

제한된 이동 범위 내에서의 손 제스처 상호작용을 위한 포인트 가속 알고리즘

김주창*, 박정우*, 김우현*, 이원형*, 정명진*
한국과학기술원*

Point Accelerating Algorithm for Hand Gesture Interaction Within the Restricted Moving Range

Juchang Kim*, Jeong-Woo Park*, Woo-Hyun Kim*, Won-Hyung Lee*, Myung-Jin Jung*
Korea Advanced Institute of Science and Technology*

Abstract - 손 제스처 인식을 위한 손 위치 검출 및 추적 알고리즘에 대한 많은 연구들이 이루어져 왔다. 검출된 손의 위치를 통한 사용자와 컴퓨터간의 인터랙션(Human Computer Interaction, HCI) 과정에서 손의 위치 좌표를 그대로 이용하는 방법은 비효율적이다. 본 논문에서는 제한된 이동 범위 내에서의 손의 위치 정보를 이용하여 컴퓨터상의 포인트를 효율적으로 컨트롤할 수 있는 알고리즘을 제안하였다. 손 위치 검출을 위해 깊이 센서를 이용하였으며 알고리즘의 효율성을 시험하기 위해 사용자의 손 움직임에 따른 포인트의 위치 변화를 관찰하였다. 실험 결과를 통해 제안된 알고리즘이 추적되는 손의 위치를 이용해 효율적으로 포인트를 컨트롤함을 확인하였다.

1. 서 론

현대사회에서 사용자와 컴퓨터간의 인터랙션(Human Computer Interaction, HCI)을 위해 사용하는 장치로는 키보드, 마우스 등이 있다. 하지만 이러한 장치 없이 HCI를 가능하게 하는 기술에 대한 연구가 활발히 이루어지고 있으며, 가장 대표적인 예로 센서를 통한 사용자의 제스처 인식을 이용한 HCI이다. 제스처 인식은 크게 사용자의 몸 전체의 제스처를 인식하는 것과 머리와 얼굴의 제스처, 그리고 손과 팔의 제스처를 인식하는 것으로 나뉜다[1]. 손의 제스처를 이용한 의사 소통은 예전부터 사람 간의 의사소통에 중요한 역할을 하였다. 현재 까지 손 제스처 인식에 관한 다양한 연구가 이루어지고 있으며, 본 논문은 손 제스처 인식을 다루고 있다.

손 제스처 인식을 통한 인터랙션 구현을 위해 다양한 센서들을 이용한 손 인식, 손 위치 검출 및 추적 알고리즘들이 개발되어 왔으며, 검출된 손의 위치에 따른 HCI가 다양한 분야에서 연구되고 있다. 손 제스처 인식은 손의 위치를 검출하고 이 위치에 따른 정보를 이용한 손동작 인식과, 손의 형상 및 자세를 파악하여 그 형상에 해당하는 의미를 인식하는 손 형상 인식으로 나뉠 수 있으며, 본 연구는 손동작 인식에 대한 알고리즘을 제시하였다. 손동작 인식을 위해서는 손의 정확한 위치 검출 및 배경과의 분리, 그리고 영상에서 손을 놓치지 않고 추적할 수 있는 추적 알고리즘이 핵심적으로 필요하다.

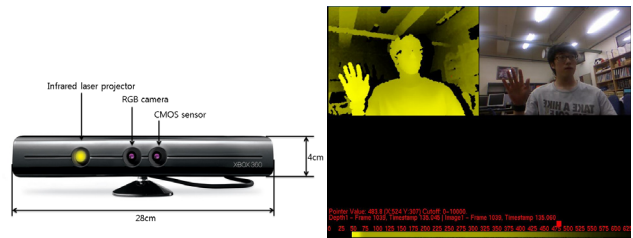
손동작 인식을 통해 얻어지는 손의 위치 정보를 바탕으로 HCI를 수행하는 방법은 다양하다. 그 예로는, 손의 위치를 컴퓨터상의 좌표에 매칭 시켜 마우스로 컨트롤 하는 것과 같은 원리로 조작하는 방법과, 컴퓨터에 미리 학습된 패턴과 동일한 동작을 입력시켜 그 패턴을 인식하고, 그 패턴에 해당하는 기능을 수행하도록 하는 패턴 인식 방법 등이 있다. 전자적 경우, 손의 위치 정보를 컴퓨터상의 포인트 좌표로 어떻게 매칭 시키느냐에 따라 포인트 컨트롤의 성능 및 효율성이 달라질 수 있다. 손의 위치 정보를 그대로 포인트 좌표로 매칭 시키게 되면 손의 이동 범위에 따라 포인트의 이동 범위가 제한된다는 단점이 있으며, 손의 위치 정보를 선형적으로 스케일링하여 포인트를 컨트롤 하게 되면 컨트롤의 분해능이 안 좋아지는 단점이 있다. 즉, 선형적인 스케일링은 포인트의 이동 가능 범위와 포인트의 분해능간의 상관관계를 불리하게 된다.

본 논문에서는 손 추적 알고리즘을 통해 검출된 손의 위치 정보를 바탕으로 컴퓨터와 인터랙션 하는 과정에서, 손의 이동 범위가 제한되었을 때 포인트를 효율적으로 컨트롤하기 위한 알고리즘을 제안한다. 손의 빠른 움직임을 asymmetric bell-shaped velocity profile에 근사하고, 손의 속도를 비선형적으로 증폭시켜 포인트의 속도를 결정한다. 이를 통해, 제한된 손의 이동 범위 내에서 분해능의 저하 없이 더 넓은 반경내에서 포인트를 컨트롤 할 수 있다. 센서를 이용해 추적되는 손의 속도를 포인트 가속 알고리즘을 통해 증폭시킨 결과로부터 알고리즘의 효율성을 확인하였다.

2. 본 론

2.1 깊이 정보 추출을 위한 센서

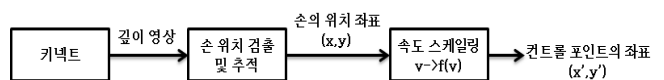
물체로부터의 거리(깊이)를 측정하기 위한 센서는 다양하다. 스테레오 카메라를 이용하거나 초음파 센서, 레이저 센서 등을 이용해 물체로부터의 깊이 정보를 추출해낼 수 있다. 본 연구에서는 손 위치 검출을 위해 적외선 레이저를 이용해 깊이 정보를 측정하는 센서인 Microsoft사의 키넥트(Kinect)를 이용하였다. 키넥트는 제스처 인식 게임을 위해 개발된 센서로, 적외선 레이저가 물체에 반사되어 돌아오는 시간을 이용해 거리 정보를 측정하는 TOF 센서이다.



〈그림 1〉 키넥트의 외형(왼쪽), 키넥트로부터 얻어진 깊이 영상과 카메라 영상(오른쪽)

〈그림 1〉은 키넥트 센서의 외형과 키넥트로부터 얻어지는 깊이 영상(depth map) 및 카메라 영상을 보여준다. 키넥트로부터 얻어지는 깊이 영상과 카메라 영상은 서로 다른 시야범위(Field of view)를 갖기 때문에, 두 영상 정보를 융합하기 위해서는 캘리브레이션이 필요하다. 키넥트는 거리 정보를 640X480의 해상도로 초당 30 프레임을 출력할 수 있기 때문에, 손의 위치 추적의 실시간(real-time) 구현이 가능하다. 본 연구에서는 키넥트로부터 얻어지는 거리 정보를 바탕으로 개발된 손 위치 검출 및 추적 알고리즘을 사용하였다.

2.2 포인트 가속 알고리즘



〈그림 2〉 포인트 가속 알고리즘의 순서도

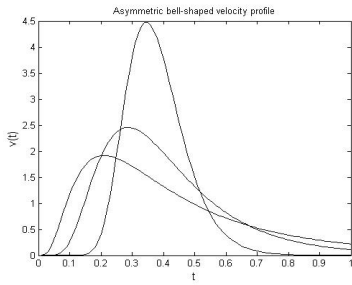
일반적인 포인트 컨트롤 알고리즘은 검출된 손의 3D 위치 좌표를 센서가 바라보는 방향의 평면으로 투사하여 얻어진 2D 좌표를 그대로 이용하거나, 선형적으로 스케일링하여 화면상의 포인트를 컨트롤 하게 된다. 본 논문에서 제안하는 포인트 가속 알고리즘은 검출된 3D상의 손의 위치 좌표를 통해 얻어지는 손의 3D 속도를 비선형적으로 스케일링하여 포인트를 효율적으로 컨트롤 한다. 손의 빠른 움직임이 asymmetric bell-shaped velocity profile에 근접하다고 가정했을 때, 속도를 비선형적으로 스케일링함을 통해 증폭된 asymmetric bell-shaped velocity profile을 나타내는 포인트의 움직임을 얻는다.

2.2.1 Asymmetric Bell-Shaped Velocity Profile

Asymmetric bell-shaped velocity profile(ABSVP)은 빠른 움직임을 근사하는데 가장 근접한 알고리즘 중 하나라고 알려져 있다[3]. 식 (1)은 5개의 변수들, 즉 \bar{V} , t_{01} , t_1 , μ , σ 로 이루어진 ABSVP를 나타낸다.

$$v(t) = \frac{\bar{V}(t_1 - t_0)}{\sigma\sqrt{2\pi}(t - t_0)} \exp\left\{ \frac{1}{2\sigma^2} \left[\ln \frac{t - t_0}{t_1 - t_0} - \mu \right]^2 \right\} \quad (1)$$

본 연구에서는 손의 빠른 움직임을 ABSVP로 가정하고, 포인트 가속 알고리즘을 거친 포인트의 속도가 어떤 velocity profile을 나타내는지 분석하였다. <그림 3>는 다양한 형태의 ABSVP들을 보여준다.



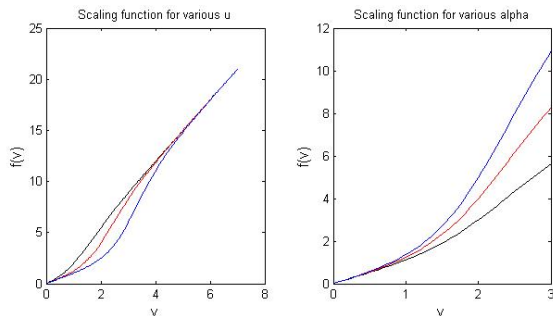
<그림 3> Various asymmetric bell-shaped velocity profiles

2.2.2 포인트 가속 알고리즘

컨트롤 하고자 하는 포인트의 위치는 현재 입력받는 손의 위치 좌표와 한 스텝 이전의 손의 위치 좌표로부터 계산된다. 두 개의 연속된 좌표로부터 손의 속도를 계산하고, 이를 스케일링 함수를 통해 새로운 속도로 변환하여 변환된 속도로부터 컨트롤 하는 포인트의 위치를 결정한다. 이 때, 손의 속도는 깊이 영상에 투사된 (x,y)좌표와, 실제 손과 센서간의 거리인 z좌표를 모두 고려한 3D 좌표로부터 계산된다. 왜냐하면, 포인트를 컨트롤하기 위한 사용자의 손 움직임이 항상 깊이 영상에 평행하지 않기 때문에, 사용자가 입력하고자하는 손의 속도는 투사된 2D 평면상의 속도보다 3D상의 속도에 더 가깝기 때문이다. 속도 스케일링 함수는 식 (2)와 같이 정의된다.

$$v' = f(v) = \left\{ \frac{\alpha}{2} \tanh(\beta(v - \mu)) + 1 + \frac{\alpha}{2} \right\} v \quad (2)$$

여기서 μ 는 임계값, a 는 가속도, α , β 는 스케일링 팩터이다. 이를 통해 작은 속도에 대해서는 속도를 증폭하지 않아 세밀한 컨트롤이 가능하게 함과 동시에, 큰 속도에 대해서는 가속도를 증가시켜 넓은 범위에서의 컨트롤이 가능하게 하였다. 스케일링 팩터 α 는 속도를 얼마나 스케일링 할 것인지를 결정하는 변수로, 사용자와 거리 센서간의 거리, 컨트롤 되는 포인트의 이동 범위 등에 따라 결정할 수 있다. β 는 임계값 부근의 기울기를 결정한다. <그림 4>은 임계값과 스케일링 팩터에 따른 스케일링 함수의 변화를 보여준다.

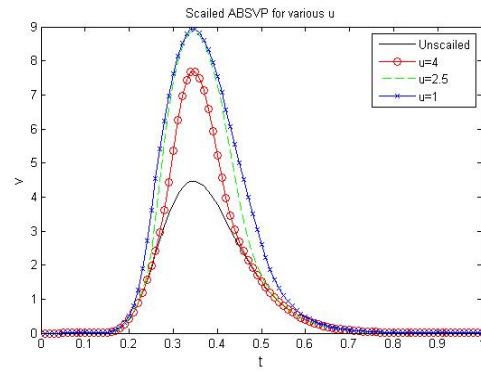


<그림 4> 임계값의 변화에 따른 스케일링 함수 (왼쪽, $\alpha=2, \beta=1, \mu=1,2,3$), 스케일링 팩터의 변화에 따른 스케일링 함수 (오른쪽, $\beta=1, \mu=1, \alpha=1,2,3$)

위에서 정의된 스케일링 함수를 통해 빠른 손의 움직임을 근사한 ABSVP를 증폭시키게 되면 컨트롤 하고자 하는 포인트의 velocity profile을 구할 수 있다. <그림 5>는 시뮬레이션을 통해 가상의 ABSVP를 여러 임계값의 스케일링 함수로 증폭시킨 결과이다. 같은 스케일링 팩터 a 를 이용해 증폭할 때, 스케일링 함수의 임계값이 클수록 증폭되는 속도의 대역폭이 작아져서 스케일링의 영향이 줄어들고, 임계값이 작을수록 증폭되는 속도의 대역폭이 커져서 스케일링의 영향이 커진다.

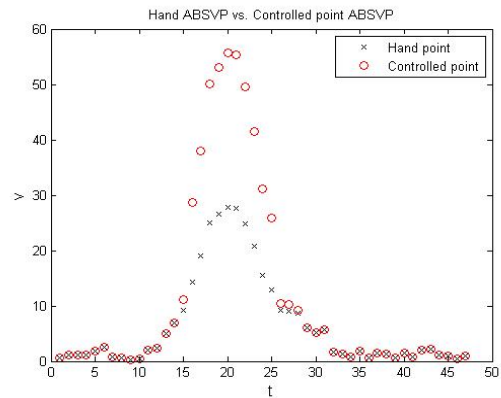
2.3 실험 결과

키넥트로부터 얻은 깊이 정보를 통해 사용자의 손을 검출하고 추적하였다. ABSVP에 맞는 손의 속도를 입력하기 위해, 손의 빠른 움직임을 선형적으로 수행하였다. 계산된 손의 속도를 스케일링 함수를 통해 증폭하여 포인트의 속도를 계산하였다. <그림 6>은 추적된 손의 속도와 포



<그림 5> 임계값의 변화에 따른 증폭된 ABSVP ($V=1, \mu=-1, \sigma=0.25$ for ABSVP, $\alpha=2, \mu=1,2,5,4$ for scaling function)

인트 가속 알고리즘을 통해 계산된 포인트의 속도를 나타내며, 이는 <그림 5>의 시뮬레이션 결과와 일치함을 확인할 수 있다. 낮은 속도에서는 포인트의 속도가 손의 속도를 따라가게 되어 추적되는 손의 좌표와 같은 분해능을 가지며, 높은 속도에서는 속도가 증폭되어 포인트의 컨트롤 가능 범위를 확장할 수 있었다.



<그림 6> 추적된 손의 좌표의 속도와 포인트 가속 알고리즘을 통해 구한 포인트의 속도 ($\mu=10, a=1, \beta=1$)

3. 결 론

본 연구의 목적은 손을 이용한 제스처 인터랙션에서, 제한된 손의 이동 범위 내에서 포인트를 효율적으로 컨트롤하기 위함이다. 실험을 통해 손의 속도를 제한된 알고리즘을 통해 비선형적으로 증폭하였을 때, 분해능의 저하 없이 포인트의 컨트롤 가능 범위를 확장할 수 있음을 확인하였다. 하지만 제한된 알고리즘을 통해 포인트의 속도를 증폭시켰을 경우, 손의 움직임이 선형적이지 않고 곡선이나 원형을 그릴 경우, 컨트롤 되는 포인트는 본래 사용자가 입력하고자 하는 패턴과 다른 패턴의 움직임을 보이게 된다. 이를 해결하기 위해 다이내믹한 손의 움직임에 대한 포인트 가속 알고리즘의 추후 연구가 필요하다. 더불어, 본 연구는 손의 움직임을 통한 포인트 컨트롤에 초점이 맞춰져 있지만, 손의 속도를 분석하고 이를 비선형적으로 증폭시키는 알고리즘은 단순한 포인트 컨트롤 이외에 손 제스처 인식이나 손 제스처로부터 사용자의 의도 파악 등에도 적용될 수 있을 것으로 예상된다.

[참 고 문 헌]

[1] Sushmita, M., "Gesture Recognition: A Survey," IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, Vol. 37, No. 3, 2007
 [2] Pavlovic, V.I., "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, Issue. 7, 677-695, 1997
 [3] Plamondon, R., "On the Origin of Assymmetric Bell-Shaped Velocity Profiles in Rapid Aimed Movements," In: Requin, J. &Stelmach G.E, eds. Tutorials in Motor Neuroscience, 283-295, 1991