한글 음절 분류를 통한 입 모양 궤적 생성

*박유신, 김종수, 김태용, 최종수
중앙대학교 첨단영상대학원 영상공학과
e-mail : {shinazer, hermes, kimty, jschoi}@imagelab.cau.ac.kr

Mouth Shape Trajectory Generation Using Hangul Phoneme Analysis

*You-Shin Park, Jong-Soo Kim, Tae-yong Kim, Jong-Soo Choi
Graduate School of Advanced Image Science, Multimedia & Film
Chung-Ang University

Abstract

In this paper, we propose a new method which generates the trajectory of the mouth shape for the characters by the user inputs. It is based on the character at a basis syllable and can be suitable to the mouth shape generation. In this paper, we understand the principle of the Korean language creation and find the similarity for the form of the mouth shape and select it as a basic syllable. We also consider the articulation of this phoneme for it and create a new mouth shape trajectory and apply at face of an 3D avatar.

I. 서론

인간은 살면서 많은 말을 하고 말을 통해 자신을 표현하고 의사소통을 한다. 이처럼 우리가 살면서 불편함 없이 살수 있는 이유는 말, 즉 언어라는 것이 있기 때문에 가능하다는 것을 알 수 있다. 이 언어란 것을 여러 언어학자들은 각기 다르게 표현하고 있지만 공통적으로 언어는 인간 서로간의 의사를 전달하는데 필요한 도구라는 것이다[1]. 의사전달 수단으로는 말, 문자와 더불어 몸으로 하는 몸짓, 손짓, 얼굴 표정 등이 있으며 이는 인간에게 있어서 중요한 의사전달 수단이 되고 있다. 따라서 이 언어를 이용한 의사 표현과 그 인식에 많은 연구가 되고 있고 의사를 전달 하고자 하는 매체로 아바타나 애니메이션으로 응용하여 표현하는 연구가 진행되고 있다[4]. 그러나 현재 사용되는 아바타나 애니메이션 캐릭터들은 2D상에서 구현되고 있는 것이 보통 이어서 부자연스러움과 그 표현이 비사실적인 면이 많다[2][3][7]. 또한 현대 산업의 발전으로 사람들은 컴퓨터와 다른 대중매체들을 통해 가상의 공간생활에 많이 익숙해져 있으며 보다 더 가상의 공간 세상을 사실적으로 느끼기를 원하고 있다. 이에 본 논문에서는 이 의사를 전달 하고자 하는 매체로서 아바타나 애니메이션을 이용하기 위하여[7][6] 모션캡춰(Motion capteru) 장비와 마커(marker)를 이용하여 기본 음소에 대한 입 모양 위치 값과 입 모양 궤적(Trajectory)값을 취득하여 이를 기반으로 사용자가 원하는 말을 문자로써 표시 했을 경우 이에 해당하는 입 모양 궤적을 생성하여 3D 애니메이션으로 표현된 캐릭터에 접목하는 것으로 실물과 같은 입 모양을 얼굴 근육의 실질적인 움직임 못지않은 자연스러움을 표현함으로써 보다 사실적으로 표현이 가능하게 된다.

II. 시스템 순서도

본 논문의 구성은 크게 두 부분으로 나눌 수 있다. 전 단계는 새로운 입 모양을 생성하기 위한 사전 작업으로 한글의 기본 음소를 취득하고 이를 분류하는 단계이고 후 단계로는 사용자를 통해 입력된 한글을 기본 음소들 간의 조합을 통해 이를 캐릭터에 접목하기 위해 입 모양 궤적을 생성하는 것이다.

그림 1. 에서 두 단계로 나누어 처리하는 전체적인 순서도를 나타내고 있다. 처음 단계로는 모션캡춰 장비를 통해 음소 데이터 취득하여 이 연속된 음소들을 각각의 음소별로 분리하고 각 음소의 키 프레임을 설정하여 키 프레임 데이터를 저장하여 데이터 베이스화 시킨다. 이를 바탕으로 두 번째 단계인 입력 문자에 대한 기본 음소들을 찾는 수행이 가능하게 된다. 이 단계까지는 입 모양의 기본 궤적들을 알고 있기 때문에 사용자가 원하는 문자에 대한 입 모양 생성에 대한 준비가 모두 끝난 것이다. 이 후 각 음소의 입 모양에 대한 동시 조음(articulation)을 고려하기 위하여 우세함수(Dominance Function)을 사용하고 이 우세함수들을 조합하기 위해 혼합함수(Blanding Function)를 사용하여 입 모양에 대한 움직임 제어 파라미터 값을 구한다[8][9]. 이를 통해 최종적으로 입력 문자에 대한 입

모양 궤적 Data를 생성하는 것이다.

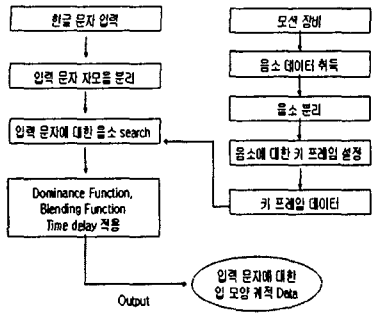

그림 1. 전체적인 순서도

Ⅲ. 한글 음절 데이터 분석

한글은 자음과 모음의 조합으로 되어진 문자이고 '초성 + 중성', '초성 + 중성 + 종성'과 같이 2가지 형태로 문자를 표현하고 있다. 자음은 초성과 종성 부분에 들어가고 모음은 중성 부분에 들어간다. 이렇게 하여 한글은 총 11,172자를 표현 하고 있다. 그러나 본 연구에서는 이와 같이 모든 한글을 사용하는 것이 아니라 입 모양에 대한 한글 유사성 분류에 의한 기본 음소를 선택하여 한글을 표현 가능하였고 그 데이터양을 줄였다.

3.1 한글의 생성 이론

먼저 한글의 생성은 기본 다섯 자의 제자원칙으로 인해 만들어 졌고, 아(牙)음, 설(舌)음, 순(脣)음, 치(齒)음, 후(喉)음으로 어금니, 혀, 입술, 이, 목구멍의 모양을 본떠서 만든 것이다. 모음의 경우는 기본 세 자의 제자 원리, 즉 천(·), 지(ㅡ), 인(ㅣ)의 삼재를 본떠서 만들었으며 기본 삼재를 조합하여 초출자 4개와 재출자 4개를 만들어 사용하였다. 현재 사용하고 있는 한글의 자음과 모음은 다음과 같다.

표.1 한글 자음과 모음

자음	ㄱ, ㄴ, ㄷ, ㄹ, ㅁ, ㅂ, ㅅ, ㅇ, ㅈ, ㅊ, ㅋ, ㅌ, ㅍ, ㅎ
모음	ㅏ, ㅑ, ㅓ, ㅕ, ㅗ, ㅛ, ㅜ, ㅠ, ㅡ, ㅣ
쌍자음	ㄲ, ㄸ, ㅃ, ㅆ, ㅉ
복자음	ㄳ, ㄵ, ㄶ, ㄺ, ㄻ, ㄼ, ㄽ, ㄾ, ㄿ, ㅀ, ㅄ
복모음	ㅐ, ㅒ, ㅔ, ㅖ, ㅘ, ㅙ, ㅚ, ㅝ, ㅞ, ㅟ, ㅢ

한글의 형태는 2가지로 나뉠 수 있는데 초성과 중성만으로 형성되는 경우와 초성과 중성, 종성으로 형성되는 경우가 있다.

초성 : 기본자음+ 쌍자음 = 19
중성 : 기본모음+ 복모음 = 21
종성 : 기본자음+ 쌍자음2 개(ㄲ, ㅆ) + 복자음 = 27

이와 같이 한글 문자로 만들 수 있는 총 가지 수는 '초성 + 중성'이므로 19*21 = 399 이고 '초성 +중성 + 종성'이므로 19*21*27 = 10,773 개, 두 경우의 수를 합해서 11,172개의 문자를 생성할 수 있다. 따라서 이와 같이 많은 한글을 데이터를 입 모양 데이터로 갖기는 그 효율성면에서 떨어진다고 볼 수 있다. 그러므로 한글 발음에 대한 입 모양의 유사성을 파악하여 데이터의 축소가 필요한 것이다.

3.2 한글 유사성 분류

한글은 자음과 모음의 조합으로 구성되어 있고 발음 시 입 모양 형태는 자음 보다는 모음의 영향을 많이 받는다. 그러나 자음의 영향을 받는 것은 양 입술이 닫혀서 발성되는 순음과 치아 사이에서 발성되는 치음은 발음 시 입 모양에 영향을 준다. 따라서 입 모양 표현에 있어 한글 유사 입 모양의 음소는 다음과 같이 40개로 나누었고 이를 취득 하였다.

실제 '가' 와 '아' 또는 '각'에 대해서 그 소리에 대해서는 다르지만 입 모양의 형태에 대해서는 'ㅏ'라는 모음에 영향을 받아 동일하다고 볼 수 있다. 따라서 한글의 유사성을 찾을 수가 있다.

표. 2 기본 음절

아	어	오	우	으	이	에	애
바(마)	버	보	부	브	비	베	배
사	서	소	수	스	시	세	새
암	엄	옴	움	음	임	엠	앰
안	언	온	운	은	인	엔	앤

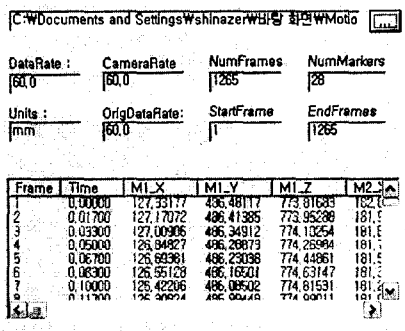

그림 2. 기본 음절에 대한 데이터

이와 같이 각 한글 유사 입 모양으로 음소를 분류한 것은 성대나 혀의 움직임 같은 동작은 소리에 대한 차이는 있어도 발음에 대한 입 모양은 유사하기 때문에 이와 같이 데이터를 분류하였다.

본 연구는 기존 연구[8]와 달리 TTS(text to speech)인 입력 문자에 대한 음성합성이 아닌 립싱크(Lip-Synch) 연구로 음성 부분은 다루지 않고 있다.

3.3 한글 조합형 코드 변환

한글은 영어와 달리 2 바이트에 해당한다. 과거 기준이 없었지만 지금은 유니 코드에 의해 한글이 표현 가능하게 되었다. 입력에 의한 코드는 완성형이다. 따라서 입력된 완성형 코드를 조합형으로 변환하여 각 초성, 중성, 종성에 해당하는 값들을 찾아 사용자가 입력한 문자에 해당하는 음소 데이터를 찾기 위한 코드 변환을 하는 것이다. 한글의 비트 표현은 다음과 같다.

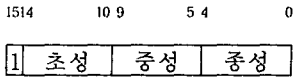

그림 3. 한글 문자에 대한 Bit 표현

한글은 최 상위 비트를 제외하고 5 비트씩 초성, 중성, 종성에 해당하는 값들을 갖고 있다. 최 상위 비트는 한글과 한자를 표시하는 비트이다.

다음은 입력된 완성형 한글을 조합형으로 분리하는 예를 보여 주고 있다.

표.3 코드값에 대한 예 (정의된 테이블)

번호	0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7
초성	ㄱㄲ ㄴㄷㄸㄹㅁㅂㅃㅅㅆㅇㅈㅉㅊㅋㅌㅍㅎ
중성	ㅏㅐㅑㅒㅓㅔㅕㅖㅗㅘㅙㅚㅛㅜㅝㅞㅟㅠㅡㅢㅣ
종성	ㄱㄲㄳㄴㄵㄶㄷㄹㄺㄻㄼㄽㄾㄿㅀㅁㅂㅄㅅㅆㅇㅈㅊㅋㅌㅍㅎ

"AC00h + (a*21+b)*28 +c" ---------- (1)

```
c := code - $AC00;
a := c div(21*28);
c := c mod(21*28);
b := c div 28;
c := c mod 28;
```

글자	코드값(헥사)	계산식
가	AC00h	AC00h+(0*21+0)*28+0
각	AC01h	AC00h+(0*21+0)*28+1
개	AC1Ch	AC00h+(0*21+1)*28+0
까	AE4Ch	AC00h+(1*21+0)*28+0
럼	B7FCh	AC00h+(5*21+4)*28+16

문자 코드에서 "가"를 처음으로 놓기 위한 값으로 정의 하고 만약 $AC00에 "가"라는 입력 문자가 들어오면 식 (1)의a, b, c 값이 0이 되어 테이블에 명시된 "가"에 대한 각각의 "ㄱ"과 "ㅏ"의 16진수 조합형 코드를 알 수가 있다.

그림 4.는 사용자가 안녕하세요라는 문자를 입력 하였을 때 초, 중, 종성으로 분리한 실험 결과이다. 이렇게 사용자의 문자 코드를 추출하면 사용자가 어떠한 문자를 입력하였는지를 알 수 있다. 따라서 사용자가 원하는 입 모양을 기본 음절을 찾아 지배함수에 적용한다.

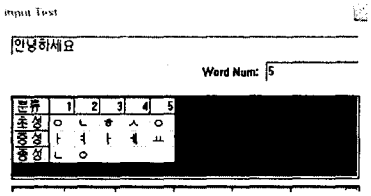

그림 4. 입력된 한글의 초성, 중성, 종성 분리

3.4 음절의 지배함수 정의

입 모양 제어 파리미터들은 Löfqvist의 스피치 생성 제스처 이론을 이용하여 입술 움직임 함수들로 정의하였다[8]. 따라서 다음 식 (2)은 제스처 이론을 기반한 시간거리 τ 에 따른 exponential의 일반적 형태이다.

$$D = e^{-\theta\tau} \quad \text{---------- (2)}$$

식 (2)의 형태를 립싱크(lip-synch.)에 사용할 수 있는 우세함수로 확장시킨 것이 다음의 식 (3)이다.

$$D_{sp} = \begin{cases} \alpha_{sp} e^{-\theta_{<-sp}|\tau|^{c}}, & \text{if } \tau \geq 0 \\ \alpha_{sp} e^{-\theta_{->sp}|\tau|^{c}}, & \text{if } \tau < 0 \end{cases} \quad \text{---- (3)}$$

$$\tau = t_{csp} + t_{osp} - t \quad \text{---- (4)}$$

$$t_{csp} = t_{starts} + \frac{duration_s}{2} \quad \text{---- (5)}$$

여기서 D_{sp}는 세그먼트 s의 p 제어 파라미터의 우세함수이고, α_{sp}는 세그먼트 s의 p 제어 파라미터의 크기이다. 또 t_{csp}는 세그먼트 s의 중심점 시간이고, t_{osp}는 t_{csp}로부터 함수가 최대가 될 때까지의 시간이다. t_{starts}는 세그먼트 시작 시간이며, duration은 세그먼트 지속 시간이다.

$$B_p(t) = \frac{\sum_{s=1}^{N} (D_{sp}(t) * T_{sp})}{\sum_{s=1}^{N} D_{sp}(t)} \quad \text{---------- (6)}$$

$B_p(t)$는 혼합 함수이고 T_{sp}는 세그먼트 s의 target point이다.

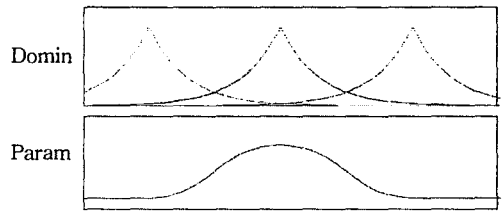

그림 5 Dominance function 와 Blending function

Ⅳ. 실험 결과

본 실험에서는 VC++ 6.0을 사용하였고 기본 음절 데이터를 취득하기 위해 25개의 마커를 사용하였다. 각 음절마다 키 프레임(Key frame)을 선정하여 선형보간(linear interpolation)을 통해 우세함수에 적용하였다. 또한 프레임 율은 초당 20프레임으로 하였다.
다음 실험 결과는 "안녕하세요"와 "반갑습니다"에 대한 문장을 실험을 통해 나타낸 것이다.

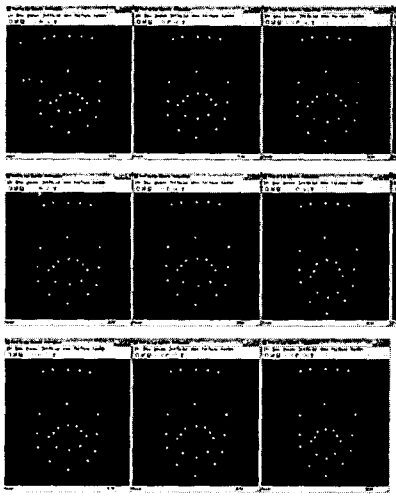

그림 6. "안녕하세요" 발음시 입 모양

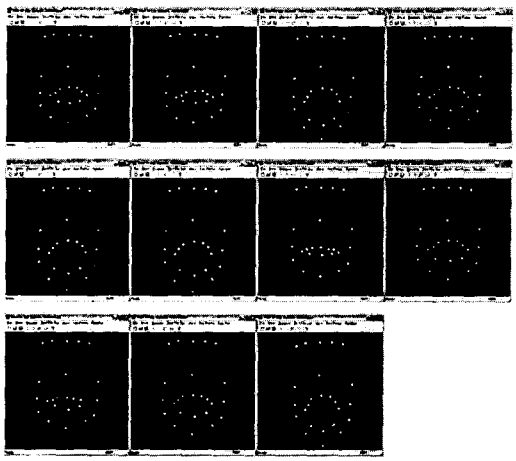

그림 7. "반갑습니다" 발음시 입 모양

Ⅴ. 결론 및 향후 연구 방향

지금까지 사용자가 원하는 말을 문자로써 표시 했을 경우 이에 해당하는 문자를 3D 애니메이션으로 표현해 실물과 같은 입 모양과 얼굴의 움직임으로 이를 표현하기 위한 방안을 제시하였다. 지금까지 진행되고 있는 얼굴 애니메이션은 사실적이지 못하고 또한 문자를 직접 입력하지 않고 얼굴에 마커를 붙여서 사람이 직접 말을 하거나 얼굴내의 특징 점을 찾아 이를 표현하려 했다. 이는 문장의 내용이 변할 경우 항상 사람이 마커를 붙이고 말을 해야 하거나 특징 점을 다시 찾는 수고를 해야 한다. 따라서 본 논문에서 자음, 모음의 데이터 조합만으로 다른 문장을 표현 할 수 있는 방안을 제안하였다.

본 연구는 기존 연구[5]와 달리 TTS(text to speech)인 입력 문자에 대한 음성합성이 아닌 립싱크(Lip-Synch) 연구로 음성 부분은 다루지 않고 있다.

현재는 이 취득한 음절 데이터를 지배 함수에 적용시키는 과정에 있고[8] 또한 이들 지배 함수들을 연결하는 혼합 함수를 어떻게 적절히 사용하는 것이 남아 있는 과제이다. 또한 향후 과제로는 TTS (text to speech)에 맞게 문자에 대한 음성을 생성하는 것이 앞으로 남아 있는 과제이다.

Acknowledgment

본 연구는 교육부의 두뇌한국21 사업(BK21) 및 과학기술부의 국가지정 연구실(M10204000079 - 02J0000 - 07310) 지원으로 수행 되었습니다

참고문헌

[1] http://www.linguistics.or.kr , " 한국 언어 학회".

[2] R. Chellappa, C. H. Wilson, and S. Sirohey, "Human and Machine Recognition of Faces : A Survey," *Proc. of the IEEE*, Vol. 83, No. 5, pp. 705-740, May 1995

[3] 한영환, 홍승홍, "연속 영상에서의 얼굴표정 및 제스처 인식," *의공학회지*, 제20권, 제4호, pp.419-425, 1999

[4] Byoungwon Choe, Hanook Lee, and Hyeong-Seok Ko. Performance-drivenmuscle-based facial animation. *The Journal of Visualization and Computer Animation*, Volume 12, Issue 2, pp. 67-79, May 2001.

[5] Waters, K. and Levergood, T. DECface: An automatic Lip-Synchronization Algorithm for Synthetic Faces, CRL Technical Report 93/4, September 1994

[6] Jun-yong Noh, Ulrich Neumann " Expression Cloning" SIGGRAPH 2001

[7] 이인서, 박운기, 전병우, "MPEG-4 FAP기반 얼굴 근육모델을 이용한 Facial Animation", 춘천멀티미디어 학술회의, pp.147-151, 2000년 2월

[8] Cohen, M.M. & Massoro, D.M. , " Modeling Coarticulation in Synthetic Visual Speech." In Thalman N.M. & Thalmann D. (Eds) Models and Techniques in Computer Animation , Tokyo : Springer-Verlag.