

다중 카메라 촬영 영상을 이용한 Object Movie 생성

최유주¹, 유효선^{0,1}, 송창용², 남윤영³

¹서울벤처정보대학원대학교, ²(주)뷰티비지, ³유비쿼터스시스템연구센터
¹yjchoi@suv.ac.kr, ¹babydigital@hanmail.net, ²songchyo@paran.com, ³youngman@ajou.ac.kr

Object Movie Construction using Images from Multi-Camera

Yoo-Joo Choi¹, Hyo-Sun You^{0,1}, Chang-Yong Song², Yunyoung Nam³

¹Seoul University of Venture and Information, ²Beauty visagism, ³

요 약

본 논문에서는 다수의 저가형 웹캠(Web Cam)으로 촬영된 다각도의 다중 영상을 입력으로 받아 이들을 자연스럽게 연결하여 임의의 각도에서 물체의 모습을 관찰할 수 있도록 하는 「다중 카메라 촬영 영상을 이용한 Object Movie 생성 기반 기술」을 제안한다. 기존 Object Movie 생성도구들이 다중의 이미지들을 스티칭(stitching)하기 위해 정확한 카메라의 위치와 방향을 요구하는 데 비해, 제안된 방법은 고가의 트랙장비를 사용하기 어려운 경우에 다수의 저가형 웹캠으로 촬영된 카메라의 위치와 방향을 보정하는 단계를 추가하여 트랙장비를 사용한 것과 같은 매끄러운 영상을 생성할 수 있도록 하였다.

1. 서 론

Object Movie는 3차원 공간상의 객체를 모든 각도에서 살펴 볼 수 있는 방법을 제공한다. 사용자 주위의 장면에 대한 넓은 시야를 제공하기 위해서 카메라를 삼각대에 고정시킨 다음 수평방향으로 회전시켜 얻어진 영상을 이용하여 이어진 영상으로 표현하는 파노라마 무비와는[1][2] 반대로, Object Movie에서 카메라는 피사체 주변의 모든 위치에서 동등하게 하나의 중심점(피사체에 있는)에 초점이 맞춰져 있다[3]. 카메라가 피사체를 회전축으로 하여 수평 이동함에 따라, 피사체의 영상은 각각의 수평각도로부터 입력된다. 카메라가 오브젝트의 위나 아래로 수직 이동하면, 모든 수직 각도로부터 볼 수 있다. 그림 1은 위도와 경도 같은 고리가 둘러져 있는 지구의 중심 안에 있는 피사체의 모습이다.

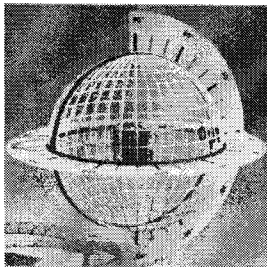


그림 1. 카메라 프레임

본 연구는 본 연구는 중소기업기술혁신개발사업의 지원과 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 정보통신부의 유비쿼터스컴퓨팅및네트워크원천기반 기술개발사업의 지원에 의한 것임.

Object Movie는 입력으로 받아들인 다중의 이미지 영상을 스티칭하여 자연스러운 인터랙티브 무비를 만든다. 각각의 카메라의 피사체로부터의 거리 및 기울어짐 정도를 일정하게 하기 위하여 기존의 Object Movie 생성도구들은 트랙 등의 보조장비를 추가적으로 사용하도록 한다. 그러므로 고가의 트랙장비를 사용하기 힘든 경우에는 Object Movie를 구현하기 위해 소프트웨어적인 카메라 위치 및 회전정보의 보정작업이 필요하다.

2. 기존 Object Movie 생성 도구

Object Movie는 수평 모션과 수직 모션의 범위가 있는 두 축의 무비이다. 무비 포맷은 모든 면으로부터, 가상의 구 위에 있는 모든 점들로부터, 오브젝트의 외관을 볼 수 있게 해 준다. 오브젝트 무비를 만드는 표준 절차에서 첫 단계는 오브젝트 주위를 일정한 양 만큼 등글게 회전하면서 수평 열을 따라 모든 위치에서 오브젝트의 사진을 찍는 것이다. 그 다음에, 카메라를 새로운 수직 위치로 상하 이동시킨다. 가상의 구에 있는 모든 위치에서 오브젝트의 사진을 찍을 때까지 이 과정이 반복될 것이다.

Object Movie를 생성하기 위하여 다양한 파일 포맷과 이를 생성하기 위한 다양한 소프트웨어들이 발표 되었다. 대표적인 Object Movie형식과 대표적인 저작 소프트웨어로는 미국 애플사의 QuickTime VR Authoring Tool, Live Picture사의 Reality Studio, PhotoVista, Object Modeler, Interactive Picture사의 IPIX BUILDER, WIZERD, IBM사의 Hot-Media등이 있다.

이들 소프트웨어들은 각각 장단점을 가지고 현재 제품화되어 사용되고 있다. 그러나 현재 상용되고 있는 프로그램들을 사용하기 위해서는 입력 이미지 영상을 찍을 때 카메라의 위치를 정확히 하여야 하므로 트랙 등의 고

가의 장비를 사용하여야 한다. 본 논문에서는 저가의 웹캠 등의 장비로 Object Movie를 만들기 위해서 카메라 위치 및 회전정보 보정작업을 위한 처리 절차 및 방법을 제안한다.

3. 입력 이미지 보정 Object Movie 생성

3.1 Threshold 값 이용한 프레임 영역 추출

한 사물의 다중 영상을 얻기 위하여 12대의 카메라를 피사체의 주변에 설치하여 12개의 이미지 파일을 생성한다. 12대의 카메라는 원형으로 제작된 촬영 부스에 30° 각도의 위치에 고정 설치한다. 12대의 카메라는 촬영 부스의 중심을 향하고 수평 각도를 최대한 유지할 수 있도록 수동으로 최대한 촬영 방향이 일정하도록 설치되지만, 촬영된 영상의 결과는 영상 중심점과 기준점이 일치하지 않고, 촬영 각도가 틀어지는 결과를 보인다. 그러므로 초기 카메라 설치 단계에서 각 설치된 카메라의 방향과 촬영각도에 대한 정보를 추출하고 이를 저장하여, 설치 단계 이후에 촬영되는 영상에 대하여 자동 위치 및 각도 조정 작업을 수행한다.

카메라의 위치 보정 정보를 추출하기 위하여, 그림 2와 같은 T자형 프레임을 제작하여, 부스의 중앙에 위치시키고, 12대의 카메라로부터 T-형 프레임을 촬영하였다.

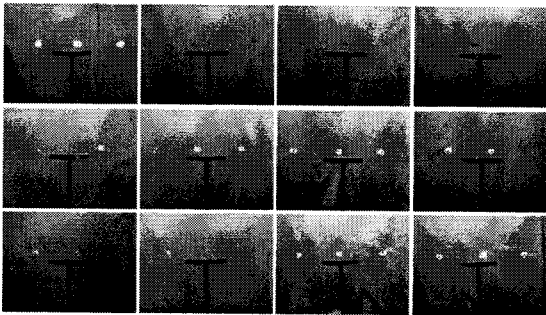


그림 2 영상 보정 정보 추출 위한 12방향에서의 T-형 프레임 촬영영상

12대의 카메라로부터 T-형 프레임 영상 각각에서 바탕영역을 제외시키고 프레임영역만을 추출한다. T-형 프레임만을 추출하기 위하여 빨강-초록-청색(RGB) 컬러 값에서 청색 값이 일정 임계값 이상인 영역을 제거하고 그 이하인 영역만을 추출하였다.

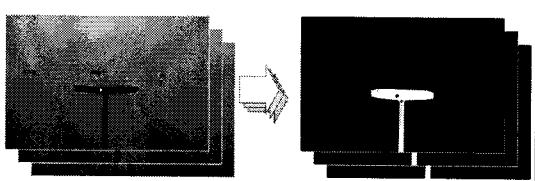


그림 3 임계값 기반 T-형 프레임 영역 추출

3.2 주성분 분석을 이용한 고유벡터 추출

주성분 분석(PCA : Principle Component Analysis)은 잘 알려진 데이터 집합을 분석하는 기법중에 하나로 다차원 변수들을 축소, 요약하는 차원의 단순화와 더불어 일반적으로 서로 상관되어 있는 변수 상호간의 복잡한 구조를 분석하는 것이 그 목적이다[4]. 주성분 분석을 하게 되면 원 데이터의 공분산 행렬로부터 정규적인 고유벡터와 해당하는 고유치를 얻게 되고 그 중 큰 값을 가지는 고유벡터들을 선택하여 선형 변환함으로써 데이터의 차원을 줄일 수 있게 된다. 이를 위하여 주성분 분석은 변수들을 변환시켜, 주성분이라는 서로 독립적인 새로운 변수를 유도한다. 이 때 각 주성분이 보유하는 변이의 크기를 기준으로 그 중요도의 순서를 생각할 수 있는데, 그들 중 첫 소수 몇 개의 주성분에 의해 원래 데이터에 내재하는 전체 변이 중 가능한 많은 부분이 보유되도록 변환시킴으로서 정보의 손실을 최소화하는 차원의 축소 (dimensionality reduction)를 가능하게 한다.

주성분 분석기법을 사용하기 위해 T-형 프레임 영역에 속하는 픽셀들에 대하여 공분산 행렬 C를 식 (1)과 같이 구하고 공분산 행렬에 대한 고유벡터를 구한다. 고유벡터 중 큰 고유값에 매칭 되는 고유벡터가 T-형 프레임의 주방향을 향하게 된다.

$$C = \begin{pmatrix} cov_{xx} & cov_{xy} \\ cov_{yx} & cov_{yy} \end{pmatrix} \quad (1)$$

여기서, 공분산 cov_{xy} 값은

$$cov_{xy} = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}), \quad (2)$$

$$\bar{x} = \sum_{i=1}^n \frac{x_i}{n}, \quad \bar{y} = \sum_{i=1}^n \frac{y_i}{n}$$

로 계산된다. 공분산 cov_{xx} 값은 x방향에 대한 분산 var_x 과 동일하며, cov_{yy} 값은 y방향에 대한 분산 var_y 값과 동일하다. 공분산 cov_{xy} 값이 0보다 작은 수의 음수의 값일 경우, x가 증가/감소함에 따라 y도 함께 증가/감소하게 됨을 의미한다. 공분산 cov_{xy} 값이 0보다 큰 양수의 값일 경우, x가 증가/감소함에 따라 y는 반대로 감소/증가하게 됨을 의미한다. 또한 공분산 cov_{xy} 값이 0일 경우는 x와 y가 서로 무관함을 나타낸다. 공분산 행렬과 같이 상호대칭인 구조를 갖는 행렬은 항상 데이터 차수에 해당하는 고유벡터를 가지며, 그림 4와 같이 고유벡터는 항상 서로 수직하게 된다. 그러나, 때때로 영상의 잡음 현상 때문에 프레임의 주축 부분이 T형 프레임의 방향과 일치하지 않는 그림 4와 같은 현상이 나타날 수 있으므로 이에 대한 보정 작업이 요구된다. 그림 4는 T-형 프레임의 기동 부분의 윤곽선 형태(흰색)와 자동으로 추출된 고유벡터 주축(빨간색)을 보여주고 있다. 상단 부분의 노이즈에 따라 프레임의 방향과 주축의 방

향이 일치하지 않고 있는 예이다.



그림 4. T-형 프레임에 대한 고유벡터

3.3 카메라 움직임 정보 추정 및 보정

프레임의 방향과 주축의 방향이 일치하도록 이를 보정하기 위하여, T-형 프레임의 중점으로부터 -x 방향으로 처음 접하게 되는 윤곽점 $a(x_1, y_1)$ 을 찾고, 주축에 해당하는 고유벡터의 끝점에서 다시 -x방향으로 처음 접하게 되는 윤곽점 $b(x_2, y_2)$ 를 찾아 벡터 $A=b-a$ 를 정의한다. 벡터 A를 T형 프레임의 주 방향벡터로 정의한다. 그림 5는 영상잡음을 포함한 상황에서 T형 프레임의 주 방향을 자동 추출한 결과이다.



그림 5 T형 프레임에 대한 보정된 고유벡터

자동 추출된 주 방향 벡터 A와 x축에 수직인 벡터 $B=(0, -1)$ 의 내각을 이용하여 식(3)과 같이 수직축에 대한 프레임의 회전 각도를 계산한다.

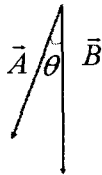


그림 6. 주 방향 벡터와 수직 벡터간의 회전 각도

$$\theta = \cos^{-1}\left(\frac{A \cdot B}{\|A\| \|B\|}\right) \quad (3)$$

프레임의 중심 이동은 수동 처리로 T-형 부분을 클릭하면 벡터의 중심이 이동 되도록 하였다. 그림 7은 추출된 주벡터와 프레임 중심을 표시한 영상이다. 프레임 중심과 주벡터 정보에 의해 계산된 회전각도를 저장하여 이후 촬영되는 영상에 대하여 대응 카메라의 회전각도 및 중심이동 정보를 반영하여 영상을 보정한다.

그림 8은 파일에 저장된 각 보정 정보를 이용하여 T형 프레임 촬영 영상을 보정한 결과 영상이다.

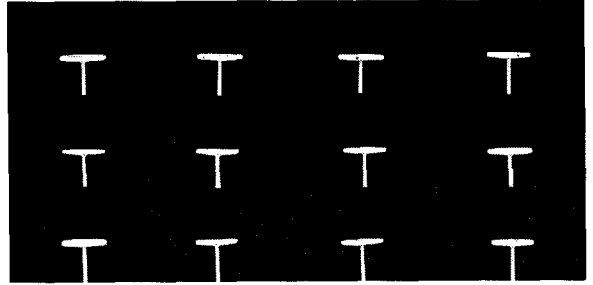


그림 7. 추출된 주벡터와 프레임 중심

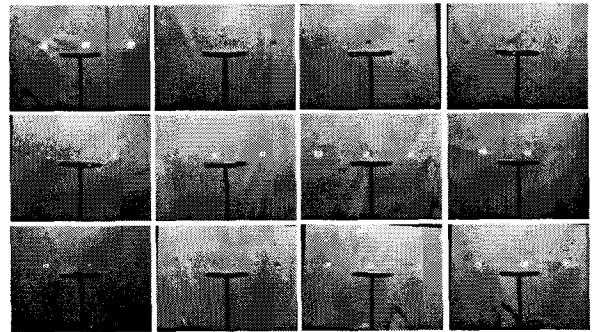


그림 8. T-형 프레임 촬영 영상에 대한 보정 결과

3.4 QuickTime Movie 생성

T-형 프레임을 통해 카메라의 위치가 보정된 이미지들을 QuickTime VR SDK를 이용하여 Object Movie를 생성하는 순서는 그림 9와 같다[5-8].

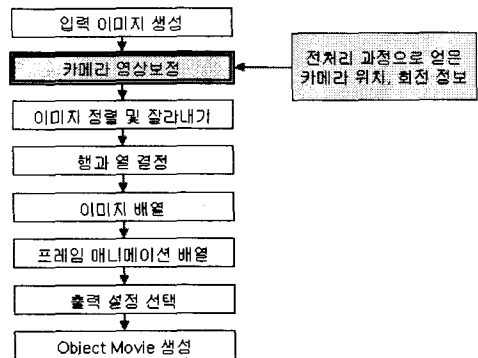


그림 9. 피사체 촬영 후 Object Movie 생성단계

무료로 뷰어가 배포되어 현재 웹기반 응용 어플리케이션 분야에서 많은 사용비율을 차지하고 있는 QuickTime VR포맷을 기반으로 Object Movie를 생성하였다. 실사를 바탕으로 한 이미지 기반 어플리케이션이기 때문에 저비

용으로 생동감 있는 인터랙티브 영상을 구현 할 수 있다. QuickTime movie 파일을 생성하기 위해서 그림 10 과 같은 순서에 따른 작업이 요구된다.

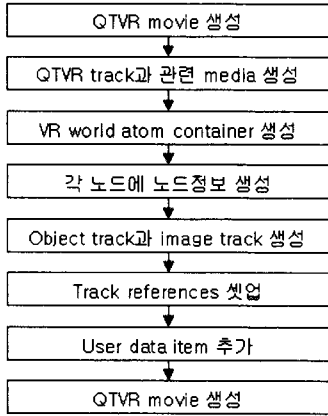


그림 10. Object Movie 변환단계

4. 실험 및 결과

4.1 실험 영상

실험자와 카메라간 거리가 80cm를 유지한 채로 30도 각도를 돌아가며 LG Viewing 360을 이용하여 촬영하였다. 촬영된 영상과 보정 정보에 의해 보정된 영상은 다음과 같다.

영상구분	영 상			
입력영상				
이미지 보정영상				

4.2 최종 Object Movie 영상

그림 11은 생성된 Object Movie파일을 QuickTime Viewer를 통하여 실행된 화면을 보여주고 있다. 사용자는 영상 상에서 마우스 드래그를 통하여 Object를 인터랙티브 하게 회전시키며 돌려 볼 수 있다.



그림 11 QuickTime VR이용한 실행 예

5. 결론

기존 상용 Object Movie 상용도구들은 정확하게 캘리브레이션된 카메라 영상들을 입력 영상으로 사용하고 있다. 본 연구에서는 정확하게 보정된 영상 입력을 위하여 고가의 카메라 트랙장비를 설치하지 않고, 초기 카메라 설치 위치를 반자동으로 파악하고, 카메라의 설치 정보를 기반으로 입력 영상을 소프트웨어적으로 보정하는 방법을 제안하였다. 소프트웨어적인 보정작업으로 사용자의 인터랙션에 따라 보다 자연스러운 객체의 회전을 지원하는 Object Movie를 생성할 수 있었다.

6. 참고문헌

- [1] R. Szeliski, "Video mosaics for virtual environments", IEEE Computer Graphics and Applications, pp.22-30, March 1996
- [2] R. Szeliski, H. Y. Shum, "Creating full view panoramic image mosaics and environment maps", Computer Graphics (SIGGRAPH'97), pp.251-258, August 1997
- [3] S. E. Chen, "QuickTime VR: an image-based approach to virtual environment navigation", Computer Graphics (SIGGRAPH'95), pp.29-38, August 1995
- [4] I. T. Jolliffe. "Principle Component Analysis (2ndedition)", Springer series in statistics. Springer-Verlag New York. 2002.
- [5] Susan A. Kitchen, "The QuickTime VR Book:Creating Immersive Imaging on Your Desktop", Peachpit Press, 1998
- [6] Apple, "QuickTime Toolkit:Basic Movie Playback and Media Types, Volume one", Elsevier, 2004
- [7] Apple, "QuickTime Toolkit:Advanced Movie Playback and Media Types, Volume two", Elsevier, 2004
- [8] Apple, "QuickTime File Format", 2001