpha 0 & \cos\alpha & 0 & 0 & 0 & -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= [-\sin\alpha \cos\alpha \sin\beta \quad -\cos\alpha \cos\beta \quad 1]
 \end{aligned} \tag{5}$$

카메라 좌표를 전체 좌표로 변환하면 그림 6에 나타난 바와 같이 되고, 식(6), 식(7)과 같이 표현된다.

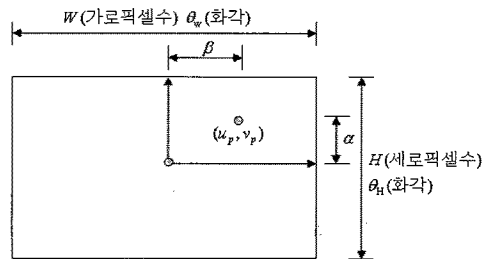
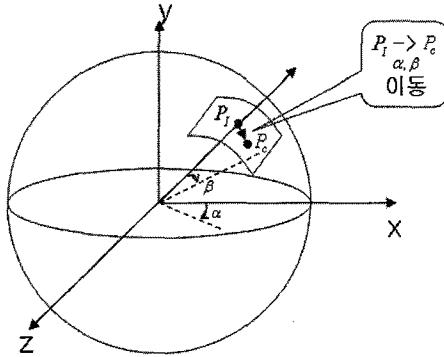


그림 4. 카메라 이미지



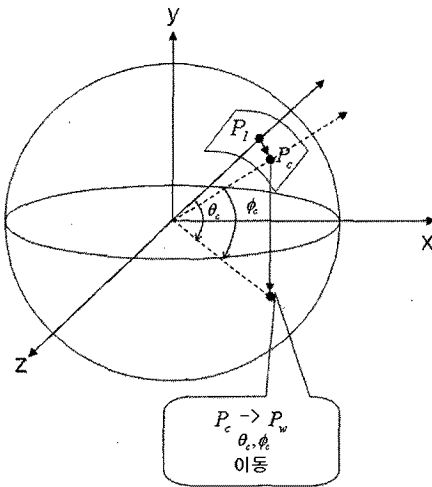
P_I : 초기 좌표 (0,0,-1)
 P_C : 카메라 좌표

그림 5. 초기좌표를 카메라 좌표로 변환

$R(x_c, \phi_c)$ 는 X축을 기준으로 ϕ_c 만큼 회전을 의미하고
 $R(y_c, \theta_c)$ 는 Y축을 기준으로 θ_c 만큼의 회전을 의미한다.

$$\begin{aligned} P_{w,\phi} &= R(x_c, \phi_c) \cdot P_C \\ P_{w,\theta} &= R(y_c, \theta_c) \cdot P_C \end{aligned} \quad (6)$$

$$\begin{aligned} P_w &= P_C \cdot P_{w,\theta} \cdot T_{w,\phi} \\ &= P_C \cdot \begin{bmatrix} \cos\theta_c & 0 & -\sin\theta_c & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta_c & 0 & \cos\theta_c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi_c & \sin\phi_c & 0 \\ 0 & -\sin\phi_c & \cos\phi_c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (7)$$



θ_c : 카메라 좌표 pan 각도
 ϕ_c : 카메라 좌표 tilt 각도

그림 6. 카메라 좌표를 전체 좌표로 변환

식(7)을 이용하여 $P_w(X_w, Y_w, Z_w)$ 값을 구하면 최종 Pan(θ), Tilt(ϕ) 위치는 식(8)과 같이 된다.

$$\begin{aligned} \theta &= \tan^{-1}(X_w/Z_w) \\ \phi &= \sin^{-1}(Y_w) \end{aligned} \quad (8)$$

PTZ 카메라를 이용한 객체 추적을 위해선 정확한 객체의 위치를 찾고 객체의 위치를 화면의 중앙으로 위치 시켜야 한다. 추적하는 객체를 화면의 중앙으로 이동시키기 위해선 Pan, Tilt의 위치를 제어함으로써 가능하다. 위의 식으로 부터 구한 Pan, Tilt 위치는 PTZ 카메라의 프로토콜을 이용하여 RS-232 방식을 사용하여 전송하게 된다.

V. Zoom 위치 제어

Zoom의 위치 제어는 식(3)의 l, w 를 이용하여 구할 수 있다. 본 논문에서 사용한 카메라 모듈은 SONY의 FCB-EX1010 모듈이 사용되었다. 이 모듈은 36배 광학 줌을 지원하고 있으며 Zoom의 위치를 0x0000 ~ 0x4000로 제어 가능하게 되어있다. 위의 O_x의 표시는 16진수를 나타낸다. 식(4)의 α, β 를 구하기 위해서는 현재의 Zoom 위치에 대한 θ_w, ϕ_H 를 구해야 한다. 본 논문에서는 Zoom의 위치를 256으로 나누어 총 0x40(64)개의 θ_w, ϕ_H 를 미리 구한다. PTZ 카메라에 따라 내부 카메라 모듈의 종류가 다양하므로 모듈이 다를 경우 θ_w, ϕ_H 의 값을 새롭게 구해야 한다. 그림 7은 각 Zoom에 대한 θ_w, ϕ_H 의 값을 나타낸다. 붉은 색은 θ_w , 파란 색은 ϕ_H 를 나타낸다. Zoom 값이 커지면 θ_w, ϕ_H 의 값이 감소함을 볼 수 있다.

전체 화면을 $A_t(W \times H)$, 화면상에 객체의 목표가

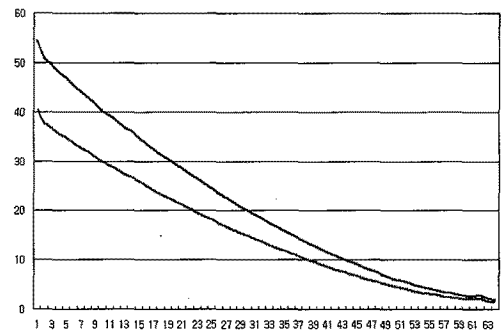


그림 7. Zoom에 대한 θ_w, ϕ_H

되는 크기를 A_d , Zoom 제어를 위한 여백을 A_m , 현재 객체의 크기를 A_c 라 한다. A_d 는 전체 화면의 1/9로 설정하고 A_m 는 A_d 의 1/4로 설정한다. 관계식으로 표현하면 식(9)와 같다.

$$\begin{aligned} A_d &= A_t/9 \\ A_m &= A_d/4 \\ A_c &= l \times w \end{aligned} \quad (9)$$

본 논문에서의 Zoom 제어식은 A_d 가 A_c 보다 클 때와 A_d 가 A_c 보다 작을 때로 나누어진다. 즉 $A_d > A_c$ 때는 Zoom In을 해야 하고 $A_d < A_c$ 때는 Zoom Out을 해야 한다. 최종 Zoom 값은 Z_t , 화면의 크기에 대한 Zoom 값의 조정 비율을 A_r 이라 할 때 $|A_c - A_d| - A_m$ 의 값이 0보다 클 경우 본 논문에서 계산한 Z_t 값은 식(10)과 같다. A_r 의 값의 범위는 $0 < A_r \leq 1$ 이 되고 값이 크면 빨리 Z_t 에 도달하고 적으면 늦게 도달한다.

$$Z_t = Z_c \pm (|A_c - A_d| - A_m) \times A_r \quad (10)$$

VI. 실험 결과 및 분석

본 논문에서는 객체가 검출되면, 컬러기반의 CAMShift 알고리즘을 사용하여 객체를 추적하는 방식을 제안하였다. Pan, Tilt 위치는 좌표표를 이용하여 측정 후 PTZ 프로토콜을 사용하여 위치 제어를 하였다. 영상은 PTZ 카메라의 Composite 신호를 USB로 바꿔주는 USB 그래픽을 사용하였고, 영상획득 및 디스플레이를 위하여 OpenCV 라이브러리를 이용하였다. 컬러는 Hue를 사용하였으며, 초기 객체는 임의로 사용자가 마우스로 지정하는 방식을 사용하였다. 화면의 크기(A_t)는 320x240로 실험을 하였다. Pan, Tilt는 현재 Zoom의 $\theta_w/48$, $\phi_H/48$ 이 넘지 않으면 화면의 중심에 있다고 가정한다. PTZ의 움직임은 PC에서 RS-232를 통하여 전송하면 RS-232 to RS-485 변환기를 거쳐 RS-485 통신방식으로 변환되어 PTZ 카메라에 전송된다. PTZ 프로토콜은 Set Position과 Get Position을 통해 이루어지며 통신 속도는 19200bps를 사용하였다. Set Position, Get Position 명령은 Pan, Tilt, Zoom을 하나의 패킷으로 만들어 PTZ 카메라로 전송하는 프로토콜이다. 실제 Set Position을 전송한 후, Get Position을 통하여 좌표의 도착 유무를 확인하

였다. 표 1은 상대적으로 높은 Zoom에서의 실험 결과를 표로 나타낸 것이고, 표 2는 낮은 Zoom에서의 실험 결과를 표로 나타낸 것이다. Zoom은 식(10)을 사용하여 구한 결과이고 표 1, 표 2에서의 time의 의미는 좌표표를 이용하여 좌표가 구해지면 PTZ 프로토콜의 Set Position 전송 후 PTZ로부터 Get Position 프로토콜을 통해 Set Position과 Get Position의 값이 동일해지는 시간을 나타낸다. CAMShift는 컬러기반의 추적이기 때문에 주변의 유사한 색상 혹은 겹침이 발생하면 오류가 발생한다. 본 실험은 주변에 유사한 색상 및 겹침이 없다고 가정한다. 그리고 Tilt의 각도 범위가 0~90도이고 객체가 이 범위 안에 있어야 한다. 위의 제약 조건을 가정하여 실험을 하였다.

표 1. 높은 Zoom에서의 좌표 결과 및 경과 시간

$\alpha(^{\circ})$	$\beta(^{\circ})$	$\theta(^{\circ})$	$\phi(^{\circ})$	Zoom	time (ms)
-0.47	-0.67	309.47	31.87	0x1b70	203
-0.47	-0.6	310.03	31.26	0x1b70	204
0.68	0.67	309.23	31.93	0x1b70	203
0.61	0.67	308.51	32.6	0x1b70	203
-0.2	-0.81	308.75	31.8	0x1afe	235
0.14	-0.63	308.58	31.17	0x18a2	297
-0.31	-0.23	308.58	31.17	0x1674	343
-0.66	-0.57	309.35	30.59	0x162d	219
0	-0.57	309.35	30.02	0x162d	172
0.66	-1.22	308.59	28.79	0x162d	234
3.71	-1.39	304.42	27.34	0x162d	297
4.7	-1.63	299.21	25.61	0x162d	297
3.54	-2.45	295.36	23.11	0x162d	297
2.8	-1.8	292.35	21.29	0x162d	266
3.46	-1.88	288.68	19.37	0x166c	296
1.24	-0.41	287.37	18.95	0x1798	313

표 2. 낮은 Zoom에서의 좌표 결과 및 경과 시간

$\alpha(^{\circ})$	$\beta(^{\circ})$	$\theta(^{\circ})$	$\phi(^{\circ})$	Zoom	time (ms)
-5.96	0	298.01	26.46	0x000	375
-8.52	1.52	307.63	27.66	0x000	406
-9.2	-1.18	317.88	26.1	0x000	391
-4.43	-1.52	322.74	24.5	0x000	328
-3.07	-0.84	322.74	24.5	0x0f3	265
-4.94	-1.35	328.11	23.05	0x21f	344
0.77	-0.61	328.11	23.05	0x39d	266
4.05	-1.49	323.76	21.5	0x535	312
5.38	-0.56	317.99	20.84	0x644	328
6.05	0.41	311.5	21.13	0x644	375
7.57	1.23	303.33	22.16	0x644	360
7.57	2.18	295.04	24.14	0x644	375
7.84	0.55	286.41	24.44	0x644	359
7.57	-0.82	278.16	23.4	0x644	375
7.71	-0.14	269.79	23.04	0x644	359
9.08	-1.23	260.02	21.51	0x644	407

표 1, 표 2의 실험 결과를 보면 Zoom이 크면 α , β 값의 크기가 작고 Zoom이 작으면 α , β 값의 크기가 커짐을 알 수 있다. 또한 α , β , Zoom 값의 차이가 클수록 시간이 길어짐을 알 수 있다. 그림 8은 붉은색 사각형을 처음에 객체로 지정한 후 실제 테스트한 영상이다. 그림에서 (a)는 (217, 15.9)이고 (b)는 (212.8, 5.2), (c)는 (219.4, 8.5), (d)는 (209.7, 13.8), (e)는 (280.2, 46.6.8), (f)는 (285.5, 42.9)의 Pan, Tilt의 좌표 값이 객체의 위치에 따라 바뀔을 알 수 있다. 각각의 실험 영상의 Zoom 값은 화면에 표시되지는 않지만 다른 값을 갖는다. 실험 결과 CAMShift를 이용한 물체추적은 실시간 처리 성능이 우수함을 알 수 있었다. 하지만 Zoom이 높은 경우 객체의 움직임이 느리게 움직이더라도 실제 화면의 객체는 빠르게 움직이기 때문에 쉽게 화면에서 놓치는 문제점이 존재한다.

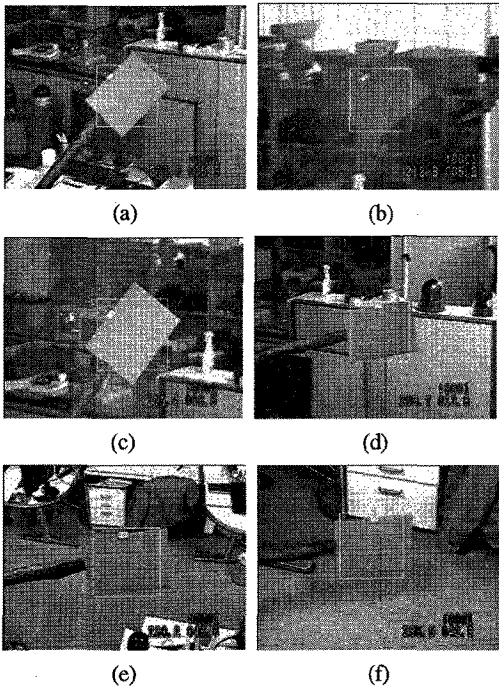


그림 8. 실험 영상

VII 결론

제안한 방법은 초기 객체를 지정한 후 CAMShift 방법을 이용하여 화면상에서 객체의 위치를 찾은 다음 구좌표를 이용하여 Pan, Tilt의 위치를 구한다. CAMShift 방법으로 구해진 객체의 면적을 이용하여 Zoom 값을 구한다. 구해진 Pan, Tilt, Zoom의 위치를

제어하여 객체를 추적하는 PTZ 카메라 시스템을 본 논문에서 제안하였다. 하지만 색 정보만을 이용하기 때문에 겹침이나 유사한 칼라에 대해 오류율이 높은 단점이 존재하며, PTZ 제어를 위해 Set 명령을 사용하기 때문에 객체를 추적할 때 화면에서 객체가 끊기는 것과 같은 느낌을 받을 수 있다. 실제 화면상의 객체 위치를 제어를 하기 때문에 객체가 움직일 경우 중앙에 위치하기 위한 객체 예측이 필요하다. 추후 연구에는 위의 문제점을 보완한 강인한 추적을 수행할 수 있는 알고리즘의 연구를 계속 진행할 예정이다.

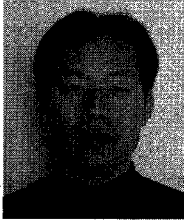
참고 문헌

- [1] D Comaniciu, V Ramesh, P Meer "Real-time tracking of non-rigid objects using mean shift" Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference Vol.2 142-149. 2000
- [2] COMANICIU, D. AND MEER, P. "Mean shift analysis and applications." In IEEE International Conference on Computer Vision (ICCV). Vol.2 pp.1197-1203, 1999.
- [3] Gary R. Bradski, "Computer Vision Face Tracking For Use in a Perceptual User Interface", Microcomputer Research Lab, 2002.
- [4] D Comaniciu, V Ramesh, P Meer "Real-time tracking of non-rigid objects using mean shift" Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference Vol.2 pp.142-149. 2000
- [5] J. G. Allen, R.YD Xu, J. S. Jin "Object Tracking Using CamShift Algorithm and Multiple Quantized Feature Spaces," Pan-Sydney Area Workshop on Visual Information Processing, pp.3-7, 2003
- [6] Ajmal Mian "Realtime face detection and tracking using a single Pan, Tilt, Zoom camera" in IEEE IVCNZ, 2008
- [7] T. Funahasahi, M. Tominaga, T. Fujiwara, and H. Koshimizu, "Hierarchical face tracking by using PTZ camera" in IEEE FG, 2004, pp.427-432.
- [8] P. Amnuaykanjanasin, S. Aramvith, and T. Chalidabhongse, "Face Tracking Using Two Cooperative Static and Moving Cameras" in

IEEE ICME, 2005, pp.1158-1161.

장 일 식 (Il-sik Chang)

정회원



2001년 2월 호남대학교 전자공학과 학사
2001년 1월~현재 (주)하이트론시스템즈 선임연구원
<관심분야> Computer Vision, 영상 인식

박 광 영 (Kwang-young Park)

정회원



2000년 2월 서강대학교 석사
1996년 12월~현재 (주)하이트론시스템즈 정보통신사업본부 팀장
<관심분야> 멀티미디어 통신, 영상인식, IT정책

안 태 기 (Tae-ki An)

정회원



1993년 2월 경북대학교 전자공학과 학사
1996년 2월 경북대학교 전자공학과 석사
2008년 2월 성균관대학교 전자전기 컴퓨터공학과 박사수료
1996년 3월~현재 한국철도기술연구원 선임연구원

<관심분야> 멀티미디어 통신, 영상분석, 인공지능

박 구 만 (Park, Goo-Man)

종신회원



1984년 2월 한국항공대학교 전자공학과 공사
1986년 2월 연세대학교 전자공학과 석사
1991년 2월 연세대학교 전자공학과 박사
1991년 3월~1996년 9월 삼성 전자 신호처리연구소 선임연구원

1996년 9월~1999년 7월 호남대학교 전자공학과 조교수

1999년 8월~현재 서울산업대학교 매체공학과 교수
2006년 1월~2007년 8월 Georgia Institute of Technology Dept. of Electrical and Computer Engineering, Visiting Scholar

<관심분야> 멀티미디어 통신, 컴퓨터비전