

---

# 벡터 기반 캐리커처에 모션 데이터를 적용한 얼굴 표정 애니메이션

## Facial Expression Animation which Applies a Motion Data in the Vector based Caricature

---

김성호  
상지대학교 컴퓨터정보공학부

Sung-Ho Kim(kimsh1204@sangji.ac.kr)

---

### 요약

본 논문은 얼굴 모션 데이터를 벡터 기반 캐리커처의 얼굴에 적용하여 캐리커처 얼굴 표정 애니메이션을 생성할 수 있도록 하는 방법론을 기술한다. 본 방법은 일러스트레이터의 플러그인 형식으로 구현하였으며, 별도의 사용자 인터페이스를 갖추고 있다. 실험에 사용된 얼굴 모션 데이터는 28개의 소형 마커를 배우 얼굴의 주 근육 부분에 부착하고 다양한 다수 개의 표정을 Facial Tracker로 캡처한 것이다. 캐리커처는 모션 데이터와의 연결을 위해 모션 캡처를 할 때 배우의 얼굴에 부착된 주요 마커의 위치와 동일한 부위에 각각의 제어점을 가진 베지어 곡선 형태로 제작되었다. 그러나 얼굴 모션 데이터는 캐리커처에 비하여 공간적인 규모가 너무 크기 때문에 모션 캘리브레이션 과정을 거쳤으며, 사용자로 하여금 수시로 조절이 가능하게 하였다. 또한 캐리커처와 마커들을 연결시키기 위해서는 사용자가 얼굴 부위의 각 명칭을 메뉴에서 선택한 다음, 캐리커처의 해당 부위를 클릭함으로써 가능하게 하였다. 결국 본 논문은 일러스트레이터의 사용자 인터페이스를 통하여 벡터 기반 캐리커처에 얼굴 모션 데이터를 적용한 캐리커처 얼굴 표정 애니메이션 생성이 가능하도록 하였다.

■ 중심어 : | 얼굴 모션 캡처 | 캐리커처 | 얼굴 표정 애니메이션 | 일러스트레이터 | 플러그인 |

### Abstract

This paper describes methodology which enables user in order to generate facial expression animation of caricature which applies a facial motion data in the vector based caricature. This method which sees was embodied with the plug-in of illustrator. And It is equipping the user interface of separate way. The data which is used in experiment attaches 28 small-sized markers in important muscular part of the actor face and captured the multiple many expression which is various with Facial Tracker. The caricature was produced in the bezier curve form which has a respectively control point from location of the important marker which attaches in the face of the actor when motion capturing to connection with motion data and the region which is identical. The facial motion data compares in the caricature and the spatial scale went through a motion calibration process too because of size. And with the user letting the control did possibly at any time. In order connecting the caricature and the markers also, we did possibly with the click the corresponding region of the caricature, after the user selects each name of the face region from the menu. Finally, this paper used a user interface of illustrator and in order for the caricature facial expression animation generation which applies a facial motion data in the vector based caricature to be possible.

■ keyword : | Facial Motion Capture | Caricature | Facial Expression Animation | Illustrator | Plug-In |

---

\* 이 논문은 2009년도 상지대학교 교내 연구비 지원에 의한 것임.

접수번호 : #100225-002

접수일자 : 2010년 02월 25일

심사완료일 : 2010년 05월 04일

교신처자 : 김성호, e-mail : kimsh1204@sangji.ac.kr

## 1. 서론

인간은 원래 매우 다양하면서 서로 다른 개성을 가지고 있는데, 특히 인간의 얼굴은 이러한 개인의 특징을 가장 잘 표현하는 대표적인 외형적인 요소라고 말할 수 있다. 또한 인간의 얼굴은 인간이 사용하고 있는 다양한 언어보다도 수많은 인간 내면의 감정을 얼굴 표정으로 표현할 수 있는 유일한 수단이며, 의사소통의 수단으로 널리 사용되고 있다. 이와 같이 우리는 인간이 가지고 있는 서로 다른 얼굴의 특징들을 추출하여 개성 있게 그림으로 표현하는 것을 캐리커처라고 한다. 캐리커처를 생성하는 기존의 방법은 비트맵 방식의 방법이 전통적으로 많이 사용되고 있으며, 최근에는 Adobe사의 일러스트레이터(Illustrator)를 이용하여 캐리커처를 제작하는 벡터 방식의 방법이 산업계에서 이루어지고 있다. 본 방법은 디자이너들이 직접 사용자의 사진으로부터 특징을 추출하여 작성하는 방법으로, 먼저 투명 종이를 사용자의 사진 위에 올려놓고 얼굴의 외곽선은 물론이고 눈, 코, 입, 귀 등의 특징을 직접 그린 다음 일러스트레이터를 사용하여 특징을 부각시킬 수 있는 부분을 추가적인 수작업으로 제작하는 것이다. 일러스트레이터는 벡터 방식을 이용한 컴퓨터 그래픽스 도구로써 이미지는 대부분 베지어 곡선(Bezier Curve)으로 구성되어져 있고, 각 부분에는 Knot이라는 제어점(Control Point)이 있어 디자이너는 이 제어점을 사용하여 이미지를 변형하게 된다.

그러나 비트맵 방식이든 벡터 방식이든 디자이너에 의해서 제작된 하나의 캐리커처에 대하여 다양한 표정을 가진 캐리커처 애니메이션을 생성하기 위해서는 각 표정들에 대한 별도의 이미지들을 필요한 양만큼 직접 편집 및 제작하고 이를 모두 비트맵 방식의 이미지로 변환한 다음 순서대로 연결시켜 보여주는 매우 비효율적인 방법을 사용하고 있다. 또한 이와 같은 기존의 방법으로는 캐리커처의 다양한 감정 표현의 변화를 자연스럽게 표현할 수 없다는 매우 큰 단점을 가지고 있는 것이 사실이다. 최근 이와 같은 비트맵 방식의 캐리커처 생성과 관련된 연구결과들[01,02,03,10,11,14,15]이 발표되었으며, 또한 비디오 영상의 프레임들로부터 캐리

커처를 자동 생성[12]하거나 조합에 의한 애니메이션 생성 기법[13] 등이 발표되었다. 그리고 단순히 얼굴 표정 애니메이션과 관련된 연구들[04,05,06,08]과 얼굴 표정에 대한 가이드[07] 등이 발표되었으며, 모션 캡처 데이터를 재사용[09]하기 위한 연구 결과들이 속속들이 발표되었다. 그러나 얼굴 모션 캡처 데이터를 벡터 기반 캐리커처에 직접 적용하여 다양하고 자연스러운 캐리커처의 얼굴 표정 애니메이션을 제작할 수 있는 기법에 관한 연구는 거의 없었다. 그러므로 본 논문에서는 캐리커처의 얼굴에 맞는 다양한 얼굴 표정들을 통하여 캐리커처에게 가장 적합한 감정 표현을 자연스럽게 인지할 수 있도록 하는 방법의 필요성을 부각시키고, 벡터 기반 캐리커처를 기반으로 얼굴 표정 애니메이션 생성이 가능한 기법을 기술하고자 한다. 이를 위해서 본 논문에서는 Motion Analysis사의 모션 캡처 장비인 Facial Tracker를 사용하여 한 명의 배우 얼굴에 28개의 소형 마커를 부착하고, 다수 개의 얼굴 표정을 캡처한 얼굴 모션 데이터를 실험에 사용한다.

본 논문의 2장에서는 얼굴 모션 데이터를 추출하고 실험에 사용하기 위한 사전 작업에 대하여 기술하고, 3장에서는 시스템 설계 및 구현에 대하여 자세히 언급할 것이며, 마지막 4장은 본 논문에 대한 결과를 기술한다.

## II. 사전 작업

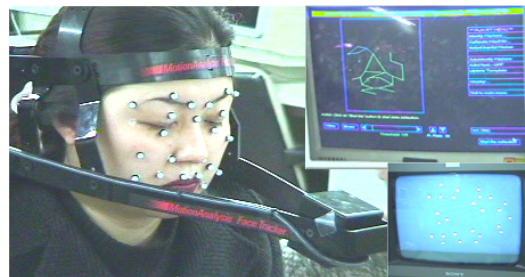


그림 1. Facial Tracker를 머리에 쓰고 소형 마커를 부착한 배우(좌), 캡처된 마커들의 연결 상황(우상) 및 마커들의 캡처 상태 모니터링(우하)

본 논문의 실험을 위하여 사용한 얼굴 모션 데이터는

[그림 1]과 같이 배우의 얼굴에 수 개 이상의 소형 마커를 주요 얼굴 근육 부위에 부착하고 한 대의 적외선 카메라로 구성된 Facial Tracker를 사용하여 캡처한 것이다.

또한 배우의 머리에 Facial Tracker를 사용하여 캡처하는 과정을 모니터를 통하여 실시간으로 확인할 수 있으며, 환경 설정에 따라 특정 마커들 간의 상호 연결을 설정하여 눈썹, 눈, 코, 입 및 볼 등을 구별하여 확인할 수 있도록 하고 있다. 배우 얼굴에 부착된 소형 마커들은 MPEG-4에서 정한 위치를 참조하여 [그림 2]와 같이 얼굴 근육의 주요 부위에 부착하였으며, 총 28개의 마커를 사용하고 있다. [그림 1]과 같이 배우의 얼굴에 부착한 마커는 반사 마커라고도 불리며, 빛을 반사하는 특수 테이프 등으로 제작되어져 있다.

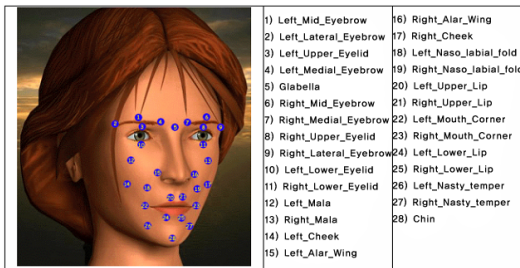


그림 2. 마커 부착 위치와 해당 마커의 명칭

[표 1]은 Facial Tracker로 캡처한 모션 데이터(\*.trc)로서 초당 30프레임으로 되어있으며, 매 프레임마다 배우의 얼굴에 부착한 마커의 X, Y, Z축 좌표 값인 벡터 값을 가지고 있다.

표 1. 얼굴 모션 데이터의 구조

PathFile	Type	6	(X,Y,0)	M2.TRC				
DataRate	CameraRate	NumFrames	NumMarkers	Units				
30.0	30.0	80	28	normalized				
Frame#	Time	1	.....	28				
		X1	Y1	Z1	....	X28	Y28	Z28
1	0.000	0.33052	0.60949	0.0	...	0.88592	0.41832	0.0
2	0.033	0.33065	0.60966	0.0	...	0.88600	0.41847	0.0
3	0.067	0.33053	0.61131	0.0	...	0.88592	0.41933	0.0
.....	.....	.....	.....	...	...	.....	.....	...
80	2.634	0.33359	0.60998	0.0	...	0.88527	0.42094	0.0

그러나 Facial Tracker는 한 대의 적외선 카메라가 얼굴에서 일정한 거리만큼 떨어진 상태에서 얼굴 정면 방향을 향하고 있기 때문에 Z축 값을 가지고 있지 않다. 그러므로 실제적인 얼굴 모션 데이터는 3차원 좌표 값이 아니라 2차원 좌표 값을 가진 얼굴 모션 데이터이다. 물론 Facial Tracker를 사용하지 않고 단순히 최신행 고가의 전신용 모션 캡처 시스템을 사용해서 얼굴 표정을 모션 캡처한 다음, Z축 좌표 값을 삭제하여 사용할 수도 있다. 그러나 본 논문에서는 3차원 캐릭터가 아닌 2차원 캐릭터의 얼굴 표정 애니메이션 생성이 목적이기 때문에 저비용으로 사용 가능한 Facial Tracker를 그대로 활용하여 캡처한 얼굴 모션 데이터를 사용하기로 한다. 또한 본 논문에서는 벡터 기반의 캐릭터 작업에 위해 최근에 가장 많이 사용되고 있는 Adobe사의 일러스트레이터를 사용하며, 캐릭터 얼굴 표정 애니메이션 생성을 위한 사용자 인터페이스 개발을 위해서는 일러스트레이터 SDK를 사용하여 Plug-in 형식으로 구현하도록 한다.

### III. 시스템 설계 및 구현

#### 3.1 시스템 구성도

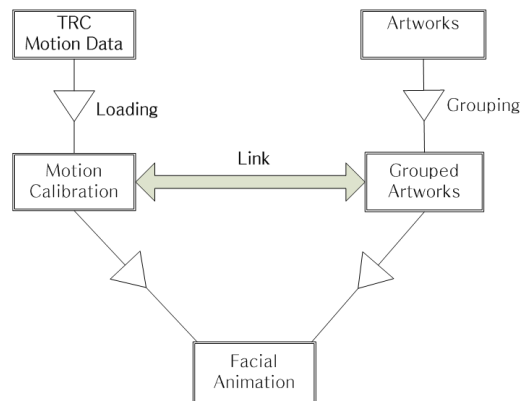


그림 3. 시스템 구성도

본 논문에서는 한 명의 배우로 하여금 다양한 얼굴 표정들을 연출하게 하고 이를 캡처한 얼굴 모션 데이터

들을 배우의 얼굴과 형태가 다른 수많은 캐리커처의 얼굴 모델에 재사용(Retargeting)하는 기법을 적용한다. [그림 3]은 본 논문에서 제안하는 캐리커처의 얼굴 표정 애니메이션 생성이 가능하도록 하기 위한 전체적인 다이어그램이다.

[그림 3]의 다이어그램을 간단히 살펴보면, 먼저 얼굴 모션 데이터를 읽어 들이고 캐리커처에 사용하도록 적합한 형태로 변환 즉, 모션 캘리브레이션(Calibration) 과정을 거쳐야 한다. 또한 캐리커처의 얼굴 표정을 위한 각 특징 부분 즉, 눈썹, 눈, 코 및 입 등의 Artwork(일러스트레이터에서 그림을 이루는 단위)를 각각 하나로 그룹화한 다음 캘리브레이션된 모션 데이터와 연결시키면 캐리커처의 얼굴 표정 애니메이션 생성이 가능해진다. 얼굴 모션 데이터를 캐리커처에 적용하기 위해 얼굴 모션 데이터(\*.trc)를 읽어 들이기>Loading) 위한 과정에서는 얼굴 모션 데이터 중에서 캐리커처에서 사용되는 실제적인 마커 데이터들만을 남기고 그 외의 잔여 마커 데이터들은 제거하여야 한다. 왜냐하면, 실험에 사용하기 위한 얼굴 모션 데이터는 [그림 1] 및 [그림 2]와 같이 총 28개의 소형 마커를 배우의 얼굴에 부착하여 캡처하였지만, 캐릭터를 제작하면서 28개의 마커를 부착할 만큼의 얼굴 근육 생성은 필요하지 않다는 것을 알게 되었기 때문이다. 그래서 본 논문에서는 캐리커처의 얼굴 표정을 표현하는 대표적인 부분 즉, 좌우 눈썹, 좌우상하 눈꺼풀, 코, 위아래 입술 등의 위치에 해당하는 18개의 마커를 실험에 사용하기로 하였다. 또한 Facial Tracker로부터 캡처한 얼굴 모션 데이터는 3차원 컴퓨터 그래픽스 분야에서 사용하고 있는 오른손 좌표계와 차이가 있기 때문에 식 (1)과 같이 좌표계의 변환과정을 거쳐야 한다.

$$\begin{aligned} result.x &= -motion.y; \\ result.y &= -motion.x; \end{aligned} \quad \text{식 (1)}$$

식 (1)에서 *motion*은 얼굴 모션 데이터에 저장된 원래의 모션 데이터를 의미하고, *result*는 본 논문을 위해서 변형시킨 결과 좌표 값을 의미한다. 즉, Facial Tracker로 캡처한 원래의 모션 데이터 좌표계는 오른

쪽 윗부분이 (0, 0)인 원점의 좌표이고, 왼쪽으로는 X축, 아래로는 Y축이 되는데, 본 논문에서는 원래의 모션 데이터를 읽어 들인 후 식 (1)과 같이 변형하여 (0,0)인 원점이 왼쪽 아래로 오게 하고 오른쪽으로 X축, 위로 Y축 방향이 되도록 각 프레임별로 다시 저장하여 사용한다.

### 3.2 사용자 인터페이스

[그림 4]는 캐리커처 얼굴 표정 애니메이션을 생성하기 위하여 일러스트레이터의 Plug-in으로 구현한 사용자 인터페이스 초기화면이다. 사용자 인터페이스의 'Select Motion'은 캐리커처에게 적용할 표정의 종류로서 얼굴 모션 데이터의 종류이기도 하다. 메뉴 'Scale Box'는 얼굴 모션 데이터의 크기가 캐리커처의 크기에 비하여 상대적으로 매우 크기 때문에 얼굴 모션 데이터를 캐리커처의 얼굴 크기에 맞게 재사용(Retargeting / Resizing)하기 위해 [그림 5]와 같이 공간의 규모를 설정하는 것으로서, 이때 모션 캘리브레이션 과정이 필요한 것이다.

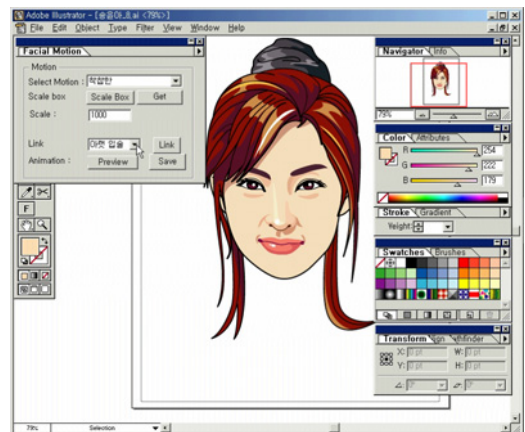


그림 4. 사용자 인터페이스의 초기 화면

메뉴 'Scale'은 모션 캘리브레이션 값을 수치적으로 표시한 것이다. 메뉴 'Link'는 캐리커처의 특징 점 즉, 좌우 눈썹, 좌우상하 눈꺼풀, 코, 위아래입술 들 중 하나를 선택하고 "Link" 버튼을 클릭하면 얼굴 모션 데이터의 특정 마커와 캐리커처의 특징 점이 각각 연결되도록 하는 것이다. 마지막 메뉴 'Animation'은 'Link'에 의해서

연결된 전체적인 캐리커처의 특징 점들이 한꺼번에 얼굴 모션 데이터에 의해서 움직이게 되는 것을 확인할 수 있으며 파일로 저장할 수 있도록 하였다. 특히 “Preview”버튼은 일러스트레이터와 사용자 인터페이스 상태에서 실시간으로 결과를 확인할 수 있는 기능이다.

### 3.3 모션 캘리브레이션

본 논문의 목적은 벡터 기반의 캐리커처에 얼굴 모션 데이터를 적용하여 표정 애니메이션을 쉽고 자연스럽게 생성할 수 있도록 하는 것인데, 얼굴 모션 데이터를 캐리커처에 적용하기 전에 반드시 거쳐야할 과정이 있다. 그것은 바로 모션 데이터를 캘리브레이션해야 한다는 것인데, 캘리브레이션(calibration)은 교정의 의미를 가지고 있다. 본 논문에서 사용하고자 하는 얼굴 모션 데이터에서 마커의 이동 값들은 모션 캡처 당시 배우의 얼굴 크기에 비례하는 값들인데, 얼굴 모션 캡처 데이터를 적용하려는 캐리커처의 얼굴 크기는 이와 매우 큰 차이가 있다.

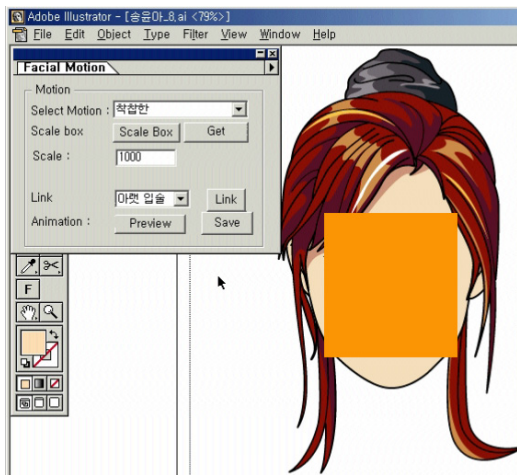


그림 5. 얼굴 모션 데이터를 캐리커처의 얼굴 영역에 맞게 캘리브레이션하기 위한 공간 영역 설정

그러므로 본 논문에서는 기존의 얼굴 모션 데이터가 캐리커처에 적용될 수 있는 같거나 유사한 크기의 영역을 가지도록 변형하기 위한 비례 값을 얻어야 할 필요가 있으며, 이를 모션 캘리브레이션이라 한다. 그러나

모션 캘리브레이션을 수행하기 위해 캐리커처의 얼굴 크기에 대한 기준을 자동적으로 얻어내기란 매우 힘든 작업이다. 왜냐하면 캐리커처를 설계하고 디자인하는 디자이너의 작업 방식과 과정에 따라서 캐리커처의 크기가 매우 다양하고 상이하기 때문이다. 그러므로 본 논문에서는 캐리커처의 얼굴 영역을 결정하는 판단 기준을 사용자가 도구를 다룰 때 직접 입력할 수 있도록 하는 방법을 취한다. 본 논문에서는 이와 같이 얼굴 모션 데이터를 캐리커처의 얼굴 크기에 맞추도록 그림 5와 같이 사용자가 사용자 인터페이스의 ‘Scale Box’ 버튼 메뉴를 직접 사용하여 얻어내도록 하였다. 즉, 사용자가 직접 캐리커처를 보고 스케일 박스를 사용하여 캐리커처의 얼굴 크기를 정해주면 정해진 스케일 박스 크기만큼의 규모로 얼굴 모션 데이터가 스케일(축소)되어 적용된다. 그리고 이와 같은 방법으로 얼굴 모션 데이터를 실험에 사용할 하나의 캐리커처의 얼굴 크기에 수십 차례 이상 맞춰본 결과 스케일 값(Scale Value)이 ‘1000’일 때 가장 적합한 규모로 스케일 된 상태이고 원래 모션 데이터에 가장 가까운 얼굴 표정을 그대로 표현할 수 있었다는 것을 알게 되었다. 그러므로 [그림 4] 및 [그림 5]와 같이 스케일 값의 기본을 ‘1000’으로 초기화 하였다. 물론 사용자는 수작업으로 직접 스케일 값을 변경해가면서 캐리커처의 얼굴 표정 애니메이션 강도를 제어할 수 있는데, 스케일 값이 커지면 원래 모션 데이터보다 표정의 강도가 강해지고, 스케일 값이 작아지면 표정의 강도도 약해진다.

### 3.4 제어점 연결

벡터 기반 캐리커처는 눈썹, 눈, 코 및 입 등을 구분시켜 각각의 영역을 이루고 있고, 모두 별도의 연속된 베지어 곡선으로 구성되어 있다. 그러므로 벡터 기반 캐리커처의 얼굴 표정 애니메이션을 얼굴 모션 데이터를 사용하여 생성하고자 한다면, 캐리커처의 각 부위별로 제어점을 설정하고 이들을 얼굴 모션 데이터의 특정 마커들의 집합과 연결시켜야 할 필요가 있다. 물론 이와 같은 제어점을 제대로 설정하기 위해서는 캐리커처 작업 전에 얼굴 모션 캡처를 위한 배우의 얼굴에 부착한 소형 마커의 위치를 파악하고 캐리커처 얼굴의 동일한



위치에 제어점을 설정해두어야 한다. 이는 원래의 얼굴 모션 데이터를 캐리커처의 얼굴에 재사용하기 위한 가장 단순하면서도 핵심적인 방법이라 할 수 있다. 그러나 캐리커처는 각 부분별로 연속된 벡터 곡선의 집합이기 때문에 캐리커처의 제어점을 마커와 일대일로 대응시키기 보다는 그룹으로 설정하여 한꺼번에 연결시켜주는 방법을 사용하기로 한다. 즉, 캐리커처의 눈썹, 눈, 코 및 입 등의 주요 부분들을 각각 마우스로 선택한 다음 별도의 그룹을 형성한다. 왜냐하면 캐리커처의 각 영역들을 별도로 그룹화하지 않고 제어점과 마커를 일대일로 연결시키게 되면 전체적인 얼굴 표정이 매우 부자연스럽게 일그러지기 때문이다. 또한 얼굴 모션 데이터에서 하나의 마커와 관련되는 캐리커처의 제어점을 일러스트레이터 파일(\*.ai)에서 자동적으로 찾아 연결시키는 것은 매우 힘든 작업이다. 그러므로 본 논문에서는 캐리커처에서 특정 마커와 관련되는 Artwork(일러스트레이터에서 그림을 이루는 단위)을 선택하고, 이 Artwork과 마커 번호를 연결하는 방법을 사용하도록 한다. 즉, 캐리커처의 눈썹, 눈, 코 및 입술 등의 영역 단위로 작업하는 것이 더 직관적이라고 생각되므로, 얼굴의 각 영역 부분인 Artwork를 선택하고 Artwork와 관련된 영역의 마커 번호들을 수 개씩 한 번에 연결시키도록 한다. 여기서 캐리커처의 Artwork에 제어점으로 연결될 각각의 마커 번호들은 [그림 3]을 참조하면 된다. 본 논문에서 실험을 위해 제작한 캐리커처는 일반 캐리커처와는 매우 상이한데, 그것은 바로 캐리커처의 자연스러운 표정을 위하여 눈과 입 부위의 Artwork는 레이어 형식으로 내부 및 외부 Artwork로 구성되어 있다는 것이다. 즉, 캐리커처가 다양한 표정을 연출할 때, 캐리커처의 눈동자(흰자)가 눈꺼풀보다 더 넓은 영역으로 내부 레이어에 숨겨서 표현하고 있으며, 윗입술과 아랫입술이 움직일 때 이빨을 포함한 입 속의 형태를 입술 내부 레이어에 숨겨서 표현하였다. 그러나 내부 레이어로 숨겨진 Artwork들은 제어점 연결과 직접적인 관련은 전혀 없으나 외부 레이어들이 움직일 때 자연스럽게 보이도록 하기 위해서는 필수적이다.

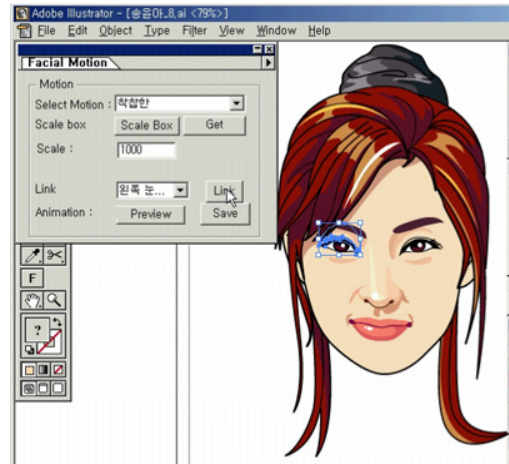


그림 6. 캐리커처의 특정 Artwork(왼쪽 눈썹 부분) 선택한 상태(좌)와 사용자 인터페이스에서 해당 부위 연결을 위한 명칭 선택(우)

사용자 인터페이스를 사용하여 제어점을 연결하기 위해서는 [그림 6]의 오른쪽 캐리커처와 같이 캐리커처의 그룹으로 정의된 특정 Artwork 부분을 마우스로 선택한 다음, [그림 6]의 왼쪽 사용자 인터페이스와 같이 메뉴 'Link'의 콤보 박스에서 해당하는 연결 부위 명칭을 선택하고 오른쪽 버튼 메뉴 'Link'를 클릭해준다. 여기서 캐리커처의 눈썹과 눈은 사용자가 바라보는 위치를 기준으로 좌우 방향 설정을 하면 되도록 하였다. 그리고 사용자는 이와 같은 과정으로 모든 Artwork마다의 제어점 연결을 완료한 다음, 사용자 인터페이스의 콤보 박스 메뉴 'Select Motion'에서 생성하고자 하는 표정의 종류를 선택하고, 'Animation' 메뉴의 "Preview" 버튼을 클릭하여 만족할 만한 표정 애니메이션이 생성되는지를 확인하면 된다. 본 논문에서는 캐리커처의 표정 애니메이션 생성을 위하여 '무표정', '착잡한 표정', '고마운 표정', '당황스러운 표정' 등을 포함하여 수십 개의 얼굴 모션 데이터를 실험에 사용하였다.

### 3.5 얼굴 표정 애니메이션

얼굴 모션 데이터의 특정 마커 번호들과 캐리커처의 그룹으로 정의된 Artwork들의 연결이 완료되면, 캐리커처의 얼굴 표정 애니메이션이 제대로 생성되어지는

지를 확인해야한다. 이는 얼굴 모션 데이터와 캐리커처가 전체적으로 제대로 연결되어졌는지를 확인하는 과정이기도 하다. 본 과정에서 중요한 것은 얼굴 모션 데이터에서 마커의 움직임을 캐리커처에 적용하기 위해 각 마커에 대한 세그먼트(Segment, 부분)의 가중치(Weight)를 구해주어야 한다는 것이다. 가령 캐리커처의 눈썹을 예로 들었을 때, 눈썹 바깥쪽 끝에 붙어 있는 마커의 움직임은 눈썹 안쪽의 세그먼트들에 대해서는 크게 영향을 주지 못한다. 반대로 캐리커처의 눈썹 바깥쪽에 가까운 세그먼트들에 대해서는 큰 영향을 미친다. 이와 같이 각 마커의 위치와 세그먼트들의 위치간의 차이를 이용해서 마커에 대한 세그먼트의 가중치를 구해준다.

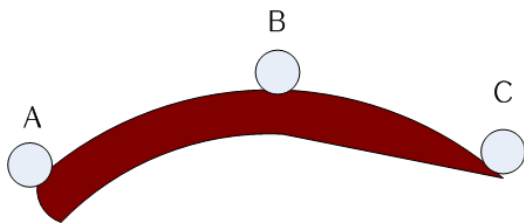


그림 7. 캐리커처의 눈썹 부위에 해당하는 Artwork와 연결된 해당 마커들의 상호작용

[그림 7]은 캐리커처의 눈썹과 그에 연결된 마커들의 예를 보여주고 있다. 마커 A의 움직임은 마커 C쪽에 있는 세그먼트들에는 거의 영향을 미치지 못하게 되고, 0에 가까운 가중치를 가진다. 반대로 마커 A쪽에 있는 세그먼트들은 1에 가까운 가중치를 가진다. 이와 같은 방법으로 얼굴 모션 데이터를 연속적으로 사용할 경우, 모션은 기준 프레임인 0번 프레임을 기준으로 변량을 사용하여 적용하고, 마찬가지로 기준 프레임에서의 캐리커처의 모양을 기준으로 매 프레임마다 적용한다. 그림 8은 본 논문에서 일러스트레이터 Plug-In으로 구현한 사용자 인터페이스를 사용하여 벡터 기반 캐리커처에 얼굴 모션 데이터를 적용하여 생성한 캐리커처의 얼굴 표정 애니메이션들 중 대표적인 표정에 해당하는 프레임들이다. [그림 8]의 (a)는 무표정 상태이고, (b)는 표정의 강도를 표현하는 스케일 값을 기본 값 '1000'으

로 설정한 상태에서의 착잡한 표정, (c)는 스케일 값을 '1500'으로 설정한 상태에서의 착잡한 표정, (d)는 기본 스케일 값의 고마운 표정, (e)는 기본 스케일 값의 당황스러운 표정을 표현한 것이다.

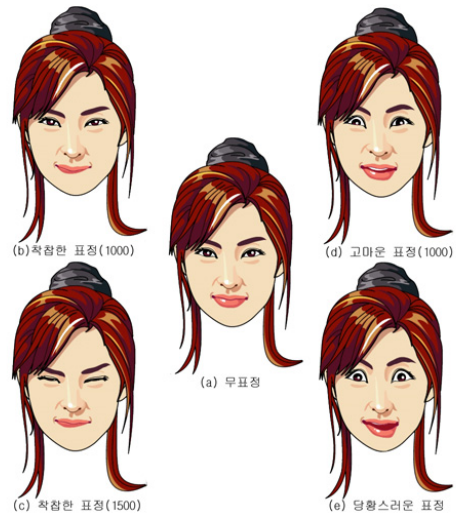


그림 8. 얼굴 모션 데이터가 적용된 캐리커처의 얼굴 표정 애니메이션 생성 결과 예제

[그림 8]에서 보는 바와 같이 벡터 기반 캐리커처의 얼굴 표정 애니메이션을 수작업으로 생성한다고 가정하면, 매우 비경제적이고 비효율적일 것임은 분명하다. 그러나 본 논문의 결과물을 이용하여 캐리커처의 얼굴 표정 애니메이션을 생성하고자 한다면, 매우 다양하고 자연스러운 캐리커처의 표정 애니메이션을 생성할 수 있을 것임은 분명하다.

#### IV. 결론

본 논문에서는 Facial Tracker로 캡처한 얼굴 모션 데이터를 벡터 기반의 캐리커처에 적용하여 캐리커처 얼굴 표정 애니메이션 생성이 가능하도록 하기 위한 방법을 제시하였다. 특히 벡터 기반 캐리커처는 Adobe사의 일러스트레이터로 제작한 파일(\*.ai)이며, 캐리커처의 얼굴 표정 애니메이션 생성을 위한 사용자 인터페이

스는 일러스트레이터 SDK를 사용하여 Plug-In 방식으로 구현하였다. 본 논문을 준비하면서 가장 어려웠던 부분은 일러스트레이터의 벡터 파일(\*.ai)의 구조를 분석하고 일러스트레이터 SDK를 사용하여 얼굴 모션 데이터를 외부로부터 읽어 들여 캐리커처의 제어점들과 직접 연결시키는 부분이었다. 또한 광학식 모션 캡처 시스템으로 캡처한 모션 캡처 데이터는 3차원 캐릭터 모델에 적용하는 것이 일반적이는데, 본 논문에서는 벡터 기반의 2차원 캐리커처에 적용하였다는 것이 타 논문이나 연구 결과들과 비교하였을 때 상이하면서 특이한 부분이라 말할 수 있다. 또한 본 논문의 결과는 단순히 정지된 캐리커처보다 활동적이고 다양한 표정의 변화를 애니메이션으로 생성 가능하다는 것에서 매우 큰 장점을 가지고 있다. 그러나 벡터 기반 캐리커처에 얼굴 모션 데이터를 적용하여 얼굴 표정 애니메이션을 생성해본 결과, 아직 해결하지 못한 부분들이 있다는 것을 알게 되었다. 즉, 얼굴 모션 데이터는 광학식 모션 캡처 시스템으로 캡처할 때 배우의 얼굴 주 근육 부위에만 소형 마커를 부착하여 다양한 표정들을 캡처하였다는 것이다. 그런 이유로 캐리커처의 다양한 얼굴 표정을 보다 더 자연스럽게 표현하려고 한다면 얼굴 윤곽선까지 표정의 변화에 따라 움직여주어야 하는데, 이를 표현하지 않았다. 그러므로 향후 연구에서는 캐리커처의 얼굴 윤곽선뿐만 아니라 헤어스타일까지 포함하는 얼굴 전체의 움직임을 표현하는 방향으로 개선시켜나갈 것으로 판단된다. 또한 전신 모션 캡처 데이터를 사용하여 캐리커처의 전신 애니메이션 생성이 가능하게 하는 것도 충분히 가능할 것으로 기대된다.

**참 고 문 헌**

[1] S. E. Brennan, "Caricature Generator : The Dynamic Exaggeration of Faces by Computer," Leonardo, Vol.18, No.3, pp.170-178, 1985.  
 [2] S. E. Brennan, "Caricature Generator," M.Sc. Thesis, MIT, 1982.  
 [3] San Mateo, CA: Morgan Kaufman Publishers.

Brennan, S. "The Caricature Generator," Leonardo 18, pp.170-178, 1985.  
 [4] S. M. Platt and N. I. Badler. "Animating facial expression," ACM SIGGRAPH Conference Proceedings, Vol.15, No.3, pp.245-252, 1981.  
 [5] P. Ekman, "Facial Expression and Emotion," American Psychologist, Vol.48, pp.384-392, 1993.  
 [6] A. Fridlund. "Human facial expression : An evolutionary view," Academic Press, New York, 1994.  
 [7] G. Faigin, "The Artist's Complete Guide to Facial Expression," Watson-Guptill, New York, 1990.  
 [8] Kiyoshi Arai, Tsuneya Kurihara and Ken-ichi Anjyo, "Bilinear interpolation for facial expression and metamorphosis in real-time animation," Visual Computer, Vol.12, No.3, 1996.  
 [9] M. Gleicher. "Retargeting motion to new characters," Proc. SIGGRAPH Computer Graphics, 1998(8).  
 [10] 이옥경, 박연출, 김성호, "얼굴 특징 정보 추출을 이용한 캐리커처 자동생성", 정보과학회, 2001.  
 [11] 이은정, 권지용, 이인권, "규칙 기반 캐리커처 자동 생성 기법", 한국컴퓨터그래픽스학회 추계학술대회, pp.47-52, 2006.  
 [12] 이은정, 권지용, 이인권, "캐리커처 비디오", 한국컴퓨터그래픽스학회, pp.65-66, 2007.  
 [13] 황지연, "캐리커처를 이용한 실시간 인터랙티브 애니메이션", 중앙대학교 첨단영상대학원 석사학위논문, 2007.  
 [14] P. Y. Chiang, W. H. Liao, and T. Y. Li, "Automatic Caricature Generated by Analyzing Facial Features," ACCV '04 Proceeding, 2004.  
 [15] L. Liang, H. Chen, Y.-Q. Xu, and H.-Y. Shum, "Example based Caricature generation with exaggeration," In Proc. the 10th Pacific Conference on Computer Graphics and



Applications, pp.386. IEEE Computer Society, 2002.

저 자 소 개

김 성 호(Sung-Ho Kim)

정회원



- 1998년 8월 : 숭실대학교 컴퓨터학과(공학석사)
- 2005년 2월 : 숭실대학교 컴퓨터학과(공학박사)
- 2006년 3월 ~ 현재 : 상지대학교 컴퓨터정보공학부 교수

<관심분야> : 컴퓨터 그래픽스, 컴퓨터 애니메이션, 모션 캡처 애니메이션, 가상현실, Web3D, 멀티미디어, 컴퓨터게임, GIS