

영상의 회전과 크기를 고려한 특징기반 영상 모자이크

고종호, 이철우
전남대학교 컴퓨터공학과

Feature-based Image Mosaicing With Rotation and Scale Change

Jong-Ho Ko, Chill-Woo Lee
Dept. of Computer Engineering, ChonNam National University

요약

본 논문은 제약되지 않은 카메라에서 얻은 영상 중 회전과 크기 변화를 가진 두 장의 디지털 영상을 자동적으로 하나의 통합된 영상으로 모자이크 하는 방법에 대해 기술한다. 기존에 제시되었던 영상 모자이크 방법은 영상의 중첩 영역이 많이 존재하거나 회전이 거의 없는 경우만을 고려하고, 카메라 제약이 많이 존재하였다. 우선, 한 쌍의 영상으로부터 각각 특징 점을 찾고, 각 특징 점에 대하여 위상을 측정하여 계층적으로 매칭을 하는 방법을 제안한다. 다음으로 비선형 이송오차 최적화 알고리즘을 이용해 최적의 변환 행렬을 구한 후, 변환 행렬에 대해 하나의 영상을 만들어 내는 과정을 기술한다.

1. 서론

최근 몇 년 전부터 영상 모자이크 기술은 컴퓨터 비전 연구에서 많이 응용이 되기 때문에 흥미 있는 연구 분야가 되어 왔다. 이 기술은 우주/항공 사진의 해석, 컴퓨터 그래픽스의 페인팅 시스템, 사진의 가공 등 영상을 다루는 분야에 폭 넓게 사용되어 왔다.

모자이크 기술은 임의의 시점에서 촬영한 영상들을 하나의 영상으로 통합시키는 물론 시점을 달리하여 영상을 표현할 수 있기 때문에 건축물 내부와 같이 복잡한 장면을 그래픽 모델을 제작하지 않고도 간단한 변환에 의해 생성할 수 있으므로 다양한 영상 생성이 필요한 각종 응용 시스템에서 사용될 수 있다. 이러한 기술을 최근에 들어 영상기반 랜더링(image based rendering) 이라고 부르며 계산의 복잡도를 줄이는 문제와 3차원 정보를 가미한 사실성(reality)의 증대가 당면 과제로 남아 있다.

지금까지 널리 알려진 모자이크 방법으로는 파노라마 영상 생성법이 있다[6]. 이 방법은 고정된 위치에 카메라를 설치하고 수평방향으로 회전시키면서 촬영한 영상을 하나의 영상으로 통합하여 원통형의 벽에 매핑 시킴으로써 관찰자에게 전 방위의 시야를 제공하는 방법이다. 하지만, 이 방법은 카메라가 자유롭

지 못하고, 영상 획득시 고려해야 할 조건이 많이 존재한다. 파노라마 영상 생성을 위해 초기에는 사용자가 수작업으로 각각의 영상 쌍마다 대응점을 지정하였으나 완전한 자동화를 실현하기 위해 많은 연구가 진행되고 있고, 더 나아가 3차원의 현실감 있는 영상을 제공하기 위해 영상 생성시의 카메라 파라미터를 복원하는 연구도 진행되고 있다.

모자이크 기술은 근본적으로 그래픽스에서 사용하는 영상 워핑(Image Warping) 이론과 컴퓨터 비전에서 사용하는 투영변환(Perspective Transformation) 이론을 결합한 기술이다. 이 기술에서는 임의 영상을 새로운 영상으로 변환하는 과정에서 각종 파라미터를 고려해야 되나 파라미터가 많으면 많을수록 계산이 복잡하여 처리시간이 길어질 뿐 만 아니라, 국소 극소점에 수렴할 가능성이 있어 바람직하지 않다. 이러한 문제점을 해결하기 위해, Szeliski[1]는 두 영상 사이의 회전만을 고려한 3-파라미터 회전 모델을 제시하였다. 이 모델은 두 영상의 초점 거리를 임의로 가정하고, 그 값은 별도의 계산이 필요한 최소화 방법 [7]에 의해 초기 추정치로부터 반복적으로 개선된다. Chiba[5]는 다중 광류(optical flow) 추정법을 이용하여 모자이크하는 방법을 소개하였다. 먼저, 초기 중첩

영역을 구하고, 광류추정의 한 방법인 Lucas-Kanade 방법을 이용 대응점을 찾는다. 이로부터 변환행렬을 구해 에피폴라 기하에 적용시켜 영상을 모자킹하는 방법을 소개하였다.

본 논문에서는 기존에 제시되었던 영상 모자이크 기술과는 달리 자유로운 카메라로부터 획득한 회전과 크기의 변화를 가진 두 장의 영상을 특징 점 추출 및 매칭을 통한 자동적인 모자이크 방법에 대해 소개한다. 먼저, 두 영상에서 각각 특징 점을 찾는다. 이 과정에서 특징 점을 최소화하기 위해 전처리 과정을 수행한다. 다음으로, 추출된 특징 점들 간의 위상을 계산하고, 계층적인 비교를 통한 정확한 매칭 점을 찾아내고, 최적의 4개의 특징 점을 투영변환을 통해서 비선형 이송오차 최적화 알고리즘을 이용해 최적의 변환 행렬을 구한 후 최종적으로 영상을 합성한다.

본 논문의 구성은 다음과 같다. 2절에서는 영상 모자이크를 수행하기 위한 전체적인 과정에 대해서 기술한다. 3절에서는 실험 결과 와 고찰에 대해서 서술하고, 4절에서는 결론을 맺는다.

2. 영상 모자이크

그림 1.은 본 논문에서 제시한 자동 영상 모자이크 생성의 전체적인 흐름을 보여 주고 있다.

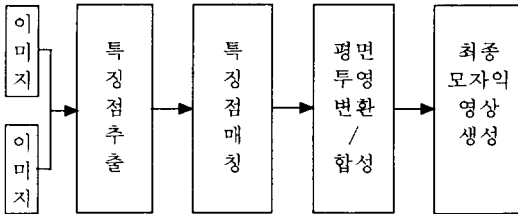


그림 1. 모자의 영상 생성 과정

2.1 특징 점 추출(Feature Point Extraction)

특징 점이란 영상 안의 물체가 가지고 있는 코너 점(Corner Point)이나 주변 화소에 비해 큰 밝기의 변화를 갖는 화소와 같은 점으로써 이 점은 영상 안에서 다른 점들과 구별되어지는 점이다. 이러한 특징 점은 여러 방향에서의 높은 미분 값을 갖는 점으로 정의되기도 하고, 특정 변환에 불변의 성질을 갖는 점으로 정의되기도 하는 등 여러 가지로 정의될 수 있다. 영상 안에서 특징 점을 뽑아내는 방법들은 다음과 같이 크게 세 가지 그룹으로 분류할 수 있다.

· 그룹 1 : 에지 지도(Edge Map)를 이용한 방법

· 그룹 2 : 그레이레벨(Gray Level) 영상을 직접 접근하는 방법

· 그룹 3 : 미분 기하 또는 모델 기반 접근법

이 가운데 대부분의 특징 점 추출은 두 번째 그룹에 속하며, 최근에는 그룹 3에서 활발한 연구가 진행되고 있다. 그룹 3에서 미분기하 방법은 Guiducci 방법으로 그룹 2에서 파생된 방법이고, 모델 기반 법은 Gaussian filter를 이용 smooth를 수행 후 분석하는 방법으로써 잘 알려진 정확한 코너를 찾는 데 이용하지만, 계산시간이 많이 걸린다는 단점을 가지고 있다.[4]

본 논문에서는 그룹 2에 속한 방법으로 Hessian Matrix를 이용했다. 이 방법은 영상의 밝기가 급격히 변화하는 점을 특징 점 후보로 선택한다. 우선, 전처리 과정이 필요한데, 입력 영상이 복잡하거나 잡음이 많은 경우를 고려하여 Sobel 오퍼레이터를 이용하여 영상의 에지 성분을 강조한 후 에지에 관한 Hessian Matrix를 적용한다. Hessian Matrix는 식 (1)과 같다.

$$H = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix} \quad \text{식 (1)}$$

여기에서, I_{xx} 와 I_{yy} 는 각각 함수 I 의 x 방향과 y 방향의 2차 미분이며, I_{xy} 와 I_{yx} 는 함수 I 의 x 방향과 y 방향의 2차 미분이다.

2.2 특징 점 정합(Feature Point Matching)

2.2.1 기존연구

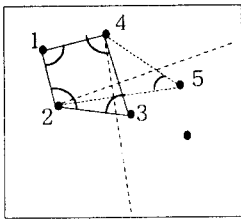
영상 안에서의 특징 점들에 대해서만 정합을 수행하는 방법을 특징점 정합(Feature Point Matching)이라 한다. 특징 점 정합은 컴퓨터 비전의 여러 분야에서 중요한 역할을 하는데 스테레오 정합, 트래킹(tracking), 움직임 추정, 영상교정, 3차원 복원, 셀프 칼리브레이션 등을 그 중요한 응용분야로 들 수 있다. Deriche[4]는 특징 점 사이의 정합 정도를 측정하는데 여러 가지 평가 기준을 사용하였다. 따라서, 정합정도를 수치적으로 나타내기가 어려우며 정합 가능한 특징 점을 찾아내는데 오히려 혼동을 증가시킨다. Zhang은 초기 정합의 결과로 발생하는 애매성의 해결을 위해 이완화(relaxation) 기법을 이용하였는데 이는 성능에 비해 많은 시간을 요구한다는 단점을 가지고 있다. 하지만, 위에서 제시한 단지 정합에 관한 연구는, 영상을 모자이크하는 데는 적합하지 않다. 우선, 많은 시간과 계산량을 요구하고, 영상의 회전과 크기에 대해서는 적용하는데 충분하지 않으며, 최적의 특징 점을 선택하여 모자이크하는 데는 불필요한 단계

를 가지고 있다.

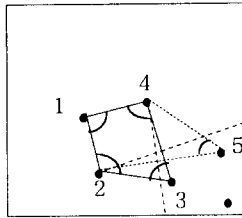
2.2.2 기하 위상을 고려한 특징 점 매칭

본 논문에서 제안하는 모자이크하는데 요구되는 매칭 방법은 특징 점과 가상 선(Virtual Line)을 이용해 그 사이의 위상 각을 알아내고 계층적으로 탐색해 나감으로써 영상의 회전과, 크기의 변화에 관계없이 매칭 점들을 찾을 수 있다. 알고리즘은 다음과 같다.

- 1) 그림 2(b)영상에서 획득한 특징 점들에 대해서 모든 위상 각을 계산한다.
 - 2) 그림 2(a)영상에서 첫 번째 특징 점을 탐색해 인접 특징 점들과의 위상을 계산한다. 이때 식 (2)의 조건에 적용해서 조건에 해당되는 것만 탐색할 수 있으므로 계산량을 줄일 수 있다.
- $$0 < \theta_1, \theta_3 < \frac{\pi}{2}, \quad \frac{\pi}{2} < \theta_2, \theta_4 < \pi \quad \text{식 (2)}$$
- 3) 만약 그림 2(a)에서처럼 하나의 사각형이 구성되었다면 그림 2(b)영상에서 탐색을 한다. 존재하지 않으면 다시 3)을 반복하고, 오차 범위 내에서 존재하면, 다른 특징 점의 각을 계산하여 3)을 반복한다.
 - 4) 찾은 모든 점과 선에 관한 매칭을 찾는다.
 - 5) 매칭 점에 대해서 가장 멀리 떨어진 점 4쌍을 결정한다.



(a) 영상 1



(b) 영상 2

그림 2. 계층적 특징점 매칭 방법

그림 2. 는 제안하는 방법에서 계층적으로 특징 점들에 대해 매칭하는 과정을 보여 주고 있다.

2.4 영상 합성

동일한 시점에서 촬영된 두 장의 영상 위치는 2차원 투영변환행렬 (M)에 의해서 결정된다. 변환행렬 M 은 다음과 같이 정의한다.

$$M = \begin{bmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & m_8 \end{bmatrix} \quad \text{식 (3)}$$

식 (3)의 증명은 [1]에 있다.

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & m_8 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} \quad \text{식 (4)}$$

여기에서, (x, y) 는 영상에서 최적의 특징 점으로 선택된 픽셀 좌표이고, (x', y') 는 대응되는 영상에서 픽셀 좌표이다. 위 식의 관계를 이용하여 식 (5)와 같은 방정식으로 앞에서 구한 4쌍의 초기 대응점을 구할 수 있다.

$$\begin{aligned} x' &= \frac{m_0x + m_1y + m_2}{m_6x + m_7y + 1}, \\ y' &= \frac{m_3x + m_4y + m_5}{m_6x + m_7y + 1}, \end{aligned} \quad \text{식 (5)}$$

본 논문에서는 비선형 이층 오차 최소화 기법으로 Levenberg-Marquardt 알고리즘을 사용하였다. 이 방법은 중첩영역 안에서 최소화되는 에러 e_i 를 구해 새로운 변환 행렬 M 을 결정하는 방법으로 알려지지 않은 운동 요소 $\{m_0, \dots, m_7\}$ 의 함수인 e_i 의 편미분 값을 이용한다. e_i 는 식 (6)과 같이 정의한다.

$$e_i = I'(x'_i, y'_i) - I(x_i, y_i) \quad \text{식 (6)}$$

각각 명암도가 다른 두 영상에서 재 구성시 경계선이 발생하게 된다. 이를 제거하기 위해서 각 영상에 가중 평균(weighted average) 알고리즘을 이용하여 눈에 보이는 경계선을 줄인다. 가중 평균 함수(w)는 식 (7) 과 같고, 식 (8)을 이용하여 최종 영상 모자이크를 수행한다.

$$w(x', y') = w_1(x')w_2(y') \quad \text{식 (7)}$$

$$I' = \frac{\sum_k w_k(x_i, y_i) I_k(x_i, y_i)}{\sum_k w_k(x_i, y_i)} \quad \text{식 (8)}$$

3. 실험 및 고찰

제약이 없는 카메라로 획득한 회전된 영상과 크기가 다른 영상에 대해서 각각 실험을 수행하였다. 실험에 사용된 영상은 모두 하나의 카메라로부터 약간의 시간 간격을 두고 획득한 영상이고, 크기는 250×250 화소이며, 전저리를 통해 256 그레이레벨을 갖도록 했다. 촬영장비는 Kodak DC120 디지털 카메라이고, 영상 모자이크에 관한 실험은 Pentium 500MHz 와 64MB의 메모리를 장착한 PC 환경에서 진행하였다. 사용 OS는 Windows 98이다.

3.1 실험 결과

그림 (3)은 회전된 입력 영상에 대해 모자이크을

수행한 결과의 영상이고 그림 4.5는 각각 크기가 다른 두 장의 입력 영상과 그에 따른 결과를 보여주고 있다.

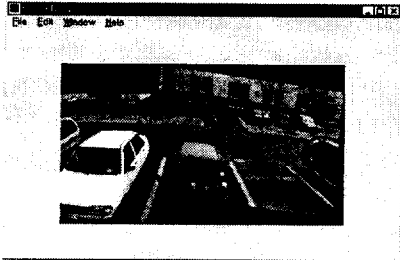


그림 3. 회전이 된 영상 모자이킹 결과

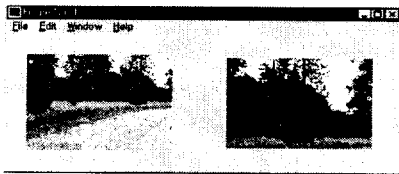


그림 4. 크기가 다른 입력 영상

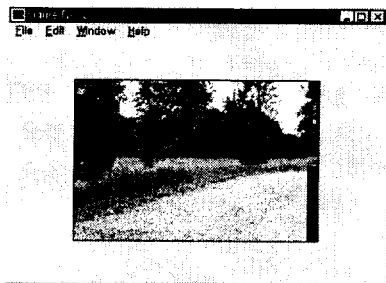


그림 5. 그림 4의 결과 영상

3.2 고찰

본 실험에서 자동적으로 영상 모자이킹을 수행하기 위해서 우선, 계산 량이 많다는 것을 단점으로 들 수가 있다. 영상을 매칭하기 위해서 소요되는 시간과 최적의 변환 행렬을 구하기 위해서는 많은 시간이 필요하다. 하지만, 제안하는 방법은 카메라의 구속 없이 획득한 영상의 회전과 크기에 대한 모자이킹은 만족할 만한 결과를 얻을 수 있고, 영상의 종류에 따라 영상의 특징 점 개수에 따른 위상을 계산 및 비교 할 때 약간의 오차 조정이 필요하다.

4. 결론

본 논문에서는 카메라의 제약조건 없이 획득된 회전과 크기를 가진 두 장의 영상으로 완성된 하나의 영상으로 만드는 자동 모자이킹 방법에 대해 기술하였다. 특징 점을 찾기 위해서 에지를 추출한 후 Hessian Matrix를 적용하였고, 추출된 특징 점에 대해서 각각 점들의 위상을 계산한 후 계층적으로 탐색하여, 최적의 특징 점 4쌍을 얻어 초기 변환행렬을 비선형 이송 오차 최소화 알고리즘에 적용해 최적의 변환 행렬을 구한 후 두 영상을 합성하였다. 합성 중에서 발생 할 수 있는 불연속 경계선은 가중평균 (Weighted average)를 적용하여 제거하였다.

[참고문헌]

- [1] R. Szeliski, "Video mosaics for virtual environments", IEEE Computer Graphics and Applications, P22-30, March 1996
- [2] R. Szeliski, "Image Mosaicing for Tele-Reality Applications", CRL Technical Report 94/2, Digital Equipment Corporation, 1994
- [3] I. Zoghlami and O. Faugeras and R. Deriche, "Using geometric corners to build a 2D mosaic from a set of images", INRIA, 2004 route des Lucioles, BP93
- [4] Thierry Blaszkza, Rachid Deriche, "Recovering and Characterizing Image Features Using An Efficient Model Based Approach", INRIA, N2422, 1994
- [5] Naoki Chiba, Hiroshi Kano, Minoru Higashihara, Masashi Yasuda, and Osumi, "Frature-Based Image Mosaicing", MVA'98 IAPR Workshop on Machine Vision Applications, pp5-10, 1998
- [6] W. Puech, A. G. Bors, J.M. Chassery, I. Pitas, "Mosaicing of Paintings on Curved surfaces," IEEE Workshop on Applications of Computer Vision, Sarasota, USA, pp.50-55, 1996
- [7] W. H. Press, B. P. Flannery, A. A. Teukolsky, and W. T. Vetterling, "Numerical Recipes in C: The Art of Scientific Computing" Cambridge University Press Cambridge, England, Second edition, 1992