

사람 행동 인식에서 반복 감소를 위한 저수준 사람 행동 변화 감지 방법

노요환[†], 김민정^{**}, 이도훈^{***}

Detection of Low-Level Human Action Change for Reducing Repetitive Tasks in Human Action Recognition

Yohwan Noh[†], Min-Jung Kim^{**}, DoHoon Lee^{***}

ABSTRACT

Most current human action recognition methods based on deep learning methods. It is required, however, a very high computational cost. In this paper, we propose an action change detection method to reduce repetitive human action recognition tasks. In reality, simple actions are often repeated and it is time consuming process to apply high cost action recognition methods on repeated actions. The proposed method decides whether action has changed. The action recognition is executed only when it has detected action change. The action change detection process is as follows. First, extract the number of non-zero pixel from motion history image and generate one-dimensional time-series data. Second, detecting action change by comparison of difference between current time trend and local extremum of time-series data and threshold. Experiments on the proposed method achieved 89% balanced accuracy on action change data and 61% reduced action recognition repetition.

Key words: Computer Vision, Human Action Recognition, Motion History Image, Action Change Detection

1. 서 론

사람 행동 인식은 컴퓨터비전 분야에서 활발한 연구가 이루어지고 있는 주제이다. 이는 인간-컴퓨터 상호작용, 비디오 검색, 비전 기반 감시등의 분야에 서 활용될 수 있다. 이미지들의 연속적인 서열인 비디오 처리는 시간적인 특징을 고려해야 하기 때문에 단일 이미지 처리와 비교해 볼 때, 매우 어려운 작업이다. 최근 많은 행동 인식 방법은 딥러닝을 통해 많

은 발전을 이루었다. 컨볼루션 신경망이나 순환 신경 망을 기반으로 한 최신 연구들은 사람 행동 인식에서 가장 좋은 성능을 보이고 있다. 그러나, 딥러닝 기반 방법들은 많은 연산량을 요구한다. 영화나 드라마 분석과 같은 작업에는 적합하나 CCTV와 같이 실시간으로 사용자의 움직임을 분석해야 하는 경우에는 환경의 제약을 받는다.

감시카메라 화면에 나타나는 사람의 움직임은 걷기, 달리기, 서있기 등의 단순한 패턴이 반복되는 경

※ Corresponding Author : DoHoon Lee, Address: (609-836) 313-312 2, Busandaehak-ro 63beon-gil, Geumjeong-gu, Busan, Republic of Korea, TEL : +82-51-510-2491, FAX : +82-51-517-2431, E-mail : dohoon@pusan.ac.kr
Receipt date : Jan. 30, 2019, Revision date : Mar. 22, 2019
Approval date : Apr. 2, 2019

[†] Dept. of Electrical, Electronics & Computer Eng., Pusan National University (E-mail : ubseeyou@pusan.ac.kr)

^{**} Office of General Education., Pusan National University (E-mail : mjkim27@pusan.ac.kr)

^{***} School of Electrical & Computer Eng., Pusan National University

※ This work was supported by a 2-year Research Grant of Pusan National University.

향이 있다. 이러한 패턴의 행동들을 저수준 행동이라고 한다[1, 2]. 영화나 스포츠에서 나타나는 변화가 많은 고수준 활동은 자주 관찰되지 않는다. 고수준 활동은 달리기, 걷기, 발차기와 같은 부분활동의 서열로 이루어진 축구와 같은 활동을 말한다. 감시카메라 환경에서는 이와 같은 저수준 행동이 반복된다.

본 논문에서는 저수준 행동에 대한 사람 행동 인식 방법 적용시 일어나는 반복적인 작업을 줄이고자 한다. 같은 행동이 반복되는데 계속해서 높은 연산량이 요구되는 행동 인식 방법을 수행하는 것은 비효율적이다. 기존의 연구에서는 고정된 프레임마다 행동 인식 방법을 적용한다. 그러나 본 연구에서는 연산량이 적은 행동 변화 감지 방법을 반복적으로 적용해 행동 변화가 감지되는 경우에만 행동을 인식하도록 하여 전체적인 연산량을 줄이는 방법을 제안한다. 제안하는 방법은 행동 인식 방법의 적용 횟수를 줄일 수 있음을 실험으로 보였다.

본 논문의 구성은 다음과 같다. 2장에서는 본 연구의 행동 변화 감지 과정과 사람 행동 인식에 대한 관련 연구를 알아본다. 3장에서는 행동 변화 감지 방법에 대해 설명한다. 모션 히스토리 이미지에서 1차원 시계열 데이터를 생성하는 과정과 시계열 데이터를 이용하여 행동 변화를 감지하는 과정에 대해 다룬다. 4장에서는 행동 변화 감지 및 인식 방법을 통합하였을 때에 대한 실험 결과를 설명한다. 5장에서는 실험결과에 따른 결론 및 향후연구 진행할 연구 방향을 제시한다.

2. 관련연구

2.1 사람 행동 인식 및 행동 변화 감지

사람 행동 인식은 사람이 행하는 행동이 무엇인지 인식하는 것을 말한다. 행동을 인식하기 위해서는 시간정보와 공간정보를 동시에 처리해야 한다. 따라서 시공간 정보를 동시에 가지는 특징이 주로 사용된다. I. Labtev [3]는 헤리스 점 검출기와 포르스트너 점 검출기를 이용해 시간 및 공간적으로 눈에 띄는 변화를 보이는 영역을 표현하는 STIP(space-time interest points)를 제안했다. H. wang [4]은 비디오를 밀집 궤적으로 표현하는 방법을 제안하였다. 밀집 궤적은 객체의 움직임을 밀집 유펜켈 플로우 필드로 시간에 대한 궤적으로 표현하였다. A.F. Bobick [5, 6]은

차영상을 이용해 시간이 많이 지난 화소는 밝기를 줄이고 최근의 움직임은 화소값을 밝게 하는 방식으로 움직임을 나타내는 모션 히스토리 이미지를 제안했다. 이러한 특징을 분류하는 알고리즘은 현재 딥러닝에 기반한 방법에 응용되어 행동 인식에 가장 좋은 성과를 보이고 있다. Y. Shi [7]는 밀집 궤적을 2차원 평면에 투사한 특징을 컨볼루션 신경망과 순환 신경망을 이용해 분류하였으며, C.B. Jin [8]은 모션 히스토리 이미지와 차영상, 가중치 평균 이미지를 각각 입력으로 사용하는 컨볼루션 신경망을 결합하여 움직임을 분류하였다. I.H. Kwon [9]은 서포트 벡터 머신을 이용하여 행동을 분류하고 가장 높은 인식률을 보이는 학습 단위를 실험을 통해 보였다.

영상의 특징 추출 과정과 딥러닝을 이용한 특징 분류 과정은 연산량이 많고, 행동의 학습 및 분류 단위는 고정된 값이기에 행동 인식 방법이 단위마다 반복 수행되며 단위 내에서 일어나는 행동 변화를 감지할 수 없다. 여러개의 객체가 동시에 나타나는 실시간 시스템에 적용하기에는 어려움이 있으며 이를 보완하기 위하여 행동의 변화를 감지하는 방법이 필요하다. Y.H. Noh [10]는 모션 히스토리 이미지를 이용한 행동 변화 감지 방법을 제안하였으나 화소와 윈도우 크기를 고정하여 행동 변화 감지가 강인하지 않다.

2.2 개념 변화 감지

행동 변화 감지는 개념 변화 감지이다. 개념 변화 감지란 시계열 데이터의 일부분이 어떠한 클래스로 분류될 때, 다음에 들어오는 데이터의 클래스가 바뀌는지 아닌지를 감지하는 것이다. 제안한 행동 변화 감지도 이전의 행동과 다음의 행동이 다른지를 판단하는 것이므로 개념 변화 감지에 해당된다. J. Gama [11]와 A. Bifet [12]은 데이터 서열을 클래스로 분류할 때 발생하는 오류율에 기반한 개념 변화 탐지 방법을 제안하였다. 시계열 데이터에서 클래스 분류시 정확도 향상을 기대할 수 있으나 지속적인 분류를 통해 변화를 찾아내는 방법이므로 행동 인식을 위한 연산량이 많다.

CCTV의 동영상을 통한 행동 변화를 감지하기 위해 오류율이 아닌 단순 수치 변화로 연산량을 줄일 필요가 있다. 이 비디오는 2차원 이미지가 단위시간마다 존재하는 3차원 데이터이며, 각 차원을 단순히

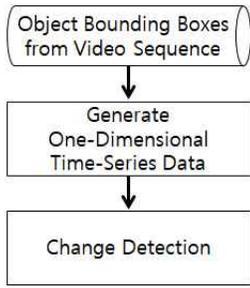


Fig. 1. Flowchart for action change detection algorithm.

수치로 표현해서는 의미있는 정보를 얻을 수 없다. 따라서 시간정보와 공간정보를 가지는 특징이 필요한데, 이러한 특징들은 오류율 정보 없이 개념 변화 감지 방법을 적용하기에 복잡하다. 따라서 동영상으로부터 단순한 수치를 추출하여 행동 변화를 감지한다.

3. 행동 변화 감지

3.1 제안한 알고리즘의 개요

비디오 데이터에서 사람의 행동변화를 감지하기 위해 움직임의 경계를 지정하여 모션 히스토리 이미지의 변화를 이용한 행동변화 알고리즘을 제안한다. Fig. 1에 제안하는 행동 변화 감지 알고리즘의 전체적인 흐름도를 나타내었다.

제안한 알고리즘은 크게 3가지 과정으로 이루어져 있다. 먼저 바운딩 박스(bounding box)에 표현되는 경계내의 이미지 서열로부터 모션 히스토리 이미지(motion history image)를 추출하여 1차원(1-D) 시계열 데이터를 생성한다. 이 데이터는 모션 히스토리 이미지에서 0이 아닌 화소의 개수이다. 다음은 시계열 데이터의 경향이 증가하거나 감소하는 정도에 따라 행동이 변하였는지 아닌지를 판단한다.

3.2 1-D 시계열 데이터 생성

모션 히스토리 이미지는 사람의 움직임이 일어나

는 위치에 적용된다. 모션 히스토리 이미지는 여러 프레임의 시간 및 공간 정보를 담고 있다. 모션 히스토리 이미지는 가장 최근에 움직임이 일어난 부분은 밝기값이 높은 화소로 표현되고 움직임이 오래될수록 밝기가 낮아진다. 움직임이 없었거나 소멸시간이 지난 움직임은 화소 밝기값이 0으로 표시된다. Fig. 2는 프레임에 따른 움직임은 사람의 모습과 추출된 모션 히스토리 이미지 예시를 나타낸다. 4번째 이미지는 20프레임에 생성된 모션 히스토리 이미지이다.

다음은 모션 히스토리 이미지를 구하는 방법이다 [6]. 시간 t 에서 프레임 $F(x, y, t)$ 는 해당 프레임의 모션 화소이다. x 와 y 는 이미지 화소의 좌표이다. 시간 t 의 프레임과 시간 $t-1$ 의 프레임간의 차영상, $D(x, y, t)$ 이라 한다. 다음 수식(1)에서 차영상 임계값을 적용하여 화소 밝기가 임계값보다 높으면 밝기를 255로 설정하고 그렇지 않으면 0으로 설정한다.

$$D(x, y, t) = \begin{cases} 255, & \text{if } F(x, y, t) - F(x, y, t-1) > \text{threshold} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

시간 t 에서 모션 히스토리 이미지 $H_\tau(x, y, t)$ 는 다음 수식(2)와 같다.

$$H_\tau(x, y, t) = \begin{cases} \tau, & \text{if } D(x, y, t) = 255 \\ \max(0, H_\tau(x, y, t-1) - \Delta\tau), & \text{otherwise} \end{cases} \quad (2)$$

이때, τ 는 모션 히스토리 이미지에서 화소 밝기의 최대값을 의미하며, $\Delta\tau$ 는 프레임마다 줄어드는 밝기값을 말한다. 움직임이 일어나지 않은 화소는 모션 히스토리 이미지에서 매 프레임마다 $\Delta\tau$ 만큼 줄어든다. 이미지 내의 모든 x 와 y 에 대해, 차영상 $D(x, y, t)$ 의 화소값이 255이면 모션 히스토리 이미지는 τ 가 된다. 차영상의 화소값이 0인 경우 모션 히스토리 이미지 화소값은 0과 이전 프레임의 모션 히스토리 이미지 화소값에서 $\Delta\tau$ 를 뺀 값 중 큰 값으로 정한다.

각각의 프레임에서 일어난 모든 행동을 모션 히스

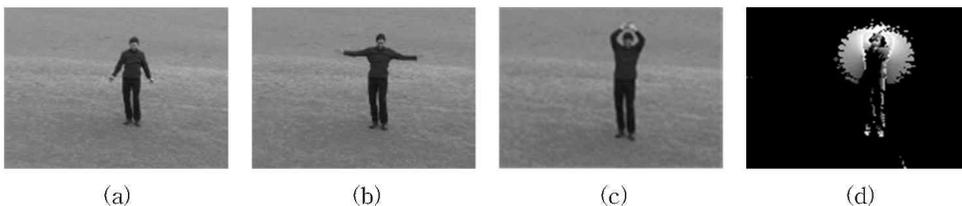


Fig. 2. 0th frame of hand waving person (a), 10th frame (b), 20th frame (c), motion history image from 0 to 20 frames (d).

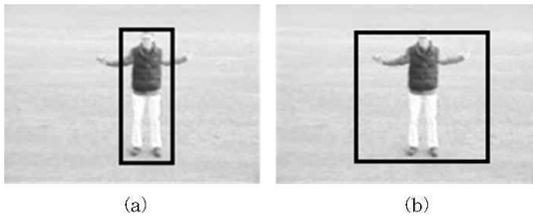


Fig. 3. (a) is the bounding box of the human body on KTH dataset, it does not cover entire human body, in (b), stretched bounding box cover entire human body.

토리 이미지에 표현하기 위해서는 사람의 움직임이 일어난 위치를 모두 포함되어야 한다. 영상에 YOLO 객체 위치 인식 알고리즘[13]과 Deep-SORT 객체 추적 알고리즘[14]을 적용하여 각각의 프레임에 대해 사람을 포함하는 바운딩 박스를 검출한다. Fig 3에서 검출된 사람의 위치는 팔과 같이 돌출된 부분이 제외된 몸통만을 검출하는 경우가 있기에 바운딩 박스를 높이와 너비중 길이가 긴 것을 선택하여 정사각형 형태로 변환한다.

Fig. 4와 같이 달리면서 바운딩 박스가 움직이는 경우, (a)와 같이 바운딩 박스의 크기가 충분히 크지 않으면 모션 히스토리 이미지가 프레임동안 일어난 모든 움직임을 포함하지 못한다. 따라서 (b)에서 검출된 모든 바운딩 박스를 포함하는 가장 작은 하나의 바운딩 박스를 계산하고 모션 히스토리 이미지를 추출한다.

C.B. Jin [8]은 모션 히스토리 이미지를 CNN으로 분류하여 사람의 행동을 인식하는 경우 1초에 25프레임으로 재생되는 영상에서는 25프레임마다 모션 히스토리 이미지를 생성했을 때 성능이 가장 높다는 것을 실험으로 보였다. 따라서 모션 히스토리 이미지 이미지를 생성하기 위한 단위시간을 25프레임으로 설정하였다. 단위시간마다 추출되는 모션 히스토리

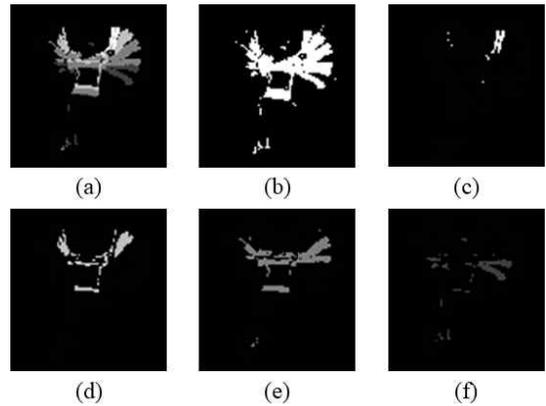


Fig. 5. Motion history image, (a) Source of MHI and (b) motion energy image of hand waving person; (c), (d), (e), and (f) denote the threshold results of MHI.

이미지에서 0이 아닌 화소의 개수를 계산하여 1-D 시계열 서열을 생성한다. Fig. 5는 손을 흔드는 사람의 모션 히스토리 이미지를 나타낸 것이다[10]. (a)는 원본 모션 히스토리 이미지이며, (b)는 0이 아닌 모든 화소의 밝기를 255로 설정한 모션 에너지 이미지 (motion energy image)이다. (c), (d), (e), (f)는 각각 화소 밝기값에 따라 임계값을 적용한 결과이다. (c)는 밝기값이 200을 초과하는 화소, (d)는 200 이하이며 140을 초과하는 화소, (e)는 140 이하이며 80을 초과하는 화소, (f)는 80 이하이며 20 초과인 화소만을 표시한 이미지이다.

Fig. 6과 Fig. 7은 행동이 손 흔들기에서 박수치기로 변할 때 모션 히스토리 이미지에서 0이 아닌 화소 개수의 변화를 나타낸 것이다. 행동이 바뀔 때 화소 개수의 급격한 변화가 일어난다. 검은색 수직선으로 표시된 부분이 행동이 바뀌는 지점이다. Fig. 6은 전체 모션 히스토리 이미지에서 0이 아닌 화소에 대한 변화에 대한 변화를 나타낸다. Fig. 7의 (a), (b), (c),

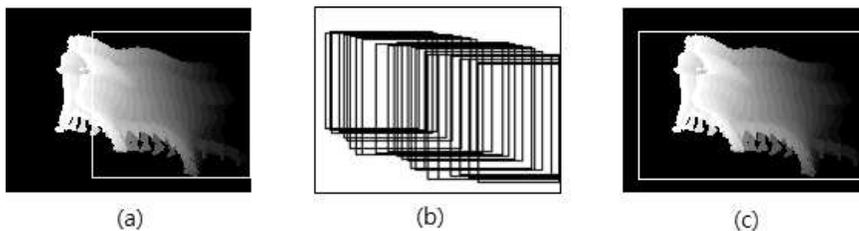


Fig. 4. (a) is an example of a bounding box that does not include the entire MHI, (b) is detected bounding boxes of multiple frames, (c) is a minimum bounding box containing all bounding boxes.

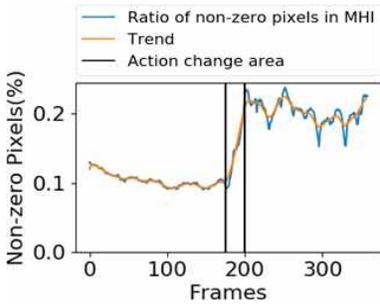


Fig. 6. Change in ratio of the number of pixels in MHI.

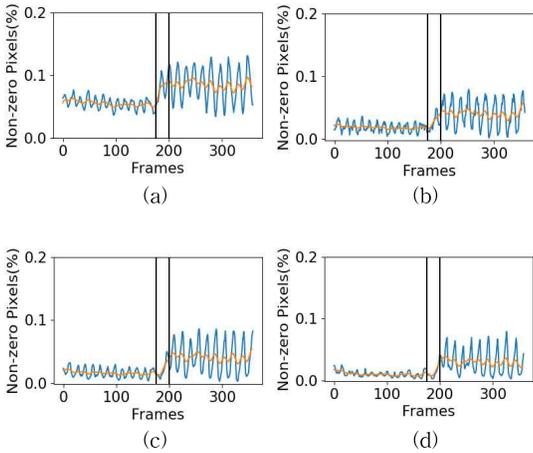


Fig. 7. Change in ratio of the number of pixels in MHI according to threshold value. greater than 200 (a), less than or equal to 200 and greater than 140 (b), less than or equal to 140 and greater than 80 (c), less than or equal to 80 and greater than 20 (d).

(d)는 각각 Fig. 3 의 (c), (d), (e), (f)와 같이 화소의 임계값이 200 초과, 200 이하 140 초과, 140 이하 80 초과, 80 이하 20 초과일 때에 대한 화소 개수의 변화를 나타낸다. 주황색 그래프는 윈도우 함수로 스무딩 된 결과이며 화소 개수 비율의 변화 경향을 나타낸다. 수직선으로 표시된 부분은 행동 변화가 일어나는 영역이다. Fig. 4에서 모든 화소값의 변화를 관찰했을 때 화소 개수의 최소값 및 최대값에 대비하여 가장 큰 변화가 일어난다. 따라서 이를 행동 변화 감지를 위한 시계열 데이터로 사용한다.

3.2 변화 감지

생성된 시계열 데이터는 Fig. 8과 같이 주기적으로 변화가 심하다. 같은 행동이 반복되는 동안에도

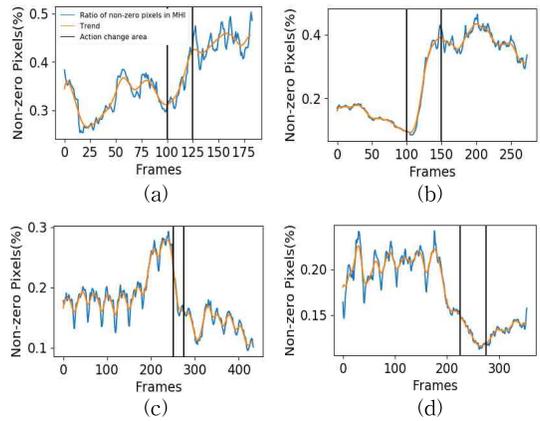


Fig. 8. Change in ratio of the number of pixels in MHI and trend. (a) is walking–running, (b) is box–ing–jogging, (c) is handwaving–walking, (d) is handwaving–boxing.

값이 크고 작게 증가와 감소를 반복한다.

데이터 변화의 경향을 나타내기 위하여 데이터 스무딩을 통해 이를 보정한다. 가장 보편적인 윈도우 함수인 하닝 윈도우(hanning window)를 사용하였다. 데이터는 하닝 윈도우를 이용한 가중 이동 평균을 이용하여 스무딩 된다. 단순 이동 평균이 아닌 가중 이동 평균을 사용하는 이유는 데이터가 평탄해지며 동시에 극점이 유지되기 때문이다. 스무딩한 결과는 데이터의 경향이라 한다. 다음 수식(3)은 n번째 하닝 윈도우 $w(n)$ 를 나타낸다. N 은 윈도우의 크기이다.

$$w(n) = \frac{1}{2} (1 - \cos(\frac{2\pi n}{N-1})) \tag{3}$$

Fig. 9는 윈도우 크기가 25인 하닝 윈도우를 나타낸 것이다. 모션 히스토리 이미지를 생성하기 위한 단위 시간이 25프레임이기 때문에 윈도우 크기도 25로 정한다. 윈도우 크기가 단위 시간보다 큰 경우에는 현재 프레임의 스무딩 결과를 얻기 위해 이후 프레임의 수치가 필요하기 때문에, 실시간으로 적용되

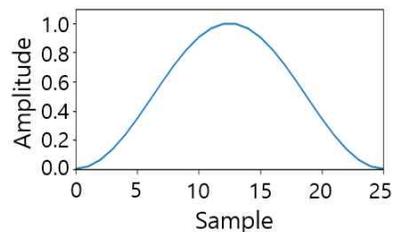


Fig. 9. Hanning window.

기 어렵다.

저수준 행동의 경우 모션 히스토리 이미지의 0이 아닌 화소 개수의 비율이 행동이 바뀌는 경우 크게 증가하거나 감소한다. 따라서 일반적인 개념 변화 감지 방법을 사용하는 대신에 경향이 변하는 특성을 이용하여 행동 변화를 감지한다. 급격한 변화를 탐지하기 위해서, 경향이 일정 값 이상 변하는 경우 행동이 변화한 것으로 판단한다. 데이터의 경향을 보면 행동이 변화하기 시작할 때와 끝날 때 극점이 생긴다. 따라서 임계값과 비교할 값은 시계열 데이터의 극점과 현재 경향간의 차이로 한다. 극점과 극점을 비교하지 않는 이유는 다음 극점이 언제 생길지 알 수 없기 때문이다.

Fig. 10은 경향을 이용한 행동 변화 감지의 흐름도를 나타낸 것이다. 먼저 임계값 θ 를 정하고, 가장 처음에 계산된 경향을 다음 경향과 비교할 기준 경향으로 정한다. 다음 프레임부터 경향을 구하고, 현재 경향과 기준 경향의 차이를 비교하여 임계값보다 높으면

행동에 변화가 일어났다고 판단한다. 그리고 기준 경향을 현재 경향으로 정한다. 만일 임계값보다 높지 않았다면 이전 시간의 경향이 극점인지를 확인한다. 극점이면 기준 경향을 이전 시간의 경향으로 정하고, 그렇지 않다면 기준 경향을 그대로 두고 위 과정을 영상의 마지막 프레임까지 반복한다.

다음은 현재 경향 $trend_t$, 기준 경향 $trend'$ 를 구하는 방법과 행동 변화 감지에 대한 위한 세부과정에 대해 설명한다. $ratio_t$ 는 시간 t 에서의 0이 아닌 모션 히스토리 이미지 화소 비율이며 다음과 같이 구한다.

$$ratio_t = \frac{Card(\{(x,y)|H(x,y,t) > 0\})}{W \cdot H} \quad (4)$$

이 때, W 는 모션 히스토리 이미지의 너비이며, H 는 높이이다. $ratio_t$ 모션 히스토리 이미지에서 0이 아닌 화소의 개수를 모션 히스토리 이미지의 넓이 $W \cdot H$ 로 나눈 값이다.

$trend_t$ 는 t 시간에서의 화소 비율 경향이다. 하닝 윈도우를 사용하여 가중 이동 평균으로 다음 수식으로 계산한다.

$$trend_t = \frac{ratio_t \cdot w(1) + ratio_{t+1} \cdot w(2) + \dots + ratio_{t+N-2} \cdot w(N-1) + ratio_{t+N-1} \cdot w(N)}{w(1) + w(2) + \dots + w(N-1) + w(N)} \quad (5)$$

다음은 행동 변화 감지를 판단하는 기준 경향 $trend'$ 수식이다.

$$|trend_t - trend'| > \theta \quad 0 < \theta < 1 \quad (6)$$

$$trend' = \begin{cases} trend_{t-1} & trend_{t-1} \text{ is local extremum} \\ trend_t & \text{action has changed} \\ trend' & \text{otherwise} \end{cases} \quad (7)$$

$trend'$ 는 변화 비율을 구하기 위한 기준값이 되며 t 시간에서 $trend_{t-1}$ 가 극대값 또는 극소값이거나, 행동 변화가 감지되었을 때 $trend_t$ 로 하고 그렇지 않으면 그대로 유지된다. 현재 경향과 기준값의 차이가 임계값 θ 보다 큰 경우에 행동 변화가 일어났다고 판단한다. 임계값 θ 는 0과 1사이의 값으로, 0이 아닌 화소의 비율이 얼마나 변해야 행동이 변한 것으로 판단할 것인지를 정한 고정된 값이다. 적절한 임계값 θ 를 정하는 것은 어렵다. 만약에 θ 가 너무 작으면 행동 변화가 일어나지 않았어도 경향 차이가 임계값보다 높아지는 경우가 많이 생기고, 반대로 θ 가 너무 크면 행동 변화가 일어났음에도 경향 차이가 임계값보다 적어 행동이 변화했다고 판단하지 못한다.

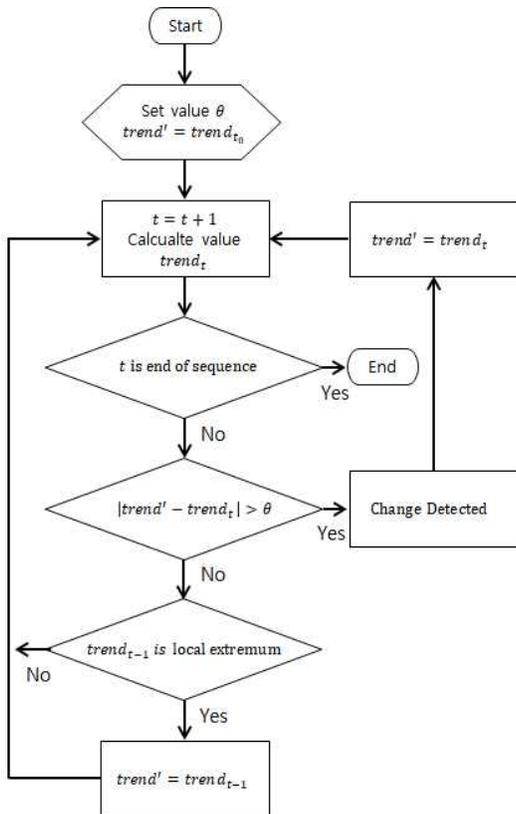


Fig. 10. flow chart of change detection.

4. 실험 결과 및 고찰

4.1 실험 환경 및 데이터

본 실험은 Intel(R) Core(TM) i5-4430 CPU@3.20 GHz의 PC와 Ubuntu 16.04 64비트 운영체제에서 수행하였다. 사용된 프로그래밍 언어는 Python이며 영상처리를 위해 OpenCV3.0.0 라이브러리를 사용하였다.

실험 영상은 자체 수집한 행동 변화 데이터 20개를 사용하였다. 저수준 행동 분류를 위한 대표적인 데이터셋인 KTH[15]는 행동이 변화하지 않고 반복되기 때문에 행동 변화 감지 성능을 측정할 수 없다. 따라서 KTH 데이터셋에서 제공하는 걷기, 달리기, 조깅, 주먹질, 박수치기, 손 흔들기의 총 6가지 분류의 행동이 중간에 변하는 데이터셋을 사용하였다. 데이터 하나당 한번의 행동 변화가 일어난다. 해상도는 640 × 480 화소이며, 프레임율은 25fps이다. Fig. 11

은 행동 변화 데이터의 예시를, Table 1은 행동 변화 데이터 20개의 구성을 나타낸 것이다.

4.2 성능 평가 방법

행동 변화 데이터의 정답은 Fig. 12와 같이 구성되어 있다. 행동 변화 감지 알고리즘의 성능 평가시, 25프레임을 기준 단위로 하여 행동의 변화 유무를 확인한다. 실제 행동이 변하지 않았고 알고리즘도 행동이 변하지 않았다고 판단할 경우 true negative, 이며 변했다고 판단하면 false positive이다. 실제 행동이 변하는 경우, 여러 단위에 걸쳐 행동 변화가 일어나도 최소 한 단위 이상 행동 변화가 일어났다고 판단할 경우 true positive, 한 단위도 변화를 감지하지 못하는 경우 false negative이다. 데이터의 특성상 행동이 변하지 않는 프레임이 변하는 프레임 보다 많기 때문에, 성능 평가 척도로 민감도(Sensitivity)



Fig. 11. Action change data.

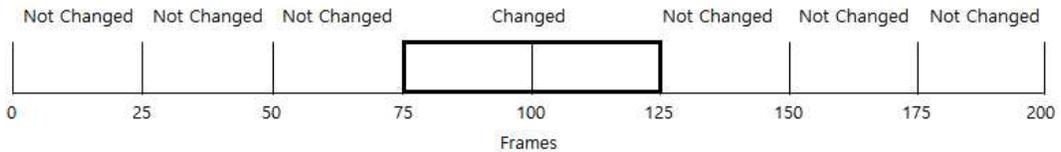


Fig. 12. Ground truth of the action change data.

Table 1. Description of action change data

Data Number	Previous Action	Next Action	Data Number	Previous Action	Next Action
1	Boxing	Hand clapping	11	Jogging	Boxing
2	Boxing	Hand waving	12	Jogging	Running
3	Boxing	Jogging	13	Jogging	Walking
4	Boxing	Walking	14	Running	Jogging
5	Hand clapping	Boxing	15	Running	Walking
6	Hand clapping	Hand waving	16	Walking	Boxing
7	Hand clapping	Walking	17	Walking	Hand clapping
8	Hand waving	Boxing	18	Walking	Hand waving
9	Hand waving	Hand clapping	19	Walking	Jogging
10	Hand waving	Walking	20	Walking	Running

와 특이도(Specificity)의 평균인 균형화된 정확도(Balanced Accuracy)를 사용한다.

행동 인식 메소드에 행동 변화 감지 방법 적용시 예상되는 행동 인식 메소드의 반복 횟수 감소 비율 및 정확도 감소 비율도 성능 평가 척도로 사용한다. Fig. 13, Fig. 14와 Fig. 15는 영상에서 단위 시간당 행동 인식 메소드가 수행되는 것에 대한 모의실험이며 Table 2는 각각의 모의실험에 대한 실험 결과를 나타내고 있다. Fig. 13에서는 매 단위 시간마다 행동 인식 메소드가 수행된다 횟수는 총 8번이다. Fig. 14

와 같이 행동 변화가 일어나는 경우에만 행동 인식 메소드를 수행하는 경우 사람의 행동이 무엇인지 모르는 첫 번째 단위 시간에 메소드가 수행되고, 이후에 행동 변화가 일어나는 경우에만 행동 인식 메소드가 수행된다. 총 3번의 행동 인식만이 수행된다. Fig. 15에서 행동 변화 감지시 false positive가 발생하는 경우 행동 인식은 틀리지 않지만 행동 변화가 없는데도 행동 인식을 수행하여 같은 행동을 다시 인식한다. 행동 변화가 일어나는 경우에는 행동 인식 메소드가 수행되어 변화된 행동을 인식해야 하지만 false

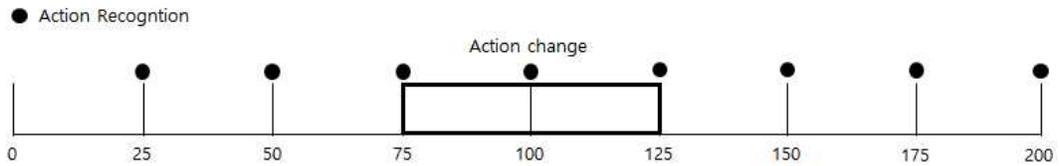


Fig. 13. Action recognition without action change detection.

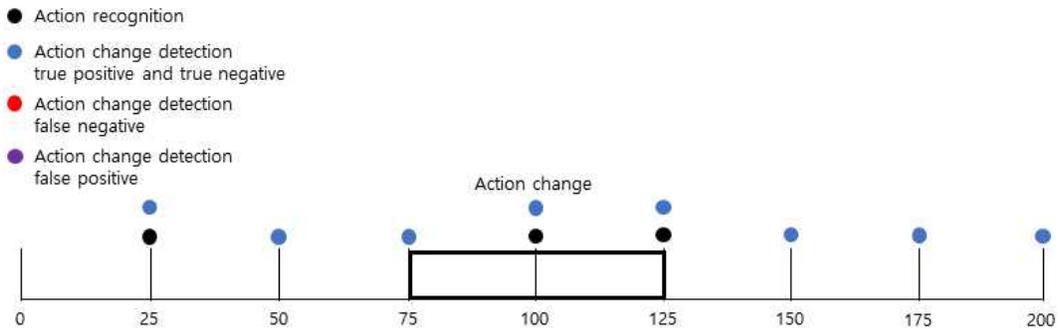


Fig. 14. Action recognition with ideal action change detection.

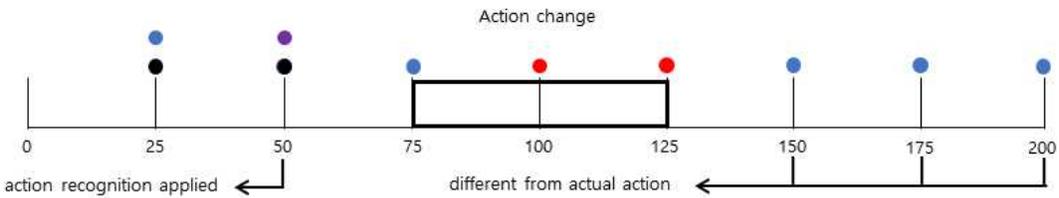


Fig. 15. Action recognition with fault of the action change detection.

Table 2. The number of action recognition repetition comparison simulation

Method	Action recognition task repetition	Action recognition fault
Action recognition	8	0
Action recognition + action change detection(ideal)	3	0
Action recognition + action change detection(fault)	2	3

negative가 발생하면 이를 수행하지 않아 정확도가 감소한다. 행동 인식은 2번만 반복되었지만 행동 변화가 탐지될 때 까지 단위 시간마다 행동 인식을 틀리게 된다. 행동 변화가 끝난 시점부터 총 3번의 행동 인식이 틀렸다.

4.2 임계값에 따른 실험 결과

Fig. 16, Fig. 17과 Table 3은 임계값 θ 에 대한 행동 변화 감지 방법의 성능을 나타내고 있다. 임계값은 0.01에서 1.0까지 0.01씩 변화시키며 성능을 평가하였다. Fig. 16은 임계값에 따른 균형화된 정확도이다. 임계값이 0.06인 경우 균형화된 정확도가 0.89로 가

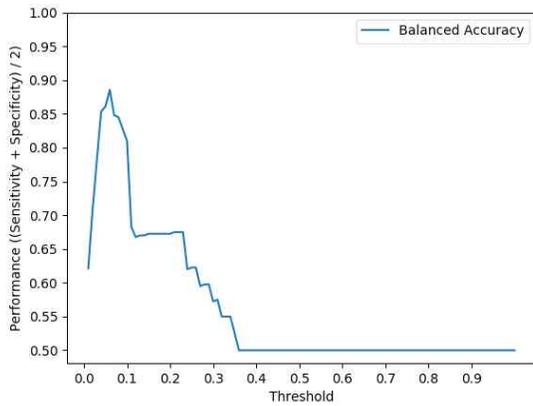


Fig. 16. Balanced Accuracy of the action change detection method on action change data according to threshold.

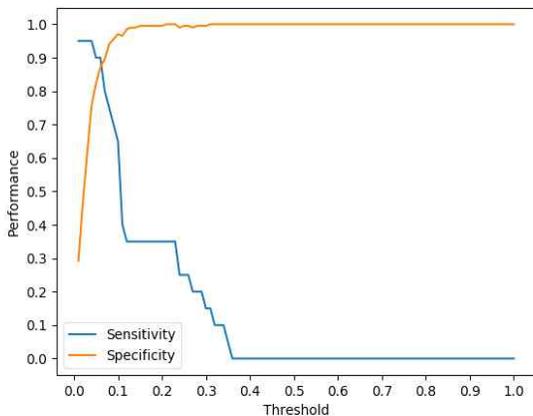


Fig. 17. Sensitivity and Specificity of the action change detection method on action change data according to threshold.

Table 3. Results of action change detection on action change data

Threshold	Balanced Accuracy	Sensitivity	Specificity
0.03	0.78	0.95	0.61
0.04	0.85	0.95	0.76
0.05	0.86	0.9	0.82
0.06	0.89	0.89	0.87
0.07	0.85	0.8	0.9
0.08	0.85	0.75	0.94
0.09	0.83	0.7	0.96
0.1	0.81	0.66	0.97

장 높게 나타났다. Fig. 17은 임계값에 따른 민감도와 특이도를 나타낸다. 임계값이 높을수록 행동이 변하지 않을 때에 대한 true negative가 증가하여 특이도가 높아지지만 행동이 변하는 경우에 대한 true positive가 감소하여 민감도가 낮아진다. 임계값이 낮으면 반대로 true positive가 증가하지만 true negative가 감소한다. 목적에 따라 임계값을 적절히 선택할 수 있다. 확실한 행동 변화를 감지하는 것이 중요한 경우 임계값을 작게 설정하고 연산량을 줄이는 것이 더 중요하면 위해서는 임계값을 크게 정할 수 있다.

Table 4는 행동 인식 메소드가 행동 변화 감지시에만 수행된다고 가정하였을 때 줄어드는 수행 횟수를 와 감소되는 행동 인식률을 임계값에 따라 나타내고 있다. 전체 데이터에 대해 행동 변화를 감지하지 않을 때 수행되는 행동 인식은 총 268번이다. 임계값이 낮을수록 감소되는 반복 횟수는 줄어들지만 행동 인식이 틀리는 횟수가 줄어드는 모습을 보인다. 임계

Table 4. Results of reducing repetitive human action recognition task on action change data

Threshold	Reduced action recognition repetition	Action recognition fault
0.03	111(41.4%)	1
0.04	135(50.4%)	1
0.05	156(58.2%)	3
0.06	165(61.6%)	3
0.07	172(64.2%)	6
0.08	176(65.7%)	9
0.09	178(66.4%)	12
0.1	179(66.8%)	16

값이 높을수록 감소되는 반복 횟수가 늘어나지만 행동 인식이 틀리는 횟수가 늘어나는 경향을 보인다.

5. 결 론

본 논문에서는 행동 인식의 연산량을 줄이기 위한 저수준 사람 행동 변화 감지 방법에 대해 다루었다. 비디오에서 사람의 영역이 주어지는 경우 모션 히스토리 이미지를 추출하여 이로부터 0이 아닌 화소 추이로 시계열 데이터를 나타낸다. 시계열 데이터의 변화 경향을 구하여 이를 행동 변화 여부를 결정하는 척도로 사용하였다. 제안한 방법은 직접 수집한 행동 변화 데이터에 대해 실험하였다. 총 20개 데이터에 대해 89%의 균형화된 정확도를 내었다. 임계값을 어떻게 설정하느냐에 따라 행동이 변하는 경우에 행동이 변하지 않는 경우에 대한 각각의 성능이 차이가 나고 행동 인식 반복 횟수와 행동 인식률 목적에 따라 적절한 행동 변화 비율을 선택해야 한다. 반드시 행동의 변화를 탐지해야 한다면 임계값을 작은 값으로 설정하여, 오탐지를 감수하고 행동 변화를 높은 정확도로 탐지할 수 있다. 연산량 감소에 목적을 두고 큰 행동 변화만 탐지하고자 한다면 임계값을 큰 값으로 설정하여 탐지한다.

향후 연구로 실제 행동 인식 방법과 행동 변화 감지 방법을 결합하여 행동 인식 정확도가 얼마나 떨어지는지에 대한 실험을 진행하고자 한다. 그리고 임계값은 고정된 값이므로 상황에 따라 행동 인식 결과가 달라질 수 있기에 상황에 따라 변하는 적응형 임계값에 대한 연구를 진행하여 제안한 방법과 성능 비교를 추후 수행하고자 한다. 현재 행동 학습에서 학습 단위를 추출하는 것이 중요하다. 따라서 제안한 방법을 활용하여 행동 학습에 적용하는 것도 향후 연구이다.

REFERENCE

[1] J.K. Aggarwal and M.S. Ryoo, "Human Activity Analysis: A Review," *ACM Computing Surveys*, Vol. 43, No. 3, pp. 1-43, 2011.

[2] G. Cheng, Y. Wan, A.N. Saudager, K. Namuduri, and B.P. Buckles, "Advances in Human Action Recognition: A Survey," *arXiv preprint arXiv:1501.05964*, 2015.

[3] I. Laptev, "On Space-time Interest Points,"

International Journal of Computer Vision, Vol. 64, Issue 2-3, pp. 107-123, 2005.

[4] H. Wang, A. Kläser, C. Schmid, and C.L. Liu, "Action Recognition by Dense Trajectories," *Proceeding of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3169-3176, 2011.

[5] A.F. Bobick and J.W. Davis, "The Recognition of Human Movement Using Temporal Templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 3, pp. 257-267, 2001.

[6] J.W. Davis and A.F. Bobick, "The Representation and Recognition of Human Movement Using Temporal Templates," *Proceeding of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 928-934, 1997.

[7] Y. Shi, Y. Tian, Y. Wang, and T. Huang, "Sequential Deep Trajectory Descriptor for Action Recognition with Three-stream CNN," *IEEE Transactions on Multimedia*, Vol. 19, No. 7, pp. 1510-1520, 2017.

[8] C.B. Jin, S. Li, and H. Kim, "Real-time Action Detection in Video Surveillance Using Sub-action Descriptor with Multi-CNN," *arXiv preprint arXiv:1710.03383*, 2017.

[9] I.H. Kwon, H. Boubenna, and D.H. Lee, "A Study on Efficient Learning Units for Behavior-recognition of People in Video," *Journal of Korea Multimedia Society*, Vol. 20, No. 2, pp. 196-204, 2017.

[10] Y.H. Noh and D.H. Lee, "Low-level Human Action Change Detection Using the Motion History Image," *Proceeding of the 9th International Conference on Software and Computer Applications*, pp. 311-314, 2019.

[11] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with Drift Detection," *Proceeding of Brazilian Symposium on Artificial Intelligence*, pp. 286-295, 2004.

[12] A. Bifet and R. Gavaldá, "Learning from

Time-changing Data with Adaptive Windowing,” *Proceeding of the 2007 Society for Industrial and Applied Mathematics International Conference on Data Mining*, pp. 443-448, 2007.

- [13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only Look once: Unified, Real-time Object Detection,” *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779-788, 2016.
- [14] N. Wojke, A. Bewley, and D. Paulus, “Simple Online and Realtime Tracking with a Deep Association Metric,” *Proceeding of IEEE International Conference on Image Processing*, pp. 3645-3649, 2017.
- [15] C. Schuldt, I. Laptev, and B. Caputo, “Recognizing Human Actions: a Local SVM Approach,” *Proceeding of the 17th IEEE International Conference on Pattern Recognition*, Vol. 3, pp. 32-36, 2004.



노 요 환

2016년 부산대학교 정보컴퓨터공학부 학사
 2018년 부산대학교 전기전자컴퓨터공학과 석사
 2019년~현재 부산대학교 전기전자컴퓨터공학과 박사과정

관심분야: 영상처리, 딥러닝, 행동인지



김 민 정

1991년 부산대학교 계산통계학과 졸업
 1993년 부산대학교 전자계산학과 석사
 1997년 부산대학교 전자계산학과 박사

1997년~2005년 밀양대학교 컴퓨터공학과 교수
 2006년~현재 부산대학교 교양교육원 교수
 관심분야: 자연언어처리, 인공지능, 행동인지



이 도 훈

1986년 부산대학교 계산통계학과 학사
 1992년 부산대학교 전자계산학과 석사
 1998년 부산대학교 전자계산학과 박사

1995년~2005년 밀양대학교 컴퓨터공학과 교수
 2004년~2005년 Indiana University 방문연구원
 2006년~현재 부산대학교 정보컴퓨터공학부 교수
 관심분야: 시각정보분석, 행위인지, 생체인식, 딥러닝