

## 정규화된 색공간과 푸리에변환특징량을 리용한 움직임검출방법

리원호, 이정환

움직임검출은 교통감시를 비롯한 동화상감시에서 많이 리용되고있는데 현재 세계적으로 촬영기로부터 입력되는 동화상에서 움직이는 물체를 실시간적으로 자동검출하는 체계를 개발하기 위한 연구사업이 활발히 진행되고있다.

움직임검출에서 가장 널리 리용되는 방법은 배경덜기법이다.

선행연구[1]에서는 배경을 가우스혼합모형으로 모형화하여 움직임검출을 진행하였다. 여기서는 화소의 특징량으로 RGB색성분을 리용하였으며 매 성분은 서로 독립이고 같은 분산을 가진다고 보았다. 즉  $\Sigma_{i, I} = \sigma_{i, I}^2 I$  이다.

선행연구[1]에서 제기한 방법은 화소단위로 RGB색성분을 직접 쓰므로 촬영기잡음과 조명변화에 불안정한 결함을 가지고있다. 또한 그림자를 제거하기도 힘들다.

선행연구[2]에서는 이러한 결함을 극복하기 위하여 특징량으로서 문양정보를 리용하였다. 여기서는 문양정보를 반영하기 위하여 LBP특징량을 리용하였는데 이 특징량도 조명변화에 불안정한 결함을 가지고있다.

선행연구[3]에서는 HSV색공간과 인공신경망을 리용한 배경덜기법으로 움직임검출을 진행하였는데 이 방법도 화소준위의 특징량을 리용하였으므로 불안정하다.

본문에서는 이러한 결함들을 극복하기 위하여 화소준위에서의 정규화된 RGB색성분과 푸리에변환결수들에 의한 문양정보를 결합하여 안정한 움직임검출알고리즘을 제기하였다.

### 1. 움직임검출에 리용되는 특징량의 구성

본문에서는 움직이는 물체의 검출을 위하여 두가지 특징량을 리용한다.

구체적인 특징량구성은 다음과 같다.

먼저 어두운 부분에서의 불안정성을 제거하기 위하여 매 R, G, B통로에 대하여 다음의 변환을 진행한다.

$$R' \leftarrow 235R/255 + 20, G' \leftarrow 235G/255 + 20, B' \leftarrow 235B/255 + 20$$

다음 색성분을  $m_1 \leftarrow (R'+G'+B')/3$ ,  $m_2 \leftarrow R'/m_1$ ,  $m_3 \leftarrow G'/m_1$  와 같이 정규화한다. 이렇게 얻어진 정규화된 RGB특징량  $m = (m_1, m_2, m_3)$  을 쓰면 조명변화에 보다 안정해진다.

다음 RGB화상을 흑백색농담화상으로 변환하고  $8 \times 8$ 블록단위로 표준화한다.

그리고 2차원리산푸리에변환을 진행하고 푸리에결수의 절대값들을 구한다. 여기서 얻어진  $8 \times 8$ 크기의 표에서 1행 1렬원소가 직류성분인데 이 표를 오른쪽으로 4칸, 아래로 4칸 회전시켜 직류성분이 중심에 놓이게 한다.

$k=1, \dots, 7$ 에 대하여  $k$ 가 들어있는 칸에 해당하는 결수들의 합을 해당 개수로 나눈 값을  $f_k$ 라고 하고 푸리에변환특징량  $f=(f_1, f_2, \dots, f_7)$ 을 구성한다.  $f_1, f_2, f_3$ 은 주파수별 특성을,  $f_4, f_5, f_6, f_7$ 은 방향별특성을 나타낸다.

해당  $8 \times 8$ 블록의 매 화소에 다같이 이 푸리에변환특징벡토르를 할당한다.

이 푸리에변환결수들은 절대값을 리용하므로 평행이동에 안정하며 문양정보를 반영하므로 조명변화에 잘 견디며 그림자검출에도 유리하다. 즉 움직이는 물체의 그림자에서는 밝기가 본래의 배경보다 훨씬 어두워지지만 문양은 같기때문에 문양정보를 리용하면 그림자검출의 성능이 높아진다.

## 2. 움직임검출알고리즘

다음으로 배경모형은 선행연구[1]에서와 같이 가우스혼합모형을 리용하였다. 즉 매 화소에서 특징량의 출현확률이  $P(X_t) = \sum_{i=1}^K \omega_{i,t} \eta(X_t, \mu_{i,t}, \Sigma_{i,t})$ 에 따른다고 가정한다. 여기서  $X_t$ 는  $t$  시각에 해당 화소에서의 특징량으로서 정규화된 RGB성분과 푸리에변환특징량으로 구성된다. 즉  $X_t = (m_t, f_t)$ 이고

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(X_t - \mu)^T \Sigma^{-1}(X_t - \mu)\right)$$

는 평균이  $\mu$ 이고 공분산행렬이  $\Sigma$ 인 가우스확률밀도함수이다. 그리고  $K$ 는 혼합성분의 개수이고  $\omega_{i,t}$ 는 매 성분의 무게이다.

논문에서는  $K=5$ 로 하고 그중 4개는 기본배경모형이고 나머지 1개는 예비모형으로 하였다. 처음에는 매 화소에 유효한 모형이 1개인데 후에 점차 모형의 개수가 4개까지 증가한다. 그리고 예비모형의 무게가 일정하게 증가하면 4개의 기본모형중에서 무게가 가장 작은 모형을 교체한다.

배경모형을 표현하는 파라미터로서  $n=1, \dots, 5, i=1, \dots, N, j=1, \dots, M$  일 때 다음의 행렬들을 보관한다.

- $\bar{m}_{nk}(i, j), k=1, 2, 3$ :  $i$ 째 행,  $j$ 째 열화소의  $n$ 번째 모형에서 특징량  $m_k$ 의 평균
- $\bar{f}_{nk}(i, j), k=1, \dots, 7$ :  $i$ 째 행,  $j$ 째 열화소의  $n$ 번째 모형에서 특징량  $f_k$ 의 평균
- $\sigma_{nk}(i, j), k=1, 2, 3$ :  $n$ 번째 모형에서 특징량  $m_k$ 의 분산
- $\sigma_{n4}(i, j)$ :  $n$ 번째 모형에서 특징량  $f_1, \dots, f_7$ 의 분산
- $\omega_n(i, j)$ :  $n$ 번째 모형의 무게
- $B(i, j)$ : 현재 유효한 모형의 개수
- $C(i, j)$ : 화소  $(i, j)$ 가 현재 프레임까지 연속적으로 예비모형으로 판정된 회수
- $R_n(i, j), n=1, \dots, 4$ :  $n$ 번째 우선순위에 있는 모형의 번호

여기서  $N, M$ 은 각각 화상의 행의 개수와 렬의 개수이다.

그리고 무게턱값과 배경학습파라미터들을 각각  $T=0.4, \alpha_0=0.0125, \beta_0=0.2$ 로 설정한다. 첫번째 프레임이 입력되었을 때 매 화소에 대하여 배경모형을 다음과 같이 초기화한다.

$$\bar{m}_{1k}(i, j) \leftarrow m_{1k}(i, j), k=1, 2, 3$$

$$\bar{f}_{1k}(i, j) \leftarrow f_{1k}(i, j), k=1, \dots, 7$$

$$\sigma_{11}(i, j) \leftarrow 20, \sigma_{12}(i, j) \leftarrow 15, \sigma_{13}(i, j) \leftarrow 15, \sigma_{14}(i, j) \leftarrow 20$$

$$B(i, j) \leftarrow 1$$

$$C(i, j) \leftarrow 0$$

$$R_1(i, j) \leftarrow 1, R_n(i, j) \leftarrow 0, n=2, 3, 4$$

$$\omega_1(i, j) \leftarrow 1/4, \omega_n(i, j) \leftarrow 0, n=2, \dots, 5$$

두번째 프레임부터는 움직임검출을 진행하고 배경모형을 갱신한다.

$t+1$  번째 프레임에서 매  $(i, j)$  에서 움직임검출과 배경모형갱신알고리즘은 다음과 같다.

걸음 1 우선순위가 높은 모형부터 탐색하면서 처음으로

$$s = \sum_{k=1}^3 w_k \frac{(m_k(i, j) - \bar{m}_{nk}(i, j))^2}{\sigma_{nk}^2(i, j)} + w_4 \frac{\sum_{k=1}^7 (f_k(i, j) - \bar{f}_{nk}(i, j))^2}{\sigma_{n4}^2(i, j)} < 16$$

인 모형번호  $n$  과  $s$  를 구한다. 여기서  $w_k$  들은 매 특징량들의 비중을 조절하는 무게인데  $w_1 + w_2 + w_3 + w_4 = 1$  이다.

그런  $n$  이 있으면 걸음 2로 이행하고 없으면 걸음 9로 이행한다.

걸음 2 걸음 1에서 구한  $n$  을  $n_0$  이라고 놓고  $k$  를 1부터  $B(i, j)$  까지 순환하면서 걸음 3-6을 반복실행하고 걸음 7로 이행한다.

걸음 3  $n \leftarrow R_k(i, j)$

$n = n_0$  이면 걸음 4로 이행하고 그렇지 않으면 걸음 5로 이행한다.

걸음 4  $\max_{1 \leq k \leq B(i, j)} \omega_k(i, j) > T$  이고  $\omega_n(i, j) < T$  이면 화소  $(i, j)$  를 전경화소로 설정하고 다음과 같이 모형을 갱신한다.

$$\alpha \leftarrow \alpha_0 e^{-s/4}, \beta \leftarrow \beta_0 e^{-s/4}$$

$$\omega_n^{(t+1)}(i, j) \leftarrow (1 - \alpha)\omega_n^{(t)}(i, j) + \alpha$$

$$\bar{m}_{nk}^{(t+1)}(i, j) \leftarrow (1 - \beta)\bar{m}_{nk}^{(t)}(i, j) + m_k(i, j), k=1, 2, 3$$

$$\bar{f}_{nk}^{(t+1)}(i, j) \leftarrow (1 - \beta)\bar{f}_{nk}^{(t)}(i, j) + f_k(i, j), k=1, \dots, 7$$

$$\sigma_{nk}^{(t+1)} \leftarrow (1 - \beta)\sigma_{nk}^{(t)} + \beta[m_k(i, j) - \bar{m}_{nk}^{(t+1)}(i, j)]^2, k=1, 2, 3$$

$$\sigma_{n4}^{(t+1)} \leftarrow (1 - \beta)\sigma_{n4}^{(t)} + \beta \sum_{k=1}^7 [f_k(i, j) - \bar{f}_{nk}^{(t+1)}(i, j)]^2$$

다음 걸음 6으로 이행한다.

걸음 5  $n$  번째 모형의 무게만  $\omega_n^{(t+1)}(i, j) \leftarrow (1 - \alpha)\omega_n^{(t)}(i, j)$  와 같이 감소시킨다.

걸음 6  $t_n \leftarrow \frac{\omega_n^2(i, j)}{\sum_{k=1}^3 \sigma_{nk}^2}$

걸음 7  $t_n$  이 작아지는 순서로 모형번호를 다시 정렬하고  $R_n(i, j), n=1, \dots, B(i, j)$  에 대입한다.

걸음 8  $C(i, j) \leftarrow 0$  이면 알고리즘을 완료한다.

걸음 9  $\max_{1 \leq k \leq B(i, j)} \omega_k(i, j) > T$  이면 화소  $(i, j)$  를 전경화소로 설정한다.

걸음 10  $B(i, j) < 4$  이면 다음과 같이 새로운 모형을 구성하고 알고리즘을 끝내며  $B(i, j) = 4$  이면 걸음 11로 이행한다.

$$B(i, j) \leftarrow B(i, j) + 1$$

$$n \leftarrow B(i, j)$$

$$R_n(i, j) \leftarrow n$$

$$\omega_n(i, j) \leftarrow 1/4$$

$$\bar{m}_{nk}(i, j) \leftarrow m_k(i, j), k=1, 2, 3$$

$$\bar{f}_{nk}(i, j) \leftarrow f_k(i, j), k=1, \dots, 7$$

$$\sigma_{n1}(i, j) \leftarrow 20, \sigma_{n2}(i, j) \leftarrow 15, \sigma_{n3}(i, j) \leftarrow 15, \sigma_{n4}(i, j) \leftarrow 20$$

걸음 11  $C(i, j) = 0$  이면 다음과 같이 예비모형을 초기화하고 알고리즘을 끝내며  $C(i, j) > 0$  이면 걸음 12로 이행한다.

$$\omega_5(i, j) \leftarrow 1/4$$

$$\bar{m}_{5k}(i, j) \leftarrow m_k(i, j), k=1, 2, 3$$

$$\bar{f}_{5k}(i, j) \leftarrow f_k(i, j), k=1, \dots, 7$$

$$\sigma_{51}(i, j) \leftarrow 20, \sigma_{52}(i, j) \leftarrow 15, \sigma_{53}(i, j) \leftarrow 15, \sigma_{54}(i, j) \leftarrow 20$$

$$C(i, j) \leftarrow 1$$

걸음 12 예비모형을 다음과 같이 갱신한다.

$$C(i, j) \leftarrow C(i, j) + 1$$

$$\alpha \leftarrow 1/[1 + \exp(0.1(75 - C(i, j)))], \beta \leftarrow \beta_0 e^{-s/4}$$

$$\omega_5^{(t+1)}(i, j) \leftarrow (1 - \alpha)\omega_5^{(t)}(i, j) + \alpha$$

$$\bar{m}_{5k}^{(t+1)}(i, j) \leftarrow (1 - \beta)\bar{m}_{5k}^{(t)}(i, j) + m_k(i, j), k=1, 2, 3$$

$$\bar{f}_{5k}^{(t+1)}(i, j) \leftarrow (1 - \beta)\bar{f}_{5k}^{(t)}(i, j) + f_k(i, j), k=1, 2, 3$$

$$\sigma_{5k}^{(t+1)} \leftarrow (1 - \beta)\sigma_{5k}^{(t)} + \beta[m_k(i, j) - \bar{m}_{5k}^{(t+1)}(i, j)]^2, k=1, 2, 3$$

$$\sigma_{n4}^{(t+1)} \leftarrow (1 - \beta)\sigma_{n4}^{(t)} + \beta \sum_{k=1}^7 [f_k(i, j) - \bar{f}_{nk}^{(t+1)}(i, j)]^2$$

걸음 13  $\omega_5(i, j) > 0.2$  이면 4개의 기본모형 중에서 무게가 가장 작은 모형을 예비모형으로 교체하고  $C(i, j)$  를 0으로 설정한다.

이상과 같이 매 프레임에서 움직임물체를 표현하는 2진화상을 얻은 다음 후처리에서 전경물체에 있는 작은 구멍들과 배경에 있는 작은 전경물체들을 제거한다.

8×8블록의 매 화소들이 같은 푸리에변환특징량을 리용하므로 이렇게 얻은 움직임 물체의 영역에서는 블록효과가 나타난다.

그러므로 위에서 얻은 물체의 경계주변에서 정규화된 RGB특징량만 리용하여 움직임 검출을 다시한번 진행한다. 이렇게 하여 자연스러운 경계를 가진 전경물체를 얻는다.

Change Detection Challenge자료기지를 리용하여 선행방법과 우리의 방법을 비교한 결과를 실시 전경영역과 거기서 전경으로 검출된 영역의 면적비는 표와 같다.

표. 실시 전경영역과 거기서 전경으로 검출된 영역의 면적비

부분자료기지	방법 [1]	방법 [2]	방법 [3]	우리의 방법
Baseline	0.73	0.78	0.80	0.89
Dynamic background	0.68	0.72	0.75	0.81
Shadow	0.65	0.74	0.73	0.79

## 참 고 문 헌

- [1] C. Stauffer et al.; In IEEE Conf. Comp. Vision & Pattern Recognition, 2, 246, 1999.
- [2] Jian Yao et al.; IEEE the CVPR Visual Surveillance Workshop Minneapolis, 6, 238, 2007.
- [3] Y. Singh et al.; International Journal of Computer Science and Network Security, 10, 3, 136, 2010.

주체104(2015)년 7월 5일 원고접수

## A Moving Object Detection Method using Normalized Color Space and Fourier Transform Features

*Ri Won Ho, Ri Jong Hwan*

Background subtraction is a method which is often used in motion detection. In the previous literatures, Gaussian mixture models based on pixel-wise color information were used.

We combined pixel-based normalized RGB color space and Fourier transform features in using Gaussian mixture model, thus improved robustness to illumination changes and camera noise.

Key words: motion detection, background subtraction