

비디오 입력을 이용한 UCC 컷아웃 애니메이션 생성 기법

UCC Cutout Animation Generation using Video Inputs

이윤진*, 양승재**, 김준호***
 아주대학교 미디어학부*, 블루홀 스튜디오**, 국민대 컴퓨터공학부***

Yunjin Lee(yunjin@ajou.ac.kr)*, Seung Jae Yang(zardeine@naver.com)**,
 Junho Kim(junho@kookmin.ac.kr)***

요약

본 논문에서는 비디오 입력으로부터 UCC 제작에 적합한 컷아웃 애니메이션을 생성하기 위한 방법을 제안한다. 본 논문의 시스템은 관절 구조 캐릭터를 기반으로 하여 사용자가 쉽게 관절 캐릭터를 생성할 수 있는 인터페이스를 제공하며 비디오 입력으로부터 생성된 관절 캐릭터의 동작 정보를 추출한다. 실제 컷아웃 애니메이션과 같이 보이도록 종이 효과 렌더링, 그림자 렌더링을 적용하여 관절 캐릭터의 애니메이션을 생성하였다. 본 논문의 시스템을 이용하여 사용자는 여러 가지 비디오 입력으로부터 다양한 캐릭터 동작을 가진 컷아웃 애니메이션을 생성할 수 있었다.

■ 중심어 : | 컷아웃 애니메이션 | 비사실적 렌더링 | 비디오 콘텐츠 | 관절 캐릭터 |

Abstract

We propose a novel non-photorealistic rendering technique which generates a cutout animation from a video for UCC. Our method consists of four parts. First, we construct an interactive system to build an articulated character. Second, we extract motions from an input video. Third, we transform motions of a character in order to reflect characteristics of cutout animations. Fourth, we render the extracted or transformed components in cutout animation style. We developed a unified system for a user to easily create a cutout animation from an input video and showed the system generated a cutout animation efficiently.

■ keyword : | Cutout Animation | Non-photorealistic Rendering | Video Contents | Articulated Character |

I. 서론

최근 디지털 카메라 및 디지털 캠코더와 같은 디지털 기기의 발달과 보급으로 영상 및 비디오 콘텐츠를 쉽게 생성할 수 있게 되었다. 뿐만 아니라, 정보통신기술의 발달로 고용량의 정보교환이 용이하게 됨에 따라 디지

털 영상 산업에서 비디오 콘텐츠가 차지하고 있는 비율이 점점 높아지고 있다. Youtube 등의 동영상 서비스를 통해 전 세계적으로 UCC (User Created Contents) 제작 붐이 일어나 비디오 콘텐츠 시장이 크게 성장하였으며 시장의 유망성 또한 높이 평가되고 있다 이에 따라, 비디오 콘텐츠 생성을 위해 다양하고 효과적인 기술에 대한 요구가

* 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임
 (No. 2009-0075105)

접수번호 : #110513-001

접수일자 : 2011년 05월 13일

심사완료일 : 2011년 06월 15일

교신저자 : 김준호, e-mail : junho@kookmin.ac.kr

높아져 관련 분야의 연구들이 활발히 이루어지고 있다.

비디오 입력은 애니메이션에 필요한 스토리, 캐릭터, 캐릭터의 움직임, 음향 등의 정보를 가지고 있다. 이와 같은 요소들을 적절하게 추출함으로써 새로운 비디오 콘텐츠를 만들 수 있는 기술은 다양한 콘텐츠를 생성하고자 하는 사용자에게 유용하다. 본 논문에서는 새로운 비디오 콘텐츠 제작을 위한 비사실적 렌더링 기법(non-photorealistic rendering)으로, 컷아웃 애니메이션(cutout animation) 생성 방법을 제시한다. 컷아웃 애니메이션은 종이와 같은 재료에서 오려 낸 평면적인 캐릭터들을 배경 위에 배열해 조금씩 움직이면서 동작을 표현하고 이를 한 프레임씩 촬영하여 애니메이션을 만드는 기법이다 [1]. 컷아웃 애니메이션은 일반적인 풀 애니메이션(full animation)에 비해 제작 과정이 단순하면서도 종이, 천 조각, 신문 등의 다양한 재료와 조명, 여러 가지 동작표현 기법에 따라 세밀하고 충분한 묘사가 가능해 상업 애니메이션 뿐 아니라 초보 애니메이터들을 위한 기초학습용 애니메이션 제작에 이르기까지 널리 이용되고 있다 [1]. 특히, 주변의 재료와 간단한 움직임만으로 애니메이션 제작이 가능하고 그림을 그리고 종이를 오리는 작업으로 어린이가 흥미를 가지고 쉽게 따라 할 수 있어 학교 수업 시간이나 어린이를 위한 애니메이션 워크숍 등에서 많이 활용된다.

본 논문에서는 비디오 입력으로부터 컷아웃 애니메이션을 생성하는 비사실적 렌더링 기법을 위해 다음과 같은 문제를 해결하였다.

- 컷아웃 애니메이션에 사용되는 관절 캐릭터 생성 방법: 컷아웃 애니메이션의 특징은 관절 캐릭터를 이용한 흰정 애니메이션 [2]이다. 사용자가 생성할 컷아웃 애니메이션에 등장하게 될 관절 캐릭터를 정의하기 위한 방법이 필요하다.
- 비디오 입력으로부터 캐릭터 동작을 추출하는 방법: 비디오 입력에서 캐릭터의 애니메이션을 위한 동작을 추출한다. 모든 프레임마다 사용자가 지정하여 동작을 추출하기 어려우므로 키 프레임에서만 정의하고 키 프레임 사이는 캐릭터의 동작에 따라 이를 적절히 추출하는 방법이 필요하다.
- 컷아웃 애니메이션의 특성에 맞게 캐릭터의 동작을

변형, 생성하는 방법: 비디오 입력에서 추출된 동작은 스톱 모션 기법으로 만들어진 컷아웃 애니메이션의 동작에 비해 매우 부드럽게 변한다. 따라서 추출된 동작을 컷아웃 애니메이션의 특성이 나타나도록 변형해야 한다.

- 컷아웃 애니메이션 기법을 모방한 비사실적 렌더링을 적용하는 방법: 컷아웃 애니메이션은 소재와 기법에 따라 여러 가지 재질적 특성을 드러내는 애니메이션 기법이다. 따라서 추출된 캐릭터를 비사실적으로 렌더링하는 방법이 필요하다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 관련 연구로 기존의 비디오 입력 기반의 비사실적 렌더링 기법들과 비디오 움직임 추적기법을 살펴본다. 3장에서는 제시된 컷아웃 애니메이션 생성 방법의 전체 과정을 간략히 설명하고 4, 5, 6장에서 각 단계를 자세히 설명한다. 7장에서 본 방법을 통해 생성된 결과를 보이며 마지막으로 8장에서 결론 및 향후 연구 계획을 논한다.

II. 관련 연구

1. 컷아웃 애니메이션 연구

컷아웃 애니메이션을 생성하는 연구로 Video puppetry [3]가 있다. 이 방법은 비디오 기반 인터페이스를 이용하여 미리 만들어진 캐릭터의 움직임을 실시간으로 추적하여 애니메이션을 생성하였다. 사용자가 컷아웃 애니메이션에서 사용되는 애니메이션 방법을 직접 실행할 수 있어 컷아웃 애니메이션의 특성을 잘 나타낼 수 있다. 그러나 주어진 비디오 입력으로부터 새로운 콘텐츠를 생성하기는 어렵다. Video tooning [4]과 SnakeToonz [5]는 비디오 입력으로부터 셀 애니메이션 혹은 컷아웃 애니메이션과 유사한 애니메이션을 생성하였다. 이 연구기법들은 주로 비디오 입력의 렌더링 방식을 바꾸어 새로운 형태의 콘텐츠를 생성하는 방법이다. 따라서 툴 렌더링이나 수채화 렌더링과 같은 일종의 비디오 필터화가 가까우며 캐릭터의 움직임이 사실적이기 때문에 컷아웃 애니메이션의 특징적인 동작을 보여 주기는 어렵다.

2. 비디오 동작 추출 연구

컴퓨터 비전 분야에서는 오래 전부터 동영상에서의 물체를 추적하는 기법을 연구해왔다. 동영상의 물체를 추적하는 기법은 폭 넓게 응용되어 실용적으로 널리 사용되고 있다. Kanade 등이 제안한 KLT 추적기법 [6]은 오늘날 가장 폭넓게 사용되는 추적 기법 중 하나이다. 이 방법은 동영상 내에서 광 흐름을 찾기 위한 방법으로, 특징 지점 검출(헤리스 코너 추출기 [7])을 이용한다. 입력 데이터로는 영상을 흑백 영상으로 변환하여 사용하게 되며 비교적 빠른 계산 시간과 세밀한 검색으로 실시간 응용 프로그램에서 사용하기에 적합한 방법이다. Lowe가 제안한 SIFT [8] 기법은 다른 영상들 간에 동일한 일부의 영상을 찾아내는 기법이다. 영상의 확대축소나 회전 왜곡에도 둔감하게 작용하기 때문에 증강현실(augmented reality) 분야에서 바코드 인식 등 영상 인식에서 많이 쓰인다. 하지만 계산 시간이 상대적으로 느리다는 단점을 가지고 있어 실시간 응용 환경에서는 제한적으로 사용되고 있다.

III. 캐릭터 생성

본 연구의 방법은 크게 네 단계로 이루어진다. 비디오 입력이 주어지면 먼저 애니메이션에 사용될 관절 캐릭터를 생성한다. 두 번째 단계에서는 생성된 관절 캐릭터에 적용될 동작 정보를 비디오로부터 추출한다. 비디오 입력 영상에서 모든 동작 정보가 의미가 있는 것이 아니므로 사용자가 관절 캐릭터와 비디오 영상 내의 캐릭터 및 물체 사이의 연결 관계를 정하면 이를 기반으로 동작 정보를 획득하게 된다. 추출된 동작 정보를 컷아웃 애니메이션의 특징이 나타나도록 동작을 변형한다. 네 번째 단계에서는 생성된 애니메이션을 비사실적 스타일로 렌더링한다.

1. 캐릭터 추출 및 생성

컷아웃 애니메이션에서 캐릭터는 한정 애니메이션 [2]이 가능한 관절 캐릭터로 생성이 된다. 관절 캐릭터는 관절 조각들과 이들 사이의 연결 관계로 정의할 수

있다. 따라서 본 논문의 방법에서도 관절 조각 생성과 연결 관계 정의의 두 과정으로 관절 캐릭터를 생성한다. 관절 조각 생성 과정과 연결 관계 설정 과정을 분리하면 같은 연결 구조를 가지면서 다른 관절 조각들로 이루어진 캐릭터를 쉽게 생성할 수 있다.

관절 조각은 이차원 영상으로 정의되며 관절 조각 영역 외의 부분은 투명하게 만들어진다. 이와 같은 관절 조각 영상은 생성하고자 하는 컷아웃 애니메이션에 따라 [그림 2]와 같이 아티스트에 의해 이미 관절 캐릭터를 이룰 수 있도록 만들어진 영상을 사용하거나 [그림 1]과 같이 비디오 영상이나 이차원 영상으로부터 영상 분할(image segmentation) 방법 [9]을 이용하여 영상을 추출함으로써 손쉽게 얻을 수 있다. 두 번째 방법은 플라쥬 스타일의 컷아웃 애니메이션을 생성할 때 이용될 수 있다.



그림 1. 영상 분할 방법에 의해 생성된 관절 조각 영상

한정 애니메이션을 위해서는 각 관절 간의 연결 관계 및 계층 관계를 정해야 하므로 이를 위해 “Video puppetry” [3]와 유사한 방법을 이용하였다. [그림 2]와 같이 결정된 관절 조각 영상들을 불러 오면 먼저 루트가 될 영상 조각을 선택하고 조각들 사이의 연결 관계를 drag-and-drop으로 결정하면 이로부터 트리 구조가 생성된다. 이때, 각 관절 조각은 부모 조각과의 연결 위치를 가지게 된다.

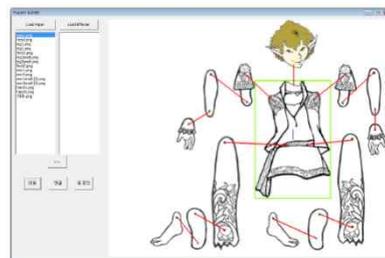


그림 2. 연결 관계 설정 시스템

IV. 비디오 입력으로부터 동작 추출

1. 캐릭터 동작 정의

관절 캐릭터의 동작은 관절 조각들의 계층 구조를 이용하여 재귀적으로 정의된다. 루트를 제외한 관절 조각은 부모 관절 조각과의 연결점과 회전각을 가지게 되며 이를 이용하여 관절 캐릭터의 동작을 정의할 수 있다. 즉, 관절 조각의 연결점의 위치는 바뀌지 않으므로 관절 캐릭터의 애니메이션은 [그림 3]과 같이 루트의 위치와 각 관절 조각의 회전각만으로 정의할 수 있다.

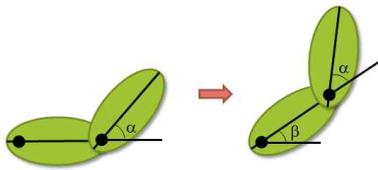


그림 3. 캐릭터 동작 정의

2. 추출 정보 정의

루트의 위치는 애니메이션 장면에서 캐릭터의 중심 위치를 나타내고 관절 조각의 회전각은 캐릭터의 세부적인 동작을 나타내게 된다. 루트의 위치 변화는 캐릭터의 중심이 될 위치를 비디오 영상 내의 한 특징점에 대응시켜 이 점이 이동될 위치를 추적함으로써 구할 수 있다. 관절 조각의 움직임도 관절 조각의 특정한 점을 비디오 영상 내의 위치에 대응시켜 구할 수 있다. 그러나 캐릭터를 구성하는 관절 조각의 모양이 비디오 영상 내에 대응될 물체의 모양과 많이 다를 경우 대응시키기가 어렵다. 따라서 관절 조각의 특정한 점을 비디오 영상 내의 특징점과 대응시키는 대신 관절 조각의 회전 정보를 추출하였다.

회전각 정보를 구하기 위해 먼저 [그림 4(a)]와 같이 비디오 영상 내의 특징점들을 정의한다. 두 개의 특징점을 연결함으로써 하나의 선분을 정의할 수 있으며 프레임 변화에 따라 특징점의 위치가 변하게 되면 그에 따라 선분의 방향이 달라진다. 따라서 비디오 영상 내의 두 특징점들을 하나의 관절 조각에 대응시킴으로써 선분의 방향으로부터 관절 조각의 회전각을 구할 수 있

고 이를 적용함으로써 [그림 5]와 같이 관절 조각을 움직일 수 있게 된다. 이때, 관절 조각들은 트리 형태의 연결 관계를 가지므로 관절 조각과 연결될 특징점들도 [그림 4(b)]와 같이 트리 형태의 계층 구조를 가지게 된다. 그러나 모든 관절 조각이 반드시 회전 정보를 가질 필요는 없으므로 특징점의 계층 구조와 관절 조각의 계층 구조가 일치하지 않을 수 있다.

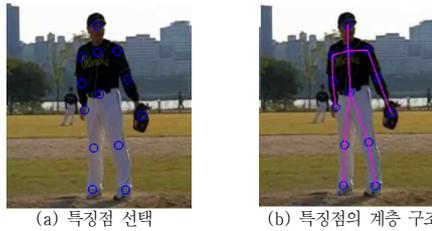


그림 4. 특징점 선택과 연결 관계 설정

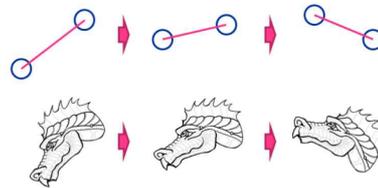


그림 5. 특징점 추적으로부터 계산된 회전각 정보 적용

3. KLT 추적 방법을 이용한 특징점 추적

캐릭터의 형태나 생성하고자 하는 동작에 따라 비디오 입력 내에서 추출하고자 하는 동작 정보가 달라지고 추출하고자 하는 동작 정보는 특징점에 의해 정의되므로 특징점은 사용자의 입력을 통해 정한다. 모든 프레임에 대해 특징점의 위치를 지정하는 대신 일정 프레임마다 지정하고 그 사이의 특징점의 위치 이동은 모션 추적 방법 [6][8][10]을 사용하여 구한다. 비디오 입력에서의 특징점의 위치를 추적하기 위해 여러 가지 방법이 사용될 수 있으나 그 중에서 Kanade-Lucas-Tomasi 특징점 추적(Kanade-Lucas-Tomasi feature tracking) 방법 [6](이하 KLT 추적 방법)은 광 흐름(optical flow)을 이용한 방법으로 비교적 빠른 시간에 특징점의 위치를 추적할 수 있어 실시간 응용 프로그램에서 많이 사용되는 방법이다. 그러나 비디오 입력에서 특징점은 프

레이미의 변화에 따라 노이즈나 조명의 변화에 의해 색상이 바뀔 수 있어 항상 추적 정보를 잘 얻기는 힘들다.

본 논문에서는 하나의 추적기(tracker) 대신 여러 개의 추적기를 사용하여 KLT 추적 방법의 성능을 개선하였다. 사용자가 비디오 영상 내에서 특징점을 선택하면 [그림 6]과 같이 선택된 점을 중심으로 원을 정의하고 그 내부에서 균일하게 점을 샘플링하여 샘플링된 점마다 추적기를 생성하였다. 그런 다음 생성된 여러 개의 추적기의 위치로부터 평균을 구하여 특징점의 이동된 위치를 결정하였다. 한 특징점만을 추적할 경우, 그 특징점의 추적이 실패할 경우, 사용자가 이를 보정해야 하지만 주변의 여러 특징점을 고려하는 경우, 추적 실패의 가능성을 낮출 수 있으며 이동 위치 또한 한 점을 추적하는 경우에 비해 노이즈에 덜 민감하여 더 정확하게 계산할 수 있다.

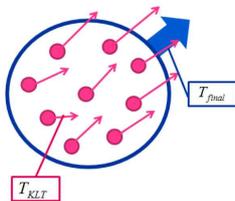


그림 6. 여러 개의 추적기를 이용한 특징점의 이동

보다 안정적인 위치 추적을 위해 각 추적기의 위치에 여러 가지 가중치를 주어 실험을 하였다. 이때, 원의 중심으로부터 거리에 반비례하여 가중치를 주거나 색상 정보, 추적기의 이동 방향 벡터를 이용하였다. 실험 결과, 추적기가 이동된 위치에서의 색상 정보와 추적기의 이동 방향 벡터가 많이 다른 추적기를 제외한 후, 제외되지 않은 추적기에 대해 동일한 가중치를 주어 계산한 평균 이동 위치가 가장 좋은 성능을 나타내었다. 이는 사용자가 선택한 특징점에 의해 정의되는 선분은 면적을 가진 물체의 움직임을 표현하는 것이므로 사용자가 선택한 한 특징점의 위치에 가중치를 크게 주는 것보다 다른 점에 대해서도 동일하게 주는 것이 더 좋으며 색상 정보를 이용하여 추적기를 제외한 것은 원의 영역을 특징점 주변의 동일한 색상의 가진 영역으로 바꾼 것과

동일하다. 이동 방향 벡터에 의한 고려는 추적기가 노이즈 및 색상 변화의 영향으로 잘못된 위치로 이동되었을 때 제거해 주는 역할을 한다.

4. 추적 실패 검출

여러 개의 추적기를 이용하여 추적 성능을 향상시킬 수 있으나 선택된 특징점이 다른 물체에 의해 가려지거나 그림자 등으로 갑작스러운 색상 변화가 발생할 경우, 특징점 추적은 실패하게 되며 사용자가 이를 수정해야 한다. 이를 위해 특징점 추적의 실패를 사용자가 때 프레임 확인할 필요 없이 알 수 있도록 추적 위치가 잘못되지 않았는지 검증하는 방법을 추가하였다. 이는 mean shift 추적 방법 [10]에서 사용된 HSV 색상 히스토그램을 이용한 것으로 처음 특징점이 정해진 프레임에서 특징점에 의해 정의된 원 내부의 영역에 대해 색상 정보로부터 히스토그램을 만들고 이를 매 프레임 이동된 추적기의 색상 히스토그램과 비교하여 차이가 많이 나는 경우, 추적이 잘못되었다고 판단하고 사용자에게 알리게 된다. 이때, 명도는 같은 지점이라 할지라도 조명에 따라 민감하게 변하므로 HSV 중 H와 S만을 이용하여 히스토그램을 생성하여 비교한다.

5. 사용자에게 의한 수정

특정점 추적이 실패하거나 사용자가 동작 정보를 수정하고 싶은 경우, 특징점의 위치를 직접 이동시킴으로써 동작 정보를 수정할 수 있다. 이때, 한 점만 이동시키는 경우, [그림 7](a)와 같이 회전각의 변화가 불연속적이 되어 동작 변화가 부드럽게 되지 않는다. 따라서 이동된 점을 중심으로 가우스 함수를 이용하여 이동된 값에 가중치를 주어 주변 프레임의 회전각에 더해 주어 [그림 7](b)와 같이 회전각이 부드럽게 변화하도록 하였다.

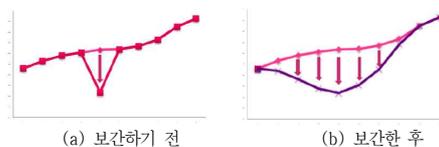


그림 7. 사용자에게 의한 회전각 정보 수정

V. 동작 변형 및 렌더링

1. 동작 변형

비디오 영상 내의 물체는 일반적으로 연속적인 움직임 가지므로 특징점 추적을 이용하여 매 프레임마다 추출된 회전각 정보는 [그림 8]과 같이 부드러운 곡선 형태를 가지게 된다. 이때, 그래프의 가로축은 프레임, 세로축은 회전각의 값이다. 이 동작을 관절 캐릭터에 적용하는 경우, 부드럽게 변하는 동작 애니메이션을 얻을 수 있다. 그런데 전통적인 컷아웃 애니메이션은 스톱 모션 애니메이션의 일종으로 관절 캐릭터의 각 부분을 조금씩 움직이고 이를 촬영한 후 촬영된 영상을 연속적으로 보여 줌으로써 애니메이션을 생성한다. 초당 프레임 수가 높으면 동작이 부드러워 보이며 반면에 초당 프레임 수가 적으면 일정 시간 동안 동일한 동작이 유지되고 프레임 간의 동작이 덜 부드러워 보인다. 본 연구에서는 이와 같은 스톱 모션의 특성을 보여 주기 위해 연속적인 회전각 값을 샘플링한 후, 샘플링된 점을 박스 필터(box filter)로 보간하여 회전각의 값을 바꾸어 동작을 변형하였다. 회전각 정보를 동일한 간격으로 샘플링할 수 있으나 회전각의 변화가 많이 일어나는 경우 샘플링 간격이 너무 크면 동작의 잘 표현하지 못 할 수 있으며 반대로 회전각의 변화가 적은 경우 샘플링의 간격이 작으면 샘플링 후에도 동작이 부드러워 보인다. 이를 개선하기 위해 회전각 정보를 분석하여 지역적으로 최소점과 최고점을 찾아 회전각 정보를 분할 후 분할된 영역별로 단위 프레임 당 회전각 정보의 변화량을 계산하여 이를 반영하여 샘플링 간격을 조절하였다. [그림 9]는 이와 같은 샘플링 방법을 이용하여 변형된 회전각 정보의 그래프이다. [그림 9]에서 알 수 있듯이 회전각의 변화가 적은 부분은 샘플링 간격이 크며 회전각의 변화가 큰 부분에서는 상대적으로 샘플링 간격이 작은 것을 알 수 있다.

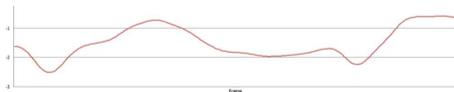


그림 8. 동작을 변형하기 전

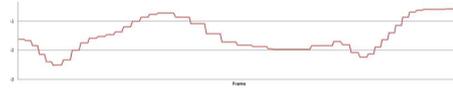
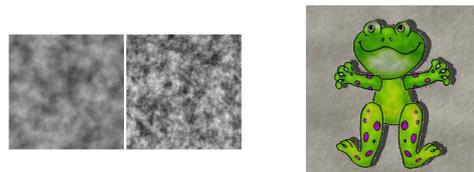


그림 9. 동작을 변형한 후

샘플링을 통한 동작 변형 외에 노이즈값을 회전각에 더하여 동작에 미세한 변화를 더해 주는 기능도 추가하였다. 이 두 가지 방법의 동작 변형은 사용자에 의해 선택이 가능하며 생성하고자 하는 컷아웃 애니메이션의 스타일에 따라 적용될 수 있다.

2. 합성 및 렌더링

동작 정보가 얻어지면 이를 캐릭터에 적용하여 렌더링 함으로써 최종적인 애니메이션을 얻게 된다. 아티스트에 의해 생성된 관절 조각의 경우, 이미 비사실적 스타일로 생성되어 있는 경우가 많으므로 특별한 렌더링을 적용하기보다 재질적 특성이 나타나도록 하였다. 컷아웃 애니메이션의 캐릭터는 종이를 이용하여 만들어지는 경우가 많으므로 관절 조각이 종이처럼 보이도록 Perline [11]의 방법을 이용하여 노이즈 텍스처를 적용하였다. 실제 관절 조각은 서로 겹쳐지면서 그림자가 생기며 이 그림자가 캐릭터에게 입체감을 줄 수 있다. 그림자는 물체가 평면에 투사되어 만들어지므로 관절 조각을 조명 방향에 따라 확대 이동하여 간단히 생성하였다. [그림 10]은 종이 효과를 주기 위해 사용된 텍스처와 종이 효과와 그림자 효과를 동시에 적용해서 캐릭터를 최종적으로 렌더링한 결과이다.



(a) 종이 효과 텍스처

(b) 종이 효과와 그림자 효과

그림 10. 합성 및 렌더링 효과

VI. 실험 결과

본 논문의 방법으로 제작된 시스템은 C++, MFC,

OpenCV, OpenGL과 GLSL을 사용하여 구현되었으며 실험은 Intel Quad Core CPU 2.33GHz 4GB RAM의 PC에서 수행되었다. 본 논문의 방법은 크게 세 부분의 시스템으로 구성이 되었다. 첫 번째 부분은 캐릭터 생성 시스템으로 [그림 2]와 같이 캐릭터 생성을 위한 인터페이스를 제공하며, 두 번째 부분은 동작 추출 및 변형 시스템으로 [그림 4]와 같이 특징점의 선택을 위한 인터페이스와 동작을 추출하면 [그림 8]의 그래프를 생성하고 [그림 7]과 [그림 9]와 같은 동작 변형을 할 수 있는 인터페이스를 제공한다. 세 번째 부분은 렌더링 시스템으로 최종 결과를 확인할 수 있으며 종이 효과 텍스처를 선택할 수 있는 인터페이스를 제공한다.

[그림 11]과 [그림 12]의 결과는 종이와 그림자 효과 렌더링을 모두 사용하여 생성되었다. [그림 11]은 동일한 비디오 입력을 이용하여 인간형 캐릭터와 개구리 캐릭터의 애니메이션을 생성한 결과이다. 관절 캐릭터의 모양과 길이가 달라도 적절하게 동작 애니메이션이 생성된 것을 볼 수 있다. [그림 12]는 얼룩말이 등장하는 비디오 입력을 이용하여 유니콘 캐릭터의 애니메이션을 생성하였다. 모든 관절 조각을 움직이지 않고 유니콘 캐릭터의 머리와 목 부분에만 특징점 추적기를 대응시켜 애니메이션을 생성하였다. 캐릭터 생성 시간은 생성하고자 하는 캐릭터의 관절 개수에 비례하며 [그림 11]의 인간형 캐릭터의 경우, 2분 정도 걸렸으며 개구리 캐릭터의 경우 1분 안에 생성되었다. 사용된 특징점의 개수는 9개로 특징점을 설정하는 시간은 30초 이내로 걸렸으며 특징점 추적 시간은 한 프레임 당 평균 0.5초로 사용자와 상호 작용이 가능한 속도이다.

VII. 결론

본 논문에서는 비디오 입력을 이용한 컷아웃 애니메이션 제작 시스템을 제안하였다. 비디오 입력을 이용하여 캐릭터 동작 정보를 얻을 수 있어, 사용자가 직접 캐릭터를 움직여서 동작을 생성해야 했던 방법에 비해 사용자가 쉽게 컷아웃 애니메이션을 제작할 수 있다. 또한, 사용자가 쉽게 수정이 가능하도록 사용자 친화적인

인터페이스를 제공하였다.

본 논문에서는 캐릭터의 동작이 복잡한 경우, 이차원 상의 회전으로만 동작을 표현하는데 한계가 있어 동작 추출의 정확성이 떨어진다는 문제가 있다. 이는 사용자의 수정으로 보완될 수 있으나 향후 연구에서 복잡한 동작 추출에 대한 연구가 필요하다. 특히, 특징점 추적 결과는 보다 최신의 특징점 추적 기법들 [12][13]을 이용하여 개선될 수 있으며 최근에는 GPU를 기반으로 한 실시간 추적 기법 [14][15]도 연구되어 이를 이용하면 동작 추출 속도를 개선할 수 있을 것으로 기대된다. 또한 미리 제작하거나 단순히 이미지 분할을 이용한 캐릭터 제작과 같이 사용자의 입력에 크게 의지하고 있는 캐릭터 생성 방법보다 비디오 입력에서 캐릭터를 보다 효율적으로 추출하기 위한 연구가 필요하다.

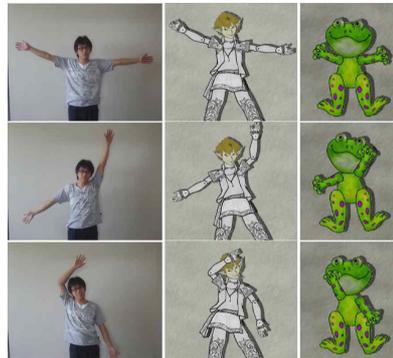


그림 11. 컷아웃 애니메이션 생성 결과

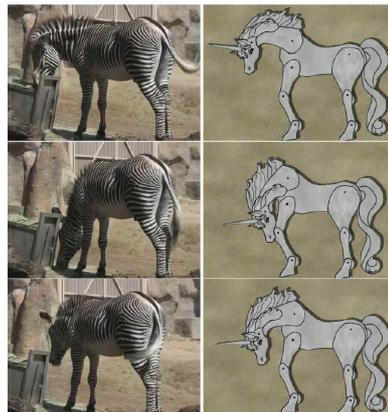


그림 12. 컷아웃 애니메이션 생성 결과

참 고 문 헌

- [1] 손영선, "유리 놀슈테인(Yuri Norstein)의 Cut-out Animation에 관한 연구", 석사논문, 숙명여대 디자인대학원, 2006.
- [2] 오송이, "컷아웃 애니메이션 제작 기법에 관한 작품 연구: 단편 애니메이션 작품 <在자re>를 중심으로", 석사논문, 세종대 공연예술대학원, 2005.
- [3] C. Barnes, D. E. Jacobs, J. Sanders, D. B. Goldman, S. Rusinkiewicz, A. Finkelstein, and M. Agrawala. "Video puppetry: a performative interface for cutout animation", SIGGRAPH ASIA, 2008.
- [4] J Wang, Y. Xu, H. Shum, and M. F. Cohen, "Video tooning," SIGGRAPH, 2004.
- [5] A. Agarwala, "SnakeToonz: a semi-automatic approach to creating cel animation from video," NPAR, 2002.
- [6] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," In Proc. 7th international joint conference on Artificial intelligence, pp.674-679, 1981.
- [7] C. Harris and M. J. Stephens, "A combined corner and edge detector," Alvey Vision Conference, pp.147-152, 1988.
- [8] D. G. Lowe, "Object recognition from local scale-invariant features," IEEE International Conference on Computer Vision, p.1150, 1999.
- [9] E. N. Mortensen and W. A. Barrett, "Intelligent scissors for image composition," SIGGRAPH, pp.191-198, 1995.
- [10] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," In Proc. IEEE Computer Vision and Pattern Recognition, p.2142, 2000.
- [11] K. Perlin, "An image synthesizer," SIGGRAPH, Vol.19, No.3, pp.287-296, 1985.
- [12] C. Tomasi and S. Jianbo, "Good features to

track," In Proc. IEEE Computer Vision and Pattern Recognition, pp.593-600, 1994.

- [13] S. B. Kang, R. Szeliski, and H. Y. Shum. "A parallel feature tracker for extended image sequences," Computer Vision and Image Understanding, Vol.67, No.3, pp.296-310, 1997.
- [14] J. M. Ready and C. N. Taylor, GPU Acceleration of Real-time Feature Based Algorithms, IEEE Workshop on Motion and Video Computing, p.8, 2007.
- [15] S. N. Sinha, J.-M. Frahm, M. Pollefeys, and Y. Genc, "GPU-based Video Feature Tracking And Matching," IEEE Transactions on Reliability, 2006.

저 자 소 개

이 윤 진(Yunjin Lee)

정회원



- 1999년 2월 ~ 2005년 8월 : 포스텍 컴퓨터공학과(공학사, 공학박사)
- 2005년 9월 ~ 2006년 5월 : 포스텍 박사후 연구원
- 2006년 5월 ~ 2007년 6월 : 미시간 대학교 박사후 연구원

- 2007년 6월 ~ 2007년 10월 : 포스텍 박사후 연구원
- 2007년 10월 ~ 2008년 2월 : 서울대학교 BK21 연구교수
- 2008년 3월 ~ 현재 : 아주대학교 미디어학부 조교수 <관심분야> : 컴퓨터 그래픽스, 비사실적 렌더링, 디지털 형상처리

양 승 재(Seung Jae Yang)

정회원



- 2008년 2월 ~ 2010년 2월 : 아주대학교 미디어학부(공학사, 공학석사)
- 2010년 3월 ~ 현재 : 블루홀 스튜디오

<관심분야> : 컴퓨터 그래픽스, 비사실적 렌더링

김 준 호(Junho Kim)

정회원



- 1998년 2월, 2000년 2월, 2005년 2월 : 포스텍 컴퓨터공학과(공학사, 공학석사, 공학박사)
- 2005년 3월 ~ 2005년 10월 : 포스텍 박사후 연구원
- 2005년 11월 ~ 2008년 2월 : 뉴욕주립대(스토니브룩 소재) 포스트닥 연구원

- 2008년 3월 ~ 2009년 2월 : 동의대학교 게임공학과 전임강사
- 2009년 3월 ~ 현재 : 국민대학교 컴퓨터공학부 전임강사

<관심분야> : 컴퓨터 그래픽스, 실시간 렌더링, 디지털 영상처리