
디지로그 북 저작을 위한 3D 객체의 In-Situ 기반의 이동 궤적 편집 기법

In-Situ based Trajectory Editing Method of a 3D Object for Digilog Book Authoring

하태진, Taejin Ha*, 우운택, Woontack Woo**

요약~디지로그 북(Digilog book)은 기존 서적과 디지털 콘텐츠를 융합함으로써, 아날로그적 감성과 디지털 오감을 함께 제공하는 증강현실기반 차세대 출판물이다. 디지로그 북을 저작할 수 있는 저작 소프트웨어인 아틀렛(ARtalet)은 증강현실 환경에서 3 차원 사용자 인터페이스를 이용한 직관적인 In-Situ 저작 환경을 제공한다. 본 논문은 디지로그 북에 증강된 3D 객체에 이동 경로를 생성하고 조작 할 수 있는, 아틀렛 저작 환경 기반의 이동 궤적 편집 기법을 제안한다. 구체적으로 이동 궤적의 조정점(Control point)을 적절히 할당하기 위해서 3차원 조작도구의 이동 좌표에 대하여 조정점 할당 검사를 한다. 그리고 부드러운 곡선 형태로 이동 궤적을 복원하기 위해서 스플라인을 이용한 보간 과정을 수행한다. 또한 작고 밀집된 이동 궤적의 조정점을 효과적으로 선택하기 위해서 동적 스코어(Score)를 기반으로 한 조정점 선택 방법을 적용한다. 실험 결과 제안한 방법은 기존 방법에 비해 오차와 완료시간은 유의한 차이가 없었지만, 조정점의 수를 약 90% 이상 감소시킬 수 있었다. 이것은 매우 적은 수의 조정점만으로도 이동 궤적을 복원할 수 있으며 추후 이동 궤적 조작에 필요한 조정점의 조작 횟수를 대폭 줄일 수 있다는 것을 의미한다. 또한 제안한 방법은 기존의 조정점 조작 방법에 비해 상대적으로 적은 손과 팔의 움직임만으로도 빠르게 이동 궤적의 형태를 변경 할 수 있었다. 본 논문에서 제안한 3D 객체의 이동 궤적 편집 방법은 몰입형 In-Situ 증강현실 환경의 교육, 게임, 디자인, 애니메이션, 시뮬레이션 등의 분야에서 드로잉 또는 곡선 편집 방법으로 응용될 수 있다.

Abstract~A Digilog Book is an augmented reality (AR) based next generation publication supporting both sentimental analog emotions and digitized multi-sensory feedbacks by combining a conventional printed book and digital contents. As a Digilog Book authoring software, ARtalet provides an intuitive authoring environment through 3D user interface in AR environment. In this paper, we suggest ARtalet authoring environment based trajectory editing method to generate and manipulate a movement path of an augmented 3D object on the Digilog Book. Specifically, the translation points of the 3D manipulation prop is examined to determine that the point is a proper control point of a trajectory. Then the interpolation using splines is conducted to reconstruct the trajectory with smoothed form. The dynamic score based selection method is also exploited to effectively select small and dense control points of the trajectory. In an experimental evaluation our method took the same time and generated a similar amount of errors as the usual approach, but reduced the number of control points needed by over 90%. The reduced number of control points can properly reconstruct a movement path and drastically decrease the number of control point selections required for movement path modification. For control manipulation, the task completion time was reduced and there was less hand movement needed than with conventional method. Our method can be applicable to drawing or curve editing method in immersive In-Situ AR based education, game, design, animation, simulation application domains.

핵심어: *Digilog Book, Immersive augmented reality, augmented reality authoring, 3D object movement path editing, tangible user interface, 3D object selection and manipulation.*

본 연구는 교육과학기술부의 글로벌프론티어연구개발사업의 연구결과물로 수행되었음.

*주저자 : 광주과학기술원 정보기전공학부 박사과정; e-mail: tha@gist.ac.kr

**교신저자 : 광주과학기술원 정보기전공학부 교수; e-mail: wwoo@gist.ac.kr

1. 서론

디지털로그 북 (Digilog Book)은 유비쿼터스 가상현실 기술을 적용하여 기존 서적과 디지털 콘텐츠를 통합함으로써, 아날로그적 감성과 디지털 오감을 제공하는 차세대 출판물이다 [1]. 아틀렛(ARtalet) 저작도구는 이러한 디지털로그 북을 쉽게 제작하기 위한 저작 소프트웨어로서 프로그래밍에 대한 전문적인 지식이 없는 사용자를 대상으로 개발되었으며, 3D 객체의 정적인 속성 편집 기능을 지원한다[2]. 이 기능을 통해 3D 객체 복사, 3차원 공간에 배치, 크기 조절, 색상 조절 기능 등을 수행하여 디지털로그 북의 장면을 생성할 수 있다. 추가적으로 증강된 3D 객체와의 상호작용을 통한 사용자의 체험과 흥미를 향상시키기 위해서는 3D 객체의 동적인 움직임을 생성하고 조작할 수 있는 방법이 필요하다.

증강현실 환경의 3D 객체 움직임 저작과 관련된 연구는 다음과 같이 크게 세 가지로 방법으로 분류된다. 1) 3D 모델링 소프트웨어를 사용하여 이동하는 3D 객체를 제작한 후 AR 라이브러리/툴킷(Toolkit)을 이용하여 로딩 하는 방법, 2) Director¹⁾[3], XNA Game Studio²⁾[4], 또는 Virtools³⁾ [5]와 같은 상용 소프트웨어로 이동하는 3D 객체를 제작한 후 AR 플러그 인(Plug-in) 모듈을 연동하여 결과를 확인하는 방법, 그리고 3) 감각형 사용자 인터페이스(TUI: Tangible User Interface)[6] 기반의 몰입형 In-Situ AR 환경에서 사용자가 직접 자신의 손의 움직임을 이용하여 이동 궤적을 설정하고 그 결과를 동일 공간인 AR 환경에서 바로 확인해 볼 수 있는 방법[7]으로 나뉜다.

3D 모델링 소프트웨어를 사용하는 방법은 사용자에게 전문적인 학습을 요구할 수 있으며, VR 환경에서 저작된 결과를 AR 환경으로 변환하여 결과를 확인해야하는 반복적인 작업이 필요하다. 한편 기존의 감각형 사용자 인터페이스기반의 In-Situ 이동 궤적 생성 방법은 사용자의 손의 움직임을 이용하여 3D 객체의 이동 궤적을 직관적으로 생성할 수 있는 장점이 있으나, 사용자 손의 떨림으로 인한 불안정한 이동 궤적의 생성, 매 프레임 저장된 수많은 이동 좌표로 인한 저장과 추후 수정의 어려움, 그리고 작고 밀집된 이동 좌표객체의 선택이 어렵다는 문제점이 존재한다.

본 논문은 기존의 상용 소프트웨어에 대한 학습을 필요로 하지 않으며, 상대적으로 사용하기 쉽고 직관적인 사용자 인터페이스를 제공하는 몰입형 In-Situ 증강현실 기반의 이동 궤적 저작 방법에 초점을 맞춘다. 구체적으로 가상 손 기법(Virtual hand technique) 기반의 6 DOF(Degree of Freedom) 조작 도구를 이용하여 직관적으로 이동 궤적을

생성하고 조작할 수 있다. 조작도구의 이동 좌표에 대한 조정점(Control point) 할당 검사를 통해 적절한 조정점을 생성하며, 저장된 조정점과 스플라인을 이용하여 부드러운 곡선 형태로 이동 궤적을 보간(Interpolation)한다. 또한 거리, 방향, 빈도, 그리고 시간 스코어(Score)를 기반으로 한 동적 선택 방법을 통해 작고 밀집된 이동 궤적의 조정점을 효과적으로 선택한다.

본 논문은 다음과 같은 순서로 구성된다. 제 2장에서는 기존의 관련연구들을 살펴보고, 제 3장에서는 제안하는 방법들의 기술적인 사항을 설명한다. 그리고 4장에서는 실험 및 구현과 관련된 사항을 언급하며 마지막으로 5장에서 토론과 결론을 맺는다.

2. 관련 연구

2.1. In-Situ AR 저작 방법

기존의 AR 콘텐츠 저작과 관련된 연구를 살펴보면, VR 환경의 모델링 툴(소프트웨어)을 이용하여 움직이는 3D 객체를 제작한 후, 이를 AR 라이브러리/툴킷 등을 이용하여 로딩 하는 것이 일반적인 방법이었다. 최근에는 VR 모델링 툴에 AR 플러그인을 연동하여, VR 환경에서 저작된 결과물을 AR 환경으로 변환하여 테스트할 수 있는 방법들이 소개되고 있다[8][9].

하지만 이동 궤적의 수정이 필요한 경우, 다시 VR 환경으로 변환을 한 후 키보드/마우스 같은 2차원 사용자 인터페이스를 이용하여 x, y, z축 각각을 변경해가며 원하는 이동 궤적으로 수정을 해야 한다. 그리고 다시 AR 환경으로 변경하여 결과를 확인해야하는 수고가 필요하다.

반면 본 논문에서 다루는 몰입형 In-Situ 증강현실 기반의 이동 궤적 저작 방법은 VR 환경으로의 변환 없이 AR 환경에서 실시간으로 이동 궤적을 수정하고 바로 결과를 확인할 수 있다. 즉 사용자의 의도를 즉각적으로 반영할 수 있다는 장점이 있다.

또한 기존 VR 환경에서의 키보드/마우스와 같은 2차원 사용자 인터페이스가 아닌 AR 환경에서 6 DOF를 제공하는 3차원 조작도구를 이용하여 3D 이동 궤적을 직조작(Direct manipulation) 할 수 있다는 것을 특징으로 한다. 즉 동일 AR 환경에서 3차원 사용자 인터페이스를 사용하여, 사용자의 공간적 직관성을 향상시킬 수 있다는 장점이 있다.

2.2. 이동 궤적 모델링 방법

이동 궤적의 외형적 특징을 잘 반영할 수 있으며 직관적으로 이동 궤적을 수정할 수 있는 방법이 필요하다. 본 논문에서

1) Director, <http://www.adobe.com/products/director> (2010.6.30)

2) XNA Game Studio, <http://www.xna.com> (2010.6.30)

3) Virtools, www.virttools.com (2010.6.30)

서는 직조작 방법을 이용하여 최소한의 입력만으로 이동 궤적의 외형을 수정할 수 있는 이동 궤적 모델링 방법을 우선적으로 고려한다.

가장 간단한 이동궤적 모델링 방법은 다항식 보간법(Polynomial interpolation)을 이용하는 것이다. 즉 n개의 조정점이 있을 경우 n-1차 다항식을 이용하여 이동 궤적을 표현하는 것이다. 그러나 조정점이 많아질수록 다항식의 차수가 높아지므로, 계산 복잡성이 증가하며 실시간 조작이 어려워진다. 또한 진동현상(Undulations)이 발생할 수 있다.

이러한 단점을 해결하기 위해서, 스플라인(Spline)을 이용할 수 있다. 즉 구간별(Piecewise)로 3차 이하의 저차(Low-degree) 다항식을 적용하여 이동 궤적을 생성하는 것이다. 스플라인 보간법은 곡선의 조정점 통과 여부를 기준으로 근사 스플라인(Approximating spline)과 보간 스플라인(Interpolating spline)으로 분류할 수 있다[10].

Bezier 곡선, B-spline, NURBS등과 같은 근사 스플라인은 조정점을 항상 통과하지 않으며, 원하는 형태로 곡선을 수정하기 위해서는 조정점과 접선벡터(Tangent vector), 노트(Knot), 가중치(Weight)등을 변경하여 곡선의 형태에 영향을 미치는 긴장도(Tension) 또는 바이어스(Bias)등과 같은 특성을 조정해야한다. 그러나 이러한 파라미터들을 직관적으로 제어하는 것은 쉽지 않으며 많은 시간이 소요될 수 있다 [11].

한편 보간 스플라인은 주어진 조정점을 통과하도록 곡선이 생성되며, 사용자는 직관적으로 조정점을 조작하여 곡선의 형태를 변경할 수 있다. 보간 스플라인의 한 종류인 카디날 스플라인(Cardinal spline)은 곡선의 긴장도 변수(0에서 1사이의 값)를 이용하여 곡선의 형태를 조정할 수 있으며, 이 값을 0.5로 설정한 것이 카드물-롬(Catmull-Rom) 스플라인이다[12]. 카드물-롬 스플라인은 기존의 여러 스플라인 표현방법에 비하여 제어할 수 있는 확장성이 다소 부족할 수 있지만, 조정점만으로도 쉽게 곡선의 외형을 조작할 수 있다는 장점이 있다.

2.3. 이동 궤적의 조정점 선택 방법

증강현실에서 가상공간의 배경이미지는 1인치 사용자 시점의 카메라로부터 실시간으로 입력받은 이미지로 대체가 된다. 이러한 조건에서는 1인치 기준 시점(Egocentric)의 3D 객체 선택 방법이 주로 사용된다. 이 방법은 크게 무한대의 가상 선과 최단 거리에서 교차되는 3D 객체를 선택하는 레이캐스팅(Ray casting) 방법과 사용자 손의 좌표와 1:1로 일치하는 가상 좌표를 이용하여 3D 객체를 선택하는 가상 손(Virtual hand) 방법으로 분류된다[13].

레이캐스팅 방법은 사용자의 손이 닿기 어려운 원거리의 객

체를 선택할 수 있는 장점이 있는 반면, 이동 궤적의 조정점과 같은 매우 작은 객체를 선택하기 어렵다는 문제점이 있다. 이를 보완하기 위해 영역 기반의 선택 기법들 [14][15][16]이 제안되었지만, 3D 객체의 6 DOF 조작에는 제약이 따른다[17]. 따라서 AR 환경에서는 3D 객체의 자유로운 6 DOF 조작을 위해 가상 손 기법들을 주로 사용되었다[7][18][19].

그리고 이동 궤적의 조정점과 같이 작고 밀집된 3D 객체를 선택하기 위한, 가장 기초적인 방법은 좌표 기반 선택 방법이다. 이 방법은 가상손의 좌표와 3D 객체의 좌표 간의 거리가 근접할 경우 해당 객체를 선택하는 것이다. 하지만 이 방법은 사용자에게 AR 공간의 정확한 3D 깊이 인식을 요구하는 것으로 정신적/물리적인 피로를 야기할 수 있다.

이러한 단점을 해결하기 위해서, 동적으로 변하는 선택 영역을 사용하는 버블 커서(Bubble cursor) 방법이 제안되었다 [20]. 그림 1(a)과 같이 보로노이(Voronoi) 다이어그램 영역을 고려하여, 가상손(조작도구)의 좌표가 포함된 영역의 3D 객체(조정점)가 자동으로 선택되도록 하며 시각피드백으로써 반투명 원구의 크기를 동적으로 변경시킨다. 이렇게 함으로써 조작도구를 조정점에 정확히 위치하지 않아도 원하는 조정점을 적절히 선택할 수 있다.

그러나 그림 1(b)과 같이 조정점들이 밀집되어 있는 경우, 선택 가능한 유효거리(Effective width)가 좁아지기 때문에 조작도구를 선택하려는 조정점 위치로 가까이 이동해야 한다. 이것은 버블 커서의 장점을 잃어버리게 할 수 있다. 또한 사용자 손의 불안정한 움직임으로 인한 조작도구의 잦은 위치 변동은 원하는 조정점의 선택을 어렵게 할 수 있다.

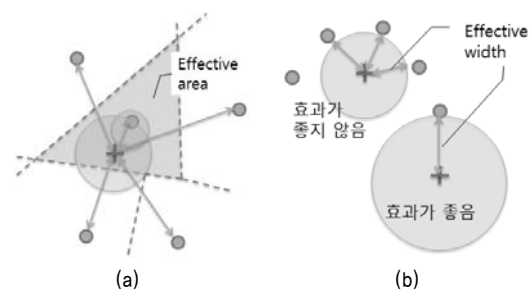


그림 1. 버블 커서의 객체 선택 방법: (a) 보로노이(Voronoi) 다이어그램 영역을 이용; (b) 3D 객체 밀집된 경우에 발생할 수 있는 현상.

3. 이동 궤적 생성 및 조작

3.1 이동 궤적 생성

조정점의 좌표는 그림 2와 같이 월드 기준 좌표계 M_{world} 를 기준으로 한, 조작도구 앞에 증강된 가상 선 끝점의 변환 행렬 M_{tip} 의 변환행렬 값을 이용하여 획득할 수 있다. 여기서 가상 선 끝점의 변환 행렬 M_{tip} 은 조작도구 변환 행렬 M_{prop} 로부터 일정거리만큼 이동된 상태이다.

기존 방법과 같이 조작도구가 인식될 때 마다 조작도구의 이동 좌표를 조정점으로 할당하는 경우, 특정 구간에서 조작도구의 위치 변동이 적다면 해당 구간 내에서 매우 많은 조정점들이 밀집되어 생성될 수 있다. 이러한 문제점을 해결하기 위해 본 논문에서는, 수식 1과 수식 3처럼 이미 저장된 조정점의 위치 $P_{CP_{i-1}}$ 와 현재 조작도구의 위치 $P_{Prop}(t)$ 를 고려하여 조정점들이 일정 거리 간격 이상의 위치에 생성되도록 한다. 이렇게 함으로써 조정점들이 특정 구간에서 밀집되는 현상을 줄일 수 있다.

또한 이동 궤적의 디테일을 반영하기 위해서, 사용자는 특정 위치에서 조작도구를 약간 더디게 움직일 수 있다. 이러한 사항을 고려하여 수식 2와 수식 3과 같이 조작도구의 위치가 일정 거리 내에서 연속적으로 머무는 경우 조정점이 할당되도록 한다.

종합적으로 수식 3과 같이 조정점과 조작도구의 거리인 D_{CP_Prop} 와 조작도구가 특정 구간에서 머무는 시간을 고려한 T_{Dwell} 값이 각각 특정 임계값 이상이 될 때, 해당되는 조작도구의 좌표 $P_{Prop}(t)$ 를 조정점 P_{CP_i} 로 할당한다.

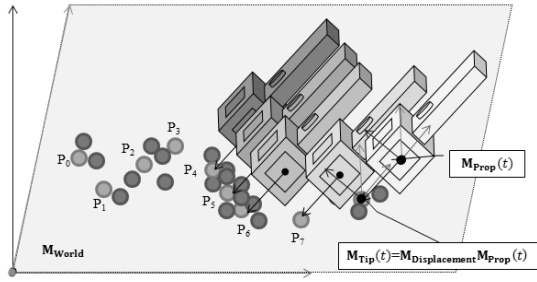


그림 2. 이동 궤적의 조정점 할당 방법. 조정점과 조작도구 사이의 거리와 조작도구의 특정 구간에서 머무는 시간을 고려하여 조정점을 할당함.

$$D_{CP_Prop} = |P_{CP_{i-1}} - P_{Prop}(t)| \quad (1)$$

$$T_{Dwell} = \begin{cases} \sum 1, & \text{if } |P_{Prop}(t-1) - P_{Prop}(t)| < D_{Thres} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$$P_{CP_i} = P_{Prop}(t), \text{ if } D_{CP_Prop} > D_{Thres} \text{ or } T_{Dwell} > T_{Thres} \quad (3)$$

저장된 이동 궤적은 스플라인을 이용한 매개변수 곡선 방정식(Parametric curve equation)으로 변환하는 모델링 과정을 통해, 부드러운 형태로 변환된다. 이것은 비전 기반 트래킹 에러 또는 사용자의 손의 떨림으로 인해 일시적으로 발생될 수 있는 이동 궤적의 지터링(Jittering) 효과를 보상할 수 있는 장점이 있다. 그리고 저장된 조정점을 이용하여 추후 이동 궤적 곡선을 복원할 수 있다.

본 논문에서는 카트몰-롬 스플라인을 이용하여 저장된 이동 궤적의 조정점들을 매개변수 곡선 방정식으로 변환한다. 주어진 4개의 조정점 CP_{i-1} , CP_i , CP_{i+1} , CP_{i+2} 과 비율 변수 t 를 이용하여 곡선 세그먼트(Curve segment) $q(t)$ 를 정의한

다(수식 4). 여기서, 변수 $t(t \in [0,1])$ 는 두 개의 조정점(CP_i 에서 CP_{i+1}) 사이의 거리 비율을 의미한다.

$$P(t) = \frac{1}{2} [1.0 \quad t \quad t^2 \quad t^3] \begin{bmatrix} 0 & 2 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 2 & -5 & 4 & -1 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} CP_{i-1} \\ CP_i \\ CP_{i+1} \\ CP_{i+2} \end{bmatrix} \quad (4)$$

3.2 이동 궤적 조작

본 논문은 기존의 정확한 좌표 일치 방법 및 버블 커서를 이용한 객체 선택 방법의 단점을 해결하고자, 그림 4와 같은 총 4개의 스코어(Score)를 이용한 동적 조정점 선택 방법을 제안한다.

첫 번째는 거리(Distance) 스코어으로써, 선택 볼륨의 중심좌표로부터 볼륨에 포함된 조정점 객체간의 거리를 반영하며 중심에 가까운 조정점 객체일수록 높은 스코어를 갖는다. 수식 5에서 D_{Radius} 는 볼륨의 반지름 길이이며, $D_{CP_i}(t)$ 는 볼륨의 중심 좌표로부터 조정점 객체 CP_i 까지의 거리이다.

$$S_{DiscP_i}(t) = 1 - \frac{D_{CP_i}(t)}{D_{Radius}}, S_{DiscP_i}(t) \in [0,1] \quad (5)$$

그리고 조작도구의 방향(Direction) 스코어는 조작도구의 방향으로부터 조정점 객체의 방위각을 계산하여 사용자가 의도하는 방향에 위치한 객체가 선택되도록 한다. 각도(Radian)는 수식 6처럼 조작도구의 방향 벡터 V_{Prop} 와 볼륨의 중심 좌표를 기준으로 한 조정점 객체 CP_i 의 방향 벡터 $V_{CP_i}(t)$ 간의 내적을 이용하여 구한다. x 는 같은 방향일 경우 1의 값을 가지며 반대 방향일 경우 -1의 값을 갖는다. 이를 응용하여 수식 7처럼 스코어 $S_{DirCP_i}(t)$ 값이 0에서 1의 값을 갖도록 조정한다.

$$\text{Angle}_{CP_i}(t) = \cos^{-1} \left(x = \frac{V_{Prop} \cdot V_{CP_i}(t)}{|V_{Prop}| |V_{CP_i}(t)|} \right) \quad (6)$$

$$S_{DirCP_i}(t) = 1 + \frac{(x-1)}{2}, S_{DirCP_i}(t) \in [0,1] \quad (7)$$

빈도(Frequency) 스코어는 볼륨 내에 조정점 객체가 포함되는 횟수를 반영한 것이다. 수식 8처럼 볼륨 내에 조정점이 포함되는 경우 해당 조정점의 빈도수를 증가시키며 포함되지 않을 경우 빈도수를 감소시킨다. 그리고 수식 9처럼 빈도수를 최소 F_{Lower} 에서 F_{Upper} 범위에 있도록 제한하고, 정규화 과정을 거쳐 0에서 1의 범위에 포함되도록 한다.

$$\text{In}_{CP_i}(t) = \begin{cases} 1, & \text{if } |P_{CP_i}(t) - P_{Prop}(t)| < D_{Radius} \\ -1, & \text{otherwise} \end{cases} \quad (8)$$

$$F_{CP_i}(t) = \sum \text{In}_{CP_i}(t), F_{CP_i}(t) \in [F_{Lower}, F_{Upper}] \quad (9)$$

마지막으로 시간 스코어는 볼륨 내에 객체가 연속적으로 포함되는 시간을 반영한 것으로, 조정점이 오랜 시간 동안 볼륨 내에 연속적으로 포함되어 있을수록 높은 스코어 값을

값이다. 볼륨 내에 포함되는 경우 횡수를 증가시키며 그렇지 않은 경우 0으로 값을 설정한다(수식 10). 즉 만약 한번이라도 볼륨 내에 포함이 되지 않으면 0으로 초기화된다. 그리고 수식 11처럼 이 값이 0에서 T_{Upper} 범위에 있도록 제한하며, 정규화과정을 거쳐 0에서 1의 범위에 포함되도록 한다.

$$In_{CP_i}(t) = \begin{cases} 1, & \text{if } |P_{CP_i}(t) - P_{Prop}(t)| < D_{Radius} \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

$$T_{CP_i}(t) = \sum In_{CP_i}(t), T_{CP_i}(t) \in [0, T_{Upper}] \quad (11)$$

그리고 수식 12와 같이 현재 프레임 t 에서 조정점 객체 CP_i 의 최종 스코어 총합인 $S_{ContribCP_i}(t)$ 를 구하고 이전 프레임의 스코어 값에 대하여 현재 스코어 값을 업데이트한다(수식 13)[21]. 여기서 C_s 과 C_g 는 각각 이전 스코어 총합의 감소율과 현재 스코어 총합의 증가율을 의미하는 가중치이며 C_s 와 C_g 의 합은 1이다($C_s, C_g \in [0, 1]$). 최종적으로 수식 14와 같이 현재 시각 t 에서의 최대 스코어 값을 가지는 조정점을 선택 가능한 조정점 CP_i 으로 결정한다.

$$S_{ContribCP_i}(t) = \alpha S_{DisCP_i}(t) + \beta S_{DirCP_i}(t) + \gamma S_{FreqCP_i}(t) + \omega S_{TimeCP_i}(t) \quad (12)$$

$$S_{TotalCP_i}(t) = C_s S_{TotalCP_i}(t-1) + C_g S_{ContribCP_i}(t) \quad (13)$$

$$CP_i = \arg \max_i S_{TotalCP_i}(t) \quad (14)$$

사용자는 조정점을 선택을 한 후, 조작도구를 이용하여 위치를 변경할 수 있다. 드래그 상태 동안 조작도구 앞에 증강된 가상 선 끝점의 변환 행렬 $M_{tip}(t)$ 로부터 조정점의 위치 속성을 설정하며[22], 이에 따라 3D 객체의 이동 궤적이 실시간으로 갱신한다.

4. 구현 및 평가

4.1 실험 설계

실험 1은 이동 궤적의 생성 과정을 평가하는 것으로, 3D 공간에 위치한 목표 이동 궤적을 따라 조작 도구를 움직이며 이동 궤적을 생성하는 것이다(그림 3). 비교 기법은 기존의 스플라인 곡선을 사용하지 않고 매 프레임마다 조작도구의 위치를 저장하는 방법(NO_SPLINE)과 카트뮐-룸 스플라인을 이용하여 이동 궤적을 생성하는 방법(SPLINE)이다.

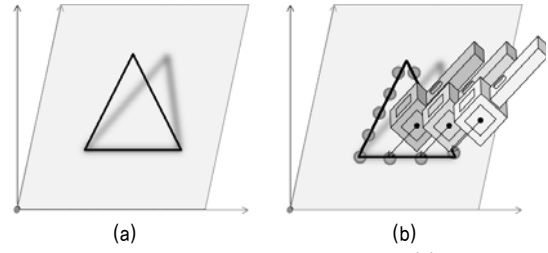


그림 3. 실험 1. 이동 궤적 생성 방법 실험: (a) 목표 이동 궤적이 주어졌 있음; (b) 한붓그리기로 목표 이동 궤적을 따라 조작도구를 이동시킴.

실험 2는 3D 공간상에 목표 이동 궤적과 최대한 비슷한 형태로 주어진 이동 궤적의 조정점을 일치되도록 하는 것이다(그림 4). 이 실험을 통해 사용자가 얼마나 빠르고 정확하게 이동 궤적의 조정점을 선택하고 조작 할 수 있는지 평가할 수 있다. 비교 기법은 포인트 기반 선택 방법(PS: Point Selection), 버블 커서의 볼륨 기반 선택 방법(VS: Volume Selection), 그리고 스코어 기반 선택 방법(SS: Score Selection)으로 구성된다.

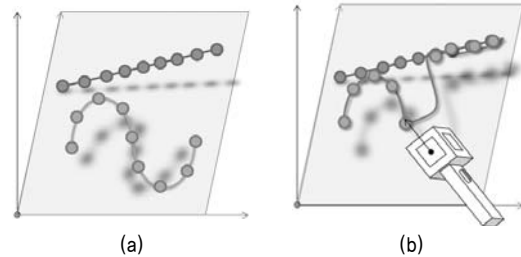


그림 4. 실험 2. 이동 궤적의 조정점 선택 방법 실험: (a) 목표 이동 궤적과 사용자가 조작해야 할 조정점들이 주어졌 있음; (b) 사용자는 주어진 조정점을 선택하여 목표 이동 궤적의 조정점 위치로 이동시킴.

상기 실험을 수행하기 위해서, 피험자 내 (Within-subject) 반복 측정(Repeated measures) 방법을 수행한다. 표 1은 실험에서 사용될 독립 변수를 보인다. 공통적으로 테스트하는 이동 궤적은 한 붓 그리기가 가능한 삼각형, 사각형, 원형, 그리고 사인파 형태와 유사한 자유곡선이며 모두 3D 공간상에 위치한다.

그리고 실험 1은 4종류의 이동 궤적과 2종류의 이동 궤적 모델링 방법으로 구성 되며(총 8번시도), 실험 2는 4종류의 이동 궤적과 3종류의 이동 궤적의 조정점 선택 방법으로 구성 된다(총 12번시도).

표 1. 독립 변수

실험	독립 변수
공통: 이동 궤적	한 붓 그리기가 가능한 도형: 정삼각형, 정사각형, 원형, 자유 곡선
실험 1. 모델링 방법	기존 이동 궤적 설정 방법 (NO_SPLINE) 카드물-롬 스플라인 방법 (SPLINE)
실험 2. 조정점 선택 방법	포인트 기반 선택 방법 (PS: Point Selection) 볼륨 기반 선택 방법 (VS: Volume Selection) 스코어 기반 선택 방법 (SS: Score Selection)
시도 횟수	실험 1: 총 8번 (4 종류 이동 궤적, 2가지 기법) 실험 2: 총 12번 (4 종류 이동 궤적, 3가지 기법)

표 2는 본 실험에서 사용될 종속 변수를 보인다. 실험 1의 종속 변수는 이동 궤적 생성을 통한 조정점의 개수, 목표 이동 궤적에 대한 오차 그리고 완료 시간이다. 오차는 생성된 조정점과 목표 이동 궤적에 이르는 최단거리를 통해 계산되며, 완료 시간은 사용자가 조작도구의 버튼을 눌렀을 때부터 버튼을 놓을 때까지의 측정된 시간이다.

실험 2의 종속 변수는 클릭횟수, 조정점을 선택할 수 있는 유효거리, 조작도구의 이동거리, 그리고 완료 시간이다. 각각의 조정점은 목표 이동 궤적의 해당 조정점과의 10mm거리 내로 사용자에게 의해 이동되어야만 작업이 완료된다. 그렇지 않은 경우 사용자는 다시 조정점을 선택해서 오차 범위 이내로 조정점을 위치시켜야 한다(클릭횟수 증가 함). 그리고 유효거리는 조정점 선택이 가능한 상태에서 사용자가 클릭을 하였을 때, 해당 조정점과 조작도구간의 거리를 의미한다. 한편 조작도구의 이동거리는 작업을 완료할 때까지의 조작도구의 이동거리이며, 완료 시간은 첫 클릭 시각부터 작업 완료 시각까지의 시간이다.

표 2. 종속 변수

실험	종속 변수
실험 1. 이동 궤적 모델링 방법	포인트 개수 오차 (단위: mm) 완료 시간 (단위: ms)
실험 2. 이동 궤적의 조정점 선택 방법	클릭 횟수 유효거리 (단위: mm) 조작 도구의 이동거리 (단위: mm) 완료 시간 (단위: ms)

4.2 구현 및 실험

실험은 마커 트래킹을 위한 광량이 충분한 실내 공간에서 수행되었다(그림 5). 카메라⁴⁾는 640x480픽셀 해상도로 초당 30프레임의 영상을 획득하며 안정적인 마커 트래킹을 위해 카메라의 자동 노출(Auto exposure)기능을 설정하였으며 마커 보드 위치로 카메라 초점(Focus)을 조정하였다. 그리고 2.0GHz CPU와 4GB 메모리를 장착한 컴퓨터에서, 비전 기

4) 웹 카메라 C905, <http://www.logitech.com> (2010.6.30)

반 마커트래킹과 그래픽 렌더링을 위해 osgART 라이브러리⁵⁾을 사용하였다.



그림 5. 실험 환경: (a) 사용자는 디스플레이 장치를 통해 증강 장면을 볼 수 있음; (b) 사용자의 앞에는 기준 좌표계가 되는 트래킹 패턴 보드가 놓여 있음. 사용자는 조작 도구를 손에 쥐고 있음.

또한 사용자의 손에 의해 패턴이 가려져도 증강을 위한 패턴 인식/카메라 자세 추정이 가능하도록 다수 패턴 보드를 사용하였으며, 카메라 시야에 트래킹 패턴이 항상 위치할 수 있도록 보드를 크기(300x600mm)로 제작하였다. 보드에 사용된 각각의 마커는 60mm크기이며 조작도구에 사용된 마커는 40mm크기이다. 손잡이 부분을 제외한 정육면체의 각 면(총 5면)에 한 개의 마커를 각각 부착하였다.

그림 6은 실험에서 사용된 이동궤적을 보인다. 모두 3D 공간상에서 패턴 보드 평면을 기준으로 수직방향 100mm 높이에 위치한다. 그리고 이동궤적의 그림자를 생성하여 사용자의 깊이인식에 도움을 주도록 하였다.

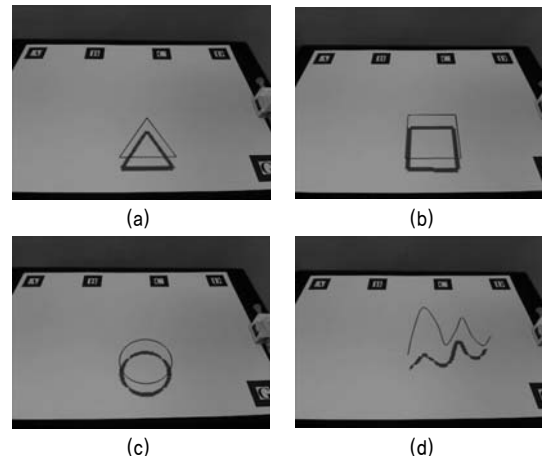


그림 6. 실험에서 사용된 이동궤적: (a) 정삼각형; (b) 정사각형; (c) 원형; (d) 자유곡선

그림 7은 이동 궤적 생성의 구현 결과의 한 예를 보인다. 그림 7(a)는 기존의 이동 궤적 생성 방법 (NO_SPLINE)을 이용한 것이며, 그림 7(b)는 본 논문에서 제안하는 스플라인 방법(SPLINE)을 이용한 것이다. 그림에서 붉은 색의 원형 객체들은 저장된 조정점을 표시한다. 실험에서는 사용된 파라미터 D_{Thres} 의 값은 30mm 이며 T_{Thres} 는 20프레임으로 설

5) osgART, www.artoolworks.com/community/osgart (2010.6.30)

6) 무선 프리젠티터 WP-8000, <http://www.3m.com> (2010.6.30)

정하였다. 즉 조작도구가 30mm미만 거리에서 20프레임 이상 계속 위치하거나, 조정점과 조작도구의 거리가 30mm이상이라면 조정점이 배치된다.

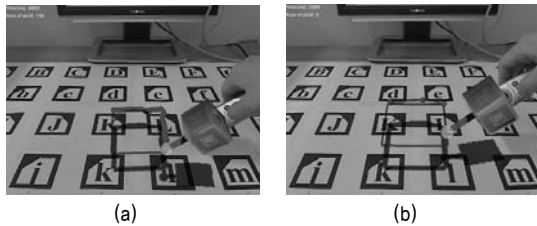


그림 7. 이동 궤적 생성 방법 실험(예: 정사각형 이동 궤적): (a) 기존 이동 궤적 설정 방법 (NO_SPLINE); (b) 카트물-롤 스플라인 방법 (SPLINE).

그림 8은 이동 궤적 조정 방법의 구현 예시를 보인다. 포인트 기반 선택 방법(PS)의 경우 조정점에 10mm거리 이내로 근접해야 조정점 선택이 가능하며, 보로노이 볼륨 기반 선택 방법(VS)은 최 근접 조정점이 선택 가능하다. 그리고 스코어 기반 선택 방법(SS)은 최적의 스코어를 가지는 조정점이 선택 가능하다. 모든 이동 궤적은 10개의 조정점으로 구성된다. SS 방법에서 D_{Radius} 는 50mm로 α , β , γ , 그리고 ω 는 각각 1로 설정하였다. 그리고 C_s 와 C_g 의 값은 각각 0.5로 설정하였다.

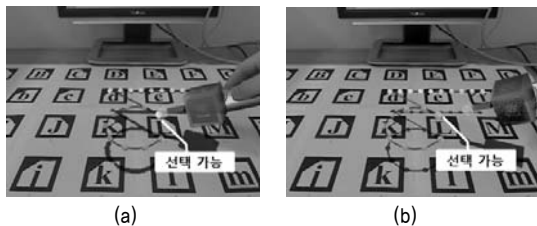


그림 8. 이동 궤적의 조정점 선택 방법 실험(예: 원형 이동 궤적): (a) 포인트 기반 선택 방법 (PS); (b) 스코어 기반 선택 방법 (SS).

사용성 평가는 컴퓨터 과학을 전공하는 대학생 5명(평균나이 22세)을 대상으로 파일럿 테스트 수준에서 수행되었다. 모두 오른손잡이였으며 증강현실 어플리케이션에 대한 사전 경험이 없었다. 실험 전에 동영상과 시연을 통하여 평가 방법들을 피실험자에게 설명한 후, 피실험자가 직접 연습할 수 있도록 하였다. 실제 실험은 약 20분 정도가 소요되었다.

샘플수가 적은 경우의 통계 분석을 위해서 비모수 검정 방법을 이용하였다. 실험 1의 두 종류의 이동궤적 생성 방법의 실험 결과를 분석하기 위하여 Mann-Whitney U 테스트를

수행하였으며, 세 종류의 이동궤적 조작 방법의 실험 결과 분석을 위하여 Kruskal-Wallis 테스트를 수행하였다.

4.3 실험 결과

그림 9는 논문에서 제안하는 스코어 기반의 조정점 선택 방법의 실험 결과를 보인다. 3개의 조정점(CP0, CP1, CP2)이 배치한 상황에서 50프레임 동안 각 스코어 값의 추이를 확인할 수 있다.

프레임 1에서 조작도구는 CPO와 가장 가까운 거리에 있으며 방향도 비슷하며 빈도 및 포함시간 스코어 값도 높다. 이때 사용자가 조작도구를 시계방향으로 조금씩 회전하면서 오른쪽으로 이동하면, CP1의 방향 스코어가 높아지며 CP1과 CP2의 거리 스코어 값이 상대적으로 증가하게 된다. 이와 동시에 빈도 및 포함시간 스코어 값도 높아지며, 17-36프레임에서는 CP1이 선택가능한 상태가 된다. 보로노이 볼륨선택 방법(VS)과 같은 거리 측도를 이용하는 방법에 비해, 스코어 기반 선택 방법(SS)을 이용하면 CP1이 선택될 수 있는 가능성이 높아질 수 있다.

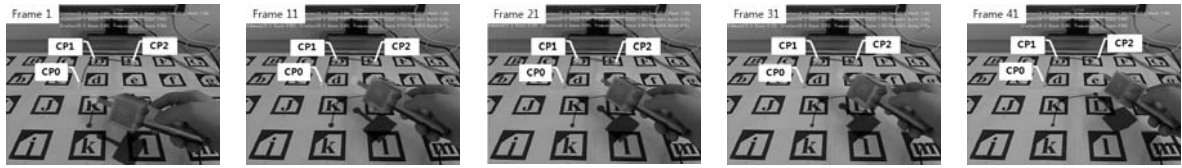
실험 1. 오차와 완료시간에서 SPLINE와 NO_SPLINE는 서로 유의한 차이를 보이지 않았다(표 3). 하지만 조정점의 개수 비교에서 SPLINE와 NO_SPLINE는 유의한 차이를 보였다. SPLINE 방법은 기존 NO_SPLINE 방법에 비해 약 10% 이내의 조정점만으로 이동 궤적 생성이 가능하였다.

실험 2. 클릭횟수와 유효거리에서 세 방법 간에 유의한 차이가 있었다(표 4). 특히 PS 방법은 VS와 SS 방법에 비하여 적게는 3회에서 많게는 7회 정도의 추가적인 클릭이 요구되었다.

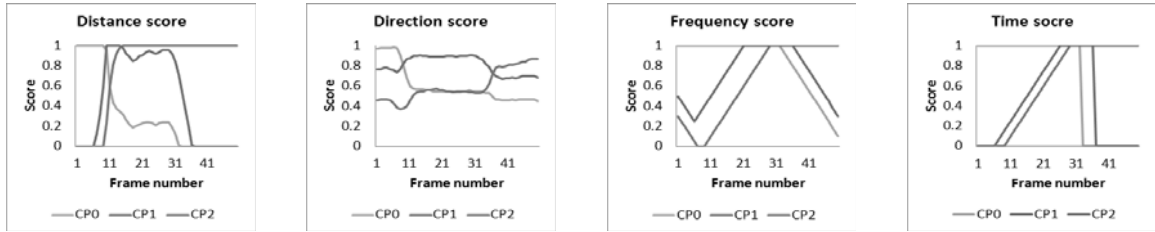
유효 거리는 PS, VS, SS 방법 순으로 증가하였으며 유의한 차이를 보였다. 구체적으로 PS의 유효 거리는 약 10mm 이내의 범위로 제한적 이었으나, 확장된 선택 영역을 이용하는 VS와 SS방법의 경우에는 유효거리가 30mm 이상으로 증가하였다.

조작 도구의 이동 거리에 대한 결과를 살펴보면, 대체적으로 PS, VS, SS 방법 순으로 감소하였다. PS방법이 가장 긴 조작 도구 이동거리가 요구되었으며, VS방법이 SS방법이 보다 좀 더 긴 이동거리를 필요로 하였다(원형의 이동궤적예외).

마지막으로 완료시간의 경우 PS, VS, SS 방법 순으로 감소하는 경향을 보였으나 상호 유의한 차이는 보이지는 않았다(자유곡선의 경우 제외).



(a)



(b)

(c)

(d)

(e)

그림 9. 이동 궤적의 조정점 선택을 위한 스코어 측정의 예시: (a) 총 50 프레임 동안의 테스트 이미지 시퀀스 (10 프레임 간격으로 정지 영상을 표시함); (b) 거리 스코어; (c) 방향 스코어; (d) 빈도 스코어; (e) 시간 스코어.

표 3. 이동궤적 모델링 실험의 결과. 평균값을 표기하였으며, () 안에 표준편차를 기입하였음. 또한 유의확률이 $p < 0.05$ 인 경우 *를 표기하였음.

	정삼각형		정사각형		원형		자유곡선	
	NO_SPLINE	SPLINE	NO_SPLINE	SPLINE	NO_SPLINE	SPLINE	NO_SPLINE	SPLINE
조정점 개수	206.4 (61.496)	17.4 (7.162)	224 (44.855)	21.4 (7.893)	199.2 (71.391)	19.8 (5.541)	247 (43.618)	21 (5.612)
	Z=-2,611, p=0,008*		Z=-2,611, p=0,008*		Z=-2,619, p=0,008*		Z=-2,611, p=0,008*	
오차 (mm)	15,921 (7,765)	15,381 (7,619)	26,364 (9,134)	16,49 (6,996)	17,725 (4,731)	17,855 (5,692)	33,425 (15,862)	29,386 (17,483)
	Z=-0,313, p=0,841		Z=-1,776, p=0,095		Z=-0,104, p=1,000		Z=-0,313, p=0,841	
완료시간 (ms)	14410 (8333,184)	13240,4 (5984,964)	16312 (7739,462)	17140,6 (7555,237)	14238,2 (9022,049)	16245 (6056,73)	18636,8 (8082,462)	17526 (5936,895)
	Z=-0,522, p=0,69		Z=-0,313, p=0,841		Z=-0,522, p=0,69		Z=0,000, p=1,000	

표 4. 이동궤적 조작 실험의 결과. 평균값을 표기하였으며, () 안에 표준편차를 기입하였음. 또한 유의확률이 $p < 0.05$ 인 경우 *를 표기하였음.

	정삼각형			정사각형			원형			자유곡선		
	PS	VS	SS	PS	VS	SS	PS	VS	SS	PS	VS	SS
클릭횟수	15,200 (4,868)	10 -	10 -	14,2 (2,28)	10,8 (0,837)	10,6 (0,548)	13,4 (2,302)	10,2 (0,447)	10,2 (0,447)	18 (7,714)	11 (0,707)	10 -
	$\chi^2_{(2)}=9,882, p=0,007^*$			$\chi^2_{(2)}=7,621, p=0,022^*$			$\chi^2_{(2)}=10,251, p=0,006^*$			$\chi^2_{(2)}=7,996, p=0,018^*$		
유효거리 (mm)	7,669 (0,664)	32,140 (8,869)	37,997 (6,445)	8,063 (0,744)	36,152 (6,29)	45,458 (6,501)	7,459 (0,783)	36,624 (9,746)	37,086 (10,513)	7,904 (1,128)	46,355 (3,768)	52,572 (15,198)
	$\chi^2_{(2)}=9,98, p=0,007^*$			$\chi^2_{(2)}=10,82, p=0,004^*$			$\chi^2_{(2)}=9,42, p=0,009^*$			$\chi^2_{(2)}=9,42, p=0,009^*$		
이동거리 (mm)	3963,582 (557,688)	3322,939 (664,569)	2874,848 (736,024)	4424,388 (482,307)	3807,927 (374,884)	3337,122 (399,033)	4460,963 (241,828)	3462,993 (360,427)	3532,772 (592,696)	6061,359 (2310,79)	4177,947 (548,115)	3613,986 (288,108)
	$\chi^2_{(2)}=3,98, p=0,137$			$\chi^2_{(2)}=7,62, p=0,022^*$			$\chi^2_{(2)}=7,94, p=0,019^*$			$\chi^2_{(2)}=10,14, p=0,006^*$		
완료시간 (ms)	56639,4 (14198,1)	37769,6 (10346,7)	36679,6 (11199,6)	56657,2 (14158,9)	39839 (13257,7)	38625 (13492,3)	53374 (14637,6)	39143,4 (12070,4)	37601,6 (9730,1)	68697,4 (18052,1)	40966,8 (12327,9)	36439,8 (10589,0)
	$\chi^2_{(2)}=4,94, p=0,085$			$\chi^2_{(2)}=4,34, p=0,114$			$\chi^2_{(2)}=3,5, p=0,174$			$\chi^2_{(2)}=6,74, p=0,034^*$		

5. 토론 및 결론

5.1. 토론

실험 1. 본 논문에서 제안한 스플라인 기반 이동 궤적 생성 방법은 기존 방법에 비해 조정점의 개수를 90%이상 감소시킬 수 있었다. 모든 이동 궤적 포인트들을 저장하는 것에 비해, 조정점만을 저장하고 이동 궤적을 복원할 수 있는 방법은 사용자에게 추후 이동 궤적을 재수정을 할 수 있는 기회를 제공할 수 있다. 또한 부가적으로 조정점 개수 감소를 통해 이동 궤적 조작에 필요한 조정점 선택 횟수를 대폭 줄일

수도 있다.

한편 정삼각형이나 원형과 같이 곡률이 큰 이동 궤적의 경우, 제안한 방법은 기존 방법과 비슷한 오차를 보였으며 완료시간이 다소 증가한 결과를 보였다. 오차와 완료시간을 더욱 감소시키기 위해서는 조정점과 조작도구간의 거리뿐만 아니라 추가적으로 이동 경로의 곡률을 고려한 조정점 할당 방법이 필요할 것이라고 생각된다.

실험 2. SS방법은 PS와 VS방법보다 조정점 선택이 가능한 유효거리가 증가하였는데, 이는 조작도구를 적게 이동하여도 조정점 선택이 가능하다는 것을 의미한다.

그리고 조작 도구의 이동거리의 실험결과에서, PS 방법이 가장 긴 이동거리를 기록하였으며, VS방법이 SS방법 보다 좀 더 긴 이동거리를 요구하였다. 이는 SS방법이 PS와 VS 방법에 비해 다양한 스코어를 추가적으로 이용하여 조정점을 선택을 효율적으로 할 수 있기 때문일 것으로 판단된다.

또한 VS와 SS방법은 PS방법보다 더 적은 시간으로 작업을 완료할 수 있었지만 VS와 SS방법 간에는 큰 차이는 보이지 않았다. SS방법이 VS방법보다 상대적으로 유효거리가 크고 이동거리가 짧음에도 불구하고 작업 완료 시간은 큰 차이를 보이지 않은 이유는, 다양한 스코어 값을 계산하고 여러 후보들 중에 최적의 값을 가지는 조정점을 결정하는데 약간의 시간 지연이 존재하기 때문일 것으로 생각된다.

5.2. 결론 및 향후 연구

본 논문은 몰입형 In-Situ 증강현실 환경에서의 3D 객체 이동 궤적 편집 방법을 제안하였다. 실험 결과 제안한 이동 궤적 생성 방법은 기존 방법에 비교하여 오차와 완료시간은 유의한 차이가 없었지만 약 90% 이상의 조정점 개수 감소 효과가 있었다. 이를 통해 추후 이동궤적에 필요한 조정점 선택의 횟수를 대폭 감소할 수 있다.

또한 이동궤적 조작의 경우, 제안한 조정점 선택 방법은 기존 방법에 비해 조정점 선택에 사용되는 유효 거리가 증가하였으며 조작 도구의 이동 거리와 작업 완료 시간은 감소하였다. 즉 기존 방법에 비해 상대적으로 적은 손/팔의 움직임만으로 빠르게 이동 궤적을 조작할 수 있었다.

본 논문에서 제안하는 이동 궤적 편집 방법을 통해 사용자는 디지로그 북의 3D 객체에 움직임 속성을 설정하고 조작을 할 수 있었다. 이를 통해 독자는 좀 더 향상된 상호작용적인 디지로그 북을 체험할 수 있을 것으로 기대한다. 제안한 이동 궤적 편집 기술은 또한 몰입형 증강현실 환경의 교육, 게임, 디자인, 애니메이션, 시뮬레이션 등의 분야에서 드로잉, 스케칭의 곡선 편집 방법으로도 응용될 수 있다.

향후 연구로써, 다양한 이동 궤적과 좀 더 밀집된 조정점 조건에서 추가적인 실험이 수행될 것이다. 한편 이동 궤적을 수정하기 위해서 조정점 각각을 개별적으로 선택하고 조작하는 것은 번거로울 수 있다. 이를 해결하기 위해 여러 개의 조정점들을 동시에 선택할 수 있는 그룹 선택 방법에 대한 연구가 진행될 것이다. 마지막으로 이동 궤적의 디테일을 보전하면서도 손쉽게 조정점을 추가하고 삭제할 수 있는 방법에 대해서도 연구가 진행되어야한다.

참고문헌

[1] Ha, T., Lee, Y., Woo, W. (2010). Digilog Book for

Temple Bell Tolling Experience based on Interactive Augmented Reality with Culture Technology. Springer Virtual Reality (spVR).

- [2] 하태진, 이영호, 우은택. (2008). 디지로그 북 저작도구 ARtalet - 3 차원 객체 속성 저작. KHCI, 1권. pp. 314-318.
- [3] MacIntyre, B., Gandy, M., Dow, S., and Bolter, J. (2004). DART: A Toolkit for Rapid Design Exploration of Augmented Reality Experiences. ISMAR, pp. 172-181.
- [4] Oda, O., Lister, L., White, S., Feiner, S. (2008). Developing an Augmented Reality Racing Game. INTETAIN, pp. 8-10.
- [5] Geiger, C., Klompaker, F., Stoecklein, J., and Fritze, R. (2007). Development of an augmented reality game by extending a 3D authoring system. ACE, pp. 230-231.
- [6] ISHII, H., ULLMER, B., (1997). Tangible bits: towards seam-less interfaces between people, bits and atoms. ACM CHI, pp. 234-241.
- [7] Lee, G., Nelles, C., Billingham, M., Kim, G.J. (2004). Immersive Authoring of Tangible Augmented Reality Applications. ISMAR, pp. 172-181.
- [8] AR-media™ Plugin for Google™ SketchUp™, <http://www.inglobetechnologies.com> (2010.6.30)
- [9] Linceo VR, <http://www.seac02.it> (2010.6.30)
- [10] Shirley, P., Ashikhmin, M., Gleicher, M., Marschner, S., Reinhard, E., Sung, K., Thompson, W., and Willemsen, P. (2005). Fundamentals of Computer Graphics. A. K. Peters, Ltd.
- [11] Bartels, R.H., Beatty, J.C., Barsky, B.A. (1987). An introduction to splines for use in computer graphics and geometric modeling. Morgan Kaufmann Publishers Inc.
- [12] Catmull, E., Rom, R. (1974). A Class of Local Interpolating Splines. Computer Aided Geometric Design. Academic Press.
- [13] Bowman, D.A., Kruijff, E., LaViola, J.J., Poupyrev, I. (2004). 3D User Interfaces Theory and Practice. Addison-Wesley.
- [14] Liang, J., Green, M. (1994). JDCAD: A Highly Interactive 3D Modeling System. Computers and Graphics, 18, 4, pp. 499-506.
- [15] Forsberg, A., Herndon, K., Zeleznik, R. (1996). Aperture based selection for immersive virtual environments. ACM UIST, pp. 95-96.
- [16] Pierce, J., Forsberg, A., Conway, M., Hong, S., Zeleznik, R. (1997). Image plane interaction techniques in 3D immersive environments. ACM

I3D, pp. 39-43.

- [17] Bowman, D., Hodges, L. (1997). An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments, ACM I3D, pp. 35-38.
- [18] Kato, H., Billinghurst, M., Poupyrev, I., Imamoto, K., Tachibana, K. (2000). Virtual Object Manipulation on a Table-Top AR Environment, ISAR, pp. 111-119.
- [19] Kato, H., Tachibana, K., Tanabe, M., Nakajima, T., Fukuda, Y. (2003). MagicCup: A Tangible Interface for Virtual Object Manipulation in Table-Top Augmented Reality, the Augmented Reality Toolkit Workshop, pp. 75-76.
- [20] Grossman, T., Balakrishnan, R. (2005). The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area, CHI, pp. 281-290.
- [21] Haan, G.D., Koutek, M., Post, F.H. (2005). IntenSelect: Using Dynamic Object Rating for Assisting 3D Object Selection, Virtual Environments, pp. 201-209.
- [22] Ha, T., Woo, W. (2010). An Empirical Evaluation of Virtual Hand Techniques for 3D Object Manipulation in a Tangible Augmented Reality Environment, IEEE 3D User Interfaces, pp. 91-98.



하 태 진

2001년 ~ 2005년 동국대학교 정보통신공학부 학사 졸업(공학사). 2005년 ~ 2007년 광주과학기술원 정보통신공학과 석사 졸업(공학석사). 2007년 ~ 현재 광주과학기술원 정보기전공학부 정보통신공학과 박사과정. 관심분야는 HCI, 증강/혼합 현실, 저작도구, 3D 사용자 인터페이스, 사용성 평가 등임



우 윤 택

1985년 ~ 1989년 경북대학교 전자공학과 학사 졸업(공학사). 1989년 ~ 1991년 포항공과대학교 전기전자공학과 석사 졸업(공학석사). 1993년 ~ 1998년 Univ. of Southern California (USC) Electrical Engineering-Systems 박사 졸업 (공학박사). 1991년 ~ 1992년 삼성종합기술원 연구원. 1999년 ~ 2001년 ATR MIC Lab. 초빙 연구원. 2001년 ~ 현재 광주과학기술원 정보기전공학부 정보통신공학과 교수. 2005년 ~ 현재 광주과학기술원 문화콘텐츠기술연구소장. 관심분야는 문화콘텐츠기술, 3D 컴퓨터 비전, 증강/혼합현실, 인간-컴퓨터 상호작용, 감정인식, 맥락 인식 컴퓨팅, 유비쿼터스 컴퓨팅 등임