



제어할 수 있다는 장점이 있다. 해부학에 기반한 방법의 경우 사람의 얼굴에 존재하는 각 근육의 수축 정도를 매개변수로 사용하여 표정을 표현하는 방법으로, 충분한 계산 시간이 주어지면 물리적 모의실험(simulation)을 통하여 사실적인 표정을 정확하게 생성할 수 있다는 장점이 있다. 지역성에 근거한 방법에서는 얼굴의 한 점이 움직일 때 주변의 부분도 유사하게 움직인다는 점에 착안하여 얼굴 모델의 무수한 정점을 제어하는 대신 몇 개의 점이나 곡선을 통해 얼굴의 움직임을 표현하는 방법이다. 이와 같은 방법을 통해 숙련된 애니메이터는 얼굴의 움직임을 자유롭게 표현할 수 있다.

본 논문에서는 많은 양의 표정 실측 데이터를 기반으로 다양한 표정을 적은 수의 매개변수로 표현하기 위한 방법을 제안한다. 동작 포착 기술 등을 이용하여 얻은 많은 양의 표정 데이터를 분석하여 데이터를 재 구성할 수 있는 최소한의 독립적인 움직임을 생성한다. 얼굴 표정 데이터에서 각 표정은 기본 표정들의 조합으로 나타난다고 가정하면 다양한 표정에 있어 얼굴 각 부분의 움직임은 기본 표정들의 가중치의 조합이라 할 수 있다. 따라서 표정 데이터의 각 정점은 표정을 제어하기 위한 신호들의 조합을 관측한 결과로 각 정점의 움직임에 맹목적 음원 분리 기술(blind source separation technique)을 적용하면 독립적인 제어 신호(control signal)를 얻을 수 있다.

제어 신호가 주어지면 각 제어 신호에 대응하는 얼굴의 움직임을 생성할 수 있어야 한다. 가정에서 얼굴의 각 정점의 변위는 제어 신호의 선형 조합으로 나타낼 수 있는 것으로 가정하였지만, 실제 얼굴의 움직임은 복잡한 근육과 피부 조직의 움직임의 조합으로 비선형성을 가진다. 따라서 표정 데이터에서 얻은 제어 신호를 이용하여 표정 데이터를 가급적 정확하게 재구성(reconstruction)하기 위해 본 논문에서는 간단한 비선형 모델을 개발하고 데이터로부터 모델을 학습하는 방법을 제안한다.

본 논문의 2절에서는 표정 생성과 복제, 표정의 매개변수화, 맹목적 음원 분리 기술이 애니메이션에 응용에 관한 기존 연구를 살펴본다. 3절에서는 표정 데이터를 이용한 표정의 매개변수화와 얼굴 변형 모델을 학습하기 위한 기술적인 접근 방법을 소개하고, 4절에서는 표정 분석의 실험 결과와 얼굴 표정 복제의 응용에 적용하기 위한 방법을 보인다. 마지막으로 5절에서는 본 논문의 결론을 맺고 향후 연구를 살펴본다.

## 2 관련 연구

### 2.1 표정 합성 및 복제

3차원 얼굴 모델은 많은 정점과 면들을 가지고 있기 때문에 직접적으로 각 정점의 위치를 조절하여 표정을 만드는 과정은 매우 많은 노력과 시간이 필요하다. 따라서 이와 같이 높은 자유도를 가지는 얼굴 모델의 자유도를 낮추는 방법으로 블렌드 웨입에 기반한 방법이 활발히 연구되었다 [2, 3, 4, 5]. 특히 관련 산업계에서는 블렌드 웨입을 이용한 애니메이션을 제작하는 방법이 가장 널리 사용되고 있다.

얼굴 애니메이션 제작에 자주 사용되는 또 다른 방법으로 연기 기반 접근 방법을 들 수 있다 [7, 8, 9, 10, 11, 12]. 연기 기반 접근 방법에서는 실제 사람의 얼굴 표정을 수치화하여 저장하고 이를 애니메이션에 이용하는 방법으로 사실적인 표정을 적은 노력으로 생성할 수 있다는 장점이 있다. 이와 같은 접근

방법 중의 하나로 Zhang 등은 데이터 기반 얼굴 역운동학 기법을 제안하였다 [13]. 이 방법에서는 얼굴에 제어점을 설정하고 제어점의 위치를 지정하면 해당되는 표정을 표정 데이터를 기반으로 합성한다.

표정 복제(cloning)는 표정 포착 등으로 얻은 표정 데이터나 사전 제작한 표정 애니메이션을 바탕으로 새로운 캐릭터의 표정 애니메이션을 생성하는 방법이다. 특히 새로운 캐릭터의 얼굴의 특성에 맞게 표정을 변형함으로써 기존의 표정 데이터를 재사용하여 사실적인 애니메이션을 효율적으로 생성할 수 있다. Noh와 Neumann은 원본 얼굴과 대상 얼굴의 기하학적 차이를 기반으로 표정을 변경하여 대입하는 방법을 제안하였다 [14]. Vlastic 등은 3차원 스캐닝 기술을 활용하여 얻은 많은 수의 얼굴 기하 모델들을 성격, 표정, 발성음 등에 따라 표정 공간을 구성하고 매개변수화 하는 방법을 제안하였다 [15].

### 2.2 얼굴 표정의 매개 변수화

복잡한 얼굴 표정을 단순하게 표현하기 위하여 표정을 몇 개의 수치로 나타내기 위한 매개변수화 방법들이 다수 제시되었다. 이중 애니메이션과 심리학 분야에서 공히 가장 보편적으로 사용되는 방법으로 FACS를 들 수 있다 [1]. FACS에서는 표정을 46개의 단위 움직임(action unit)들의 조합으로 본다. 따라서 각각의 단위 움직임을 블렌드 웨입 등으로 제작하여 애니메이션에 적용할 수 있다. 하지만 FACS는 시각적인 특징을 기초로 사람이 직관을 이용하여 단위 움직임을 분석하였기 때문에 실질적으로 중복되는 움직임을 나타내는 단위 움직임들이 다수 존재한다. 최근 이와 같은 FACS의 문제점을 해결하기 위해 새로운 FACS 모델을 개발하기 위한 연구가 활발하다.

얼굴 모델의 기하학적인 특징을 바탕으로 표정을 매개변수화하는 연구도 활발히 진행되어 왔다 [16, 17, 18, 19, 20, 21]. Sifakis 등은 FACS 등이 얼굴 표정을 의미의 기반으로 매개변수화 할 때 나타나는 문제점을 해결하기 위해 얼굴의 물성을 파악하여 해부학적으로 사실적인 얼굴 모델을 이용하기 표정을 생성하는 방법을 제안하였다 [22]. 이후에 Sifakis 등은 해부학적으로 정확하게 모델링 된 근육 모델은 얼굴 표정을 정확하게 매개변수화 함을 보였다 [23].

### 2.3 애니메이션을 위한 독립 요소 추출

독립 요소 분석은 많은 데이터로부터 특징적인 성질을 추출하는데 널리 사용되었고 애니메이션 분야도 예외는 아니다 [24]. Mori와 Hoshino는 동작 데이터에 독립 요소 분석 기법을 적용하여 동작으로부터 의미 있는 정보를 추출하는 연구를 제안하였다 [25]. Cao 등은 음성을 위한 움직임과 표정을 위한 움직임을 구분하기 위해 독립 요소 분석을 사용하였다 [26]. Shapiro 등은 같은 방법을 동작 데이터에 적용하여 동작에서 주된 움직임과 스타일을 분리하기 위한 방법을 제안하였다 [27]. Fidaleo 등은 독립 요소 분석을 이용하여 제스처의 크기를 구분하고 이를 통하여 관련 부분(coarticulation region)들을 분석하는 방법을 제안하였다. 얼굴 영상의 각 부분을 분할하고 독립 요소 분석을 통해 얻은 동작 수준에 따라서 표정을 매개변수화 하였다 [28].

### 3 데이터 기반 표정 매개변수화

표정 매개변수화의 목적은 사람이 표현할 수 있는 다양한 표정을 적은 수의 매개변수로 나타내기 위한 모델을 개발하는 것이다. 데이터베이스에 존재하는 많은 수의 표정으로부터 표정들을 구성하는 기본 움직임의 조합으로 표현될 것이다. 따라서 이와 같은 기본 움직임을 추출할 수 있다면 각 표정을 나타내기 위해 각각의 움직임을 조합하기 위한 가중치를 매개변수로 사용하여 표정을 제어할 수 있다.

표정을 나타내는 기본 움직임을 결정하는 방법에 따라서 다양한 매개변수화가 가능하다. 예를 들면 FACS에서는 많은 사람의 다양한 표정을 일일이 관찰하고 정리함으로써 사람이 기본 움직임을 추출한 것으로 볼 수 있다. 또한 블렌드 셰입의 경우에는 경험이 많은 애니메이터가 자신의 경험을 바탕으로 다양한 표정을 생성할 수 있다고 믿어지는 움직임들을 임의로 결정할 것이다. 본 논문에서는 데이터를 수치적으로 분석함으로써 주요 움직임을 추출한다.

#### 3.1 매개변수화의 요구사항

애니메이션에 적합한 표정의 매개변수화에 사용되는 기본 움직임은 다음과 같은 조건을 만족하여야 한다.

- **완결성(completeness):** 표정 매개변수화에 사용되는 기본 움직임은 이들의 조합으로 사람이 나타낼 수 있는 가능한 많은 범위의 표정을 나타낼 수 있어야 한다. 특히 얼굴 전체로 나타내는 표정 뿐만 아니라 미묘한 감정을 나타내기 위한 작은 표정의 변화도 효과적으로 표현할 수 있어야 한다.
- **간결성(compactness):** 사람의 표정을 가능하면 적은 수의 매개변수의 수로 나타낼 수 있어야 한다. 즉, 실제 사람의 표정의 움직임을 파악하여 움직임의 기초가 되는 최소한의 개수의 기본 움직임을 이용하여야 한다.
- **사실성(soundness):** 표정 매개변수를 이용하여 생성한 표정은 대부분 정상적인 사람이 나타낼 수 있는 자연스러운 표정이어야 한다. 즉 각각의 매개변수에 해당하는 기본 움직임은 정상적인 사람의 표정의 대응되어야 하고 적절한 범위의 매개변수 조합은 정상적인 표정으로 나타나야 한다.
- **유일성(uniquness):** 임의의 표정은 기본적으로 한 가지 매개변수 조합으로 나타낼 수 있어야 한다. 이는 각 기본 표정이 다른 표정들의 조합으로 표현되지 않아야 한다는 조건으로 표현된다.
- **직관성(intuitiveness):** 각각의 매개변수는 얼굴의 움직임을 구체적으로 표현하여야 한다. 즉 각각의 매개변수가 나타내는 움직임을 사람의 표정과 연관하여 이해할 수 있어야 하고 임의의 표정이 주어지면 이를 표현하기 위한 매개변수 조합을 직관적으로 어렵지 않게 추측할 수 있어야 한다.
- **효율성(eficiency):** 매개변수의 조합이 결정되면 이 매개변수 조합에 의해 나타나는 표정을 빠르게 생성할 수 있어야 한다. 특히 애니메이션 제작 과정에서 캐릭터의 미묘한 감정 변화를 나타내기 위해서는 표정을 지정하고 수정하는 과정을 반복하는 경우가 많기 때문에 상호작용이

가능한 정도의 속도로 표정을 생성할 수 있어야 한다.

이러한 조건들을 기준으로 보면 기존의 매개변수화 방법들은 일부 조건에 최적화 되어 있음을 확인할 수 있다. 먼저 블렌드 셰입을 이용하는 방법은 각각의 표정을 최대한 직관적으로 이해할 수 있도록 구성되어 있으며 표정을 생성하는 과정이 가장 효율적이다. 따라서 특정한 표정을 생성하는 과정에서 어떤 조합을 이용하여야 하는지를 직관적으로 추측할 수 있고 결과가 빠르게 가시화될 수 있다. 또한 각각의 블렌드 셰입을 숙련된 애니메이터가 정교하게 제작한다면 어느 정도의 사실성을 확보할 수 있다. 하지만 이와 매개변수가 특정 애니메이터의 직관에 의존하여 결정되었기 때문에 캐릭터의 모든 표정을 나타낼 수 없을 경우가 많이 있고, 가능한 표정을 생성하는 데 필요한 매개변수에 비해 더 많은 수의 매개변수가 필요한 경우가 많이 있다.

FACS의 경우에는 수많은 관찰을 통하여 완결성을 확보하고자 노력한 것으로 생각할 수 있다. 또한 각각의 기본 움직임이 직관적인 의미를 가지며 각각의 표정이 정상적인 표정에 대응된다. 뿐만 아니라 FACS의 기본 움직임에 기반하여 블렌드 셰입을 제작할 경우 효율적으로 표정을 생성할 수 있다. 반면 간결성이나 유일성에 한계가 있다. 몇명의 전문가가 시각적인 얼굴의 움직임을 기초로 얼굴의 기본 움직임을 파악하였기 때문에 최소 움직임을 보장할 수 없다. 실제로 최근에는 기존의 FACS에서 지정하였던 기본 움직임 중 다수를 더욱 간결한 형태로 축약하기 위한 연구가 활발히 진행 중이다.

근육 모델을 이용한 접근 방법은 실제 사람의 얼굴 근육이 표정을 나타내기 위한 기본 요소로 보기 때문에 높은 수준의 유일성과 완결성, 사실성이 확보된다. 반면 직관성에 문제가 있을 수 있는데, 이는 근육의 움직임이 복잡한 조직의 움직임을 통해 얼굴 표정으로 나타나기 때문이다. 약간의 경험으로 근육의 움직임을 표정의 변화로 연결하기 위한 지식을 습득할 수 있기는 하지만 전술한 복잡한 얼굴 조직의 변형을 계산하는 데는 많은 계산량이 요구되는 물리적 시뮬레이션이 사용되어야 하기 때문에 상호작용을 통하여 표정을 수정할 수 있는 수준의 효율성을 얻기 어렵다.

마지막으로 몇 개의 간단한 제어점을 기반으로 기하모델을 직접 변형하여 표정을 생성하는 방법은 가장 높은 직관성과 완결성을 보인다고 할 수 있다. 하지만 실제 사람이 실현할 수 있는 독립적인 표정에 비해 상당히 많은 자유도를 가지기 때문에 낮은 간결성을 가진다고 할 수 있다. 또한 이러한 높은 자유도에 기인하여 사실적인 표정을 생성하기 위해서는 숙련된 애니메이터가 많은 제어점들을 정밀하게 조절하여 하기 때문에 사실성을 유지하기 어렵다.

#### 3.2 접근 방법

본 논문에서는 표정 데이터베이스로부터 높은 완결성, 간결성, 사실성, 효율성을 얻기 위한 매개변수화 방법을 제안한다. 표정은 기본적으로 얼굴 근육과 목 주변의 근육의 움직임에 의해 결정된다. 이때 특별한 훈련을 거치지 않은 일반적인 사람들은 각각의 근육을 자유롭게 움직이지 못한다. 예를 들면 눈썹을 위로 움직이는 근육을 외곽 프론타리스(outer-frontalis) 한 쌍과 내측 프론타리스(inner-frontalis)의 세 개의 근육으로 구성이 되어 있지만 많은 사람들이 이들 각각을 독립적으로 움직이지 못한다. 이는 얼굴의 움직임이 근육의 개수보다 적은 수의 매개변수로 표현될 수 있을 가능성을 시사한다. 또한 일

굴의 움직임이 얼굴의 넓은 영역에 의해 나타난다. 예를 들어 앞서 지적한 눈썹을 올리는 움직임의 경우 눈썹과 이마 뿐만 아니라 코등, 심지어는 귀까지 움직이게 한다. 이는 애니메이션이 터가 눈썹을 움직이는 애니메이션을 자연스럽게 제작하고자 한다면 귀와 코등의 피부의 움직임까지 고려하여야 한다는 것이다.

본 논문에서는 이와 같은 관찰 결과에 기반하여 얼굴의 움직임을 분석한다. 많은 수의 표정 데이터를 분석하여 연관성(Correlation)이 없는 독립적인 가장 적은 수의 요소를 추출하여 기본 움직임으로 삼는다. 연관성에 기반하여 매개변수화를 수행할 경우 다음과 같은 장점을 예상할 수 있다. 먼저 연관성을 기반으로 수치적으로 움직임을 분리하기 때문에 특별한 사용자의 노력 없이 매개변수화를 진행할 수 있다. 특히 수치적으로 움직임을 분리하는 대표적인 방법인 주성분 분석(principle component analysis)에 의한 방법보다 연관성을 기반으로 분리할 경우 각각의 움직임이 직관적일 가능성이 높다. 그리고 연관성이 있는 모든 움직임을 한 개의 매개변수를 이용하여 조절하게 하면 최소한의 매개변수로 표정을 나타낼 수 있다. 또한 전역적으로 나타나는 얼굴의 움직임을 한개의 매개변수로 나타내며 따라 보다 자연스러운 표정을 생성하기 쉽다.

따라서 본 논문에서 제안하는 분리 방법은 지정된 데이터에 대해 완결성을 확보할 수 있다. 또한 연관성이 높은 모든 움직임을 하나의 매개변수로 나타내기 때문에 간결한 매개변수 집합을 얻을 수 있으며 각 움직임은 데이터에서 항상 동시에 나타나는 모든 움직임을 포함하기 때문에 여타 방법에 비해 쉽게 사실적인 애니메이션을 생성하는데 도움을 줄 수 있다. 그리고 기존에 시도되었던 자동적으로 매개변수를 얻기 위한 방법들에 비해 직관적인 움직임을 나타낼 가능성이 높다. 마지막으로 본 논문에서는 표정 분석 단계와는 별개로 적절한 변형 모델을 학습하기 위한 방법을 제안한다. 이때 높은 효율성을 얻기 위한 방법을 적용함으로써 효율적으로 애니메이션을 생성할 수 있게 한다.

이와 같은 가설을 바탕으로 본 논문에서는 표정 데이터베이스에 포함된 표정으로 부터 독립적이고 상호 관련성이 전혀 없는 몇 개의 기본 움직임의 조합으로 표현한다. 기본 표정들이 조합될 때 각각의 기본 움직임을 조합하기 위한 가중치를 매개변수로 하면 매개변수를 조절함으로써 표정을 생성할 수 있다. 따라서 본 논문에서 제안하는 표정 분석 방법은 얼굴 표면의 많은 점점들의 움직임을 신호로 생각하고 각각의 신호를 재 생성할 수 있는 독립적인 신호의 집합을 얻는 것이다. 각각의 독립적인 신호는 매개변수의 신호와 관련된다. 특히 한 개의 매개변수를 최대 값으로 지정하면 매개변수에 해당하는 기본 표정을 얻을 수 있다.

### 3.3 표정 분석

표정 데이터베이스로부터 상호 독립적인 매개변수를 추출하고 매개변수를 이용하여 표정을 생성하기 위해 본 논문에서 제안하는 기본 아이디어는 표정 데이터에 맹목적 음원 분리 기법의 하나인 독립 요소 분석 기법(ICA)을 적용하는 것이다. 이때 독립 요소 분석은 선형적인 정보에 적합하기 때문에 본 논문에서는 먼저 표정 데이터 베이스에 저장되어 있는 표정 애니메이션을 선형화하고 동기화하는 전처리 과정을 수행한다. 전처리된 표정 애니메이션 정보를 독립 요소 분석 기법을 통하여 분석하여 애니메이션에 대응하는 매개변수 신호를 얻는다. 이와 같은 과정으로 얻은 매개변수 신호와 데이터베이스에 존재

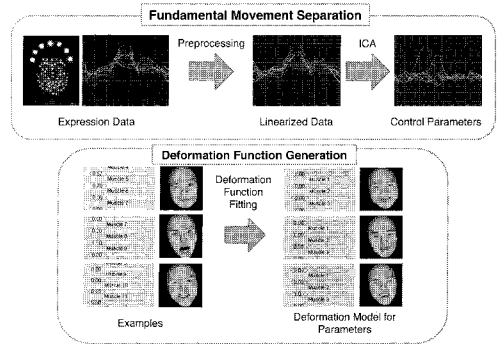


그림 1: 표정 분석 기법의 개요

하는 표정을 기초로 표정의 비선형성을 고려하여 매개변수에 따른 표정 변화를 얻는 과정을 거친다. 본 논문에서 제안하는 표정 분석 방법의 개략적인 과정은 그림 1에 나타나있다.

표정 데이터 베이스를 이용하여 매개변수 공간을 추출하고 애니메이션에 적용하기 위해서 본 논문에서는 다음과 같이 표정을 나타낸다. 먼저 3차원 표정 포착 등의 기술을 통해 얻은 표정이나 3차원 기하모델에 주어진 표정 정보는 기본적으로 표정 포착에 사용된 마커(marker)나 3차원 기하모델의 정점(vertex)의 변위(displacement)로 나타난다. 각 정점(혹은 마커)의 위치를  $\mathbf{p}_i$ 라 할때 표정은  $\mathbf{x} = (\Delta x_0 \Delta y_0 \Delta z_0 \Delta x_1 \Delta y_1 \Delta z_1 \dots \Delta x_n \Delta y_n \Delta z_n)^T$ 로 나타낼 수 있다. 이때  $(\Delta x_i, \Delta y_i, \Delta z_i)$ 는  $\mathbf{p}_i$ 의 좌표 축 방향으로의 변위를 나타낸다. 각 정점이나 마커의 변위는 무표정한 상태를 기준으로 변화량으로 나타낸다.

각각의 기본 표정을  $\mathbf{x}_i$ 라 할 때 매개변수 벡터  $\mathbf{s}$ 가 주어지면  $\mathbf{s}$ 에 대한 표정을 다음과 같은 선형 조합으로 근사할 수 있다.

$$\mathbf{x} = \sum_i \mathbf{s}_i \mathbf{x}_i \quad (1)$$

따라서 시간에 따라 변화하는 표정과 이에 해당되는 매개변수의 신호는 다음과 같은 관계가 성립한다.

$$\mathbf{x}(t) = \begin{pmatrix} \mathbf{x}_0 & \mathbf{x}_1 & \dots & \mathbf{x}_k \end{pmatrix} \mathbf{s}(t) \quad (2)$$

$$= \mathbf{A}\mathbf{s}(t). \quad (3)$$

매개변수와 표정과의 관계가 위와 같이 근사되면 독립 요소 분석 기법을 이용하여  $\mathbf{x}(t)$ 로부터  $\mathbf{s}(t)$ 와  $\mathbf{A}$ 를 얻을 수 있다. 독립 요소 분석 기법은 식을 만족하는  $\mathbf{A}$ 와  $\mathbf{s}(t)$  중  $\mathbf{s}_i(t)$ 의 가우시안(Gaussianity)를 최소화 하는 방향으로  $\mathbf{s}(t)$ 의 해를 계산한다. 이는 독립적인 신호들이 조합되면 조합된 신호는 점점더 가우시안 분포를 가지기 때문에 각 신호의 가우시안성을 최소화 함으로써 각 신호가 상호 독립성을 가지게 된다는 특징을 이용하여 독립 요소를 추출하는 방법으로 이해할 수 있다. 따라서  $\mathbf{x}(t)$ 를 독립 요소 분석 기법을 적용하여 음원(source)를 획득하면 얼굴에 나타나는 독립적인 움직임을 추출할 수 있다.

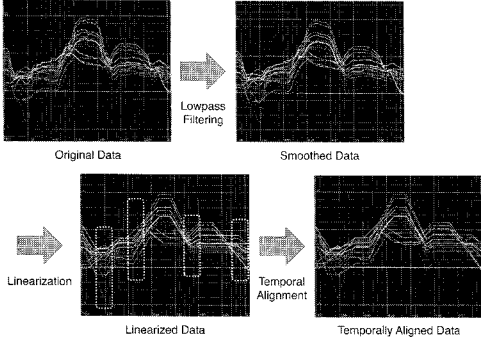


그림 2: 표정 정보에 독립 요소 분석 기법을 적용하기 위한 전처리 과정

### 3.3.1 전처리 및 독립 요소 분석

독립 요소 분석 기법을 사용하기 위해서는 입력 데이터에 몇 가지 가정이 필요하다. 먼저 독립 요소를 정확하게 분석하기 위해서는 상호 연관성이 있는 것으로 예상할 수 있는 움직임, 예를 들면 눈썹 주위의 점들의 움직임이 서로 선형적이어야 한다. 즉, 특정한 점의 움직임이 다른 특정한 점의 움직임의 배수로 나타나야 한다. 하지만 얼굴의 움직임의 비선형성 때문에 항상 정확한 배수 관계를 가지지 못한다. 또한 독립 요소 분석은 특정 점의 변위의 평균을 매개변수값 0에 대응시키는 반면 본 논문에서는 매개변수를 0으로 지정하면 무표정을 만들 수 있으면 보다 효과적으로 표정을 생성할 수 있다. 이와 같은 독립 요소 분석 방법의 요구사항에 따라 표정 데이터를 전처리하여 분석 결과를 보다 정확하게 할 수 있다.

먼저 입력 데이터의 평균 값을 0으로 만들기 위해 본 논문에서는 표정 데이터  $\mathbf{x}(t)$ 를 직접 사용하는 대신  $\mathbf{x}(t)$ 와  $-\mathbf{x}(t)$ 를 연결(concatenate)하여  $\mathbf{x}'(t)$ 를 만들고 이를 입력 신호로 한다. 이러한 방법을 이용하면 각 점점의 변위의 평균 값은 0이 되고 분석 후 매개변수 신호를 0으로 지정할 경우 무표정을 얻을 수 있다. 물론  $\mathbf{x}(t)$  대신  $\mathbf{x}'(t)$ 를 입력신호로 사용하였기 때문에 결과로 얻는 매개변수 신호 역시  $\mathbf{s}(t)$ 와  $-\mathbf{s}(t)$ 를 연결한  $\mathbf{s}'(t)$ 가 된다. 따라서 이 중 앞의 반만을 매개변수 신호로 사용하여 후술할 얼굴 변형 모델 학습에 사용한다.

상호 연관성이 있는 움직임의 선형성을 확보하기 위해 움직임을 선형화 하는 과정은 다음과 같다. 먼저 지역적으로 연관성 있는 움직임을 추출하기 위해 얼굴의 움직임의 흐름을 표정을 짓기 위해 움직이는 부분, 표정이 완성되어 일시적으로 멈춰있는 부분으로 구분한다. 이는 각 점점의 움직임의 속도를 측정하고 속도가 사용자가 지정한 임계 값 보다 작은 경우 멈춰있는 부분으로 판단하고 그렇지 않으면 움직이는 부분으로 판단하는 간단한 방법을 적용하여 해결한다. 이런 과정을 통하여 각 신호에서 정점이 움직이기 시작하는 시점과 멈추는 시점을 얻을 수 있고, 이런 시점의 다른 점들과 정렬하여 선형화된 신호를 얻을 수 있다.

이와 같이 전처리 된 신호에 독립 요소 분석 기법을 적용하면 기본 표정 집합  $\mathbf{A}$ 과 각각의 표정을 얻기 위해 사용되는 시간에 따라 변화하는 가중치  $\mathbf{s}(t)$ 를 얻을 수 있다. 이때  $\mathbf{s}(t)$ 는 얼굴의 움직임을 나타내는 신호로 볼 수 있다. 예를 들어 데이터 수집에 참여한 연기가가 얼굴의 모든 근육을 각각 움직일 수

있고 모든 움직임이 각각 데이터베이스에 포함되었다고 하면 각각의 움직임은 시간적으로 독립이기 때문에 각각의 근육을 제어하기 위한 신경 신호에 대응하는 값이  $\mathbf{s}(t)$ 에 나타나게 될 것이다. 하지만 일반적으로 사람은 각각의 근육을 독립적으로 움직일 수 없기 때문에  $\mathbf{s}(t)$ 의 각각은 동시에 움직이는 움직임 을 축약한 제어 신호로 볼 수 있다.

### 3.3.2 변형 모델 학습

전술한 바와 같이 식 (3)은 얼굴의 각 부분의 비선형적 움직임을 선형으로 근사한 결과이다. 따라서 독립 요소 분석을 통해 얻은  $\mathbf{A}$ 를 기본 움직임으로 직접 사용하면 비선형적인 움직임을 정확하게 모델링 할 수 없다. 따라서 본 논문에서는 각 매개변수의 변화에 따른 얼굴의 각 점점의 움직임을 비선형 함수로 나타내고 비선형 함수를 데이터베이스에 기반하여 학습함으로써 보다 자연스러운 얼굴의 움직임을 얻는다.

본 논문에서는 특정 매개변수  $s_i$ 에 대해 특정한 점점의 특정 축 방향의 움직임을 3차 베지어 곡선(Bezier curve)로 근사한다. 3차 곡선을 사용하는 이유는 먼저 선형 모델에 비해 복잡한 비선형 움직임을 표현할 수 있고, 복잡한 비선형 모델에 비해 빠르게 계산될 수 있다는 점이다. 3차 베지어 곡선으로 표현된 움직임은 다음과 같이 수식화 될 수 있다.

$$x_i = \sum_j (a_{ij}s_j^3 + b_{ij}s_j^2 + c_{ij}s_j). \quad (4)$$

이때  $x_i$ 는 점점의 특정한 자유도 방향으로의 변위를 나타내고  $a_{ij}, b_{ij}, c_{ij}$ 는 매개변수와 점점의 움직임을 연결하는 미지의 계수이다. 여기서 상수항을 배제한 이유는 매개변수의 값이 0일 때 무표정을 표현하기 위해서이다. 위의 식은 행렬 식의 형태로 다음과 같이 표현된다.

$$x_i = \begin{bmatrix} \vdots \\ \vdots \\ \vdots \end{bmatrix} \begin{bmatrix} a_{ij} \\ b_{ij} \\ c_{ij} \end{bmatrix} = \tilde{\mathbf{s}}_j \mathbf{c}_i \quad (5)$$

시간에 따라 변화하는  $x_i$ 와  $s_j$ 에 대하여 행렬식을 구성하면

$$\begin{bmatrix} \vdots \\ x_i(t-1) \\ x_i(t) \\ x_i(t+1) \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ \tilde{\mathbf{s}}_j(t-1) \\ \tilde{\mathbf{s}}_j(t) \\ \tilde{\mathbf{s}}_j(t+1) \\ \vdots \end{bmatrix} \mathbf{c}_i \quad (6)$$

$$= \mathbf{S} \mathbf{c}_i$$

을 얻을 수 있다. 이때  $\mathbf{S}$ 는 매개변수를 통한 제어 신호 행렬로 각 점점에 대하여 변화하지 않는 행렬이고  $\mathbf{c}_i$ 는  $i$ 번째 점점의 자유도에 해당되는 계수 벡터가 된다. 따라서 각각의 점점 자유도에 대하여 이 선형 방정식의 해를 최소 자승해(least squares solution)을 계산하면 계수 벡터를 얻을 수 있다. 이때  $\mathbf{S}$ 가 정점에 따라 변화하지 않기 때문에 특이 값 분해(singular value decomposition) 등의 방법을 적용하면 효율적으로 계수를 계산할 수 있다.



그림 3: 실험 표정 집합

## 4 실험 결과

본 연구에서는 표정 데이터 베이스를 구성하기 위해 연기자의 얼굴에 68개의 마커를 부착하고 표정을 포착하였다. 가급적 사람이 지을 수 있는 많은 수의 표정을 저장하기 위해 연기자는 FACS에 나타나 있는 각각의 표정을 짓도록 하였고, 30초간 임의의 표정을 연기하도록 하였다. 표정 데이터는 표정의 미묘한 움직임을 포착하기 위해 초당 120회를 측정된 결과를 사용하였다. 얻어진 3차원 마커 정보는 머리에 부착한 마커를 기준으로 목의 움직임에 의한 얼굴 전체의 움직임을 상쇄하여 정면을 바라보는 상태로 조절하였다. 그림 3에는 실험에 사용된 표정의 일부분을 보여준다.

이와 같은 데이터는 전술한 전처리 과정을 거쳐 독립 요소 분석에 적용되었다. 전처리 전에 마커 정보에 포함하여 있을 수 있는 미세한 잡음을 제거하기 위해 가우스 필터(Gaussian filter)를 적용하였다. 얼굴의 각 부분의 움직임의 크기가 부분에 부분적으로 매우 다르기 때문에 각 마커의 움직임을 마커의 최대 변위를 기준으로 정규화(normalize) 하였다. 전처리에서 표정 신호를 선형화 하기 위해 표정을 지은 상태에서 멈추어 있는 부분을 측정하기 위해 신호의 기울기를 계산하였다. 실험에서는 초당 0.6 밀리미터 이하로 움직이는 경우 정점이 거의 멈추어 있는 것으로 판단하고 선형화를 진행하였다. 독립 요소 분석을 수행하기 위해 본 논문에서는 수학 패키지인 *IT++*에 구현되어 있는 빠른 독립요소분석 알고리즘(FastICA algorithm)을 이용하였다 [29].

그림 4는 본 논문에서 제안하는 방법을 통해 얻은 15개의 기본 표정을 보여준다. 표정은 각각 왼쪽 상단에서부터 오른쪽으로 왼쪽 뺨을 올리는 표정, 왼쪽 눈을 찡그리는 표정, 오른쪽 뺨을 올리는 표정, 오른쪽 눈을 찡그리는 표정, 양쪽 눈을 감는 표정, 턱을 내리는 표정, 입을 오무리는 표정, 입꼬리를 당기는 표정, 코를 찡그리는 표정, 볼을 훌쩍하게 하는 표정, 입술을 내미는 표정, 아래 입술로 위 입술을 덮는 표정, 눈으로 웃는 표정, 눈을 크게 뜨는 표정, 눈썹을 올리는 표정으로 이름 붙일 수 있다. 이들 표정은 실제적으로 각각이 분명하게 의미 있는 경우도 있고, 그렇지 않은 경우도 있다. 이는 데이터에서 각각의 표정이 분명하게 구분되지 않았음을 시사한다. 하지만, 설명한 바와 같이 각각의 표정에 적절한 이름을 붙일 수 있

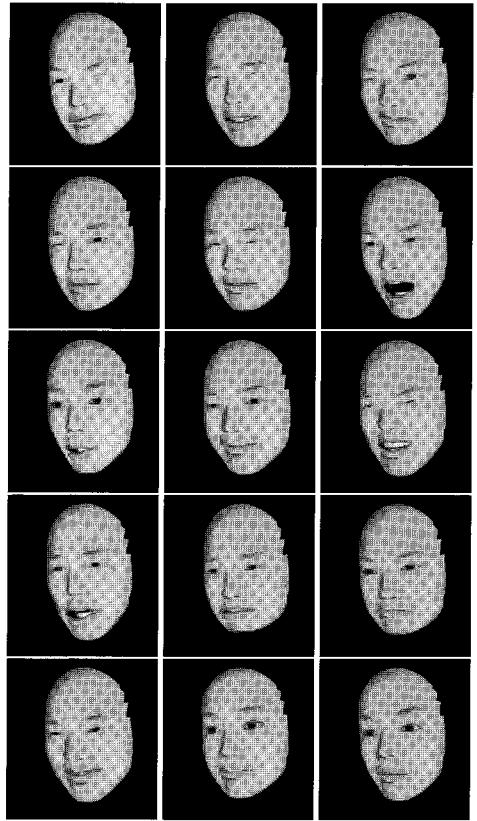


그림 4: 추출된 기본 표정 집합.

때문에 사용자는 매개변수의 조절에 대응되는 표정 변화를 어느 정도 예상할 수 있다.

본 연구에서는 독립 요소 분석의 결과의 완결성을 검증하기 위해 얻어진 독립 요소를 조합하여 데이터베이스 내의 표정을 재구성(reconstruction)하였다. 그림 5는 독립 요소의 개수를 다르게 하면서 독립 요소를 추출하고 이들로 데이터베이스 내부의 표정을 재구성 했을 때의 오차를 보여준다. 푸른 점은 모든 마커에 대하여 최대 오차를 보여주고, 붉은 점은 모든 마커의 평균 오차를 보여준다. 그래프에 나타난 바와 같이 10개 독립 요소를 추출한 경우에는 모든 정점이 3mm 이내의 오차를 보였고, 평균적으로 0.5mm 내의 오차를 가져 적은 수의 독립 요소로 대부분의 표정을 정확하게 생성할 수 있음을 알 수 있다.

그림 6에서는 본 논문에서 제안하는 비선형 변형 기법의 결과를 보여준다. 볼을 올리는 표정에서 볼은 먼저 위쪽으로 올라간다. 충분히 올라가 피부와 근육이 늘리면 서서히 볼의 살이 앞으로 나와 부푸는 것을 볼 수 있다. 실제로 본 논문에서 제안한 방법은 실제 표정의 변화를 독립성에 따라서 함축하여 매개변수화 한 것으로 볼 수 있고, 따라서 비선형성이 크게 나타나지 않는다. 비선형 변형 모델에 의한 결과도 수치적으로 선형 모델과 크게 다르지 않다. 하지만 사람의 얼굴 표정에서 이와 같은 미묘한 차이는 표정에 사실성을 부여하는데 중요한 요소라 할 수 있다.

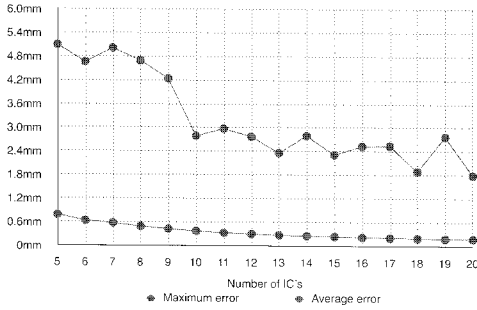


그림 5: 독립 요소의 수에 따른 재구성 오차

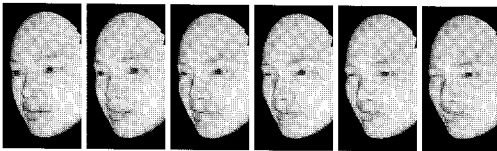


그림 6: 비선형 변형의 예.

그림 7은 매개변수의 값을 조절하여 표정을 생성한 결과를 보인다. 각각의 매개변수가 기능적, 직관적 움직임과 어느 정도 대응되기 때문에 간단한 의미 있는 표정을 쉽게 생성할 수 있었다. 특히 얼굴의 각 부분을 따로 움직이는 것이 아니라 얼굴의 주요 움직임에 따라 다른 부분의 미묘한 움직임이 자동으로 생성되기 때문에 사실적인 표정을 큰 어려움 없이 생성할 수 있다. 하지만 매개변수 각각은 시간적인 애니메이션에서 독립적인 움직임을 나타내지만 같은 부위의 움직임을 나타내는 경우가 있고, 실제로 특정한 기본 움직임은 동시에 나타나지 않는 경우가 있다. 따라서 이러한 조합을 매개변수로 지정할 경우 그림 8에서처럼 부자연스럽거나 물리적으로 불가능한 표정이 나타나기도 한다.

## 5 결론 및 향후 연구

본 논문에서는 실측 데이터를 기반으로 사람의 다양한 표정을 매개변수화 하기 위한 방법을 제안했다. 독립 요소 분석 방법을 이용하여 표정 데이터로부터 독립적인 움직임들을 나타내는 제어 신호를 추출하여 매개변수로 삼았다. 특히 얼굴에 나타나는 각 점점들의 움직임은 비선형적인 특징을 가진다는 점에 착안하여 독립 요소 분석을 효과적으로 수행하기 위한 전처리 과정과 매개변수에 따른 얼굴 움직임의 변화를 나타내기 위한 효율적인 비선형 모델을 제안하였다. 이와 같은 매개변수화 과정을 표정 포착으로부터 얻은 표정 데이터에 적용하여 매개변수와 기본 움직임을 추출한 결과 각각의 기본 움직임은 직관적으로 이해할 수 있는 수준으로 나타났고 특히 적은 수의 매개변수만을 이용하여 대부분의 표정을 적은 오차 범위에서 나타낼 수 있는 것을 보였다.

제안한 방법의 가장 중요한 문제점은 얻어진 매개변수의 특징이 주어진 데이터에 따라 다르다는 점이다. 이는 연기자에 따라 독립적으로 움직일 수 있는 근육의 종류와 조합이 다르기 때문으로 해석될 수 있다. 또한 사람에 따라서 동일한 표정에

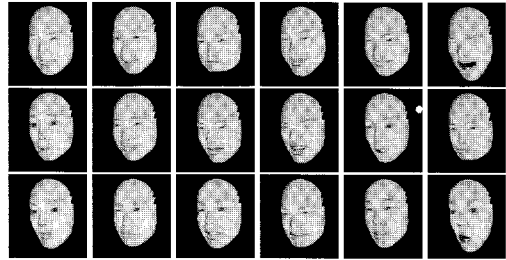


그림 7: 매개변수를 통해 생성한 표정



그림 8: 매개변수 조절에 의한 표정 생성의 나쁜 예

서도 각 정점이 움직이는 방향과 크기가 다르기 때문에 많은 사람의 표정을 동시에 처리하여 독립 성분 분석을 적용하는 것은 적절하지 않다. 따라서 일반적으로 많은 사람들의 일반적인 표정을 전체적으로 매개변수화 할 수 없다. 하지만 본 논문에서 제안하는 방법은 각각의 연기자의 특징과 가능한 표정을 적절하게 매개변수화 하는 목적으로 사용할 수 있다. 따라서 향후 연구에서는 일반적인 표정을 매개변수화 하기 위해 얼굴의 움직임을 부분별로 정규화하고 통합하여 일반적으로 사람이 취할 수 있는 표정의 전체를 매개변수화 하기 위한 연구가 필요하다.

마지막으로 본 논문에서 제안한 매개변수화 방법은 얼굴 전체를 움직이는 매개변수가 다수 나타난다. 이는 실제 표정 데이터에 이러한 움직임이 항상 동시에 나타난 것을 의미한다. 즉 데이터에서는 각각의 움직임이 독립적이지 않다는 것을 의미하고 실제 애니메이션에서도 이들을 동시에 움직이는 것이 자연스러울 수 있다는 점을 시사한다. 하지만 기존의 애니메이션들은 얼굴의 각 부분을 따로 움직여 전체적인 표정을 만드는 데 익숙하기 때문에 본 논문에서 제안하는 방법을 통해 얻은 매개변수가 적절하지 않을 수 있다. 따라서 사용자의 요구에 따라서 지역적인 표정과 지역적인 표정을 선택적으로 매개변수화 하기 위한 방법으로 확장하기 위한 연구가 필요하다.

## 감사의 글

본 연구는 2005-1학기 아주대학교 정작연구비와 방위사업청과 국방과학연구소(UD060048AD)의 지원으로 수행되었습니다.

## 참고문헌

- [1] P. Ekman and W. V. Friesen, "Facial action coding system : A technique for the measurement of facial movement," *Consulting Psychologist Press: Palo Alto*, 1978.
- [2] L. Torresani, D. B. Yang, E. J. Alexander, and C. Bregler, "Tracking and modeling non-rigid objects with rank con-

- straints,” in *2001 Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Dec. 2001, pp. 493–500.
- [3] B. Choe, H. Lee, and H.-S. Ko, “Performance-driven muscle-based facial animation,” *The Journal of Visualization and Computer Animation*, vol. 12, no. 2, pp. 67–79, May 2001.
- [4] P. Joshi, W. C. Tien, M. Desbrun, and F. Pighin, “Learning controls for blend shape based realistic facial animation,” in *2003 ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, Aug. 2003, pp. 187–192.
- [5] E. Chuang and C. Bregler, “Mood swings: expressive speech animation,” *ACM Transactions on Graphics*, vol. 24, no. 2, pp. 331–347, Apr. 2005.
- [6] V. Blanz and T. Vetter, “A morphable model for the synthesis of 3d faces,” in *Proceedings of SIGGRAPH 99*, ser. Computer Graphics Proceedings, Annual Conference Series, Aug. 1999, pp. 187–194.
- [7] L. Williams, “Performance driven facial animation,” in *Proceedings of SIGGRAPH 90*, 1990, pp. 235–242.
- [8] P. Kalra, A. Mangili, N. Magnenat-Thalmann, and D. Thalmann, “Simulation of facial muscle actions based on rational free form deformations,” A. Kilgour and L. Kjelldahl, Eds., vol. 11, no. 3, 1992, pp. 59–69.
- [9] H. Li, P. Roivainen, and R. Forcheimer, “3-d motion estimation in model-based facial image coding,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 6, June 1993, pp. 545–555.
- [10] I. Essa, S. Basu, T. Darrell, and A. Pentland, “Modeling, tracking and interactive animation of faces and heads: Using input from video,” in *Computer Animation '96*, June 1996, pp. 68–79.
- [11] M. Escher, I. Pandzic, and N. Magnenat-Thalmann, “Facial deformations for mpeg-4,” in *Computer Animation '98*, June 1998.
- [12] D. Fidaleo, J.-Y. Noh, T. Kim, R. Enciso, and U. Neumann, “Classification and volume morphing for performance driven facial animation,” in *International Workshop on Digital and Computational Video*, 2000.
- [13] L. Zhang, N. Snavely, B. Curless, and S. M. Seitz, “Spacetime faces: high resolution capture for modeling and animation,” *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 548–558, Aug. 2004.
- [14] J.-Y. Noh and U. Neumann, “Expression cloning,” in *Proceedings of ACM SIGGRAPH 2001*, ser. Computer Graphics Proceedings, Annual Conference Series, Aug. 2001, pp. 277–288.
- [15] D. Vlastic, M. Brand, H. Pfister, and J. Popović, “Face transfer with multilinear models,” *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 426–433, Aug. 2005.
- [16] F. I. Parke, “Parameterized models for facial animation,” *IEEE Computer Graphics & Applications*, vol. 2, no. 1, pp. 61–68, Nov. 1982.
- [17] Y. Lee, D. Terzopoulos, and K. Waters, “Realistic modeling for facial animation,” in *Proceedings of SIGGRAPH 95*, ser. Computer Graphics Proceedings, Annual Conference Series, Aug. 1995, pp. 55–62.
- [18] K. Waters, “A muscle model for animating three-dimensional facial expression,” in *Computer Graphics (Proceedings of SIGGRAPH 87)*, July 1987, pp. 17–24.
- [19] K. Waters and J. Frisbie, “A coordinated muscle model for speech animation,” in *Graphics Interface '95*, May 1995, pp. 163–170.
- [20] S. M. Platt, “Animating facial expressions,” in *Computer Graphics (Proceedings of SIGGRAPH 81)*, Aug. 1981, pp. 245–252.
- [21] E. S. Chuang, F. Deshpande, and C. Bregler, “Facial expression space learning,” in *10th Pacific Conference on Computer Graphics and Applications*, Oct. 2002, pp. 68–76.
- [22] E. Sifakis, I. Neverov, and R. Fedkiw, “Automatic determination of facial muscle activations from sparse motion capture marker data,” *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 417–425, Aug. 2005.
- [23] E. Sifakis, A. Selle, A. Robinson-Mosher, and R. Fedkiw, “Simulating speech with a physics-based facial muscle model,” in *2006 ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, Sept. 2006, pp. 261–270.
- [24] A. Hyvärinen and E. Oja, “Independent component analysis: Algorithms and applications,” *Neural Networks*, vol. 13, no. 4–5, pp. 411–430, 2000.
- [25] H. Mori and J. Hoshino, “Independent component analysis of human motion,” in *Proceedings of International Conference on Acoustics Speech and Signal Processing*, vol. VI, 2002, pp. 3564–3567.
- [26] Y. Cao, P. Faloutsos, and F. Pighin, “Unsupervised learning for speech motion editing,” in *2003 ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, Aug. 2003, pp. 225–231.
- [27] A. Shapiro, Y. Cao, and P. Faloutsos, “Style components,” in *Graphics Interface 2006*, June 2006, pp. 33–39.
- [28] D. Fidaleo and U. Neumann, “Analysis of co-articulation regions for performance-driven facial animation,” *Computer Animation and Virtual Worlds*, vol. 15, no. 1, pp. 15–26, 2004.
- [29] T. Ottosson and A. Piatyszek, “Welcome to IT++.” [Online]. Available: <http://itpp.sourceforge.net/>
- [30] G. A. Kalberer, P. Mueller, and L. V. Gool, “Speech animation using viseme space,” in *Vision, Modeling, and Visualization VMV 2002*. Akademische Verlagsgesellschaft Aka GmbH, Berlin, 2002, pp. 463–470.
- [31] B. Yin, K. Wang, S. Ma, and J. Guo, “General face animation expression based on ica,” no. 1, 2005, pp. 181–185.



〈저자 소개〉



송 지 혜

2007년 2월 아주대학교 미디어학부 (학사)  
2009년 2월 아주대학교 미디어학과 (석사)  
2009년 3월 ~ 현재 삼성전자 DMC 부문 사원



신 현 준

1995년 2월 한국과학기술원 전산학과 (학사)  
1997년 2월 한국과학기술원 전산학과 (석사)  
2002년 2월 한국과학기술원 전산학과 (박사)  
2002년 3월 ~ 2002년 8월 한국과학기술원 박사후연구원  
2002년 9월 ~ 2003년 12월 University of Wisconsin Visiting Scholar  
2004년 1월 ~ 2004년 2월 한국과학기술원 박사후연구원  
2004년 3월 ~ 2005년 2월 서울대학교 박사후연구원  
2005년 2월 ~ 현재 아주대학교 미디어학부 교수  
〈관심분야〉 컴퓨터그래픽스, 영상처리, 그래픽시스템 등