사용자 인터페이스 에이전트 환경을 위한 국어 발음 애니메이션

최승걸, 이미승, 김웅순
시스템공학연구소, 컴퓨터그래픽스연구실

Korean Talking Animation for User Interface Agent Environment

Seungkeol Choe, Miseung Lee, Woongsoon Kim
Computer Graphics Laboratory, System Engineering Research Institute
{skchoe, miseung, woong}@seri.re.kr

요 약

사용자가 컴퓨터와 자연스럽고 인간적으로 대화할 수 있고, 사람의 요구에 지능적인 해답을 능동적으로 제시할 수 있는 사용자 인터페이스 에이전트가 활발히 연구되고 있다. 음성, 펜, 제스쳐인식 등을 비롯한 다양한 방법을 통하여 사람의 의사전달방식을 컴퓨터의 입력수단으로 구현하여 사용자 편의성을 도모하고 있다. 본 논문에서는 컴퓨터를 블랙박스로 하고, 표면적으로 지능형 3차원 그래픽 얼굴 에이전트와 사용자가 의사소통을 하는 사용자 인터페이스를 대상으로 하였다. 컴퓨터가 단순문제 해결을 위한 도구에서 많은 정보를 다양한 매체를 통해 제공하는 보조자의 역할을 수행하게 되었기 때문에 위의 방법은 보다 적극적인 방법이라 할 수 있다. 이를 위한 기반 기술로써 국어를 발음하는 얼굴 애니메이션을 연구하였다. 발음을 표현하기위한 데이터로써 디지털 카메라를 사용하여 입술 운동의 특징점의 위치를 조사하였고, 모델링 시스템을 개발하여 데이터를 입력하였다. 적은 데이터로도 복잡한 자유곡면을 표현할 수 있는 B-Spline곡면을 기본데이터로 사용하였기 때문에 애니메이션을 위한 데이터의 양 또한 줄일 수 있었다. 그리고 국어음소의 발음시간 수열에 대한 입술모양의 변화를 조사하여 발음소리와 입술 움직임을 동기화시킨 발음애니메이션을 구현하였다.

1. 서론

1.1 개요

에이전트는 날로 증가하는 많은 정보와 어플리케이션 사이에서 사용자에게는 보다 편리하고 효율적인 정보의 이용방법을 제시하고, 정보의 유통과 관리가 적절하고 빠르게 수행될 수 있도록 하고자하는 시도에서 발생한 개념이다. 따라서 에이전트는 사용자가 제시한 요구사항으로부터 지능적인 판단으로 해당정보에 접근하여야 하며, 정보의 중복을 피하고 다른 에이전트와의 상호협조를 통하여 정보를 관리할 수 있어야 한다. 이를 지능형 에이전트(intelligent agent)라고도 부른다. 지능형 에이전트는 사용자가 요구하는 정보를 대상으로 하고, 정보를 얻기위한 방법을 자동적

으로 찾아내며, 다른 에이전트와 상호 대화, 협조, 타협 등을 수행하며, 예측할 수 없고 동적인 상황에 대하여 지능적이고 유연하게 대응하는 특징을 가진다. 그리고 학습에 의한 지식 습득과 분산 개방형 환경에서 수행됨을 전제로 하는 소프트웨어적 형태로 정의된다[15].

현재 이러한 에이전트 개념이 응용되는 분야는 정보검색, 사용자 보조역할, 네트워크 운용, 그리고 사용자 인터페이스 등 다양하다. 그 중 인터페이스 에이전트 환경의 필수 요소로는 특정 어플리케이션과 이를 다루는 사용자, 그리고 사용자를 보조하기 위한 지능형 컴퓨터 프로그램이다. 인공지능 기법이 가미된 이 프로그램은 어플리케이션과 사용자와의 중간에 위치하여, 이 프로그램들을 제어하는 다른 에이전트 프로그램 중에 독자적 개체로서, 그리고 사용자의 작업을 보조, 동역하는 개인 비서로서의 기능을 수행한다[7]. 이 인터페이스 에이전트는 문서, 음성, 제스쳐 인식, 음성합성 등의 다양한 (multimodal) 형태로 수행 되는데, 본 논문에서 중점적으로 다루는 내용은 이러한 인터페이스 에이전트의 한 형태로서 3차원 그래픽스를 이용한 가상 얼굴의 모델링과 애니메이션에 대하여 다루려고 한다.

최근 3차원 얼굴 애니메이션 분야는 컴퓨터 게임, 홈쇼핑 등을 비롯한 많은 응용분야와 고유의 연구가치 때문에 특정 어플리케이션들을 포함하는 일반적인 응용분야[9], 또는 물리적, 생물학적 시뮬레이션[6]이나 실세계를 사실감 있게 표현하는 방향[8]으로 연구되어져 왔다. 본 논문의 목적은 인터페이스 에이전트 구현이라는 특정 응용분야 내에서 위의 세가지 연구 방향들의 장점을 취하고, 현실적인 제약점을 조사, 해결하는 데에 있다.

논문의 구성은 다음과 같다. 제 1장의 나머지 부분은 관련 연구로써 일반적인 얼굴 애니메이션의 동향과 그 특징, 에이전트 환경을 고려한 연구의 방향을 논하고, 제 2 장에서는 국어의 발음을 위한 입술 모양의 변화, 그리고 입술 모양 데이터베이스 구축 및 발음시간과의 동기화에 대하여 논하고, 제 3 장에서는 얼굴 곡면 모델의 데이터베이스 구축, 제 4 장에서는 데이터베이스를 통한 국어발음의 애니메이션, 그리고 제 5 장에서는 실험결과에 대한 고찰로써 처리속도를 증가시키고 데이터베이스의 질적 확대와 양적 최적화 방법에 대하여 논한다.

1.2 얼굴애니메이션의 연구 동향

얼굴애니메이션을 위해서는 3차원 가상얼굴의 모델링과 이를 기반으로하는 애니메이션 기법으로 나눌수 있다. 하지만 이 두가지 연구들이 독립적이어서 모델링과 애니메이션의 연구들이 제각각 연구되는 것은 아니고, 모델링 작업을 할 때에 애니메이션 기법을 고려하는 방법이 있고, 어떠한 애니메이션은 특수하게 모델링된 데이터를 필요로 할 때가 있다. 따라서 모델링 기법과 애니메이션 간에는 밀접한 관계가 있다.

얼굴 모델링의 시초는 1974년 Parke[9]의 파라메트릭 방법이라 할 수 있다. 이 모델은 3차원 삼각형 폴리곤의 모임이며 애니메이션은 얼굴의 각 부위의 움직임을 정의하는 파라메터들이 있고, 이러한 파라메터에 의해 조절된 값들이 삼각형 메쉬의 각 정점(Vertices)의 위치값을 변화시킨다. Parke의 얼굴 모델은 그 데이터가 간단하고 다루기 편하여 이후 다른 시스템의 기본 모델이 되고 있다.

Platt[11]는 1981년 얼굴의 실제 표면과 내부구조를 표현한 모델을 제시하였다. 이 모델은 모델링된 근육을 피부표면 밑에 위치시키고, 시뮬레이션과 같은 기법으로 애니메이션을 구현하였다. 따라서 피부표면이 외부의 힘에 의하여 튀어나오고, 들어가는 것, 늘어져있는 모양들이 구현될 수 있었다. 하지만 그 당시의 하드웨어 성능의 부족으로 일반적으로 사용되지 못하였다. 이와같은 근육기반의 애니메이션은 1987년 Nadia Thalman[8]에 의해서도 연구되었는데 얼굴 모양을 추상화한 후, 근육에 대한 애니메이션 파라메터를 순차적(procedual)으로 정의하였다. 또한 기본 데이터에 독립적으로 작용하므로 여러 모델에 대하여 애니메이션이 적용될 수 있다. 근육기반 애니메이션은 1987년 K.Waters[13]에 의해서도 연구되었는데, 모델의 형식은 그대로 3차원 폴리곤이었으며, 애니메이션은 비틀림 등의 변형기법으로 표현하였다.

한편 1987년 Waite는 3차원 얼굴 모델을 B-Spline곡면으로 표현하였다. 적은 데이터로 복잡한 곡면을 표현할 수 있는 이 곡면의 장점 때문에 비교적 적은 데이터로 눈썹, 입술의 경계선 등, 정교한 부분을 표현할 수 있었고, 반면 표면의 위치제어가 까다로운 점 때문에 해부학적으로 정확한 애니메이션은 부족하였다.

1990년 Terzopoulos와 Waters는 물리적 기반의 얼굴 모델을 연구하였다[12]. 이들은 실제 얼굴 표면은 탄성력을 가진 여러겹의 표피로 이루어진 점을 이용하여 스프링 모델로 수직연결된 3겹의 피부층 모델을 제시하였다. 그리고 이 스프링의 탄성계수는 실제 얼굴의 탄성계수로 정하였다. 앞서 언급한 근육기반의 얼굴 모델과 같이 표면 위치의 변화량을 근육의 움직임으로 유도하였다. 더우기 근육의 움직임에 물리적 성질까지 고려하여 얼굴 표면의 움직임을 계산하였다는 측면에서 보다 실제적인 접근이라 할 수 있다. 얼굴 전체에 이러한 물리량의 계산에 의한 애니메이션의 속도가 감소할 가능성 때문에 얼굴 표면의 처리는 폴리곤 메쉬를 이용하였다. 따라서 이 방법은 얼굴의 정확한 모델링 보다는 실시간의 자연스런 애니메이션에 중점을 두었다고 볼 수 있다.

한편 D.Forsey[5]는 기존의 폴리곤 메쉬에 의한 얼굴 모델이 아닌 Hierarchical B-Spline을 이용하여 얼굴 모델링 시스템인 "Face Maker" 를 개발하였다. 얼굴과 같이 변형 자유도가 매우 높은 곡면은 B-Spline곡면의 성질상 조절격자(control vertices)의 수가 매우 많아지므로 곡면의 변형이 많은 곳에는 격자점을 첨가하여 곡면을 구성하고, 변형이 적은 곳에는 적은 격자점으로 표현할 수 있는 곡면을 제안하여, 자료의 갯수도 줄이고 사실감 있는 얼굴 모델을 구현하였다.

국내의 연구로는 이선우, 문보희 등이 개발한 다중제어레벨의 입모양 중심의 표정생성 연구[2]가 있다. 이 연구는 특히 국어의 발음을 4단계의 정확도 레벨로 분류하였으며 음성합성을 이용한 발음 시스템을 개발하였다. 가상공간 내에서의 빠른 실행을 위하여 폴리곤 메쉬로 구현하였기 때문에 정확한 발음과 표정연출에 제한이 있었다.

1.3 연구방향

이 절에서는 인터페이스 에이전트로서의 얼굴애니메이션을 위한 주변환경과 제약점 등을 살펴보고 시스템의 설계를 위한 기본적 방향의 정의를 내린다. 개발중인 인터페이스 에이전트의 환경은 멀티미디어를 기반으로 하는 원격서비스를 주된 기능으로 한다. 이를 위하여 통신과 멀티미디어 운영체제를 탑재하고 에이전트 서버를 운용하는 대용량 고성능 서버를 중심으로 무선 및 동작(Mobile)환경에서의 멀티미디어의 처리를 신속, 효율적으로 처리하는 하드웨어를 기반으로 한다. 이 시스템은 원격 지에 분산되어 있는 메일, 비디오, 음성인식, 데이터베이스, 사용자 인터페이스를 관리하는 각각의 에이전트를 관리하며 주어진 서비스 요청에 대하여 에이전트간의 제어, 분산처리, 문제해결 등의 다중작업과정을 지능적(intelligent)으로 처리하게 된다.

사용자의 측면에서는 소형 단말기를 휴대하고 원하는 서비스를 무선을 통하여 중앙 서버에 연결하면 서버는 지능적방법으로 분산에이전트간 통신을 통하여 문제를 해결 사용자에게 알려주고, 다시 사용자의 인터랙션을 기다리게 된다. 사용자의 소형 단말기에서는 3차원 얼굴의 인터페이스 에이전트가 등장하여 사용자와의 대화, 시스템의 알림 기능 등을 표정, 발음 등을 통하여 수행하게 된다. 따라서 이 시스템은 여러 가지 얼굴모양이나 모양의 특징정보를 데이터베이스화 하여 관리하여야 하고 빠른 검색기능, 데이터 크기의 최적화 등 에이전트의 기능에 수반하는 고유의 환경을 갖게된다.

얼굴 애니메이션 연구의 내용은 데이터베이스에 저장될 데이터를 기본모델의 기하적 데이터와 애니메이션을 위한 데이터로 나누고, 기본모델은 B-Spline곡면 데이터로 하며 애니메이션 데이터는 기본모델로부터 변형된 곡면의 조절격자 데이터들 간의 보간에 의해서 생성하도록 하였다. 일반적으로 얼굴모델의 데이터는 그 얼굴 표현의 일반성과 저장소에 담길 데이터와는 서로 비례하기 때문에 그 비례의 정도가 폴리곤 모델보다는 곡면모델이 효율적이기 때문이다. 그러나 곡면

표 2.1 입술 벌림 실측 데이터 단위 : mm

	이	에	애	위	외	으	어	아	우	오
위 (O,L_u)	0	0	1.7	0	0.5	0	5.7	6.0	0	3.5
아래(0,L_l)	6.0	7.5	11.8	0.9	4.9	3.1	5.2	12.0	2.1	5.2
합계	6.0	7.5	13.5	0.9	5.4	3.1	10.9	18.0	2.1	8.7
가로(M_r,M_l)	54.7	52.0	52.0	15.6	21.5	36.3	45.0	48.5	18.0	27.7

데이터는 화면에 그려지기 전에 세밀한 폴리곤으로 전환되기 때문에 이에 따르는 계산시간이 존재한다. 하지만 하드웨어성능의 발전으로 이 시간은 점점 짧아질 것으로 예상된다.

2. 음소발음 입술의 모델링 및 동기화

2.1 음소의 모델링

국어를 발음하는 얼굴의 모델링을 위해서는 기존의 3차원 데이터에 입술과 그 주위의 좌표값을 변화시켜 해당 음소가 발음하는 모양을 만들어내야 한다. 이때 좌표의 값을 얼마나 정확히 측정하고 데이터에 잘 적용하였는 지의 여부가 정확한 발음 애니메이션을 결정한다. 따라서 음소를 정확히 발음하는 사람으로부터 혹은 여러 사람으로부터 얻은 데이터에 통계적 조작을 통하여 데이터를 얻는 방법이 중요하다. 현재 입술변화에 대한 입술의 상하좌우의 운동과 오므림, 내밈운동에 대한 최대치의 데이터가 존재한다[1].

이 데이터는 정면 시점에서는 카메라를 이용하고 측면시점에서는 X선을 이용하여 생성한 자료이다. 따라서 입술의 모양을 적당한 이심율의 타원으로 표현할 때에는 충분한 데이터이다. 하지만 입술 그 자체도 자유곡면이므로 곡면의 모양을 결정할 수 있는 추가 데이터가 필요하다. 이를 위하여 실제 사람의 발음순간을 디지탈 카메라로 촬영하여 분석하였다.

그림 2.1과 같이 모델의 정면과 측면에서 디지탈 카메라를 설치하고 조명 2대와 모델의 뒷면에 흡광판을 설치한다. 그리고 자음과 모음을 차례로 발음하도록 하여 두 대의 카메라가 동시에 입술모양이 최대가 되는 순간을 찍는다.

발음을 연출하는 모델은 입술의 움직임을 결정하는 입술 주변의 주요지점에 점을 찍는다.(그림 2.2) 그리고 모음을 발음하는 경우에는 그대로 발음하고, 자음을 발음하는 경우에는 자음만을 따로 발음할 수 없으므로 자음의 입모양을 가장 적게 변화시키는 모음과 함께 발음하도록 하였다. 이때 자음의 입모양을 관찰하기 가장 좋은 입모양은 함께 발음되는 모음의 발음시간이 자음의 발음시간보다 현저히 짧을 때 나타난다. 표 2.1에 의하면 윗입술 중앙 하단부를 L_u, 아랫입술 중앙

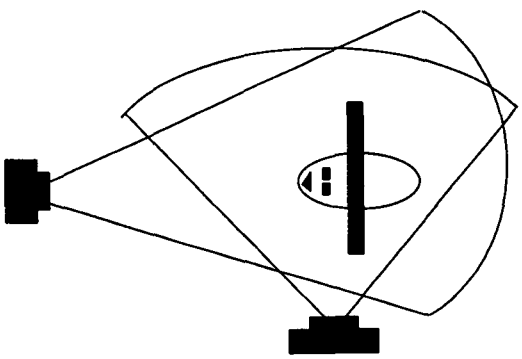


그림 2.1 디지털 카메라를 이용한 음소데이터의 생성

표 2.2 발화위치에 따른 자음의 분류

	양순음	치조음	경구개음	연구개음	후두음
폐쇄음	ㅂ,ㅃ,ㅍ	ㄷ,ㄸ,ㅌ		ㄱ,ㄲ,ㅋ	
마찰음		ㅅ,ㅆ			ㅎ
파찰음			ㅈ,ㅉ,ㅊ		
비음	ㅁ	ㄴ		ㅇ	
유음		ㄹ			
활음	w		y		

상단부를 L_l 입술의 양 끝 지점을 각각 M_r, M_l 이라 했을 때, O부터 L_u까지의 거리 dist(O,L_u)가 가장 작은 모음은 ' ㅣ', ' ㅖ', 'ㅜ', 'ㅡ'였으며 dist(O,L_l)이 가장 작은 모음은 'ㅜ', 'ㅡ'이고, dist(M_l,M_r)값이 가장 평균에 가까운 모음은 ' ㅓ', 'ㅡ'가 된다. 따라서 'ㅡ', ' ㅓ'를 자음과 함께 발음하면 상대적으로 자음의 발음시간이 가장 길기 때문에 정확한 자음 모양 측정에 도움이된다.

또한 21개에 달하는 자음에 대한 데이터를 모두 생성하는 것은 아니고 입안이나 목안에서 동일발원지를 갖는 자음들은 입술모양 또한 타 발원지를 갖는 자음들과는 차이를 보인다는 가정으로 표 2.2에서와 같이 5가지로 분류하였다. 왜냐하면 자음발음은 모음에 비해 상대적으로 입술 변형에 거의 영향을 주지 않기 때문이다.

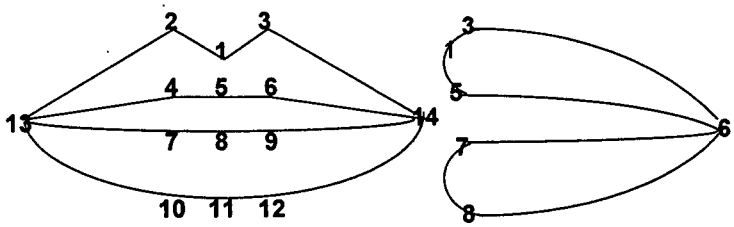
사진 촬영된 데이터는 모두 고정된 카메라와 모델의 얼굴에서 생성되었기 때문에 데이터 이미지 내의 동일 좌표위에 입술 부분이 위치하게 된다. 하지만 촬영도중 흔들림이나 시간의 흐름에 따라 장비와 사람의 위치등이 변하므로 데이터의 보정작업을 통해서 각 데이터의 기준점들을 일치시켜야 한다.이를 위하여 Adobe Photoshop을 이용하여 발음과정에도 변하지 않는 귀부분, 코부분 등의 윤곽선을 추출하여 서로 맞추는 작업을 수행하였다.

2.2 발음 시간에 대한 입술 변형의 동기화

사람의 말소리를 결정하는 음성기관은 말소리의 생성을 담당하는 목안, 소리를 전달해 주는 입안 그리고 소리를 방사(radiation)시키는 입술로 되어있다. 특히 입술은 입안으로부터의 발음기류를 통과시키는 제어나 기류의 속도와 위치의 조절을 담당하며 말소리에 영향을 끼친다. 또한 어떤 음절이 정해졌을 때, 그 음절을 이루고 있는 음소들을 발음하는데 있어서, 한 음소에서 그 다음의 음소발음으로 입술모양을 변화시킨다. 이처럼 소리로 이루어진 음소의 발음 시간 내에 사진으로부터 얻은 그 음소의 고정된 키프레임을 유지하는 것이 아니라 다음 음소의 키프레임으로 변화하며 앞음소를 발음하는 것이 특징이다. 따라서 발음시간에 대한 입술 변형은 주어진 음소의 발음시간동안 해당음소의 데이터를 표현하는 것이 아니라 그 음소의 앞부분과 뒷부분 음소들의 데이터와 조화롭게 변하도록 하는 일이다. 발음하는 동안 음소들은 각각 초성, 중성, 종성 중 한 가지이므로 초성, 중성, 종성, 공백으로 나누어 입술 변형을 조사하였다.

(가) 초성

초성의 발음이 일어나는 시간에는 실제 입 모양을 초성이나 중립의 모델에서부터 중성의 모양으로 변하고 있고, 그 입 속에서 혀의 움직임으로 초성의 소리가 나는 것이다. 따라서 시각 t 에 발음해야 할 음소가 초성이라면 그 시각에 앞쪽 키프레임은 초성에 해당하는 모델 데이터로 하고, 뒤쪽 키프레임은 뒤에 따르는 중성의 모델 데이터가 된다. 그리고 보간 파라메터는 그 초성이 발음되는 시간에 대한 {현재 프레임의 발생 시각(t) - 초성의 발생 시각}의 비율이 된다.



<정면> <측면>

그림 2.2 입술 변화 추적을 위한 특징점들

(나) 중성

한 음절이 발생될 때 그 입술의 모양은 주로 중성의 모음에 의해 좌우된다. 더욱이 모음이 발음될 때에는 입술의 모양이 소리가 나는 동안에도 그대로 유지되고 있다. 그러므로 주어진 시각이 모음을 발음하는 시간의 구간 내에 있으면 이때의 입 모양은 보간이 일어나지 않는 모음 그대로의 모습이 된다.

(다) 종성

종성을 발음할 때의 입술모양은 중성의 입술모양에서 종성의 입술 모양으로 바뀐 뒤 다시 다음 음절의 초성의 입술 모양으로 바뀌게 된다. 그래서 다음 음절이 발음될 때 그 초성의 소리와 입술모양이 일치되도록 한다. 따라서 주어진 시각이 종성이 발음되는 시간의 구간 내에 있으면 다음의 두 가지 경우로 나누어진다. 그 시각이 시간구간의 전반부에 있으면 입술 모양은 중성이 발음될 때의 입술 모양과 종성 그 자체의 입술 모양의 중간 형태이다. 그리고 보간을 위한 파라메터는 그 종성이 발음되는 시간의 이분의 일에 대한 {현재 프레임의 발생 시각(t) - 종성의 발생 시각}의 비율이 된다. 그리고 그 시각이 시간구간의 후반부에 있으면 앞쪽 키프레임은 종성, 그리고 뒤쪽 키프레임은 뒤음절의 초성을 지정한 다음, 보간을 수행하면 된다.

(라) 공백

앞에서 살펴본 바와 같이 프레임의 발생 시각 t가 공백 시간의 구간에 있을 때는 그 앞 음소의 영향을 받는다. 앞 음소가 자음, 모음인 경우 모두 앞 키프레임은 앞 음소의 키프레임 데이터를 사용하고 뒤쪽 키프레임은 공백의 키프레임 데이터를 사용하면서 빠른 속도로 공백 모델로 보간하는 방법을 사용하였다.

각 음소는 자신의 발음 속도가 정해져 있기 때문에 문장이 입력되는 동시에 음소별 배열과 음소의 발음 시간의 수열이 생긴다. 또한 애니메이션이 시작되는 동시에 시각 프레임의 수열이 생긴다. 하지만 규칙적으로 지나가는 시각 프레임의 수열에서 임의의 발음 시간의 수열 원소가 모두 정확히 시각 프레임의 원소와 일치하면 애니메이션에서 얼굴 모델은 모든 원소를 정확히 발음하는 것이다. 하지만 적당한 음소의 요소가 있어서, 그 음소에 정확히 대응하는 시각 프레임의 원소가 존재하지 않고, 그 음소의 앞쪽과 뒤쪽에만 시각 프레임 수열들이 대응한다면 애니메이션이 동작하는 동안 아무리 정확히 관측하여도 그 음소의 정확한 모델은 볼 수가 없다. 최악의 경우, 모든 음소 프레임에서 이런 일이 발생한다면 결국 문장을 정확히 연출할 수 없을 것이다. 따라서 만약 시각 프레임 수열의 원소가 음소 수열의 원소를 지나친 첫 번째 원소라면 이 시각 프레임에는 막 지나친 음소 수열의 원소를 표현해야 한다. 간단한 기호들로 살펴보면 다음과 같다.

F_T : 초당 계산할 프레임의 수

$\{P_n\}$, P_n : 음소(공백 포함),

m : 그 음소에 대응하는 모델의 키프레임 데이터,

t : 각 음소의 발음 시간.

$\{F_m\}$, F_m : 번째 시각 프레임,

I : 그 시각의 앞쪽 키프레임,

F : 그 시각의 뒤쪽 키프레임,

frm : 실제로 실행한 프레임.

임의의 자연수 n에 대해서

$$\sum_{j=0}^{n} P_j::t \le F_m < \sum_{j=0}^{n} P_j::t + \frac{1}{F_T}$$

인 m이 있으면

$$F_m::frm = P_n::m$$

이다. 만약

$$F_{m+1}::I \equiv F_m::I$$

이면 이에 따른 $F_m::frm$과 $F_{m+1}::frm$ 의 격차를 완화하기 위해 $F_{m+1}::frm$을 구하는 파라메터 $param$을 다음과 같이 정한다.

$$param(m+1) = \frac{F_{m+1} - \sum_{j=0}^{n} P_j::t - \frac{1}{F_T}}{P_n::t - \frac{1}{F_T}}.$$

3. 얼굴 곡면 모델의 생성

3.1 기본 얼굴 모델

3차원 얼굴 모델을 컴퓨터상의 가상의 공간상에 가시화하기 위해서는 얼굴 데이터를 표현할 수 있도록 해주는 시스템이 필요하다. 이 시스템의 사용으로 사용자가 원하는 모양의 얼굴을 생성시켜야 한다.

이때 실제 상용화되어 있는 모델링 소프트웨어처럼 완벽한 편집 기능을 가진 시스템을 사용한다면 사용자는 눈, 코, 귀, 입 등의 얼굴 요소뿐만 아니라 전체 얼굴의 윤곽까지도 편집기를 통해서 생성시켜 나가야 한다. 하지만 사람의 얼굴은 얼굴 요소들과 윤곽만으로도 만들 수 있는 서로 다른 얼굴이 엄청나게 많을 수 있다. 이것은 피부의 윤곽이나 주름의 상태 등에 따라서 나타날 수 있는 얼굴이 주는 인상이 매우 다양하다는 것을 의미한다. 그리고 이를 위한 편집기를 제작하는 것은 상용 모델링 소프트웨어의 주 기능인 편집 기능 이상의 기능을 가진 소프트웨어를 제작하는 것과 같다. 왜냐하면, 얼굴 모델을 모델링 하는 작업은 얼굴 모델이 폴리곤들의 집합이라 할지라도 얼굴 표면의 경사와 굴곡을 고려하면 폴리곤 하나 하나의 크기와 방향을 결정하는 일도 많은 시간을 요하기 때문이다. 그리고 곡면 형태의 얼굴 모양을 모델링할 때에도 비록 곡선을 이용하여 눈, 코, 귀, 입의 모양을 결정하는 데에는 폴리곤 형태의 얼굴을 모델링하는 일에 비해서 효율적이지만, 얼굴과 같이 거의 폐곡면과 같은 곡면을 U축 방향과 V축 방향으로 동일 선상에서 시작하여 동일 선상까지 이어지는 곡선을 손으로 편집하기는 몹시 어려운 일이다. 따라서 얼굴 모양을 생성하는 데에는 다른 방법이 필요하다. 개발된 얼굴 모델링 시스템은 기본 얼굴에 대한 데이터를 선택해 놓고 이 얼굴에 사용자가 변형을 가해서 원하는 인상을 가진 얼굴로 바꾸

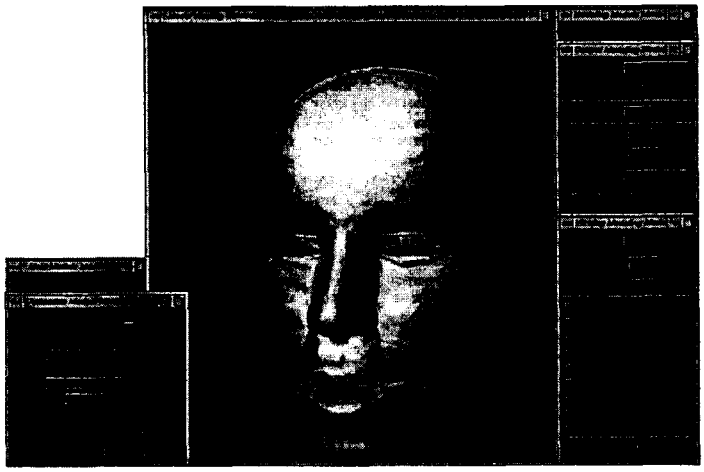


그림 3.1 시스템의 전체 화면

는 방법의 모델링을 택하였다. 일단 기본 얼굴의 데이터는 다른 연구 그룹이나 소프트웨어를 다루는 회사의 데이터베이스에서 취하였다. 주로 이 데이터 방법은 실제로 사람이 Digitizer에 의해 3차원 모델로 생성되거나 석고상과 같은 실제 모델을 모델링 해놓고 이 모델을 디지털 데이터로 입력하여 생성한 것이다.

3.2 키프레임 생성을 위한 모델링 시스템

이 모델링 시스템의 주된 목적은 애니메이션을 위하여 키프레임이 되는 얼굴모델을 생성시켜 그 곡면의 조절점들을 찾아내어 저장하는 일이다. 그러므로 얼굴 곡면을 고려하지 않은 기본얼굴의 조절점들로부터 키프레임 조절점을 구해내는 일과 같다. 이 절에서는 다음의 3가지 모델링 방법을 제시한다. 그 하나는 조절점을 (x, y, z)세방향으로 이동시키면서 얼굴 표면의 변화를 살피는 것이고, 파라메터 도메인의 u, v각각의 iso-parametric 곡선을 구하여 곡선상의 조절점들을 한 번에 변형 시키고 전체얼굴에 대입하는 방법이다. 이 방법들이 수동입력인데 반하여 특징적 조절점들을 자동으로 이동 시키고 나머지 점들은 적당한 위치로 보간하는 방법도 있다.

(가) Iso-Parametric 곡선을 이용한 모델링

주어진 B-Spline곡면

$$S(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_i^p(u) B_j^q(v) P_{i,j}$$

여기서 $\{P_{i,j}\}_{i,j=0}^{n,m}$는 조절 격자점, $\{B_i^p(u)\}_{i=0}^{n}, \{B_j^q(v)\}_{i=0}^{m}$는 B-Spline 기저함수이고, k, l을 각각 u, v 방향의 knot 벡터의 개수라 하면,

$$n = k - p - 1, \quad m = l - q - 1$$

로 주어진다. 이 곡면 $S(u, v)$의 u방향(v방향) iso-parametric 곡선이란 고정된 점 v_0(u_0)에 대한 B-Spline 곡선

$$c(u) = S(u, v_0) = \sum_{i=0}^{n} B_i^p(u) \left\{ \sum_{j=0}^{m} B_j^q(v_0) P_{i,j} \right\},$$

$$\left(c(v)=S(u_0,v)=\sum_{j=0}^{m}B_j^q(v)\left\{\sum_{i=0}^{n}B_i^p(u_0)P_{i,j}\right\}\right)$$

이다. 따라서 $v_0(u_0)$값만 결정된다면 iso-parametric 곡선은 약간의 linear combination에 의하여 정해진다. 본 시스템에서는 $v_0(u_0)$값을 정할 때, 조절점들이 움직일 때 가장 크게 움직이는 표면의 (u,v)에 대하여 iso-parametric 곡선을 구하였다. 이는 기존 모델의 충분한 데이터량과 iso-parametric 곡선을 다루는 사용자가 직관적으로 곡선을 선택하고 다루기 위함이다.

예를 들어, v_0를 구하고 u방향 iso-parametric 곡선을 구할 때, 사용자는 먼저 조절점 $P_{r,s}$를 선택한다. 다음은 파라메터 v_0를 구하는 과정이다. 조절점 $P_{r,s}$가 $P_{r,s}'$의 위치로 이동하면 곡면상에서는 다음과 같은 변화가 생긴다.

$$K(u,v)\equiv B_r^p(u)B_s^q(v)[P_{r,s}'-P_{r,s}]$$

은 두 곡면 $S(u,v)$과

$$S'(u,v)\equiv \sum_{i=0}^{n}\sum_{j=0}^{m}B_i^p(u)B_j^q(v)Q_{i,j},$$

$$Q_{i,j}=P_{i,j},\ \text{만약}\ (i,j)\neq(r,s)\ \text{그리고}\ Q_{i,j}=P_{i,j}',\ \text{만약}\ (i,j)=(r,s),$$

의 차이이다.

가장 많이 움직인 곳의 (u,v)를 계산하기 위하여 방향미분계수(directional derivative)를 0이라 하면

$$(a_1,a_2)\cdot\nabla K(u,v)=a_1K_u(u,v)+a_2K_v(u,v)=0,$$

여기서 (a_1,a_2)는 임의의 실수 단위 벡터이다. 그러므로 $(a_1,a_2)=(1,0),(0,1)$인 경우

$$K_u(u,v)=0=K_v(u,v)$$

라 할 수 있다. 그런데 각각의 B-spline B_r^p과 B_s^q이 의미를 가지려면 (u,v)는 $supp(B_r^p)\times supp(B_s^q)$ 내에 위치해야 한다. 그런데 $supp(B_r^p)\times supp(B_s^q)\subseteq[r,r+p+1)\times[s,s+q+1)$ 이므로 $B_r^p(u)\neq0$ 그리고 $B_s^q(v)\neq0$이다. 그러므로 $B_r^{p'}(u)=0$, $B_s^{q'}(v)=0$이다.

현재 구하려는 것은 v방향의 값이므로 B_s^q의 극치점을 구하기만 하면된다. 일반적으로 B-Spline 기저함수는 차수가 0인 경우를 제외하면 유일한 극대값을 갖는다. 그 극대값을 구하기 위해서 비순환(Nonperiodic) 균일(uniform) B-spline의 대칭성을 이용하여 계산시간을 줄일 수 있다. 이 때 $q\leq s$ 그리고 $s+q+1\leq m-q$ (m은 v방향 knot 벡터) 를 만족해야 한다. 그러면

$$v_0=s+\frac{q+1}{2}$$

이 고정된 값이 된다. 그리고 q와 s가 위의 조건을 만족하지 않으면 다항식의 미분을 이용하여 도함수를 계산하고 수치해석적인 방법을 이용하여 도함수의 근을 구해낸다.

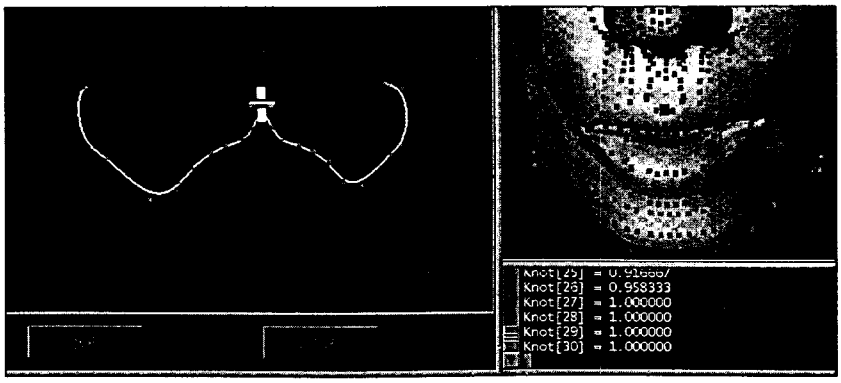


그림 3.2 Iso-parametric 곡선에 의한 모델링

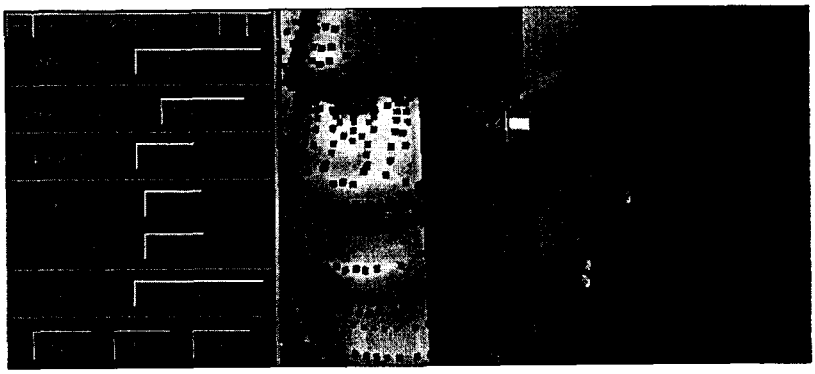


그림 3.3 조절점 이동을 통한 모델링

(나) 조절점 이동에 의한 모델링

얼굴의 모델링은 빰과 같이 단순하고 넓은 부분에 걸친 곡면도 있고, 눈썹이나 눈꺼풀 등 미세한 굴곡을 가진 부분들이 있다. 정밀한 모델링 방법으로 iso-parametric곡선에 의한 방법은 표현에 한계도 있고 사용자 측면에서도 직관성이 부족하다. 따라서 변형을 원하는 미세한 부분은 전체얼굴을 보며 조절점을 이동할 필요가 있다.

얼굴의 곡면을 제어하기 위하여 사용자가 직접 조정점 을 가시화시키고 마우스로 그 조정점을 클릭하면 클릭 한 하나의 조정점 을 (x, y, z)방향으로 평행이동 시킬 수 있는 사용자 인터페이스가 나타난다. 사용자가 이 인터페이스를 원하는 방향으로 마우스를 드래깅 하면서 움직이면 얼굴 표면 또한 움직이게 된다. 이러한 조작은 크게 조정점의 가시화, 마우스 button press Event의 처리 그리고 평행이동을 위한 3차원 사용자 인터페이스의 생성의 단계로 동작하게 된다.

4. 얼굴모델의 국어발음 표현

4.1 입력자료의 처리

사용자가 입력할 수 있는 음소는 약간의 제한을 갖고 있다. 즉 이 시스템은 최종적으로 초성, 중성, 종성과 빈칸을 탐지하여 이 구문에 맞게 발음을 하기 때문에 음소의 모음으로부터 발음을 내지 않는 음소 즉 숫자, 특수문자 등은 입력을 받을 수 없게 되어 있다. 예를 들어 1996을 입력하였을 때 천구백구십육을 발음 할 수도 있고 일구구육으로 발음할 수도 있기 때문에 모호함을 없애기 위하여 입력 받는 음소를 제한한 것이다.

입력된 문장을 음소 array의 형태로 저장한 뒤 각 array 원소들에 대한 조사를 행한다. 먼저 자음, 모음, 빈칸인지를 확인하고 각 경우에 대하여 다음과 같이 처리한다.

(가) 음소가 자음인 경우

2장에서 살펴본 것과 같이 단자음은 5가지 분류에 따라서 입모양을 갖게 되고 쌍자음인 'ㄲ', 'ㅃ', 'ㅃ', 'ㅆ', 'ㅉ'의 실제 음성은 단자음 'ㄱ', 'ㄷ', 'ㅂ', 'ㅅ', 'ㅈ'들과 각각 구별되는데, 이것은 각 음소의 조음부에 강세(stress)가 더해지고 보다 강한 기류가 폐로부터 전달되기 때문에 일어나는

현상이다[1]. 따라서 5개의 쌍자음은 단자음의 입술모양 분류에 포함된다.

입술모양의 변화는 발음하는 문장의 문법적 구조로부터 유도된 음운적 구조에 직접적으로 의존한다. 따라서 복자음과 같이 단일 음소로서의 소리를 갖지 않고, 뒤따르는 음소와 결합하여 변형되거나 분리되는 자음은 앞부분의 자음만을 입력하였다. 그런데 변형되어 발음이 되는 음소는 원래의 조음부와 다른 위치에서 발음되는 경우가 존재하므로, 텍스트와 같은 문장을 발음하려면 중간 단계로써 소리나는 대로 문장을 전환하여 그 문장들을 입력하는 기능이 필요하다.

(나) 음소가 모음인 경우

입력된 음소가 모음인 경우에는 다음과같은 처리가 필요하다. 'ㅒ', 'ㅖ'는 각각 'ㅐ', 'ㅔ'와 구별되어야 하지만 이때에도 모두 'ㅐ'와 'ㅔ'로 처리된다. 왜냐하면 'ㅐ'와 'ㅒ'의 차이와 'ㅔ' 와 'ㅖ'의 차이는 발음상의 반자음 'y'가 개입되었는지에 대한 여부에 달려 있는데 다른 자음소에 비해 반자음을 수치적으로 최고 12.5%에서 최저 33%까지 발음 시간이 짧다. 더우기 입모양으로 그 미세한 발음을 일일이 나타내는 것 보다는 'y'를 발음하는 음성으로 대처하는 것이 효율적이다. 이에따라 [y] 발음이 섞인 'ㅛ', 'ㅕ', 'ㅑ', 'ㅠ' 등도 각각 'ㅗ', 'ㅓ', 'ㅏ', 'ㅜ' 등으로 전환되어 처리되도록 하고 있다.

(다) 음소가 빈칸인 경우

빈칸을 발음할 때의 입술 모양은 매우 중요하다. 일단 빈칸이 오면 말소리는 잠시 그치지만 입술의 모양은 이 말소리에 완전히 영향받지 않고 독자적으로 행동한다. 실험에 따르면 빈칸의 앞에 있는 음소가 종성이었을 때는 종성을 발음하는 입술 모양을 그대로 유지하고 있다. 하지만 앞 음소가 모음일 경우는 어느 정도까지는 모음의 입술 모양을 유지하다가 중립 상태로 돌아와야 한다. 'ㅇ' 자음은 본래 그 자체의 입술의 모양이 그리 크지 않아서 종성의 입 모양에서 다시 초성의 입 모양으로 바뀌는 시간과 거리가 매우 짧다. 하지만 공백에서 모음의 입 모양을 그대로 유지할 경우 입 모양이 큰 모음 발음과 크지않은 초성의 발음 사이에 눈에 띌 정도의 불연속성이 있게 된다. 이를 감쇄하기 위해서 공백의 앞 음소가 이 모음이면 공백의 발음 후반기에 입술이 중립 상태로 되돌아 가도록 한 것이다. 음소가 공백일 경우는 공백을 그대로 처리한다. 하지만 앞으로 보간을 행한 키프레임을 정하는 부분에서 위 문제에 대한 고려가 이루어질 것이다.

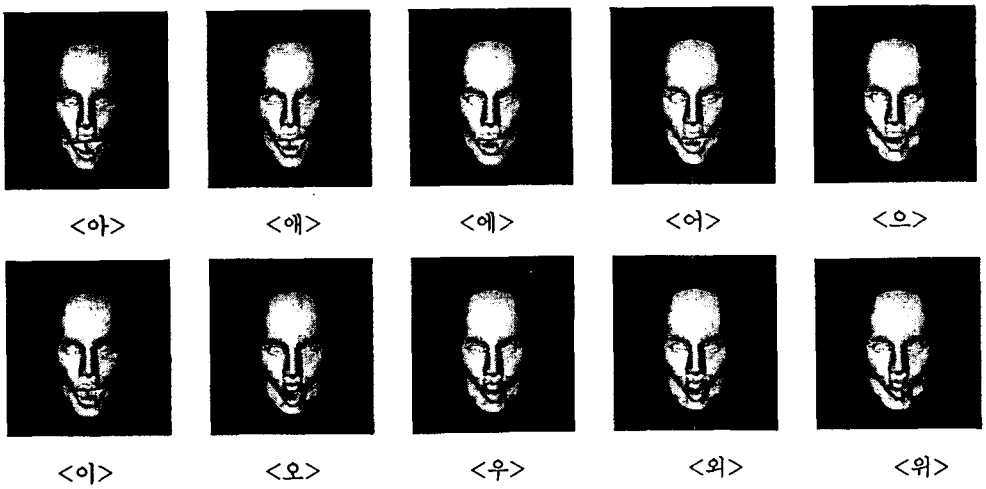


그림 4.1 모음을 발음하는 애니메이션

4.2 보간을 통한 애니메이션

임의의 두 점 P_1과 P_2가 직선상의 서로 다른 곳에 있을 때, 임의의 $t\in[0,1]$에 대하여 $\{\cos(t), t^2, t, t^{\frac{1}{2}}\}$으로부터 유도 되는 집합 $U=\{a\cos(t)+bt^2+ct+dt^{\frac{1}{2}} \mid a,b,c,d\in R\}$의 부분집합 $S=\{at^k \mid k=2,1,1/2,\ a\in R\} \cup \{b\cos(t) \mid b\in R\}$의 임의의 원소 pt^q나 $r\cos(t)$에 대하여 $t_0\in[0,1]$에서의 중간 값 P_0 는 다음과 같이 구한다.

$$P_0 = (1 - pt_0^q) P_1 + pt_0^q P_2 \text{ 또는 } P_0 = (1 - r\cos(t)) P_1 + r\cos(t) P_2$$

이 때, $t_0=\frac{1}{2}$ 이면 k값이 증가 할 수록 P_0의 값은 더욱 빠른 속도로 P_2로 접근한다. 따라서 k값이 작을 수록, t가 증가 할 수록 P_0의 값이 P_2로 변화하는 폭이 작은 것이다. 그리고 p 또는 r이 1 보다 크면, t가 1이 되기 전에 P_0의 값은 P_2가 되어 버려서 P_0가 P_2가 되는 지점보다 t가 큰 값에는 $P_0=P_1$으로 하여야 한다.

그리고 p 또는 r이 1보다 작으면 $t \equiv 1$인 곳에서도 P_0는 가 되지 않을 수 있다. 이러한 현상 모두 애니메이션에서의 특별한 효과를 낼 수 있으며, 발음하는 모델의 동작 상태를 제어할 수 있다. 그림 4.1은 이런 과정을 통해 얻은 애니메이션 중, 모음을 최대로 발음하는 프레임들을 나타낸다.

5. 결론 및 향후 연구계획

5.1 실험결과 및 고찰

제2장에서 살펴본 얼굴 모델링 및 애니메이션에 대한 여러 가지 기법 중에서 구현 방법은 3장에서 제시한 기본 모델의 변형에 의한 모델링과 4장에서 논한 곡면의 조절점 데이터의 보간을 이용한 애니메이션이다. 개발시 다음과 같은 실험을 수행하였다.

(가) 기본 모델 데이터의 형식과 크기

기존의 연구에서는 주로 폴리곤 형태의 기본 모델[6,9,10,13,14]을 사용하였다. 폴리곤 형태의 데이터는 화면에 그려지는 시간이 곡면보다 빠르다는 장점이 있으나, 복잡한 자유곡면의 표현을 위해서는 많은 양의 삼각형 폴리곤이 필요하다. 이는 곧 많은 저장소를 요구하게 된다. 시스템의 그래픽성능과 계산시간의 단축으로 인하여 적은 데이터로 복잡한 곡면을 표현할 수 있는 B-Splin e곡면 데이터가 유용하게 사용될 수 있다.

얼굴 모델링은 기본 얼굴 모델에 변형을 가할 때 목표로 하는 발음 모양이나 표정 또는 인위적 얼굴을 연출할 때 실제 사용자가 생각했던 얼굴을 그대로 생성할 수 있는 데이터가 더욱 요구되었으므로 이러한 실감 있는 곡면 표현에 장점을 갖고 있는 곡면 모델 데이터를 중심으로 모델링 시스템을 개발하여 자연스러운 입술 모양을 생성할 수 있었다.

(나) 디지탈 카메라를 이용한 음소단위 데이터의 생성

애니메이션이 임의의 방법으로 구현될 때 그 방법의 차이에 따라 실제감이나 효율성 면에서 다소 차이는 있게 마련이다. 하지만 가장 중요한 요소는 순간 정지 상태에 나타날 수 있는 모델의 표정을 가장 정확히 표현하는 것이다. 이러한 측면에서 볼 때 얼굴 변화의 각 모습을 데이터

로 가시화시킨 일은 의미가 있다. 충분하진 않지만 기존의 데이터들과도 비교할 수 있고, 더 많은 사람의 데이터를 생성하여 분석한다면 더 정확한 데이터를 얻을 수 있을 것이다. 그러면 개발한 결과에 대하여 신뢰성을 부여할 수 있고 이를 기반으로 하여 더욱 효과적이고 체계적인 개발과정을 설계할 수 있을 것이다.

(다) 애니메이션 파라메터의 생성

애니메이션을 설계할 때 한국어의 각 음소의 기본 발화 시간을 데이터로 이용하였다. 하지만 입술 모양과 각 음소의 발화시간과의 관계는 새롭게 세워야 했다. 그래서 초성이 발화될 때는 초성에서 중성으로 변하는 과정으로의 입술 모양, 중성이 발화될 때는 고정된 모음 입술 모양에서 종성의 입술모양, 그리고 종성이 발화될 때는 중성에서 종성으로 변하는 과정 상에서의 입술 변화를 구현하였다. 실제로 발화하고자 하는 문장의 중간에서 일정시간 발화하는 자음을 표현 할 때의 입술 모양에 대한 연구는 미미하였다. 이와 같이 위의 방법은 실험적인 시도였다고 볼 수 있으며, 앞으로의 연구를 통해서 검증되어야 할 문제이다.

5.2 향후 연구계획

본 연구와 관련하여 다음과 같은 연구가 필요하다. 첫째, 표현할 수 있는 국어의 범위를 확장하는 문제이다. 음소단위의 발음 애니에이션에는 음절 단위의 발음에서 나타나는 음운법칙 등이 생략되어 있다. 더욱이 발음하는 상황에 따라서 변화하는 말소리의 속도 변화도 표준속도에 맞추고 있다. 따라서 최적의 발음범위를 규정하는 연구가 뒤따라야 한다. 둘째, 데이터 베이스 원소의 간략화이다. 데이터 베이스에는 기본 곡면모델의 조절점 데이터와 각 음소에 해당하는 조절점 데이터, 그리고 애니메이션 속도를 위한 음소의 기본 발음 속도들이 있다. 발음할 때의 얼굴표면을 분석하여 얼굴의 각 부위 별 움직임을 수식화 해야 한다. 그래서 애니메이션 데이터를 얼굴 각 부위에 대한 분할과 그 움직임 등으로 구조화 해야한다. 셋째, 기본 데이터의 최적화 및 귀, 머리카락, 눈동자, 이, 혀 등의 모델링이다. 기본 데이터를 얼굴 각 부위에 따라서 여러 개의 곡면으로 분할하고, 애니메이션을 실행할 때 변형되지 않는 곡면의 수가 최대가 되면서 전체 곡면의 개수는 최소화 할 수 있도록 행한다. 또한 머리카락의 자연스런 움직임, 눈동자, 혀의 움직임들도 제어해야 하며, 발음할 때 '이'의 역할이 활발하도록 해야 한다.

참고문헌

[1] 김영송, "우리말 소리의 연구", 과학사, 1975

[2] 문보희, 이선우, 원광연, "다중 제어 레벨을 갖는 입모양 중심의 표정생성", HCI'96 학술대회 발표논문집, pp. 257-270, 1995

[3] Piegl L, Tiller W, "The Nurbs Book", Springer-Verlag, 1995

[4] Farin G, "Curves and Surfaces for Computer Aided Geometric Design A Practical Guide", Third Edition, Academic Press, 1993

[5] Forsey D., Bartel R., "Hierarchical B-Spline Refinement", Computer Graphics, 22(4), pp.205-212, 1988

[6] Lee Y., Terzopoulos D., Waters K., "Realistic Modeling for Facial Animation", Proceedings

of SIGGRAPH 95(LA U.S. August 6-11 1995). In Computer Graphics Proceeding, Annual C onference Series, 1995, ACM, SIGGRAPH, pp.55-62

[7] Maes P., "Social Interface Agents : Acquiring Competence by Learning from Users and Ot her Agents". In Etzioni, O., editor, Software Agents - Papers from the 1994 Spring Sympo sium(Technical Report SS-94-03), pp.71-78. AAAI Press,1994

[8] Magnenat-Thalmann N., Preamu E., Thalmann D., "Abstract Muscle Action Procedures for Human Face Animation", The Visual Computer, Vol.3, No.5, pp.290-297, 1987

[9] Parke, F. I. "A Parametric Model for Human Faces", Tech.Report UTEC-CSc-75-047 Salt Lake City:University of Utah, 1974

[10] Parke, F. I., "Parametrized Models for facial animation", IEEE Computer Graphics, 2(9) p p. 61-68, 1982

[11] Platt S., Badler N., "Animating Facial Expressions", Computer Graphics(SIGGRAPH), pp. 245-252, 1981

[12] Terzopoulos D., Waters K., "Physically-Based Facial Modeling, Analysis, and Animation", Journal of Visualization and Computer Animation, 1(4), pp.73-80, 1990

[13] Waters K., "A Muscle Model for Animating Three-Dimensional Facial Expression", Comp uter Graphics Proceedings, Annual Conference Series, 1987, ACM, SIGGRAPH'87, pp.17-2 4

[14] Waters K., Lebergood T.M., "DECface: An Automatic Lip-Synchronization Algorithm for Synthetic Faces", Canbridge Research Lab. Technical Report Series, CRL 93/4 1993

[15] Wooldridge M., Jennings N. R., "Intelligent Agents: Theory and Practice", Knowledge En gineering Review, October, 1994