

키 프레임 추출을 통한 영상 정합 기법

*김종호 **유지상

광운대학교

*jjong3927@kw.ac.kr **jsyoo@kw.ac.kr

Image matching methods through key frame extraction

*Jongho Kim **Jisang Yoo

Kwangwoon University

요약

본 논문에서는 카메라로 촬영한 동영상에서 키 프레임을 추출하고 특징점을 기반으로 영상을 정합하는 파노라마 영상 생성 기법을 제안한다. 제안한 기법에서는 다양한 동영상의 히스토그램, 에지 등의 정보를 이용해 강인한 키 프레임을 추출하고 추출된 다수의 키 프레임 영상에 실린더 투영 방법과 FAST(Feature from Accelerated Segment Test) 기법을 적용하여 자연스러운 정합 영상을 획득할 수 있다. 정합된 특징점의 오차율을 최소화하기 위해 RANSAC(Random Sample Consensus)을 사용하고 여러 장의 다른 시점 영상을 정합할 때 생길 수 있는 경계선을 제거하고 보정하기 위해 선형가중치 함수도 사용한다. 실험을 통해 제안하는 기법으로 자연스러운 파노라마 영상을 생성할 수 있었다.

1. 서론

최근 VR(Virtual Reality)에 대한 관심이 많아지고 다양한 어플리케이션이 개발되면서 VR 기술의 중요성이 커지고 있다. 본 논문에서는 180도, 360도 카메라를 사용하지 않고, 평범한 디지털 카메라나 스마트폰으로 촬영한 동영상을 이용하여 파노라마 영상을 만드는 방법을 제안한다. 파노라마 영상이란 영상처리를 이용해 여러 장의 영상을 하나의 큰 영상으로 정합한 고해상도 영상을 말한다. 파노라마 영상은 VR뿐만 아니라 로드뷰, 의료영상, CCTV영상 등에 활용되고 있다.

파노라마 영상은 중첩되는 다수의 영상을 정합하고 스티칭하는 방법으로 생성된다. 다수의 영상은 각도와 방향을 다르게 하여 여러 장의 정지 영상을 촬영하여 획득한다. 제안한 기법에서는 동영상에서 중요한 프레임이라고 판단되는 키 프레임들을 자동으로 추출하여 파노라마 영상을 생성하게 된다. 기존의 정지 영상을 여러 번 촬영하는 것보다 쉽고 빠르게 고품질의 파노라마 영상을 생성할 수 있다는 장점이 있다.

동영상에서 키 프레임을 추출하는 방법에는 단순히 고정된 프레임 수마다 키 프레임을 추출하거나 영상간의 색상 차이를 이용하여 추출하는 방법 등이 있다. 이러한 방법들은 영상의 재생 속도, 영상의 밝기 등에 따라 성능이 변하기 때문에 비효율적이다. 제안하는 방법에서는 영상으로 부터 히스토그램, 에지 성분, 차 영상, 컬러 값 등의 정보를 복합적으로 이용하여 키 프레임으로 추출한다. 그 결과 다양한 환경의 영상에서도 효율적인 키 프레임 추출이 가능하다.

파노라마 영상을 생성하는 방법도 여러 가지가 있다. 먼저 SIFT(Scale Invariant Feature Transform) 기법으로 특징점을 추출하여 호모그래피 행렬을 구하고 파노라마 영상을 생성하는 기법이 있다[1]. 이 방법은 SIFT의 연산량이 많기 때문에 수행 속도가 매우 느

린 단점이 있다. 각 화소 값의 정합을 통해 파노라마 영상을 생성하는 방법도 있으나 전체 영상의 화소값을 탐색하기 때문에 속도가 느리고, 작은 잡음에도 에러가 많이 발생한다[2]. SIFT 기법을 개선한 SURF(Speed Up Robust Feature) 기법은 수행 속도가 빨라졌지만 실시간으로 파노라마 영상을 생성하기 위해 더 빠른 특징점 추출 기법이 필요하다[3].

기존의 영상 정합 방법들은 영상을 정합 순서대로 입력해서 스티칭해야 한다. 그렇지 않으면 파노라마 영상을 만들 때 영상이 휘어지는 왜곡현상이 발생 할 수도 있다. 이러한 문제점을 해결하기 위해 2차원 영상을 3차원 공간으로 투영해 왜곡을 최소화 하려는 시도가 있었으나 수행 시간이 오래 걸리는 단점이 있다[4]. 제안하는 기법에서는 정합된 특징점의 위치를 이용해 영상의 정합순서를 자동으로 계산할 수 있다.

파노라마 영상 생성에 있어서 가장 중요한 것은 영상에서 특징점을 정확하게 찾고 호모그래피 행렬을 계산하는 것이다. 각각의 영상들은 촬영한 각도와 초점거리가 다르기 때문에 정합할 때 영상이 휘어지는 현상이 발생할 수 있다. 이를 보정하기 위해 입력 받은 영상을 기존 좌표계에서 원통형 모양의 좌표계로 투영하는 실린더 투영을 수행한다. 투영된 영상에서 FAST기법을 이용하여 기존의 방법보다 빠르게 특징점을 추출할 수 있다[5].

영상간의 정합을 위해 특징점의 벡터 서술자를 빠르게 계산 할 수 있는 SURF 기술자를 이용한다. 기술자를 이용해 영상간의 특징점을 정합하고, 그 위치를 이용해 영상의 정합 순서를 자동으로 계산한다. 특징점을 정합할 때 RANSAC 기법을 적용하여 오차가 있는 점은 제거한 뒤 가장 정확하게 정합된 네 점을 이용해서 호모그래피 행렬을 계산한다[6]. 영상을 정합하는 과정에서 발생하는 경계선의 색상 불일치와 왜곡을 자연스럽게 보정하기 위해 선형 가중치 함수도 이용한다[7].

본 논문의 2장에서는 동영상에서 키 프레임을 추출하는 방법과 파노라마 영상을 생성하기 위한 방법을 설명한 뒤 실험 결과를 비교한다. 그리고 3장에서 결론을 내린다.

2. 본론

2.1 키 프레임 추출

입력 동영상은 일반적인 디지털 카메라, 스마트폰으로 촬영하여 획득할 수 있다. 본 논문에서는 촬영된 동영상의 크기는 물론이고 촬영 환경, 촬영 의도와 무관하게 파노라마 영상을 생성하기 위한 효율적인 키 프레임 추출 기법을 제안한다.

본 논문에서는 화소 값, 히스토그램, 에지 정보 등을 모두 고려하여 키 프레임을 추출한다. 제안하는 기법을 동영상의 모든 프레임에 적용하면 계산량이 많고 비효율적이기 때문에 다섯 프레임 간격을 두고 적용하여 성능은 유지하면서 계산 속도를 향상시킬 수 있다. 키 프레임으로 선택하기 위한 첫 기준이 되는 것은 현재 프레임의 화소값을 합산한 값과 이전 프레임에서 합산한 값과의 차이이다. 식(1)의 결과 값이 실험적으로 구한 임계값 이상이면 키 프레임 후보로 선택하게 된다.

$$Clr(f_1, f_2) = \left| \sum_{i=0}^T f_1(i) - \sum_{i=0}^T f_2(i) \right| \quad (1)$$

식 (1)을 이용하여 키 프레임 후보로 추출된 프레임에 두 번째 기준인 히스토그램의 차이를 적용한다. 식 (2)를 이용하여 이전 프레임과의 히스토그램 차이 값을 합산 한다. 히스토그램 차이의 합이 임계값 이상이면 키 프레임 후보 상태를 유지하고 아니면 후보에서 제외한다.

$$Histo(h_1, h_2) = \sum_{k=0}^{255} |h_1(k) - h_2(k)| \quad (2)$$

마지막으로 식 (2)에서 키 프레임 후보로 유지된 프레임에 Canny 에지 추출 방법을 적용한다[8]. 키 프레임 후보 프레임들 중에서 전 프레임과 현재 프레임의 에지의 차이를 식 (3)과 같이 계산한다. 식 (3)의 결과가 임계값 보다 크면 해당 프레임을 최종적으로 키 프레임으로 선택한다.

$$Edge(e_1, e_2) = \left| \sum_{i=0}^T e_1(i) - \sum_{i=0}^T e_2(i) \right| \quad (3)$$

2.2 실린더 투영

동영상에서 추출된 키 프레임은 서로 초점거리의 차이가 있기 때문에 정합 시 옆으로 휘어지는 왜곡 현상이 발생할 수 있다. 이 현상을 보정하기 위해 기존의 평면 좌표계에 있는 영상을 원통형 모양의 좌표계로 투영하며, 이 과정을 실린더 투영이라고 한다.[9].

식 (4)를 이용해 실린더 투영을 수행한다. 그림 1은 초점거리 f 를 변화시키면서 역방향 실린더 투영을 한 결과 영상이다.

$$\begin{aligned} x' &= s\theta + x_c = s \tan^{-1} \frac{x}{f} + x_c \\ y' &= sh + y_c = s \frac{y}{\sqrt{x^2 + f^2}} + y_c \end{aligned} \quad (4)$$

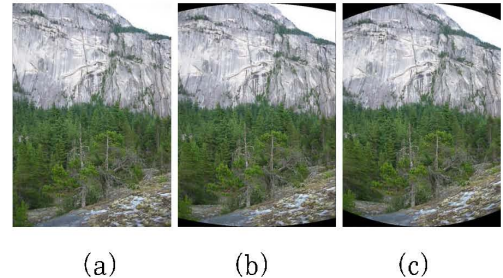


Fig. 1. 실린더 투영 결과 영상
(a) 원본 영상 (b) $f = 700$ (c) $f = 400$

2.3 특징점 추출 및 정합

실린더 좌표계로 투영된 영상에서 특징점을 추출하여 영상간의 일치하는 정합점을 찾는다. 제안하는 기법에서는 특징점을 추출하기 위해 FAST 기법을 사용한다. FAST 기법은 기존의 Harris, SIFT, SURF와 같은 특징점 추출 기법보다 빠른 속도로 특징점 검출이 가능하다. 그림 2은 FAST를 이용해 각 영상에서 추출한 특징점의 위치를 빨간 점으로 표시한 결과 영상이다.



Fig. 2 FAST기법으로 추출한 특징점 결과 영상

각 영상에서 특징점을 검출한 후, 영상간의 정합을 위해 특징점의 정보를 포함하는 서술자를 생성한다. 제안하는 기법은 SURF 기법에서 사용하는 서술자를 이용하여 객체의 회전과 크기에 강한 서술자를 표현할 수 있다[10].

2.4 호모그래피 계산

각 영상의 특징점을 정합하기 위해 RANSAC 기법을 이용한다. RANSAC 기법은 잡음이 섞여있는 데이터로부터 적절한 모델을 예측하여 오차를 최소화하는 기법이다. 추출된 특징점 중에서 임의로 4개의 특징점을 선택하고 호모그래피 행렬을 계산한다. 호모그래피 행렬이란 기준 영상의 좌표와 대응되는 영상의 좌표 간의 관계를 식 (5)와 같이 행렬로 나타낸 것이다. 계산된 호모그래피 행렬을 이용하여 기준 영상을 변환한 뒤 대응하는 영상의 특징점과 비교한다. 위의 과정을 반복 수행하여 오차가 최소가 되게 하는 4개의 특징점을 찾고, 그 특징점으로 계산된 호모그래피 행렬을 영상에 적용하면 파노라마 영상을 얻을 수 있다.

$$\begin{bmatrix} \hat{x}_i w_i \\ \hat{y}_i w_i \\ w_i \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \quad (5)$$

2.5 색상 블렌딩

다수의 영상을 정합할 때 각 영상마다 조명과 시점이 달라 정합 후 부자연스러운 경계선이 발생할 수 있다. 이러한 경계선을 제거하기 위해 그림 3의 선형가중치함수(Bilinear Weighted Function)를 사용한다. 겹쳐지는 경계선 주변의 화소의 가중치를 다르게 부여하는 방법이다. 그림 4은 영상을 정합했을 때 선형가중치 함수의 효과를 나타내는 결과 영상이다.

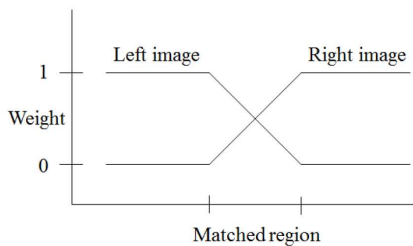


Fig. 3 색상 블렌딩을 위한 선형가중치 함수

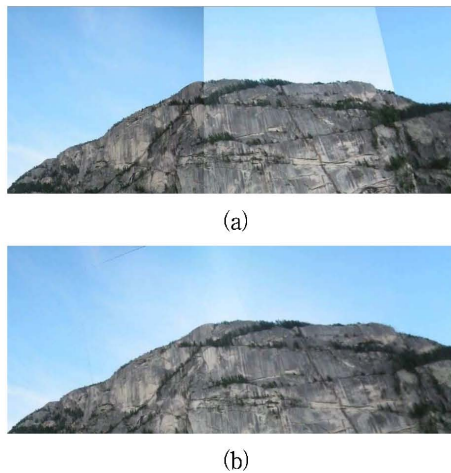


Fig. 4 (a) 색상 블렌딩 수행 전의 파노라마 결과 영상
(b) 색상 블렌딩 수행 후의 파노라마 결과 영상

2.6 실험 결과

실험 환경은 i5 인텔 CPU, 8GB 램, GeForce GTX460 그래픽 카드를 이용하고, 비주얼 스튜디오 2013으로 구현하였다. 영상 내 FAST 알고리즘을 구현하기 위해 OpenCV 2.4.11 라이브러리를 사용한다. 본 논문에서는 직접 촬영한 동영상으로 실험을 진행하였다.

그림 6의 (a)는 “아파트” 동영상에서 추출된 키 프레임의 임의의 방향과 순서로 입력하는 그림이고, (b)는 추출된 키 프레임 6장으로 생성한 파노라마 결과 영상이다. 제안하는 기법에서는 정합할 영상의 순서와 방향에 상관없이 자연스러운 파노라마 영상을 생성하는 것을 알 수 있다.



(a)



(b)

Fig. 6 (a) 임의의 방향과 순서를 가지는 입력 영상
(b) 파노라마 결과 영상

표 1은 SIFT, SURF 기법과 제안하는 기법을 이용하여 10장의 키 프레임으로 파노라마 영상을 생성할 때 과정별 처리 속도를 비교한 것이다. 특징점을 추출하는 과정에서 제안하는 FAST 기법의 수행 시간이 많이 감소한 것을 확인할 수 있다.

Table. 1 10장의 키 프레임 정합시 특징점 추출 방법에 따른 과정별 처리 속도 비교

Process	10 frames		
	SIFT (ms)	SURF (ms)	Proposed method (ms)
Cylindrical projection	281	281	281
Feature extraction	30157	6012	824
Feature matching	5112	2644	954
Calculate homography	891	891	891
Color blending	846	846	846
Total	37287	10674	3796

2. 결론

본 논문은 동영상에서 자동으로 키 프레임을 추출하고, 추출된 키 프레임을 이용하여 파노라마 영상을 생성하는 방법을 제안하고 있다. 제안하는 기법은 기존의 키 프레임 추출 방법보다 다양한 환경의 영상에서 강인하게 키 프레임을 추출할 수 있다. 또한 기존의 방법보다 빠르게 특징점을 추출할 수 있는 FAST 기법을 이용하여 파노라마 영상을 생성하는 방법을 제안하였다.

또한 실린더 투영과 색상 블렌딩 기법을 통한 경계 영역의 색상 불일치 영역을 보정하는 과정을 추가하여 자연스러운 파노라마 영상을

생성할 수 있었다. 제안하는 기법은 입력 영상의 순서와 방향에 상관 없이 정상적인 파노라마 영상 생성이 가능하다. 앞으로 더 많은 영상을 정합할 때 병렬 처리를 이용하여 실시간으로 정합할 수 있을 것으로 예상된다.

Reference

- [1] M. Brown, and D.G. LOWE, "Automatic panoramic image stitching using invariant features", *International journal of computer vision*, vol. 74, no. 1, Dec. 2006
- [2] R. Szeliski, "Image alignment and stitching: a tutorial," *Computer graphics and vision*, Vol. 2, No.1, pp.15-16, Jan. 2006
- [3] H. Bay, A. Ess, T. Tuytelaars and L. V. Gool "Speeded-Up Robust Features (SURF)," *Computer vision and image understanding (CVIU)*, Vol. 110, No. 3, pp. 2-8, Jun. 2008
- [4] Yongwoo Cho, Joo Myoung Seok, Doug Young Suh, "3D-based monitoring system and cloud computing for panoramic video service", *The Journal of Korean Institute of Communications and Information Sciences*, Vol.39, No.09, pp. 590-597, August. 2014
- [5] E.Rosten, T.Drummond, "Machine learning for high-speed corner detection", *European conference on computer vision*, Vol.1, pp.430-443, 2006
- [6] K-W Kwon, A-Y Lee, U. Oh, "Panoramic image composition algorithm through scaling and rotation invariant features," *Information processing society journal*, Vol. 17, No. 5, Jun. 2010.
- [7] P. J. Burt and E. H. Adelson, "A multiresolution spline with application to image mosaics," *ACM Transaction on graphics*, Vol. 2, No. 4, pp. 2-5, Oct. 1983.
- [8] Canny. J "A computational aproach to edge detection", *IEEE Transactions on pattern analysis and machine intelligence*, Vol.8, No.6,pp.679-698, 1986
- [9] R.Szeliski, H.Y.Shum, "Creating Full View Panoramic Image Mosaics and Environment Maps", *Computer Graphics*, pp.251-258, August 1997.
- [10] David G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, Vol. 60, No. 2, pp. 5-16, Jan. 2004