

論文2000-37SP-5-3

3D 수화 애니메이션을 위한 팔과 손의 통합된 키 프레임 에디터 (An Integrated Keyframe Editor of Arms and Hands for 3D Sign-Language Animation)

金商雲*, 李種雨*, 青木由直**

(Sang-Woon Kim, Jong-Woo Lee, and Yoshinao Aoki)

요 약

서로 다른 언어 사이의 언어 장벽을 극복하기 위한 한 방법으로 CG 애니메이션 기술을 이용한 수화통신 시스템이 연구되고 있다. 이 시스템에서 하나의 수화 애니메이션을 생성하기 위해서는 그 제스처에 해당하는 팔과 손의 관절 각을 결정해야 한다. 그러나 현재까지 시행착오적인 방법으로 관절 각을 계산하였기 때문에 애니메이터의 숙련도에 의지할 수밖에 없었다. 따라서 본 논문에서는 이러한 문제점을 해결하기 위하여, 쉽고 빠르게 3차원 수화 애니메이션을 위한 팔과 손의 키 프레임들을 함께 생성할 수 있는 통합된 키 프레임 에디터를 제작한다. 구현된 키 프레임 에디터에서는 관절 각 계산에 역 운동학을 이용하였으며, 양팔과 20개의 손가락 관절에 동일한 변환 행렬식을 이용 할 수 있도록 표준화하였다. 실험 결과 본 논문에서 제시한 에디터를 이용할 경우 이중언어간의 수화통신에 사용되는 수화 파라미터 사전을 좀더 효율적으로 생성할 수 있음을 확인하였다.

Abstract

As a means of getting over the linguistic barrier between different languages, a sign-language communication system using CG animation techniques has been studied. In order to generate a kind of sign-language animation in the system, the joint angles of the arms and the hands corresponding to the gesture have to be determined. Up to date, however, the values of joint angles have been decided by trial-and-error methods based on the animator's experiences. To overcome the drawback, in this paper, we design an integrated keyframe editor of the arms and the hands for 3D sign-language animation with which we can easily and quickly generate the keyframes of the sign-language animation required to build up. In the implemented keyframe editor, the values of joint angles are calculated by using inverse kinematics, and the same transformation matrix is applied to the joints of two arms and twenty fingers. Experimental results show a possibility that the editor could be used efficiently for making up the sign-language communication dictionaries needed for inter-communication between different languages.

I. 서 론

멀티미디어 통신시대에서 수화통신은 언어의 벽을 넘을 수 있는 한 방법이 될 수 있다. 또한 인터넷은 사회 구성원간의 중요한 통신 수단으로 인터넷 수화통신은 가상공간에서 자신을 상징화시킨 아바타(avatar)들간의 대화 시스템으로 발전시킬 수 있다. 특히 수화에서는 동사나 형용사의 품사를 구분하지 않으며, 구문구조가 간단하기 때문에 자연어 처리와는 달리 다른 나

* 正會員, 明知大學校 컴퓨터學部

(Div. of Computer Science & Engineering, Myongji University)

** 正會員, 北海道大學 工學研究科

(Graduate School of Engineering, Hokkaido University)

※ 이 논문은 1999년 韓國學術振興財團의 학술연구비

(과제번호: 1998-001-E00979)에 의하여 지원되었음.

接受日字:2000年1月7日, 수정완료일:2000年8月4日

라의 수화로 번역이 용이하다는 특징이 있다^[1]. 이러한 점에 주목하여 인터넷상에 수화통신 시스템을 구현하여 그 가능성에 대한 연구가 진행되고 있다^[2]. 문헌 [2]의 수화통신은 지적통신(intelligent communication) 방식으로, 송수신측에 각각 수화 애니메이션을 위한 3차원 모델을 준비하고, 실제의 수화영상을 전송하는 대신에 애니메이션을 생성할 수 있는 파라미터만을 전송하는 방식을 이용하였다. 따라서 효율적인 수화 통신을 위해서는 수화 애니메이션을 생성할 수 있는 정확한 관절 각 파라미터를 손쉽게 도출할 수 있어야 한다. 또한 서로 다른 언어사이의 수화통신을 위해서는 수화 파라미터 변환 사전을 작성하여야 한다.

그러나 지금까지의 연구에서는 관절 각을 바꾸어 가면서 제스처 프레임을 생성하여 보고 적절한 관절 각 각도를 결정하는 정 운동학(forward kinematics) 방식을 이용하였다^[3]. 따라서 하나의 키 프레임을 생성하는데 많은 노력이 필요하였으며, 자연스럽게 정확한 키 프레임을 얻기 위해서는 애니메이터의 숙련도에 의지할 수밖에 없었다. 이러한 단점을 보완하기 위해서 역 운동학(inverse kinematics)을 사용하여 제스처 애니메이션 생성을 위한 필요한 관절 각 파라미터 값을 자동적으로 결정할 수 있는 방법을 제안한 바 있다^[4]. 그러나 이 에디터에서는 팔 관절 각만을 계산하고 손 모양은 기존의 방식대로 정 운동학 방식을 이용하였다. 따라서 본 논문에서는 제스처 뿐만 아니라 손 모양의 생성까지 역 운동학을 확대 적용하여 전체 인터페이스를 통합한 키 프레임 에디터를 제작하고, 이를 수화통신 시스템의 구현에 적용하는 문제에 대하여 검토한다. 이 논문의 에디터에서는 툴바에 모드 변환 버튼을 준비하여 제스처와 손모양, 정 운동학과 역 운동학 방법을 각각 선택할 수 있도록 하였으며, 또한 크기가 작아 작업이 어려웠던 손 모양 제작 윈도우의 크기를 바꿀 수 있는 기능을 추가하여 애니메이터의 의존도를 낮추고 키 프레임의 생성 시간을 줄일 수 있도록 한다. 또한, MetaCreations사의 Poser, 3D Planet사의 3D Create 및 Blaxxun사의 Avatar Studio 등^[5] 기존의 상용 에디터에서도 역 운동학을 이용한 경우가 있으나 지적통신을 위한 관절 각 파라미터를 추출할 수 있는 에디터는 찾을 수 없었다. 따라서 본 논문에서는 역 운동학을 이용하여 수화 애니메이션을 재생할 수 있는 팔과 손의 파라미터를 효율적으로 도출할 수 있는 통합된 에디터를 제안한다.

역 운동학^[6]이란 기구의 끝점(end-effector)의 좌표 값을 입력으로 받아 시작점에서 끝점까지의 각 관절의 파라미터 값을 계산할 수 있는 방법으로서, 이를 이용한 키 프레임 에디터로부터 얻을 수 있는 장점은 다음과 같다. 첫째, 손의 위치를 직접 제어할 수 있기 때문에 손을 특정 위치로 보내기 위한 반복되는 보정 작업이 필요치 않게 된다. 둘째, 각 관절 각 값의 수치를 직접 입력하지 않고 스크롤 디바이스(mouse)를 이용하므로 애니메이터의 숙련도에 대한 의존도를 감소시킬 수 있다. 셋째, 키 프레임 생성 시간을 크게 줄일 수 있다. 본 논문의 구성은 다음과 같다. 제 2장에서는 3D 수화 애니메이션의 생성을 위한 팔과 손의 관절 각 파라미터 구조를 고찰하고, 제 3장에서는 역 운동학을 이용하여 수화 관절 각 파라미터 값을 계산하는 방법에 대하여 기술한다. 제 4장에서는 수화 애니메이션을 생성하는 키 프레임 에디터를 설명한다. 제 5장에서는 제작한 에디터로 수화 애니메이션을 생성하는 실험에 대하여 고찰 후, 제 6장에서 결론을 맺는다.

II. 3D 수화 애니메이션을 위한 관절 각 파라미터

1. 팔 관절 각 파라미터

수화는 손의 모양, 방향, 움직임, 위치로 의미를 전달한다. 이 중에서 손의 위치는 어깨의 각도와 회전, 팔꿈치의 각도에 따라 결정되며, 이에 대한 관절 각 파라미터 구조는 그림 1과 같다.

그림 1(a)에서 $\omega_1, \omega_2, \omega_3$ 는 어깨 관절을 중심으로 (X, Y, Z) 좌표 계를 설정했을 때의 각 축을 중심으로 회전하는 각도이고, ω_4, ω_5 는 팔꿈치 관절의 각도, ω_6, ω_7 은 손목 관절의 각도이다. 3차원 모델 관절의 속성과 움직일 수 있는 제한범위로 7개의 자유도 중 $\omega_1 \sim \omega_4$ 는 팔의 자세와 손의 위치를 결정하는 요소이고, $\omega_5 \sim \omega_7$ 은 손의 방향을 결정하는 요소이다. 따라서 팔의 제스처를 얻기 위해서는 $\omega_1, \omega_2, \omega_3, \omega_4$ 에 해당하는 관절 각을 결정해야 한다.

그림 1(b)는 역 운동학을 이용하여 끝점(손의 위치)의 좌표 값으로부터 필요한 네 개의 파라미터 값을 계산하기 위하여 그림 1(a)의 실제 모델을 단순화시킨 것이다. 여기서 L_1 : 상박의 길이, L_2 : 하박의 길이,

Bx_H, By_H, Bz_H : 손의 절대위치 좌표 값이며, 역 운동학 함수를 이용하여 계산해야 할 관절 각 축은 실선으로 나머지는 점선으로 표시하였다. 이 과정에서 기존 로봇공학에서 문제시되던 역 운동학에 관련된 singular point 문제 등을 해결하기 위해서 ω_2 관절의 각도 파라미터 값인 γ 값을 역 운동학 산출과정에서는 고정된 값을 이용하고 후에 γ 값을 수정하는 방식을 사용하였다. 또한 계산해야 할 관절 각 파라미터를 θ_1 ($\omega_3: -40^\circ \sim +180^\circ$), θ_2 ($\omega_1: -40^\circ \sim +170^\circ$), θ_3 ($\omega_4: -10^\circ \sim +150^\circ$), γ ($\omega_2: -70^\circ \sim +80^\circ$) 로 정의하였다^[7].

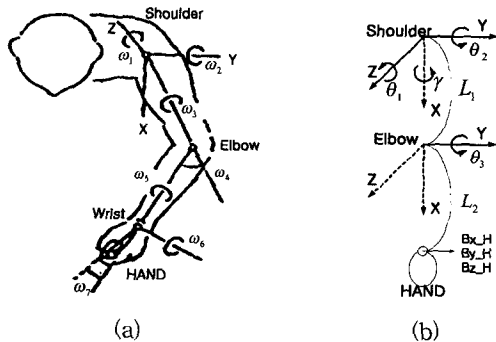


그림 1. (a) 팔 관절 각의 자유도 구조, (b) 관절 각 계산을 위한 좌표 계
Fig. 1. (a) Degree of freedom of the arm, (b) the coordinate system for calculating joint angles.

2. 손 모양 관절 각 파라미터

손 모양을 결정하기 위한 손가락의 자유도에 대한 파라미터 구조는 그림 2와 같다.

그림 2(b)는 역 운동학을 이용하여 끝점의 좌표 값으로부터 필요한 네 개의 파라미터 값을 계산하기 위하여 인지를 표본으로 실제 모델을 단순화시킨 것으로, 여기서 L_1 : 첫 번째 마디의 길이, L_2 : 두 번째 마디의 길이, L_3 : 세 번째 마디의 길이, Bx_I, By_I, Bz_I : 인지의 절대위치 좌표 값이다. 또 θ_1, θ_2 는 각각 MP (Metacarpus Phalanx) 관절을 중심으로 (X, Y, Z) 좌표 계를 설정했을 때의 (Y, Z) 축을 중심으로 회전하는 각이고, θ_3 는 PIP (Proximal Inter-Phalanx) 관절의 각도, β 은 DIP (Distal Inter-Phalanx) 관절의 각도이다. 또한 관절을 표시하는 MP_I, PIP_I, DIP_I 의 첨

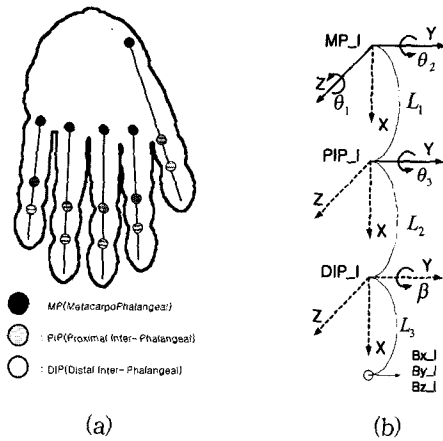


그림 2. (a) 손 모양 관절 각의 자유도 구조, (b) 인지의 관절 각 계산을 위한 좌표 계
Fig. 2. (a) Degree of freedom of hand shape, (b) the coordinate system for calculating joint angles of the index finger.

자 I는 인지 (Index finger)를 나타내는 것으로, 다섯 손가락 각각에 대하여 같은 좌표 계를 적용할 수 있다. 그림 2(b)에서 계산해야 할 파라미터는 θ_1 ($-30^\circ \sim +30^\circ$), θ_2 ($-15^\circ \sim +90^\circ$), θ_3 ($-15^\circ \sim +90^\circ$), β ($-15^\circ \sim +90^\circ$)이다^[7]. 이 때 엄지를 제외한 네 손가락의 DIP는 특별한 경우를 제외하고는 스스로 움직이지 못하고, 상위 관절인 PIP의 움직임에 영향을 받는다. 따라서

$$\beta = \frac{2}{3} \theta_3 \tag{1}$$

로 하여 각 손가락의 PIP관절에서 움직임이 발생하면 그 움직임의 2/3만큼 DIP관절이 움직이도록 할 수 있다^[8]. 또한 PIP 관절에서 손가락 끝까지의 직선거리를 L_2' 라 할 때, 식(1)과 손가락 좌표 계의 L_2, L_3 및 L_2 와 L_3 의 사이각 $(180^\circ - \beta)$ 의 삼각함수를 이용한 기하학적 관계해석으로부터

$$L_2' = \sqrt{L_2^2 + L_3^2 - 2L_2L_3 \cos(180^\circ - \beta)} \tag{2}$$

가 성립한다. 따라서 그림 1(b) 팔 좌표 계의 L_2 에 식(2)의 L_2' 을 대응시키면 그림 1(b)와 그림 2(b)의 관절 각 파라미터 계산에 동일한 역운동학 계산식을 적용할 수 있게 되어 팔과 손에 관한 관절 각 계산식을 표준화할 수 있다.

III. 역 운동학에 의한 관절 각 계산

그림 1(b) 및 그림 2(b) 좌표 계에 적용할 수 있는 변환 연산은 어깨(그림 2(b)에서는 MP에 해당한다)의 Z축과 Y축에 대한 회전변환 $Rot(Z, \theta_1)$, $Rot(Y, \theta_2)$ 와 팔꿈치(그림 2(b)에서는 PIP)의 Y축에 대한 회전변환 $Rot(Y, \theta_3)$ 및 L_1 , L_2 에 대한 이동 연산 $Tran(X, L_1)$, $Tran(X, L_2)$ 로 모두 5개이다. 한편, 역 운동학에서 기구의 끝점(end-effector) 좌표(기준좌표)가 주어졌을 때 이를 상대좌표로 변환시키기 위해서는 각 관절의 변환 연산을 차례로 적용시켜야 한다. 따라서 그림 1에서 손목의 기준좌표가 주어졌을 때 각 관절의 파라미터 값을 구하기 위한 역 운동학 변환 식은 식(3)이 된다.

$${}^5_0 H = {}^1_0 H \cdot {}^2_1 H \cdot {}^3_2 H \cdot {}^4_3 H \cdot {}^5_4 H \quad (3)$$

여기서, 변환 연산 ${}^{i+1}_i H$ 는 각각 식(4) 와 같으며, 이 때 간결한 표현을 위하여 $\sin \theta_n = S_n$, $\cos \theta_n = C_n$, $\cos \theta_n \times \cos \theta_m = C_{nm}$, $\sin \theta_n \times \sin \theta_m = S_{nm}$ 로 치환하여 표기하였다.

$${}^1_0 H = Rot(Z, \theta_1) = \begin{pmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4a)$$

$${}^2_1 H = Rot(Y, \theta_2) = \begin{pmatrix} C_2 & 0 & S_2 & 0 \\ 0 & 1 & 0 & 0 \\ -S_2 & 0 & C_2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4b)$$

$${}^3_2 H = Tran(X, L_1) = \begin{pmatrix} 1 & 0 & 0 & L_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4c)$$

$${}^4_3 H = Rot(Y, \theta_3) = \begin{pmatrix} C_3 & 0 & S_3 & 0 \\ 0 & 1 & 0 & 0 \\ -S_3 & 0 & C_3 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4d)$$

$${}^5_4 H = Tran(X, L_2) = \begin{pmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4e)$$

또한, 그림 1(b)나 그림 2(b)의 좌표 계에 식(4)의 변환 연산 식을 적용하였을 경우 결과되는 좌표 계는 그림 3과 같다.

여기서, 회전변환 연산 $Rot(Z, \theta_1)$ 을 적용하면 기준 좌표 계 (X, Y, Z) 는 (X_1, Y_1, Z_1) 좌표 계로 변하고

다시 $Rot(Y, \theta_2)$ 을 적용하면 (X_2, Y_2, Z_2) 좌표 계가 되고, 계속하여 $Tran(X, L_1)$, $Rot(Y, \theta_3)$, $Tran(X, L_2)$ 을 적용하면 좌표 계는 최종적으로 상대좌표 (X_5, Y_5, Z_5) 로 변환됨을 나타낸다. 즉, 식(3)은 (X_5, Y_5, Z_5) 좌표 계에서 끝점의 위치를 기준 좌표 계 (X, Y, Z) 에서의 위치로 변환하는 식이 된다. 따라서 상대 좌표 계의 끝점 위치 $(0, 0, 0)$ 를 기준 좌표 계의

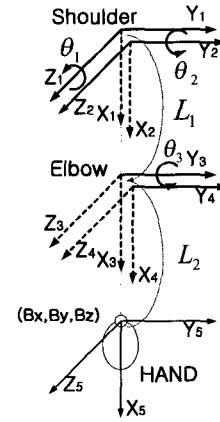


그림 3. 회전 및 이동 연산에 따른 기준 좌표 계의 변환과정

Fig. 3. The translation process of the base coordinate system according to the operations of rotation and translation.

위치 (B_x, B_y, B_z) 로 변환하는 식은

$${}^5_0 H \cdot \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} B_x \\ B_y \\ B_z \\ 1 \end{pmatrix} \quad (5)$$

이 된다. 여기서 ${}^5_0 H$ 는 식(3)에 식(4) 를 대입하여 정리한 것으로

$${}^5_0 H = \begin{pmatrix} H_{11} & H_{12} & H_{13} & H_{14} \\ H_{21} & H_{22} & H_{23} & H_{24} \\ H_{31} & H_{32} & H_{33} & H_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (6)$$

이 된다. 이 때 간결한 표현을 위하여 $H_{11} = C_{123} - C_1 S_{23}$, $H_{12} = -S_1$, $H_{13} = C_{12} S_3 + C_{12} S_2$, $H_{14} = (C_{123} - C_1 S_{23}) L_2 + C_{12} L_1$, $H_{21} = S_1 C_{23} - S_{123}$, $H_{22} = C_1$, $H_{23} = S_{12} C_2 + S_{12} C_3$, $H_{24} = (S_1 C_{23} + S_{123}) L_2 + S_1 C_2 L_1$, $H_{31} = -S_2 C_3 - C_2 S_3$, $H_{32} = 0$, $H_{33} = -S_{23} + C_{23}$, $H_{34} =$

($-S_1C_3 - C_2S_3$) $L_2 - S_2L_1$ 로 치환하여 표기하였다.

식(6)를 식(5)에 대입하여 정리하면 식(7)을 얻을 수 있다.

$$B_x = (C_{123} - C_1S_{23})L_2 + C_{12}L_1 \quad (7a)$$

$$B_y = (S_1C_{23} - S_{123})L_2 + S_1C_2L_1 \quad (7b)$$

$$B_z = (-S_1C_3 - C_2S_3)L_2 - S_2L_1 \quad (7c)$$

또한 그림 1(b) 또는 그림 2(b)에서 L_1 , L_2 및 L_1 와 L_2 의 사이각 ($180^\circ - \theta_3$)의 삼각함수를 이용한 기하학적 관계해석으로부터

$$B_x^2 + B_y^2 + B_z^2 = L_1^2 + L_2^2 - 2L_1L_2\cos(180^\circ - \theta_3) \quad (8)$$

를 얻을 수 있다. 따라서 먼저 식 (8)로부터 관절 각 파라미터 θ_3 는

$$\theta_3 = 180^\circ - \cos^{-1}\left(\frac{B_x^2 + B_y^2 + B_z^2 - L_1^2 - L_2^2}{-2L_1L_2}\right) \quad (9)$$

이 된다. 그리고 식(7a), (7b)를 C_1 과 S_1 에 대하여 정리한 후 식(7b)를 식(7a)로 나누면

$$\frac{S_1}{C_1} = \frac{B_y}{B_x} \quad (10)$$

이 되고, $\tan \theta = \sin \theta / \cos \theta$ 의 관계식을 이용하면 θ_1 은

$$\theta_1 = \tan^{-1}\left(\frac{B_y}{B_x}\right) \quad (11)$$

이 된다. 또한 식(7)을 $C_3L_2 + L_1 = t_1$, $S_3L_2 = t_2$ 로 치환하여 C_2 과 S_2 에 대하여 정리하면

$$C_2 = \frac{t_1(C_1B_x + S_1B_y) + t_2(C_1B_x + S_1B_y - B_z)}{t_1^2 + t_2^2} \quad (12a)$$

$$S_2 = \frac{t_2(C_1B_x + S_1B_y) + t_1(C_1B_x + S_1B_y - B_z)}{t_1^2 + t_2^2} \quad (12b)$$

이 된다. 따라서 관절 각 θ_2 는

$$\theta_2 = \tan^{-1}\left(\frac{S_2}{C_2}\right) \quad (13)$$

이 된다. 식 (9), (11), (13)을 이용하여 손의 위치 (B_x, B_y, B_z)가 주어지면 필요한 관절 각 파라미터

$\theta_1, \theta_2, \theta_3$ 을 계산할 수 있다. 그런데 이는 그림 1(b)의 경우 위의 어깨 X축을 고정시키고(γ 을 고정시키고) 어깨 Z, Y축, 팔꿈치 Y축 회전에 관한 파라미터 값을 계산한 것이다. 그러나 γ 값을 고정하고서는 자연스러운 팔의 자세를 결정할 수 없다. 따라서 손의 위치 (B_x, B_y, B_z)에 따라 자연스러운 팔의 자세(어깨 X축 회전을) 얻을 수 있는 data를 준비하여 두는 방법으로 관절 각 계산 과정을 자동화하였다. 그러나 어깨 X축의 회전을 자동화시키는 데에는 한계가 있다. 따라서 γ 값을 사용자가 원하는 크기로 증감시킬 수 있도록 하였다. 이때 어깨 X축 회전 량의 결정은 마우스의 드래그(오른쪽 클릭, 좌우방향)로 증감시킨다. 끝으로, 손의 방향은 손의 위치나 모양과는 관계가 없으므로 직접 마우스의 드래그를 이용하여 관절의 회전 값을 결정하도록 하였다.

IV. 수화 애니메이션 키 프레임 에디터

3차원 수화 애니메이션을 생성하기 위한 키 프레임 에디터의 윈도우 영상은 그림 4와 같다. 그림 4(a)는 팔 모양 제스처를 만들기 위한 에디터 윈도우이고, (b)는 손 모양을 제작하기 위한 에디터이다.

여기서 좌측은 마우스를 이용하여 팔이나 손의 형상을 제작하는 윈도우이고, 우측은 좌측의 제스처를 생성하기 위해 필요한 관절 각 파라미터의 값을 나타내는 패널이다. 이 논문에서는 오른 손과 왼손을 자세히 관찰할 수 있도록 창 모드를 바꿀 수 있는 기능을 추가하였고, 툴바에 토크스위치를 추가하여 정 운동학 방식과 역 운동학 방식을 선택할 수 있도록 하였다. 즉 정 운동학 방식을 이용할 경우에는 우측의 패널에 관절 각의 값을 입력한 후 좌측의 윈도우에 나타나는 제스처나 손 모양을 관찰하고 원하는 키 프레임이 얻어질 때까지 관절 각의 값을 바꾸어 입력하는 방식으로 원하는 파라미터 값을 구하는 방식이다. 따라서 주어진 애니메이션을 생성할 수 있는 파라미터 값을 구하기 위해서는 관절 각 입력을 반복해야 하며 이 때 반복횟수는 애니메이터의 숙련도에 따라 결정된다. 반면에 역 운동학 방식에서는 좌측의 윈도우에서 마우스를 움직여서 생성하고자하는 팔 및 손 모양의 키 프레임을 작성한 다음 이에 해당하는 각 관절 각 값을 역 운동학 변환 식으로 계산하여 오른쪽 패널에 나타내는 방법으

로, 원하는 애니메이션 키 프레임의 파라미터 값을 손쉽게 계산할 수 있다. 그림 4에서 오른쪽 패널의 파라미터 수는 한쪽 팔과 손에 7개와 20(=5×4)의 파라미터 관절 각이 있기 때문에 패널에 있는 전체 관절 각 개수는 54(=27×2)가 된다.

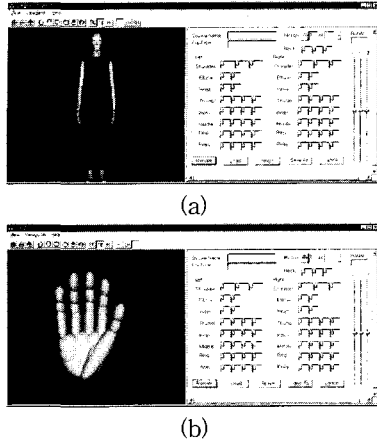


그림 4. 키 프레임 에디터 윈도우. (a) 팔 제스처 에디터 화면, (b) 손 모양 에디터 화면
 Fig. 4. Keyframe editor window. (a) a view of keyframe editor for arm gesture, (b) a view of keyframe editor for hand shape.

V. 실험

1. 키 프레임 에디터를 이용한 수화 애니메이션 생성 실험

제안 시스템의 구축환경은 PentiumII 300MHz에서 Visual C++ 과 Operinventor를 사용하여 제작되었다. 먼저 제안한 키 프레임 에디터를 이용하여 팔의 관절 각을 계산하여 키 프레임을 작성한 다음 이를 이용하여 수화 애니메이션을 생성하는 실험을 하였다. 그림 5는 그림 4(a)의 키 프레임 에디터를 이용하여 제작한 수화 애니메이션의 정면 영상과 측면 영상이고, 표 1은 이 키 프레임들을 생성하기 위한 두 팔의 관절 각이다.

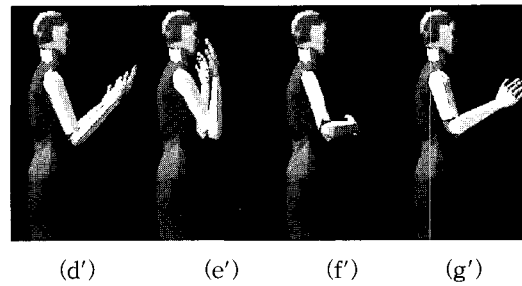
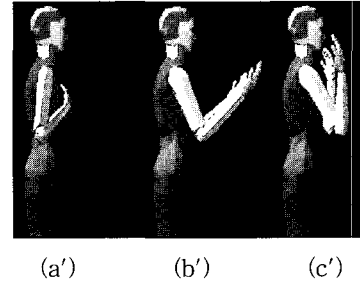
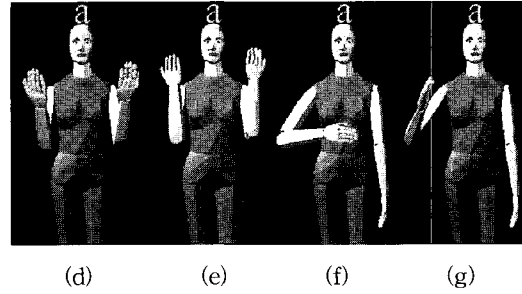
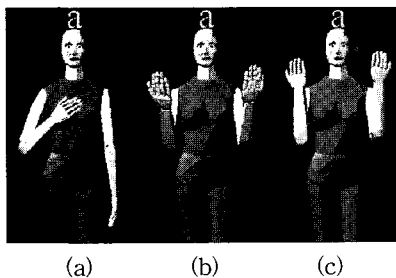


그림 5. “NaNeun HakKyoEa KamNiDa”에 대한 수화 애니메이션 생성 예. (a)~(g) 정면 영상, (a')~(g') 측면 영상

Fig. 5. Example of sign-language animation for “NaNeun HakKyoEa KamNiDa”. (a)~(g) front view, (a')~(g') side view.

표 1. 그림 5. 수화의 키 프레임별 팔 관절 각

Table 1. Arm joint angles of sign-language keyframes for Fig. 5.

수화단어	프레임	왼팔				오른 팔			
		θ_1	γ	θ_2	θ_3	θ_1	γ	θ_2	θ_3
"NaNeun"	(a)	0	0	0	0	-20	60	-10	150
	(b)	0	0	20	120	0	0	20	120
"HakKyoEa"	(c)	0	0	30	160	0	0	30	160
	(d)	0	0	20	120	0	0	20	120
	(e)	0	0	30	160	0	0	30	160
"KamNiDa"	(f)	0	0	0	0	-30	60	10	120
	(g)	0	0	0	0	10	10	-30	120

표 2. 그림 6. 지숫자 모션의 손가락 관절 각 파라미터(단위 : 도)

Table 2. Finger joint angles of digit sign-language animation designed for Fig. 6 (Units: degree).

지숫자	"1"	"2"	"3"	"4"	"5"	"6"	"7"	"8"	
엄지	θ_1	-8	8	8	32	43	43	42	43
	θ_2	-6	-6	-6	18	33	33	33	33
	θ_3	-6	106	104	15	14	14	14	14
	β	180	180	180	150	-18	18	18	18
인지	θ_1	0	0	0	0	0	0	0	0
	θ_2	0	0	0	0	59	22	22	22
	θ_3	0	0	0	0	97	9	9	9
	β	0	0	0	0	65	6	6	6
중지	θ_1	-1	-12	-4	-4	1	1	0	0
	θ_2	53	-15	33	33	53	53	0	0
	θ_3	102	13	21	21	102	102	0	0
	β	68	8	14	14	67	67	0	0
약지	θ_1	8	8	-9	-9	8	8	8	0
	θ_2	59	59	-9	9	59	59	59	0
	θ_3	95	95	9	9	95	95	95	0
	β	63	63	6	6	63	63	0	0
소지	θ_1	15	15	15	12	15	15	15	15
	θ_2	44	44	44	27	44	44	44	44
	θ_3	70	70	70	87	70	70	70	70
	β	47	47	47	8	47	47	47	47

지금까지의 정 운동학 방법에서는 표 1과 같은 키 프레임별 관절 각을 얻기 위해서는 경험적으로 그리고 반복적인 방법으로 관절 각을 직접 변화시켜 가면서 모델의 변화를 관찰하여 결정하는 방법을 이용하였다. 그러나 본 논문의 역 운동학 방법에서는 단지 인터페이스 윈도우 상에서 손의 위치를 원하는 곳에 끌어놓으면 그에 해당하는 관절 각 값을 자동으로 구할 수 있게 하였다. 따라서 스크롤 디바이스의 움직임만을 이용하여 실시간으로 보다 자연스러운 키 프레임을 만들 수 있었으며, 조작이 간단하여 애니메이터의 숙련도와 상관관계를 줄일 수 있었다. 이 방법을 이용하면 3차원 수화 파라미터 사전 DB의 양질의 키 프레임 데이터를 좀더 빠르고 쉽게 구축을 할 수 있게 되었다.

다음은 본 논문에서 제안한 에디터를 이용하여 손의 제스처를 생성하는 실험을 하였다. 그림 6은 손 모양만으로 그 의미를 나타내는 1~8의 지숫자 애니메이션 키 프레임으로, 그림 4(b)의 좌측 윈도우에서 마우스를 이용하여 생성한 모습이며, 표 2는 우측 패널로부터 얻은 각 손 모양 제스처에 해당하는 관절 각 파라미터

값이다.

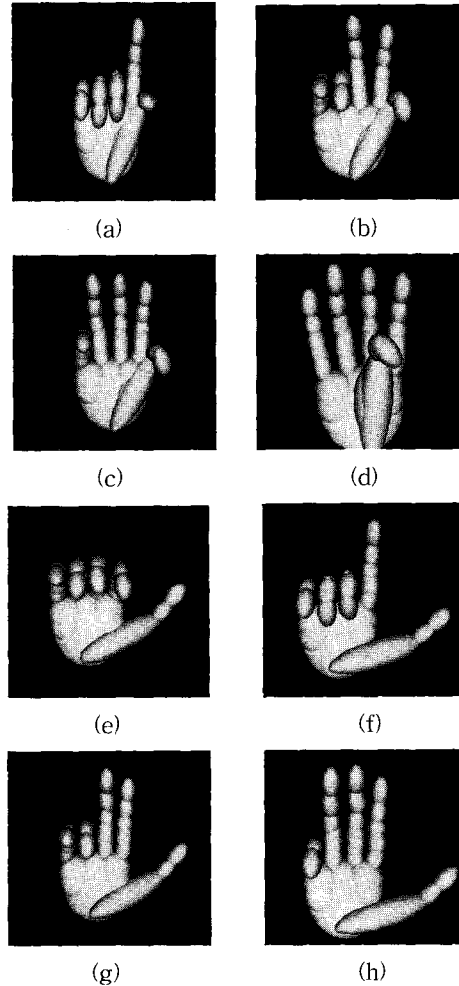


그림 6. 손 모양 에디터로 작성한 지숫자 키 프레임 예. (a) "1", (b) "2", (c) "3", (d) "4", (e) "5", (f) "6", (g) "7", (h) "8".

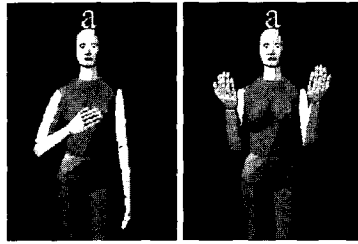
Fig. 6. Examples of digit sign-language keyframes generated with the hand shape editor. (a) "1", (b) "2", (c) "3", (d) "4", (e) "5", (f) "6", (g) "7", (h) "8".

지금까지의 연구에서는 다양한 손 모양을 미리 제작하여 놓은 후 필요할 때마다 적절한 것을 선택하여 사용하였다. 그러나 제안 방법에서는 제스처 키 프레임을 제작하였던 에디터를 손 모양 에디터로 모드 변환한 다음 필요한 손 모양 키 프레임을 손쉽게 생성할 수 있게 되었다.

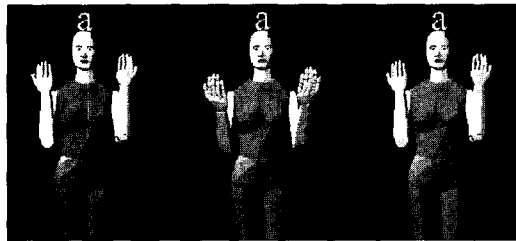
2. 수화 키 프레임 생성시간 비교 실험

기존의 정 운동학 에디터와 이 논문에서 제안한 역

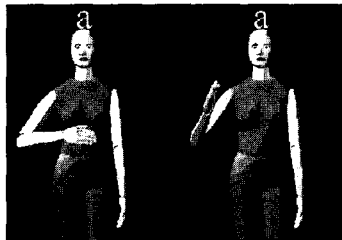
운동학을 이용한 에디터의 사용 편의성에 대한 기초 실험으로 초보자를 대상으로 그림 7의 제스처 키 프레임 생성하도록 한 뒤 키 프레임 제작시간을 비교하였다. 실험과정은 다음과 같다. 먼저 키 프레임 에디터를 처음 접하는 학생 10명을 대상으로 약 10분 정도의 적응시간을 가진 후 그림 7과 같이 5개의 수화 단어를 구성하는 총 10개의 키 프레임을 제작하도록 하였다.



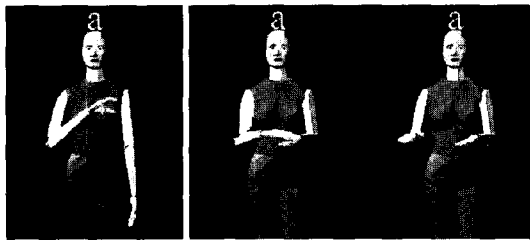
(a) (b)



(b)



(c)



(d) (e)

그림 7. 제작시간 측정을 위한 샘플 키 프레임. (a) “Na(I)”, (b) “HakKyo(school)”, (c) “GaDa(to go)”, (d) “Irum(name)”, (e) “ImNiDa(to be)”
 Fig. 7. Sample keyframes for measuring the generation time. (a) “Na(I)”, (b) “HakKyo(school)”, (c) “GaDa(to go)”, (d) “Irum(name)”, (e) “ImNiDa(to be)”.

실험에 참가한 학생은 컴퓨터학부 1학년 학생으로, 컴퓨터 사용이 능숙한 학생도 있었으나 PC 활용 능력이 부족한 학생도 일부 포함시켰다. 실험 결과 각 학생별 키 프레임 작성 시간은 표 3 및 표 4와 같다.

표 3, 4에서 전체적인 키 프레임 생성 평균시간은 비슷하나 그림 7 (a), (c), (d), (e) 키 프레임은 역 운동학 에디터의 생성시간이 짧은 반면 (b)의 경우는 정운동학 에디터가 짧음을 보이고 있다. 그 이유는 (b) “HakKyo(school)” 키 프레임의 경우 같은 동작이 2회 반복되기 때문에 동일한 관절 각 열을 그대로 다시 적용할 수 있었기 때문이다. 따라서 단순 반복동작이 많은 수화 제스처일 수록 정 운동학 에디터가 효율적이고 제스처가 복잡하거나 키 프레임간의 유사함이 적을 수록 역 운동학 방식이 효율적임을 알 수 있었다.

표 3. 정 운동학 에디터로 그림 7의 키 프레임을 생성한 시간 (단위: 초)

Table 3. Keyframe generation times of Fig. 7 using the forward kinematics editor (unit: sec.).

실험자 키프레임	p1	p2	p3	p4	p5	평균
(a)	61	94	170	270	94	137.80
(b)	142	118	130	141	117	129.60
(c)	90	70	663	134	210	233.40
(d)	300	434	374	256	363	345.40
(e)	75	97	150	228	270	164.00
합	668	813	1,487	1,029	1,054	1,010.2

표 4. 역 운동학 에디터로 그림 7의 키 프레임을 생성한 시간 (단위: 초)

Table 4. Keyframe generation times of Fig. 7 using the inverse kinematics editor (unit: sec.).

실험자 키프레임	p1	p2	p3	p4	p5	평균
(a)	39	150	42	42	94	73.4
(b)	168	810	460	320	198	391.2
(c)	85	92	72	115	136	100.0
(d)	451	251	187	286	345	304.0
(e)	113	123	216	98	128	135.6
합	856	1,426	977	861	901	1,004.2

또한 본 실험에서는 실험 대상자의 수화 인지도, 컴퓨터 조작능력 등이 고려되지 않아 다소 생성시간의 편차가 증가하였다. 즉 실험 도중에 습득되는 툴의 적응 시간을 고려하지 않아 1인당 두 가지 방법의 편집

작업을 실행하는 도중 후반부의 생성 시간이 초반부의 키 프레임 작성 시간 보다 적은 시간이 소요되는 결과를 보였다. 그리고 또 역 운동학을 이용한 방법에서 구속조건(constraints)이 적절히 적용되지 않아 사용편의성과 효율성에 악영향을 주었으며, 실험 대상자의 에디터 적응 시간이 짧아 실험 결과가 다소 부정확하였다.

3. 인터넷상에서 한-일간 수화 영상통신 실험

끝으로 본 논문의 에디터를 이용하여 한-일간 수화 통신을 위한 수화 파라미터 번역 사전을 작성하고 인터넷상에 클라이언트-서버 시스템을 구축하여 기초적인 이중 언어간 수화통신을 실험하였다^{[1][2]}. 이 시스템에서 대화자는 IP주소를 이용하여 서버에 접속한 후 수화 문장을 입력하면 이 문장은 서버로 전송되어 수화 파라미터로 변환(대화자의 선택에 따라 한글 또는 일본어 수화 파라미터로)되어 다시 클라이언트로 전송되고, 클라이언트는 이 수화 파라미터를 이용하여 애니메이션을 생성하여 디스플레이하게 된다. 예를 들어 한글 수화를 사용하는 클라이언트 A에서 일본어 수화를 사용하는 클라이언트 B로 "NaNeun HakKyoEa KamNiDa"에 대한 한글 수화를 전송하면, B에서는 이 수화 영상에 대한 애니메이션 파라미터를 번역사전을 이용하여 일본어 수화 파라미터로 변환한 후 "WatashiHa KakKyoE Ikimasu"의 일본어 수화를 디스플레이 하게 된다. 이러한 통신방식을 이용할 경우 언어가 서로 달라도 수화는 매우 비슷하기 때문에 번역 과정이 자연어와 달리 단순하게 된다. 또한 클라이언트들 간에는 실제의 수화 영상이나 압축영상 대신에 수화 애니메이션 파라미터만을 전송하기 때문에 실시간 통신이 가능하게 된다.

VI. 결 론

본 논문에서는 역 운동학을 이용하여 3차원 수화 애니메이션을 위한 키 프레임을 효율적으로 생성할 수 있는 에디터를 제작하였으며, 팔과 손의 관절 각을 손쉽게 계산하여 한-일간의 지적 수화통신을 위한 수화 번역사전을 만들어 한-일간 수화통신 시스템을 구축하는 실험을 하였다. 제작한 에디터로 실험한 결과 복잡한 제스처의 키 프레임을 생성하는 경우에 정 운동학을 이용한 에디팅 방법보다 효율적으로 생성할 수 있었다. 그러나 각각의 방식에는 장단점이 있었다. 따라서

제스처의 성격에 따라 두 방식을 적절히 선택한다면 제스처 생성을 보다 쉽고 빠르게 생성할 수 있을 것이다. 이 논문에서 구현한 에디터에서는 각 관절이 움직일 수 있는 범위의 한계(구속조건)를 극히 일부만 적용하였다. 따라서 앞으로 각 관절의 운동 범위를 고려한 인체모델의 정적 구속조건(static constraints)과 관절간의 상호 간섭 등을 고려한 동적 구속조건(dynamic constraints)을 적절히 이용하여 인간의 움직임과 유사한 제스처를 생성할 수 있는 에디터를 구현하는 연구가 필요하다. 또한 수화통신에서 얼굴의 감정표현은 중요한 역할을 한다^[9]. 따라서 팔과 손의 에디팅 기능을 확장하여 대화자의 얼굴 표정의 효율적인 전달을 위하여, 감정표현에 필요한 AU의 개수를 최소화면서 과장된 감정표현을 생성하는 'Comic' 적인 요소를 적용한 표정 에디터의 개발이 필요하다.

참 고 문 헌

- [1] 李今燦, 青木由直, 金商雲, "日韓手話通信のためのCGアニメーションシステム構築の基礎的検討," 電子情報通信學會論文誌 D-I, vol. J82-D-I, no. 5, pp. 613 - 621, 1999. 5
- [2] 金商雲, 吳芝英, 青木由直, "인터넷상에 3차원 모델을 이용한 한-일 실시간 수화 통신 시스템의 구축을 위한 기초적인 검토," 대한전자공학회논문지-S, vol. 36, no. 7, pp. 811 - 820, 1999. 7
- [3] 李鍾雨, 吳芝英, 金商雲, 青木由直, "스플라인 보간법을 이용한 3차원 수화 애니메이션의 생성," 대한전자공학회추계종합학술대회 논문집, vol. 21, no. 2, pp. 931-934, 1998. 11
- [4] 李鍾雨, 金商雲, 青木由直, "역운동학을 이용한 3차원 수화 애니메이션의 키 프레임 에디터," 대한전자공학회하계종합학술대회 논문집, vol. 22, no. 1, pp. 655-658, 1999. 6
- [5] S. K. Wilcox, Web Developer.Com Guide to 3D Avatar, John Wiley & Sons, Inc., pp. 99 - 184, 1998.
- [6] K .S. Fu, R. C. Gonzalez, and C .S. G. Lee, Robotics: Control, Sensing, Vision, and Intelligence, McGraw-Hill, pp. 12 - 81, 1987.
- [7] 최혜옥, 송만균, 전병민, "구속조건 기반의 손 모

- 델,” 한국컴퓨터그래픽스학회 논문지, vol. 3, no. 1, pp. 1-7, 1998
- [8] H. Rijkeman and M. Girard, "Computer animation of knowledge-based human grasping," *Computer Graphics*, vol. 25, no. 4, July 1991.
- [9] 高光勳, 金商雲, 棚橋眞, 青木由直, "효율적인 수화통신을 위한 코믹한 감정 표현," 대한전자공학회 하계종합학술대회 논문집, vol. 22, no. 1, pp. 651 - 654, 1999. 6

저 자 소 개



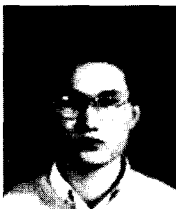
金商雲(正會員)

1978年 2月 韓國航空大學 通信情報工學卒. 1980年 2月 延世大學校 大學院 碩士課程卒 (工學碩士). 1988年 2月 同大學院 博士課程卒 (工學博士). 1989年 - 現在 明知大學校 助教授, 副教授, 教授 (Computer工學專攻). 1992年 - 1993年 北海道大學 情報工學科 客員研究員. 主關心研究分野는 Pattern 認識 및 學習, 假想空間에서의 Avatar通信



青木由直(日本 電子情報通信學會, 情報處理學會 正會員)

1964年 北海道大學 工學部 電子工學科卒. 1966年 同大學 大學院 修士課程卒. 1969年 - 1971年 Canada Laval 大學 留學. 1966年 - 現在 北海道大學 講師, 助教授, 教授 (大學院 工學研究科 情報Media工學 講座). 中國 瀋陽大學 및 黑龍江大學 客員教授. 主關心研究分野는 信號畫像處理, Computer Graphics, 衛星通信. 工學博士. 主著書는 「波動信號處理」(北森出版), 「オペレータ法デジタル信號處理」(コロナ社), 「コンピュータグラフィックス講義」(コロナ社).



李種雨(學生會員)

1994年 3月 - 1998年 2月 남서울大學校 電算學科卒 (工學士). 1998年 3月 - 2000年 8月 明知大學校 産業大學院卒 (工學碩士). 2000年 9月 現在 明知大學校 大學院 博士課程 在學中. 主關心研究分野는 假想空間에서의 Avatar通信