

재활 훈련 서비스를 위한 실시간 모션인식 알고리즘

최 완[†], 김태영^{**}, 임철수^{***}

요 약

본 논문은 모션 인식 분야 중 지체 장애인의 재활훈련과 같은 치료 분야를 위한 실시간 모션 인식 알고리즘을 제안한다. 본 방법은 빠르고 정확한 모션인식을 위하여 사용자의 특징점 위치에 적외선 LED 밴드를 착용하고 적외선 필터가 장착된 카메라로 좌표 값을 얻어낸 후 미리 정의되어 있는 동작과의 일치 여부를 판별한다. 또한 적외선 LED의 특징점 좌표 들을 꼭지점으로 가지는 다각형의 무게중심과 특징점 좌표들의 벡터를 구한 후 일정한 거리와 각도 구간으로 구분하고 각 LED의 속한 구간을 알아낸다. 이를 이용하여 삼육 재활학교에서 구성한 재활훈련 시나리오의 동작에 대하여 실험을 수행한 결과 초당 20프레임 이상의 빠르면서도 98%에 가까운 인식률을 얻을 수 있다.

A Real-Time Motion Recognition Algorithm for a Rehabilitation Service

Wan Choi[†], Tae-Young Kim^{**}, Cheol-Su Lim^{***}

ABSTRACT

The objective of this paper is to develop the real time motion recognition method for the rehabilitation program of the people with physical disabilities. Using the infrared ray LED, this method gets the real time information of the user's motion and tells the difference between the information we get and the pre-defined information. In order to get the fast recognition and comparison, this method positions the specific coordinates of infrared ray LED in the pre-classified sector which keeps certain distance and angle from the point of origin. The result in our experiment shows the speed over 20 frames per second in every motion, and we are satisfied the rate for the recognition of the motions which is near 98%.

Key words: Infrared ray LED(적외선 LED), Real-Time Motion Recognition(실시간 모션 인식)

1. 서 론

본 연구는 전 국민의 10%를 차지하는 장애인들 중 대부분의 비율을 차지하는 지체 장애인들의 재활 훈련 서비스를 위한 실시간 모션 인식 처리 알고리즘을 제시한다. 장애인의 숫자는 매년 증가하지만 현재 장애인을 대상으로 한 재활 훈련 프로그램은 다른 나라에 비하여[1] 거의 전무한 상태이며 장애인의 수

자에 비하여 재활교사의 비율이 낮기 때문에 재활훈련 프로그램의 필요성은 극히 높다. 이는 직접 또는 비디오를 이용한 재활훈련은 재활교사가 지속적으로 훈련동작을 점검해야 하고 사용자는 한가지 시점으로만 훈련동작을 확인하게 되지만 재활훈련 프로그램 사용 시 사용자는 여러 시점으로 훈련동작을 확인할 수 있어 동작에 대한 이해도가 높아지게 되며 재활교사는 프로그램의 자체 평가로 인하여 다수의

* 교신저자(Corresponding Author) : 김태영, 주소 : 서울특별시 성북구 정릉동 16-1(136-704), 전화 : 02)913-1362, FAX : , E-mail : tykim@skuniv.ac.kr

접수일 : 2007년 5월 8일, 완료일 : 2007년 8월 7일

[†] 정회원, 서경대학교 컴퓨터공학과

(E-mail : wanne@skuniv.ac.kr)

^{**} 정회원, 서경대학교 컴퓨터공학과

^{***} 정회원, 서경대학교 컴퓨터공학과

(E-mail : cslim@skuniv.ac.kr)

* 본 연구는 문화관광부 및 한국문화콘텐츠진흥원의 문화콘텐츠기술연구소(CT)육성사업의 연구결과로 수행되었음

훈련자를 관리할 수 있게 되는 것으로 재활훈련의 효율이 높아지게 된다. 특히 뇌졸중, 소아마비 등으로 움직임에 제한이 있는 지체 장애인의 재활훈련 프로그램의 경우 지속적인 반복 훈련이 필요하며, 빠르게 결과를 확인해야 하는 특성상 실시간 처리는 반드시 필요하다. 또한 지체 장애인들은 정확한 움직임을 하기 힘들기 때문에 재활 훈련을 위해서 상황에 따른 모션 인식의 필요성이 크다.

기존의 모션 인식 연구들[2-8]은 대부분 비장애인을 대상으로 동작 데이터를 얻어낸다. 그 이유는 정확한 동작 데이터를 얻어냄으로써 사용자의 동작 인식의 정확도를 높이기 위해서 이다. 그러나 이것은 장애인을 대상으로 하는 재활훈련에는 적합하지 않다. 이 중 사람의 걷는 동작을 인식하는 방법의 경우, 신체의 중요 관절 14곳에 센서를 장치하는데[2] 고가의 센서를 사용하지만 장비에 전선이 연결되어 장애인의 행동에 지장을 주고, 장치된 각 센서의 이동과 좌표축의 기울기를 서로 다른 센서와 비교하여 위치를 추적하는 것을 개별적으로 계산해야하므로 실시간 처리가 어렵다는 제한이 있다. 또한 단일 카메라를 이용하여 신체의 부분 별로 나누어 인식하는 방법의 경우[3] 이미 가지고 있는 모델을 신체에 윤곽을 맞춘 후 구역을 맞추는 것으로 추적속도는 빠르지만 수동으로 초기화를 해주어야 하며 사람의 걷는 옆모습만을 추적할 수 있다. 또한 신체의 반대쪽 부위가 가려졌다 나타나는 경우 오류율이 큰 편이라는 문제가 있다. 센서와 카메라를 동시에 이용한 모션 인식 방법[4]은 센서와 카메라로 촬영한 영상을 이용하여 3차원 깊이 정보와 센서와 영상의 정보를 구하여 3D 모델로 표현한 후 각 정보를 종합하는데 카메라는 신체 움직임의 양 손끝과 머리의 위치를 알아내고, 센서는 신체의 윤곽을 알아내는데 쓰이게 된다. 이때 쓰이는 장비가 고가일 뿐만 아니라 카메라로 획득한 정보는 정확하지 않은 동작으로 인식할 확률이 높기 때문에 센서로 획득한 정보와 다를 경우 정확한 모션 인식이 되지 않을 경우가 있으므로 본 연구의 취지에 적합하지 않다.

여러 카메라를 사용하여 각 카메라에서 얻어진 영상들을 이용한 모션 인식 방법[5,6]은 각 카메라의 영상들의 정보를 얻어 종합하게 되는데, 각 카메라에서 대상의 윤곽까지의 깊이 값을 획득하여 종합함으로써 대상의 윤곽을 추정한다. 이때 추정하는 절차의

계산이 복잡하여 인식까지의 시간이 오래 걸리게 되므로 실시간으로는 사용할 수 없다. 반면 실시간으로 모션 인식을 하는 방법[7,8]은 빠른 모션 인식을 위하여 인식하는 신체 부위를 상반신만으로 제한을 두고 부위별로 미리 정의되어 있는 모델에 맞추어 대상을 인식한다. 이것은 인식 자체는 빠르게 되지만 장애인 재활훈련은 신체 전체를 인식해야하므로 본 연구의 취지에는 부합하지 않으며 여러 장면을 획득하여 움직임을 추정하고 그 평균치를 이용하여 오류를 줄이기 때문에 비슷한 동작으로 인식하지만 정확한 동작으로 인식하지 못하는 경우도 있으므로 장애인 재활훈련 프로그램에 적합하지 않다. 이와 같은 이유들로 지체장애인을 위한 재활 훈련 프로그램에 기존 연구를 적용하기에는 문제가 존재한다.

본 논문에서는 기존 연구에서 사용된 센서와 카메라를 사용하되 사용자의 움직임에 제한을 주는 센서 대신에 가볍고 정확한 데이터를 얻어낼 수 있는 무선 적외선 LED 밴드를 사용하여 정확하고 빠르게 모션 인식을 할 수 있는 기법을 제안한다. 이것은 적외선 필터를 거쳐 배경과 잡음이 배제된 적외선 LED 좌표값을 입력으로 받아 각 LED의 위치에 따라서 사용자의 동작을 결정하는 방법이다. 이 방법은 LED의 위치를 구간 별로 나누어 어떤 구간에 위치하는가를 계산 하므로 별도의 깊이(Depth)값을 얻어내거나 여러 카메라를 이용하여 얻은 값들의 동기화가 필요하지 않으므로 모션 인식 속도가 빨라지는 장점이 있다. 이것은 재활훈련에 사용되는 동작들이 팔을 들어올리는 등의 2차원적인 비교로 동작의 인식이 용이한 동작이기 때문이다. 그러나 LED 좌표값 비교 방법은 좌표값에 의존하기 때문에 카메라의 위치 변화에 따라 다른 좌표값을 얻을 수도 있고, 이로 인해 정확하지 못한 결과 값을 낼 수도 있다. 여기서는 단순히 LED 좌표값만으로 동작을 인식하지 않고 이를 이용하여 일정한 구간으로 나눈 후 LED좌표의 구간값을 알아내어 카메라의 위치변화뿐만 아니라 다른 외부 변화에 의하여 결과값의 변화를 방지하는 방법을 제안한다. 또한 구간이 넓어 모션인식의 정확도가 낮은 경우를 방지하기 위하여 바깥쪽 일수록 넓어지는 비교 구간을 조밀하게 하여 정확한 모션 인식을 가능케 하였다. 실험 결과, 제안한 방법은 2차원 LED 좌표 이외의 값은 필요 없고 모션 인식을 할 때 구간 비교만을 수행 하므로 기존 방법보다 속

도상의 성능이 높아짐을 확인 할 수 있었다.

본 논문에서는 1장의 서론에 이어 2장에서는 본 연구에서 제시하는 시스템의 구조에 대한 개괄적인 설명을 하며 3장에서는 모션 인식 및 매칭 알고리즘에 대하여 자세히 살펴본다. 4장에서는 본 연구의 시스템에서 실험한 결과와 성능에 대하여 기술한 후 마지막으로 5장에서는 본 연구의 결론에 대하여 기술한다.

2. 지체장애인을 위한 재활 훈련 시스템

본 연구의 지체장애인 재활 훈련 시스템은 가상 환경 상에서 지체장애인의 재활에 필요한 기본동작(사용자의 근력 강화 및 보완, 유연성 및 균형감각 향상)을 보며 교사의 도움이나 공간의 제약 없이 반복훈련을 수행할 수 있어 훈련 능률 향상에 도움을 준다.

그림 1은 본 연구에서 제시하는 재활훈련 시스템의 구조로 다음과 같이 5개의 구성으로 이루어진다.

2.1 인터페이스 모듈(Interface Module)

인터페이스모듈은 데이터베이스를 통해 사용자 로그(신상기록, 훈련기록 등)를 관리하며 사용자 별로 훈련 과정을 설정하고 사용자의 훈련 동작, 모범 동작 및 기타 훈련에 관한 정보를 화면에 표시하는 역할을 한다. 또한 훈련 설정 정보를 주 제어 모듈에 전달하여 현재 훈련에 대한 사용자의 평가 결과를 확인할 수 있다. 인터페이스 모듈의 주 기능은 사용자의 정보 관리와 훈련 세부 설정 및 디스플레이로 구분된다.

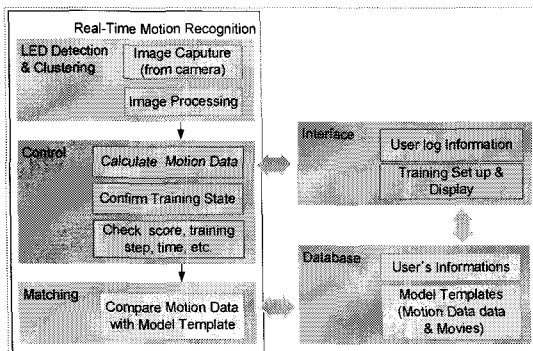


그림 1. 재활훈련 시스템 구조도

2.2 특징점 인식 모듈(LED Detection & Clustering Module)

사용자의 훈련 동작과 모범 동작의 비교를 위한 특징점을 획득하기 위한 처리 과정이다. 적외선 필터가 부착된 카메라를 통해 훈련 동작과 모범 동작의 비교를 위한 특징점을 획득하기 위한 처리 과정이다. 적외선 필터가 부착된 카메라를 통해 사용자의 동작영상을 획득하고 이를 영상 처리하여 사용자가 착용한 적외선 LED 밴드의 위치(특징점)를 찾아낸다. 특징점 인식 모듈은 영상 획득을 위한 카메라 장치 관련 설정과 영상 처리(Image Processing)로 구분된다.

2.2.1 영상 획득(Image Capture)

영상 획득 부에서는 응용 프로그램에서의 카메라 장치 획득 및 선택이 이루어진 후 카메라 장치를 초기화 시킨다. 이후 일정 단위로 장면을 캡처하게 된다.

2.2.2 영상 처리(Image Processing)

획득한 영상을 미리 정의된 밝기 값으로 1차 영상 처리(영상 노이즈 제거)하여 LED 검출한 후 특징점이 되는 적외선 LED 밴드의 착용점을 구별하기 위해 Clustering 처리(K-Means 알고리즘을 이용하여 일정한 거리를 가진 LED 좌표를 그룹화)를 하게 된다.

2.3 주 제어 모듈(Main Control Module)

입력된 LED 좌표(특징점)를 기반으로 모델 템플릿과 매칭을 하기 위한 준비와 매칭 후의 정보를 처리한다. 5개의 특징점을 각 꼭지점으로 하는 다각형의 무게중심 좌표를 계산하고 무게 중심좌표와 각 특징점 간의 거리, 각을 구하여 동작 비교를 위한 데이터로 사용한다. 또한 매칭 결과에 따른 훈련 결과 정보를 업데이트한다.

주 제어 모듈은 입력된 특징점 좌표를 이용하여 중심 좌표 및 거리, 각을 구하는 동작 데이터 계산과 훈련 상태 확인 및 업데이트로 구분할 수 있다.

2.3.1 동작 데이터 계산

그룹화 된 LED 좌표(특징점)로 구성되는 다각형

의 무게 중심 좌표를 결정한 후 무게 중심 좌표와 각 LED 좌표들 간의 거리 및 각도를 계산하게 된다. 이후 계산 결과 가장 큰 값을 기준으로 거리구간을 5등분하고 각 LED가 위치하는 구간을 계산하게 된다. 마지막으로 무게 중심점을 원점으로 하는 2차원 평면상의 각(360°)을 16등분하고 각 LED가 위치하는 구간을 계산하는 것으로 LED 구간을 결정하게 된다.

2.3.2 훈련 상태 확인

응용 프로그램에서 선택된 사용자의 훈련 메뉴에 따라 실행 하게 될 훈련의 최근 세 동작을 결정한 후 매칭에서의 결과에 따른 점수, 훈련 동작, 현재 시간을 확인 및 업데이트 하게 된다. 또한 중간 목표 매칭 성공 시 다음 동작 매칭, 최종 목표 매칭 성공 시 종료 메시지를 전달하고 매칭 실패 시는 실패 메시지를 전달한다.

2.4 매칭 모듈(Matching Module)

주 제어 모듈에서 계산된 무게 중심좌표와 각 특징점 간의 거리, 각도를 이용하여 동작비교를 수행한다. 모델 템플릿에서 매칭 될 동작 정보를 검색하여 얻어온 후 주 제어 모듈에서 계산된 요소들과 비교를 통해 다시 주 제어 모듈로 매칭 결과를 통보한다.

2.4.1 모델 템플릿과 현재 동작 데이터와의 비교

주 제어 모듈에서 특징점 좌표를 통해 계산 된 요소들과 함께 동작 메뉴 정보를 전달 받은 매칭 모듈은 모델 템플릿에서 비교할 동작 데이터를 획득한 후 동작 데이터와 주 제어 모듈에서 넘겨받은 요소들을 비교한 후 남아있는 동작 데이터에 따라 중간 목표 달성과 최종 목표 달성을 구분한다.

2.5 데이터베이스(Database)

데이터베이스 모듈은 사용자(지체장애인)의 신상 기록, 훈련 기록, 훈련 결과 평가 정보 등의 로그 정보와 각 훈련 항목에 대한 모범 동작(동영상, 3D 모델링), 동작 비교를 위한 모범 동작의 데이터(Model Template) 저장 매체이다.

3. 실시간 모션 인식 알고리즘

그림 2는 실시간 모션 인식을 위한 알고리즘의 처리 과정이다. 단계 1에서는 적외선 LED를 검출하고 그룹화 시켜 대표 특징점 좌표를 얻어내는 과정이고, 단계 2에서 전 단계에서 얻은 특징점 좌표를 이용하여 무게 중심을 구한 후 이를 원점으로 한 각 좌표들의 벡터값을 계산한다. 단계 3은 계산된 벡터를 기반으로 각각을 거리 구간과 각도 구간으로 나눈다. 마지막으로 단계 4에서는 단계 3에서 구해진 거리 및 각도 구간과 템플릿의 동작 데이터가 일치 하는가를 비교 한다. 비교 결과 일치 경우 일치 여부를 알리며 불일치의 경우 제한된 시간의 한도 내에서 다시 단계 1부터 반복한다.

본 방법에서는 실시간으로 모션 인식 및 비교를 하기 위하여 각 좌표들의 구간을 간결하면서도 정확한 비교를 할 수 있도록 나누었기 때문에 단계 4의 템플릿의 동작 데이터들과의 비교 시간을 감소 시킨다.

3.1 각 LED 좌표 획득 및 클러스터링(LED Detection & Clustering)

그림 3(왼쪽)와 같이 사용자의 LED를 검출하기 위해 카메라에 적외선 필터를 장치하여 LED 외의 밝은 부분과 배경을 제거하였다. 검출된 LED는 각 밴드 중 여러 개(6~8)이므로 이 중 1개의 대표중심 좌표만을 얻기 위해 K-Means 알고리즘[9]을 적용 하였다.(그림3의 오른쪽)

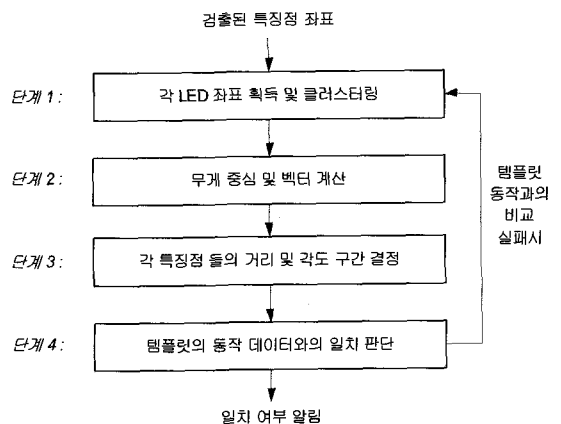


그림 2. 모션 인식 처리 과정



그림 3. LED 디텍션 과 클러스터링

신체 부위의 통합 혹은 교차 현상 및 해결책으로 는 신체 각 부위의 영상에서의 위치를 그림4와 같은 운동방정식(dynamics)을 통하여 추적하고 예측함으로써 부위가 가까워져도 오류가 발생하지 않도록 독립적으로 처리하였다. 즉, 신체의 움직임은 임의의 한 방향으로 운동할 때, 계속해서 그 방향으로 운동하고자 하는 성질을 운동방정식을 통해서 적용한 것이다. 이러한 공식 (1)을 이용한 운동방정식을 통한 추적 및 예측은 교차현상(신체의 한 부위가 다른 부위와 순간적으로 겹쳐 뒤바뀌어 인식되는 현상)을 해결하는데도 유용하게 사용되었다.

$$\begin{array}{ll}
 X_m = aX_1 + b & Y_m = cY_1 + d \\
 X_1 = aX_2 + b & Y_1 = cY_2 + d \\
 a = (X_m - X_1)/(X_1 - X_2) & c = (Y_m - Y_1)/(Y_1 - Y_2) \\
 & \downarrow \text{상수 a,b,c,d 계산} \\
 X_e = aX_m + b & Y_e = cY_m + d
 \end{array}
 \tag{1}$$

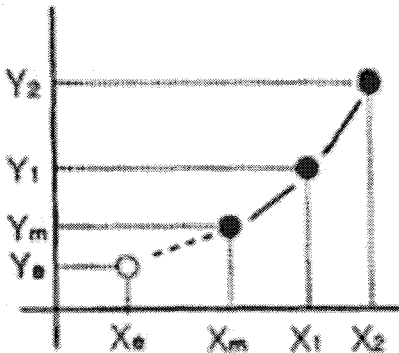


그림 4. 운동방정식을 이용한 각 부위의 추적 결과

- $(X_1, Y_1), (X_2, Y_2)$: 3 frame 차이로 측정된 이전의 clustering 좌표값
- (X_e, Y_e) : 예상되는 clustering 좌표값
- (X_m, Y_m) : 현재 측정된 clustering 좌표값

3.2 무게 중심 계산 및 구간 결정

사용자의 모션인식을 위하여 클러스터링 된 LED 좌표 간의 관계 데이터를 획득한다. 각 LED(특징점) 좌표가 이루는 다각형의 무게 중심을 계산한 후 무게 중심 좌표를 기준으로 각 LED의 벡터 값(거리, 각도)을 측정된 후 이를 기반으로 거리 및 각도 구간을 결정하는 알고리즘을 제시한다.

3.2.1 무게중심 계산

LED좌표는 카메라와 사용자와의 거리, 카메라의 위치 등에 의하여 변화될 수 있다. 이러한 변화에도 일정한 벡터 값(거리, 각도)을 얻어낼 수 있도록 LED좌표들의 무게중심을 구한 후 그것을 중심으로 구간을 측정하도록 한다.

그림 5는 각 LED 좌표를 꼭지점으로 하는 다각형의 무게 중심을 구하는 예시를 보여주고 있다. 다각형의 무게 중심은 원점을 중심으로 각각의 삼각형의 넓이와 무게 중심의 위치 벡터를 곱한 후 다각형의 면적 전체로 나누어 주면 다각형의 무게 중심을 구할 수 있게 된다. 이를 먼저 임의의 삼각형 OP1P2에 대해서 구하게 되면

$$\begin{aligned}
 \text{넓이 } A_1 &= \frac{x_1 y_2 - x_2 y_1}{2} \\
 \text{무게 중심 } \bar{G}_1 &= \left(\frac{x_1 + x_2}{3}, \frac{y_1 + y_2}{3} \right)
 \end{aligned}
 \tag{2}$$

공식 (2)와 같이 나타낼 수 있게 되며 이는 아래와 같이 공식 (3)로 나타낼 수 있게 된다.

$$\begin{aligned}
 A_i &= \frac{x_i y_{i+1} - x_{i+1} y_i}{2} \\
 \bar{G}_i &= \left(\frac{x_i + x_{i+1}}{3}, \frac{y_i + y_{i+1}}{3} \right)
 \end{aligned}
 \tag{3}$$

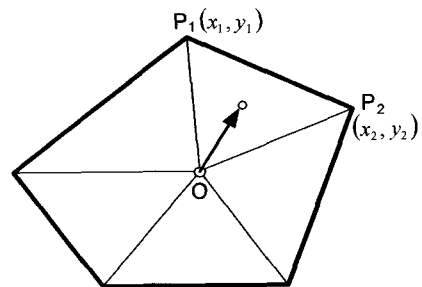


그림 5. 다각형의 무게 중심

이를 이용하여 공식 (4)와 같이 다각형의 무게 중심을 구할 수 있다.

$$\bar{G} = \frac{\sum A_i G_i}{A} = \left(\frac{\sum (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i)}{6A}, \frac{\sum (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i)}{6A} \right) \quad (4)$$

이는 각 삼각형의 넓이와 무게중심 벡터를 곱하여 합한 후, 전체 넓이로 나눈 것이다.

3.2.2 거리 구간 계산

LED 좌표의 거리 구간은 LED들의 무게중심에서 각 LED까지의 거리를 나타낸다. 이들 거리 구간은 무게중심에서 멀어질수록 조밀해진다. 이것은 약간의 동작 차이만으로도 거리 구간이 달라질 수 있으므로 이를 방지하여 모션인식의 정확도를 높이기 위함이다.

거리 구간을 구하기 위하여 입력 받은 LED 좌표를 기반으로 구한 무게 중심을 원점으로 하는 2차원 평면에 각 LED를 위치시킨다. 원점을 기준하여 다시 정해진 LED좌표의 x, y 값을 기반으로 원점과 각 LED의 거리를 구한 후 가장 큰 값을 기준하여 n등분 하게 된다. 이것은 카메라와 사용자의 거리에 따라서 거리 구간의 변동이 없도록 하기 위함으로써 n등분의 공식은 (5)와 같다.

$$\log \left\{ \frac{op_x \times n}{\max \cdot op_x} \right\} \quad (5)$$

또한 균등한 비율로 거리 구간을 나눌 시 중심 좌표에서 멀어질수록 동작에 대한 상대적 면적이 넓어짐으로 인하여 모션 인식률의 저하를 막고 정확한 비교를 하기 위하여 그림 6과 같이 원거리 일수록 구간이 조밀해지는 모양을 취하고 있다.

3.2.3 각도 구간 계산

LED좌표들의 각도 구간은 LED들의 무게중심을 원점으로 한 2차원 평면에 재배치된 LED들의 좌표들이 일정한 기준에서 얼마나 회전된 위치에 있는가를 측정하는 것이 기본이 된다. LED좌표는 움직임에 따라서 무게중심을 기준으로 회전 하게 되며 이는 모션인식에 중요한 입력 값이 된다.

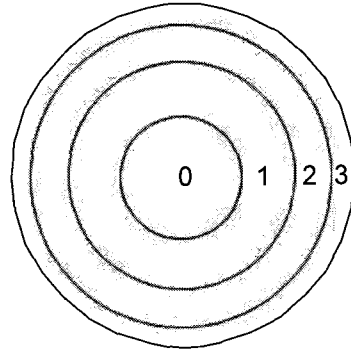


그림 6. 거리에 따른 구간 영역

무게 중심을 원점으로 하여 배치된 LED 좌표들의 거리 별로 나누어진 구간을 0구간부터 n구간, 거리 0구간을 각도 m구간으로 나누는 경우 공식 (6)과 같이 각 거리 구간별로 단위각을 구한다.

$$\angle a = \frac{360}{m \times 2^n} \quad (6)$$

0구간을 16으로 나눌 경우 0구간 16등분, 1구간일 경우 32등분, 2구간일 경우 64등분으로 거리 별로 구간을 조밀하게 나누어 단위각을 구한다.

이것은 같은 단위각으로 원거리의 각도 구간을 나눌 경우 원거리의 각도 구간은 근거리에 비해 너무 넓어져 정확도가 떨어질 우려가 있으므로 그림 7과 같이 원거리 일수록 조밀한 단위각으로 각도 구간을 측정하게 된다. 원점과 임의의 LED 좌표가 이루는 선과 평면의 양의 가로축이 이루는 각 θ 를 atan2함수를 이용하여 알아낸다.

$$\theta = \text{atan2}(y, x) \quad (7)$$

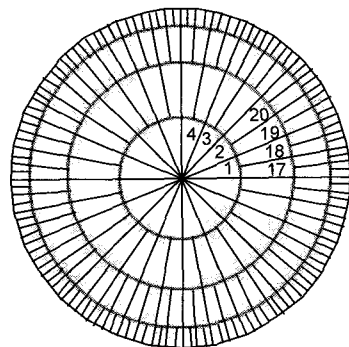


그림 7. 거리에 따른 각도 구간

공식(7)에서 나온 결과값을 단위각으로 나눈 결과의 정수부만을 취함으로 거리에 따른 조밀한 각도 구간이 결정 된다.

$$\text{각도 구간} = \frac{\theta}{\Delta a} + \sum (2^n \times m) + 1 \text{의 정수부} \quad (8)$$

3.3 템플릿의 동작 데이터와의 일치 판단

3.3.1 템플릿의 생성

예제 동작의 템플릿은 그림 8과 같은 현재 사용자가 행하고 있는 동작의 장면과의 일치 여부를 가리는 기준이 되며 사용자와 같은 환경에서 수십 회의 모션 인식을 통해 각 특징점이 존재하는 구역을 표시하게 된다. 또한 하나의 장면에서 유사도를 측정하기 위하여 같은 장면을 여러 번 동작하여 모션 일치의 오차를 줄이게 된다. 그림 9는 임의의 한 장면의 템플릿을 생성하는 과정을 보인 것이다. 한 장면의 좌표를 여러 번 획득하여 유사한 동작에도 일치 판단을 내릴 수 있도록 좌표들을 정리하여 템플릿으로 생성한다. 이는 구간이 조밀하기 때문에 아주 약간의 차이에도 다른 구간으로 인식할 수 있다는 것에 착안한 것으로, 정확한 동작일 경우에는 정확한 동작의 일치 판단을, 유사한 동작에는 유사 판단을 할 수 있도록 템플릿을 생성하는 과정에서 조절 할 수 있다. 작성된 템플릿은 별도의 파일로 저장되며 시스템 장치의 호출 시 사용된다.

3.3.2 템플릿의 호출

사용자의 모션을 인식하기 전 시스템은 사용자가 해야 할 동작의 템플릿을 호출하게 된다. 호출된 템플릿은 별도의 파일에서 호출되어 각 구간의 유사도 값을 저장하게 되며 미리 정의된 동작에 대한 동작

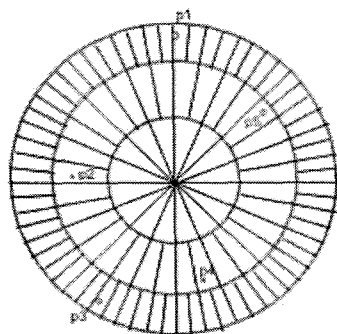


그림 8. 사용자의 움직임을 구간에 따라 나뉜 예

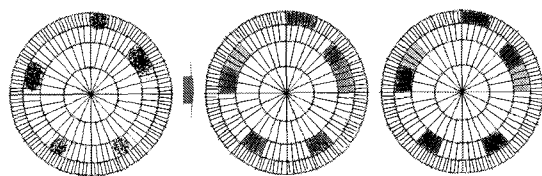


그림 9. 비슷한 동작을 포함한 동작 비교를 위한 템플릿 생성

비교 값은 정확한 동작 구간일수록 높아지게 된다. 예를 들어 그림 10과 같이 19구간이 모션 템플릿에서 정확한 동작 구간인 경우 19 구간은 10의 유사도 값을 가지며 53, 54 구간은 5의 유사도 값을 시스템에 저장하게 되며 나머지 부분은 0으로 표시된다.

3.3.3 템플릿과의 비교

그림 11은 동작의 템플릿과 사용자의 동작 벡터와의 일치 여부를 판단하는 한 예이다. 템플릿과의 비교는 벡터가 위치한 구간의 유사도 값이 0이 아닐 경우 일치로 판단되며 그 중에서도 유사도 값이 10일 때 정확한구간의 일치로 판단한다. 또한 모든 구간의 유사도 값의 일정한 값 이상을 가질 때 모션 일치로 판단하게 된다. 동작 데이터는 별도의 표시를 할 필요는 없지만 이해가 쉽도록 구간 간의 비교 그림을 사용하여 비교 하였다. 실제 시스템에서는 3구간, 19구간에 동작 데이터가 있는 경우, 바로 3번째 구간, 19번째 구간 만을 검색하여 데이터가 유사도 값을 가지고 있는지 확인 후 모션 일치를 판단한다. 사용자의 동작이 템플릿과 일치 할 때 시스템은 다음 장면이 있을 경우는 다음 장면의 템플릿을 읽어 들여 다음 장면의 모션 인식 및 비교를 준비하고, 최종 장면이 일치할 경우는 한 동작의 일치가 완료 되었음을 알리게 된다.

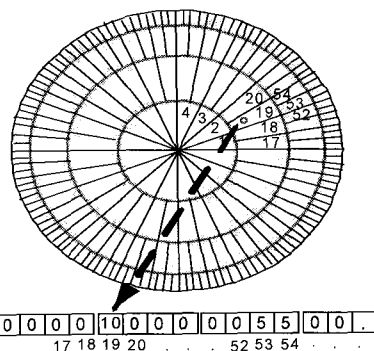


그림 10 템플릿 호출 시 유사도 확인 과정

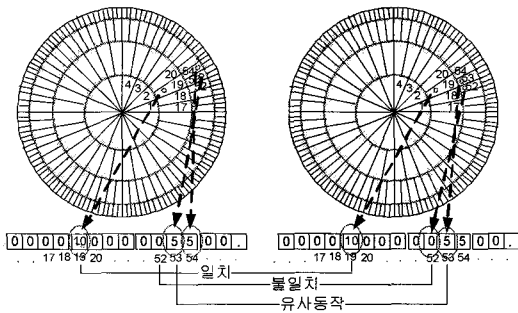


그림 11. 템플릿과 실제 동작과의 비교

4. 실험 및 결과

그림 12는 본 연구에서 사용된 시스템의 구성도이다. 본 실험은 별도의 재활훈련 전용 하드웨어를 사용하지 않고 1GB의 주 메모리를 가진 펜티엄 4 환경에서 Visual Studio .Net 2003에서 구현하여 성능 실험을 수행하였다. 본 실험환경으로는 사용자의 동작을 인식하기 위하여 적외선 필터를 장치한 카메라 1대와 사용자가 자기 자신을 볼 수 있도록 하는 카메라 1대가 사용되었으며 각각의 카메라는 사용자의 전체 몸 상태를 찍을 수 있을 정도의 거리에 고정되어 위치시켰으나 인식용 카메라는 고정되어 있지만 측면의 LED를 검출해야 하는 경우 필요에 따라 이동 가능하다. 또한 동작인식을 위한 컴퓨터와 사용자가 자신의 모습을 볼 수 있도록 HMD Viewer를 실행시키는 컴퓨터가 서로 통신을 사용하여 연결하였다. 본 연구의 실험 동작인 재활훈련 시나리오는 사용자의 근력, 유연성 그리고 균형각각을 훈련하기 위한

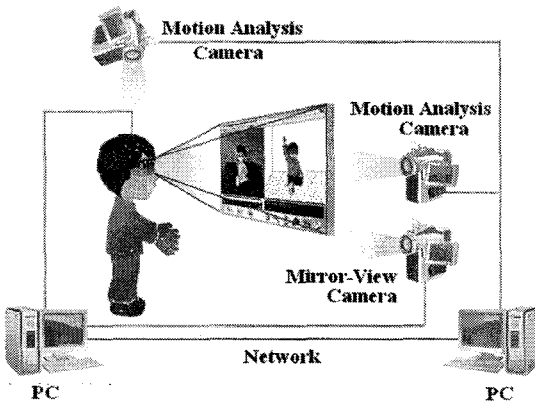


그림 12. 재활훈련 서비스 프로토타입의 시스템 구성도

기본동작으로 이루어져 있으며 본 동작들은 지체장애인 재활훈련으로서의 신뢰성을 위하여 장애인 재활학교인 삼육재활학교의 연구진에 의해 작성되었다. 본 실험에서는 표 1에서 보는 바와 같이 이 시나리오에 따라 제시된 동작들의 39가지 동작 중 가장 복잡한 2가지 유형의 동작을 사용하였다.

동작 1은 의자에 앉아 팔을 자연스럽게 늘어뜨린 자세로부터 좌우로 팔을 뻗어 머리 위로 들어올린 후 내리는 동작이며, 동작 2는 바닥에 앉아 다리를 쭉 뻗은 상태에서 좌우에 놓여진 물체를 양손으로 번갈아 가며 만지는 동작이다. 이상의 동작을 웹캠(2종류), 캠코더 등으로 촬영하여 미리 준비되어 있는 모범 동작의 Reference와 비교 하였다. 운동 동작들은 확실한 동작 일치 여부를 가리기 위해 한 동작당 n가지의 장면으로 나누어 장면마다 동작 일치 여부를 판단하였다. 한 동작을 n가지 장면으로 나누는 것은 동작들의 중간 움직임 궤도를 인식하기 위한 것으로 동작 중간 중간을 인식하여 동작이 잘못된 동작이 되지 않는가를 확인하기 위함이다. 본 연구에서는 삼육재활학교에서 제시하는 4등급의 동작 평가에 적절한 평가가 가능한 4장면으로 나누었다. 그림 13과 같이 30여 회의 실험 결과, 다음과 같은 결과를 나타내었다.

4.1 인식률

그림 13은 각 동작의 실험에 대한 실제 영상을 각

표 1. 동작 인식을 위한 훈련 동작

훈련 동작	시작 자세	종료 자세
앉은 자세에서 양팔을 머리 위로 들어 올리는 동작(동작 1)	양발을 바닥에 붙이고 의자에 앉은 자세	원위치
앉은 자세에서 물체를 만지는 동작(동작 2)	바닥에 엉덩이를 붙인 상태에서 다리를 펴고 앉은 자세	원위치

표 2. 동작 인식률에 대한 실험 결과

훈련 동작	템플릿 생성 시에 시험한 횟수에 따른 매칭 확률		
	10회	20회	30회
동작 1	97%	98%	98%
동작 2	96%	98%	98%

각 보여준다. 기존 연구 중 센서를 사용하는 방법은 센서의 전선이 사용자의 동작에 방해가 되는 것에 비하여 본 실험에서는 무선 밴드를 사용함으로써 동작 방해 요인을 배제한다. 또한 운동 동작들은 중간 중간 정확한 동작을 하고 있는지 알아내기 위하여 한 동작 당 4장면으로 나누어 장면마다 모션 일치 여부를 판단하고 최종 장면이 일치 되었을 때 한 동작의 일치를 결정한다. 사용자의 신체 각 부위의 LED 밴드들은 각 원으로 표현되었으며 원의 중심 좌표가 각 특징점의 좌표가 된다.

본 연구에서 제시한 알고리즘을 수십여 회 실험한 결과, 표2와 같이 시나리오에서 복잡한 자세에 해당하는 본 실험 자세에서 높은 인식률을 보였다. 또한 정확하지 않은 동작에 대하여는 훈련 실패의 결과를 얻어내었다. 템플릿 생성 시 시험한 횟수에 따라서 매칭 판단 확률에 차이가 생겼으나 이는 유사도 정보의 차이에 따름으로 20회 이상의 횟수에서는 같은 정확도를 보였으며 20회 이상에서의 오류율은 자연 광에서 생기는 적외선으로 인한 노이즈 때문으로써 인식에 큰 영향을 주는 것은 아니지만 사용자의 가림 현상이 발생하였을 때 노이즈를 신체로 인식하는 경우가 생김으로 오류가 발생하였다.

4.2 속도

표 3과 4는 기존 연구에 쓰인 다중 카메라를 사용한 인식 방법과 센서와 카메라의 정보를 종합하는 방식의 동작 인식 결과를 보인 것이다. 표 5는 본 연구에서 사용된 알고리즘을 이용한 동작인식의 결과 속도로서 기존 연구에 비하여 빨라졌음을 알 수 있

표 3. 기존 연구의 다중 카메라 방식을 이용한 동작 인식 결과

Nb. of Cameras	4	4	4	11
동작Sec. per Frame	0.961	0.666	0.686	0.806

표 4. 기존 연구의 센서와 카메라 정보를 종합하는 방식을 이용한 동작 인식 결과

Current Framerate	appr. 10-14
Background points Removed	appr. 13~15

표 5. 동작 인식속도에 대한 실험 결과

훈련 동작	속 도		
	캠코더 사용 시	웹캠 사용 시 (anicam)	웹캠 사용 시 (Logitech Quick cam Pro 5000)
동작 1	20 fps	8 fps	18 fps
동작 2	21 fps	7 fps	18 fps

다. 본 연구에서 제시한 알고리즘의 수행 속도는 카메라의 속도에 영향을 받지 않는 바는 아니지만 아주 느린 카메라를 제외하고는 빠른 속도를 보였다.

5. 결 론

재활훈련에서 동작 인식을 지체장애인 분야에 활용하려면 원하는 때에 훈련 결과를 확인 할 수 있어야 하고, 정확하게 동작을 인식할 수 있어야 한다. 본 연구에서는 이러한 조건을 만족하는 재활훈련 시스템을 설계하고, 사용자의 모션 인식 및 모범 동작과의 모션 일치 알고리즘을 제시하였다. 실험 결과 제시된 알고리즘으로 재활훈련 시나리오 상의 모범 동작과 일치하는 사용자의 동작을 높은 확률로 인식할 수 있었다. 동작 인식의 정확도 면에서는 3차원적인 비교가 좋지만 본 연구는 실제 응용에서 실시간으로 재활훈련을 서비스 하는 것을 목적으로 하고 재활 훈련 동작들이 2차원적인 비교만으로도 동작의 인식이 용이하였으므로 2차원적인 비교를 선택하게 되었으며 실험 결과 좋은 결과를 얻을 수 있었다. 기존의 연구들이 10프레임 내외의 속도를 보이던 것에 비하여 본 연구에서는 20프레임 내외의 속도 향상이 확인 가능하였다. 또한 본 연구의 최종 목표인 모델기반의

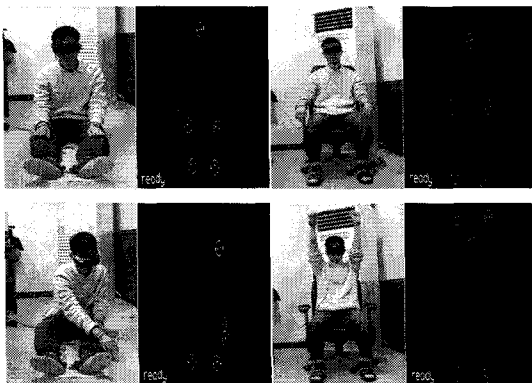


그림 13. 동작 1과 2의 특징점 추출 및 클러스터링 예

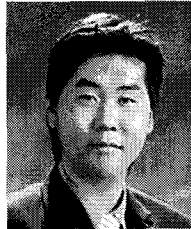
모션 인식으로의 기반을 다지며 차후 다방면의 재활 훈련에 적용하여 시간적 공간적으로 제한되어 있는 지체 장애인들의 재활 훈련을 좀 더 쉽게 활용 할 수 있도록 하는데 도움이 될 것이다. 그러나 복잡한 동작을 인식하기 위한 한계를 극복하기 위하여 차후 연구과제로 모델기반의 매칭이나 다중카메라를 이용한 각 특징점의 벡터 추적에 의한 매칭 방법에 관한 연구를 수행할 예정이다.

참 고 문 헌

[1] <http://www.visrehab.com/>.
 [2] N. Ichiguchi and D. Komaki, "Experimental Analysis on Human Walking Animation of Virtual Robot in 3D Visual Space," *IEEE SMC '99 Conference Proceedings*, pp. 918-923, 1999.
 [3] Huazhong Ning and Liang Wang, "Model-based Tracking of Human Walking in Monocular Image Sequences," *IEEE TEN-CON '02. Proceedings*, 2002.
 [4] Steffen Knoop and Stefan Vacek, "Sensor Fusion for 3D Human Body Tracking with an Articulated 3D Body Model Sensor Fusion for 3D Human Body Tracking with an Articulated 3D Body Model," *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, May 2006.
 [5] Kristen Grauman and Gregory Shakhnarovich, "A Bayesian Approach to Image-Based Visual Hull Reconstruction," *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Madison, WI*, June 2003.
 [6] German K.M. Cheung and Simon Baker, "Visual Hull Alignment and Refinement Across Time:A 3D Reconstruction Algorithm Combining Shape-From-Silhouette with Stereo," *IEEE Computer Vision and Pattern Recognition*, 2003.
 [7] Joachim Schmidt and Jannik Fritsch, "Kernel Particle Filter for Real-Time 3D Body

Tracking in Monocular Color Images," *IEEE Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition (FGR'06)*, 2006.

[8] Takashi Yamamoto and Rama Chellappa, "Shape and Motion Driven Particle Filtering for Human Body Tracking," *IEEE ICME 2003*, pp. 61-64, 2003.
 [9] David A. Forsyth and Jean Ponce, *Computer Vision A Modern Approach, International Ed.*, Prentice Hall, 2003.



최 완

2006년 서경대학교 컴퓨터공학과 학사 졸업
 2007년 현재 서경대학교 대학원 컴퓨터공학 석사과정 재학



김 태 영

1991년 이화여자대학교 전자계산학과 학사 졸업
 1993년 이화여자대학교 전자계산학과 석사 졸업
 2001년 서울대학교 컴퓨터공학부 박사 졸업
 2007년 현재 서경대학교 컴퓨터공학과 조교수 재임



임 철 수

1985년 서울대학교 계산통계학과 학사 졸업
 1988년 (미) 인디애나주립대 컴퓨터과학과 석사 졸업
 1995년 서강대학교 전산과 박사 졸업
 2007년 현재 서경대학교 컴퓨터공학과 부교수 재임