

모폴로지 코너 검출을 이용한 고속 파노라마 비디오 제작 기법

정회원 이 정 호*, 이 관 수*, 양 원 근*, 진 주 경*, 종신회원 정 동 석**

Fast Panoramic Video Generation Method Using Morphological Corner Detection

Jungho Lee*, Kwansu Lee*, Wonkeun Yang*, Jookyung Jin* *Regular Members*,
Dongseok Jeong** *Lifelong Member*

요 약

본 논문은 다수의 카메라로부터 입력받은 동영상을 하나의 파노라마 동영상으로 합치는 기법을 제안한다. 서로 인접하면서 겹치는 영상들을 하나의 큰 영상으로 만들어주는 파노라마 영상은 사진 측량법, 위성사진, 컴퓨터 그래픽 등의 분야에서 응용된다. 일반적으로 영상을 모자이크할 때 발생하는 왜곡을 최소화하기 위해 투시 변환(perspective transformation)을 사용하는데, 변환 추정에는 필요한 특징점을 얻어내는 방법이 결과물의 품질을 결정하게 된다. 본 논문에서는 코너점을 특징점으로 사용했으며, 결과물의 품질이 우수하고 연산 속도가 빠르게 하기 위해 모폴로지 구조를 사용해서 코너점을 추출하였다. 그리고 거의 모든 상황에서 안정적으로 코너점을 검출하기 위해 코너점의 강도를 구분하는 방법을 이용했다. 인접한 영상의 대응점으로부터 추정된 8-변수 투시 변환 값으로 영상 모자이크를 했으며, 결과 영상의 잔상을 제거하기 위해 쌍일차 색상혼합을 적용했다. 실험결과 제안한 방법은 여러 조건에서 빠른 속도와 좋은 화질의 결과를 보였다.

Key Words : Panorama, Mosaic, Perspective

ABSTRACT

This paper proposes a method of building a panoramic video from several videos captured from adjacent cameras. The panoramic image which constructed from adjacent and overlapped images is used for photogrammetry, satellite photo or many computer graphic applications. The perspective transformation, which is estimated from the appropriate corresponding pairs of images, can be used to construct the panoramic image without unwarranted distortion and its quality is decided by how to find the features needed for transform estimation. We used the corner points for the corresponding features, and morphological structures were utilized for fast and robust corner detection. We used the criterion of the corner strength, which guarantees the robust detection of the corner in most situations. For the transformation, 8 parameters were estimated from perspective equations which use matched points of adjacent images, and bilinear color blending was used to construct a seamless panoramic video. The experiments showed that the proposed method yields fast results with good quality under various conditions.

* 인하대학교 전자공학과 멀티미디어 연구실 (*{julian, kwansu78, jukyong77, aida}@inhaian.net, ** dsjeong@inha.ac.kr)
논문번호 : KICS2005-12-491, 접수일자 : 2005년 12월 9일, 최종논문접수일자 : 2006년 3월 22일

I. 서론

고화질의 파노라마 영상을 자동으로 제작하는 기술은 사진 측량법, 컴퓨터 비전, 컴퓨터 그래픽스 등에서 활발히 연구되고 있는 분야이다. 이 기술을 바탕으로 생성된 영상은 항공, 위성사진 합성, 화면 안정화, 변화 검출, 비디오 압축, 그리고 비디오 인덱싱 등에서 카메라의 화각과 화질을 개선하기 위해 사용되고 있다¹⁻⁵. 파노라마 영상을 얻기 위해 몇 가지 기술들이 개발되었는데, 가장 간단한 하드웨어 기반의 기술은 화각 자체가 넓은 어안렌즈를 사용하는 것이다. 또 다른 하드웨어 기반의 기술 중, 긴 필름에 원통형의 파노라마 영상을 저장하는 방법이 있다⁶. 그러나 이런 방법들은 그 설정을 바꾸기가 쉽지 않고 영상에 왜곡이 많다. 때문에 컴퓨터 그래픽스처럼 실시간으로 결과가 필요하지 않은 경우에는 영상 모자이크 혹은 영상 재단(stitching) 방법이 널리 사용되고 있다. 템플릿 정합, 에지 검출, 또는 투시 변환 등을 사용하는 영상 모자이크 방법을 이용하면, 하드웨어의 선택을 자유롭게 할 수 있고 결과 영상의 왜곡 또한 줄일 수 있다^{7, 9, 12}. 대부분의 영상 모자이크 방법들은 투시 변환을 사용하기 위해, 부분적으로 겹쳐진 이웃한 영상에서 대응점을 찾는다^{7, 8, 12, 13}. 템플릿 정합을 이용한 SA(simulated annealing)¹⁴방법이나 윤곽에서 높은 곡률점을 찾는 방법¹⁷ 등은 가장 널리 사용되는 방법들이다. 그러나 이 방법들은 구현에 필요한 연산량이 많기 때문에, 비디오 압축에서의 변화 검출 또는 배경 추출과 같이 속도가 중요한 응용에 적용할 경우 심각한 단점이 된다.

본 논문에서는 대응점 탐색을 위해 모폴로지를 이용한 코너 검출을 사용하여 효율적이고 빠른 영상 모자이크 방법을 제안한다. 그리고 이를 이용하여 파노라마 비디오 제작 시스템을 구현하였다. 모폴로지를 이용한 코너 검출¹¹은 팽창연산(dilation)과 침식연산(erosion)을 이용하여 코너의 강도를 표현하는 방법이다. 그러나 기존의 방법은 특정 코너 지점을 명확하게 찾는 것이 아니라 각 화소가 가지고 있는 코너적 특성만을 표시했기 때문에, 이웃한 화소의 정보가 고려되지 않아서 상황에 따라 너무 많거나 너무 적은 코너가 검출되는 단점을 가지고 있었다. 따라서 본 논문에서는 이런 단점을 보완하기 위해 코너강도 지도(corner strength map)에서 국부적 첨단(local peak)을 찾는 방법을 제안했다. 또한 두 영상에서 찾은 코너점에서 대응점을 효과

적으로 찾기 위해 템플릿 정합뿐만 아니라 코너점 간의 상대적인 거리 및 이웃한 코너점과의 상관관계 등을 이용했다. 구현된 시스템을 이용하여 실험한 결과 제안한 방법은 속도가 빠르면서도 모자이크 영상의 왜곡이 적은 결과를 보였다.

2장에서는 영상 모자이크 방법에 대해 설명하고, 제안된 방법에 사용된 자세한 알고리즘은 3장과 4장에서 설명한다. 5장에서는 구현된 시스템과 실험 결과에 대해 설명하고 향후 방향에 대해 이야기한다.

II. 영상 모자이크의 진행 순서

같은 시간 및 장소에서 이웃한 카메라 혹은 수평 회전(pan)하는 카메라로부터 얻은 몇 장의 영상으로 이용하여 모자이크 과정을 거치면 파노라마 영상을 얻게 된다. 모자이크를 할 때 서로 이웃한 영상들은 어느 정도의 겹쳐져있는 구간이 있어야만 한다. 제안하는 모자이크 방법은 크게 네 부분으로 구성된다.

첫번째, 이웃한 영상에서 겹쳐진 구간을 측정해야 한다. 이 구간은 뒤에 수행하게 될 알고리즘을 적용할 경계가 된다. 이웃한 여러 대의 카메라에서 얻은 영상들, 또는 수평 회전하는 카메라로부터 얻은 영상들은 중복되어 촬영된 구간이 존재한다. 뒤에서 자세히 알아보겠지만, 중복된 구간의 범위를 알아내게 된다면 대응점을 찾는 시간이 많이 감소된다. 두 번째, 이웃한 두 영상의 겹쳐진 영역에서 서로 대응되는 네 개의 대응점을 결정한다. 이 대응점들은 간단한 코너점 검출 방법 및 패턴 정합을 이용하여 결정된다. 세 번째, 앞서 구해진 네 개의 대응점의 상대적인 좌표를 이용하여 투시 변환의 매개 변수를 계산한다. 간단하게 첫 번째 단계에서 얻은 정보만을 이용하여 영상을 이어 붙여서 파노라마 영상을 만들 경우, 렌즈의 왜곡으로 인해 결합된 영상이 어긋나 보이게 된다. 렌즈의 왜곡은 화각이 넓어질수록 심해지며, 촬영면의 중심부 보다는 바깥쪽에 맺히는 영상의 왜곡이 더 심하다. 이런 왜곡을 제거하기 위해 투시 변환을 사용하는데, 평행 이동, 회전, 크기, 어파인 변환 등을 모두 표현할 수 있는 장점 때문에 일반적으로 8-변수 변환이 이용된다^{7, 8, 12, 13}. 이 변환의 변수 값들은 네 쌍의 대응점 좌표로부터 수학적 근사를 이용하여 얻어진다. 마지막 단계는 색상 혼합으로, 변환된 영상을 하나로 합할 때 렌즈의 차이나 조명의 변화 등으로 발생한 불연속적인 부분을 제거하기 위해 쌍일차 변환을 이용한다. 제안된 모자이크 방법의 블록 다이

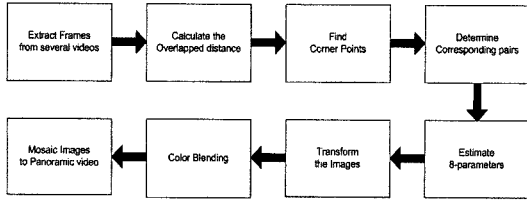


그림 1. 파노라마 비디오를 제작하는 과정의 블록 다이어그램
Fig. 1. A block diagram of the process for panoramic video generation

어그램이 그림 1에 있다.

Ⅲ. 대응점 탐색

이번 장은 인접한 영상에서 겹쳐진 부분의 길이를 계산하는 것과 코너점 검출, 그리고 검출된 코너 점을 이용한 템플릿 정합에 대해 설명한다.

3.1 중복 영역 계산

주어진 영상에서 겹쳐진 부분이 어디인지 알게 되면, 대응점을 탐색할 영역을 한정시킬 수 있게 되어 계산량을 줄이는 효과가 있다. 보통 이웃한 영상들끼리 10%~30% 내외의 영역이 겹쳐지기 때문에, 전체적으로 볼 때 상당한 계산량 감소를 얻을 수 있으며 대응점의 정확도도 어느 정도 향상시킬 수 있게 된다. 본 논문에서는 중복 영역을 계산하기 위해 평균 제곱 오차(Mean Square Error)를 사용했다. 이웃한 두 영상에서 겹치는 부분의 길이를 한 화소씩 늘려가며 이동시켰을 경우, 겹쳐진 부분의 평균 제곱 오차는 두 영상이 정확히 겹쳐졌을 때 가장 작게 된다. 그리고 카메라를 설치할 때, 겹치는 부분의 대략적인 길이를 알 수 있다면 계산량을 줄이는데 도움이 된다. 본 논문에서 사용한 평균 제곱 오차를 식 (1)에 보인다.

$$\sigma_{MSE}^2 = \frac{1}{OA} \sum_{n=0}^{height} \sum_{m=0}^{OL} |u(m, n) - u'(OL - m, n)|^2 \quad (1)$$

OA: Overlap Area, OL: Overlap Length.

$u(m, n)$ 은 기준이 되는 영상의 그레이 값이고, $u'(m, n)$ 은 이웃한 영상의 그레이 값이다. 또한 OA는 겹쳐진 부분의 화소 숫자로 영상을 이동시킴에 따라 증가하게 된다. 이 방법을 이용할 경우 비교적 간단하게 구현할 수 있으나, 영상이 큰 경우 계산량이 매우 많아지는 단점이 있다. 그래서 이를 해결하기 위해 영상의 크기를 변화시키며 계산을 수행하

는 영상 피라미드 구조를 적용했다. 우선 주어진 영상들의 면적을 1/16 크기로 줄인다. 그리고 이웃한 영상이 대략 어느 정도 겹쳐지는지의 설치정보와 평균 제곱 오차를 이용하여 중복 영역을 계산한다. 1/16 영상에서 얻은 중복 영역을 기준으로 1/4 영상에서 조금 더 정밀하게 중복영역을 계산한다. 마지막으로 1/4 영상에서 계산한 중복 영역의 길이를 이용하여, 원 영상의 중복 영역을 가장 정밀한 정도로 계산하게 된다.

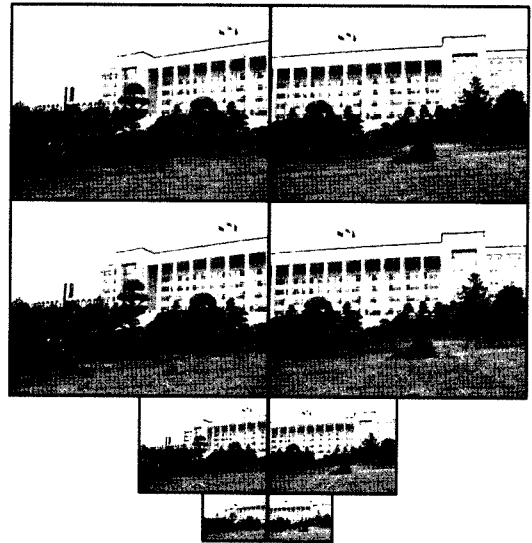


그림 2. 원본영상으로부터 면적을 1/4씩 줄여가며 만든 피라미드 영상. 가장 작은 면적에서부터 중복된 영역을 계산하기 시작해서 점차적으로 정확도를 높여나간다

Fig. 2. Pyramid images with scaled down into quarter size of area. Overlapped area is calculated from the smallest size and gradually increase the accuracy

이때 사용하는 1/16 및 1/4 영상의 크기 배열이 그림 2처럼 피라미드와 같은 형태를 띤다고 해서 영상 피라미드라는 이름을 가지게 되었다^[10]. 이 방법을 이용할 경우, 원 영상에서 직접 계산하는 방법에 비해 절반 이하의 계산량으로도 중복 영역의 길이를 얻을 수 있다.

3.2 코너점 검출

영상을 모자이크하기 위해서는 특정 카메라로부터 얻은 영상을 변형시켜야 하는데, 이를 위해 크기 변환이나 회전 변환, 어파인 변환 또는 투시 변환 등을 이용한다. 이런 변환들은 각 영상간의 위치적 상관관계를 알아내기 위해 몇 쌍의 대응점이 필요하며, 이를 찾아내기 위해서 경계(edge), 센서스 변환값, 굴곡점 또는 코너점 등을 이용할 수 있다. 코

너점은 열거한 특징 값들 중에서도 모자이크 될 영상의 명암차이나 영상의 왜곡에 강하다는 장점이 있다. 게다가 검출에 필요한 계산량이 적어서 고속으로 처리할 수 있기 때문에, 대응점을 찾아내기 위한 특징 단위로써 코너점을 이용하게 되었다. 같은 장면을 서로 다른 카메라로 촬영하더라도 담고 있는 물체나 배경의 구성이 같기 때문에 코너점의 분포 또한 매우 유사하다. 따라서 어떤 카메라로 촬영한 영상의 특징 코너점은 다른 카메라로 촬영한 영상의 어떤 코너점과 서로 대응될 수 있다. 물론 영상을 촬영한 조건에 따라 대응이 되지 않고 어느 한쪽 영상에만 나타나게 되는 코너점도 존재하지만, 이런 부분은 후처리를 거치면서 제거할 수 있다.

코너점을 검출하는 방법은 여러 가지가 있는데, 적은 계산량으로도 검출률을 좋게 하기위해 비대칭 모폴로지 닫힘 연산^[11]을 이용한 방법을 사용했다.

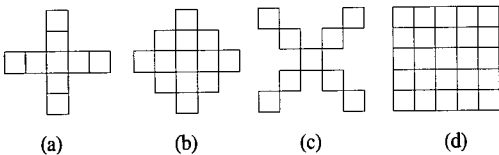


그림 3. 코너검출에 사용된 모폴로지 마스크 (a) cross, (b) diamond, (c) X, (d) square
Fig. 3. Morphological Masks used for corner detection

이 방법은 그림 3과 같은 네 개의 마스크를 이용하여 코너의 강도를 계산하는 것으로 그레이 영상에 적용했을 때, 코너일 확률이 높은 화소는 코너 강도 값이 높게 된다. 코너의 강도는 마스크들을 이용하여 비대칭적으로 닫힘 연산을 하고 그 값들의 차이로 결정되게 된다.

비대칭적인 닫힘 연산(asymmetric closing)은 식 (2)와 같이, 어떤 마스크를 이용하여 팽창연산(dilation)을 적용한 결과에 다른 마스크를 이용하여 침식연산(erosion)을 함으로써 얻어지게 된다. 이때 그레이 영상에 대한 팽창연산은 이진 영상에서의 팽창연산과는 다르게 마스크의 중심에 위치한 화소의 값을 마스크 영역에서 가장 큰 그레이 값으로 대체해주는 방법이다. 그레이 영상에 대한 침식연산도 마찬가지로 마스크의 중심에 위치한 화소의 값을 마스크 영역에서 가장 작은 그레이 값으로 바꿔주는 방법이다. 식 (2)에 비대칭 닫힘 연산의 한 방법을 제시하였다.

$$I_{cross, diamond}^c = \left(I_{cross}^\delta \right)_{diamond}^\epsilon \quad (2)$$

코너의 강도는 식 (3)에 나와 있는 것처럼, 주어진 영상에 그림 3(a)와 그림 3(b)의 마스크로 비대칭 닫힘연산을 한 결과와 그림 3(c)와 그림 3(d)의 마스크로 비대칭 닫힘연산을 한 결과의 차이 값으로 계산할 수 있다.

$$C_{cross, X}(I) = \left| I_{cross, diamond}^c - I_{X, square}^c \right| \quad (3)$$

이 방법은 어떤 화소에 대한 닫힘 연산의 과정값들을 저장해서, 이웃한 다른 화소의 닫힘 연산을 계산할 때 이를 이용하는 방식으로 계산량을 줄일 수 있다. 또한 코너 검출에 필요한 서로 다른 두 개의 비대칭 닫힘 연산은 상호간에 관계가 없으므로 스레드나 멀티 프로세스를 이용하여 속도를 높일 수 있다.

실험을 통한 결과를 보면, 실제 코너와 유사한 부분의 코너 강도값이 크게 나오는 것을 알 수 있다. 그러나 연산의 특성에 의해 코너에 해당하는 화소뿐만 아니라, 코너 근처의 다른 화소들도 큰 값을 가지게 된다. 따라서 코너의 강도가 그림 4처럼 봉우리 형태로 이루어져 있는 것을 알 수 있다.

모폴로지를 이용한 기존의 방법은 이런 코너 강도에 특정한 문턱 값을 적용하여 코너점을 검출했다. 이런 방법은 단순하다는 장점이 있는 반면, 영상의 조건이 변할 경우 그 결과가 달라진다. 즉, 상황에 따라 코너점이 과잉 검출되거나 검출된 숫자가 부족하게 되며, 인접한 지역에 많은 코너점들이 몰릴 수도 있다. 검출된 코너점의 개수는 대응점을 찾는데 있어서 매우 중요한 역할을 한다. 코너점의 숫자가 너무 많을 경우 연산시간이 많이 걸리며, 여러 코너점들이 겹쳐서 대응되기 때문에 정확한 대

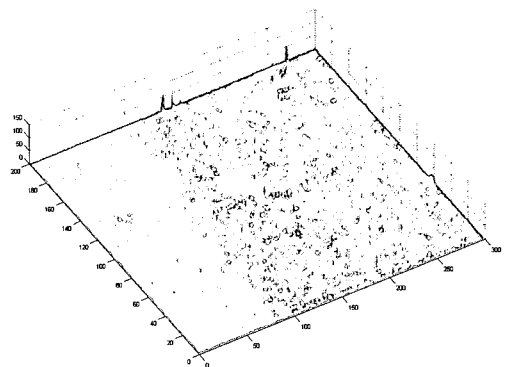


그림 4. 모폴로지 닫힘연산의 차이값을 표시한 코너강도 지도
Fig. 4. A corner strength map which represents the difference between morphological closings

응점을 결정하는데 어려움이 있다. 코너점이 불충분할 경우에는 대응점의 후보가 줄어들게 되므로 투시 변환의 변수 값을 유추할 수가 없다. 따라서 어떤 화소를 코너로 검출할지를 결정하기 위한 척도가 필요하며, 본 논문에서는 코너 강도지도에서 국부적 점단을 검출하는 방법으로 이를 해결했다. 이 방법은 그림 4에서 봉우리 부분에 해당하는 화소를 찾는 방법으로, 재귀 함수를 이용하여 계산량이 많지 않으면서 좋은 결과를 보였다. 그림 5에서 제안한 방법으로 찾은 코너점과 기존의 방법을 비교해 볼 수 있다.

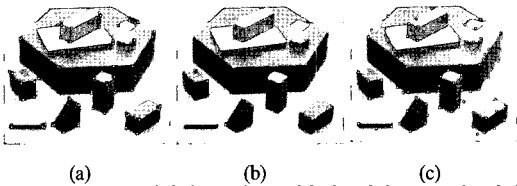


그림 5. Blox 영상의 코너를 검출한 결과 (a) 기존방법 (Threshold=20), (b) 기존방법 (Threshold=40), (c) 제안한 방법
 Fig. 5. Detected corners of Blox image (a) existing method (Threshold=20), (b) existing method (Threshold=40), (c) proposed method

3.3 템플릿 정합

투시 변환에 사용되는 8-변수를 예측하기 위해서는 모자이크된 두 영상에서 대응점의 상대적 좌표 4쌍이 필요하다. 이전에 설명했던 코너점은 단지 두 영상의 국지적인 특징들을 뽑아낸 것이기 때문에 한 영상의 어떤 코너점이 다른 영상의 어떤 코너점과 대응된다는 정보는 존재하지 않는다. 따라서 두 영상에서 추출한 코너점들의 상관관계를 알아내야 하고, 본 논문에서는 템플릿 정합을 이용했다.

먼저 좌측에 해당하는 영상에서 얻어낸 코너점 중 하나를 선택하고, 그 코너점을 중심으로 특정 크기의 템플릿을 얻어낸다. 그리고 오른쪽에 해당하는 영상의 모든 코너점에 대해서, 각 코너점을 중심으로 템플릿과 크기가 같은 샘플들을 뽑아내어 템플릿과 MSE(Mean Square Error)를 비교하는 방식으로 대응점을 찾아내게 된다. 이 방법은 간단하면서도 그 결과가 비교적 정확하지만, 모든 코너점에 대해서 수행할 경우 계산량이 많아진다는 단점이 있다. 여기서는 이를 보완하기 위해 템플릿의 코너점과 샘플의 코너점간의 거리를 이용했다. 즉, 두 영상의 겹쳐진 영역에 대한 좌표계에서 어떤 템플릿의 좌표에서 비교될 샘플의 좌표까지의 거리가 특정 문턱치 이상의 값을 가지게 될 경우에는 정합을 수행

하지 않도록 하는 것이다. 이렇게 하면 템플릿의 코너점과 상대적으로 인접한 부분의 샘플들만을 이용하게 되므로 연산량이 상당히 줄일 수 있으며, 실험에서는 거리 측정을 위해 euclidean distance를 사용했다.

일반적으로 서로 대응되는 코너점간의 템플릿 정합은 아주 적은 MSE 값을 가진다. 그러나 어느 한 쪽 영상에만 존재하는 고립 코너(isolated corner)의 경우에는 대응되는 템플릿이 있다고 하더라도 MSE 값이 커지게 된다. 그래서 좌표간의 거리와 함께 MSE값 또한 대응점을 판별하는데 이용했다.

템플릿 정합을 끝마치게 되면 여러 개의 대응점들이 찾아지게 되는데, 이 중 투시 변환의 변수값을 찾아내기 위해서 4개의 대응점을 선택해야 한다. 일반적으로 어떤 렌즈를 이용하여 촬영한 영상이든지 구조적으로 왜곡을 가지고 있으며, 예측으로 얻은 투시 변환의 변수들은 실수 값이기 때문에 완전한 투시 변환을 할 수는 없다. 그렇지만 실험결과 4개의 대응점이 이루고 있는 사각형의 크기가 크고, 그 형태가 직사각형에 가까울수록 변환시에 발생하는 왜곡의 정도가 줄었다. 따라서 템플릿 정합으로 얻은 여러 개의 대응점 중에서 면적이 넓으면서 직사각형에 가까운 형태가 되도록 4개의 대응점을 결정하는 과정이 필요하다. 이를 위해 두 영상이 겹쳐져 있는 영역을 그림 6과 같이 4개의 사분면으로 분할하여, 중심점을 기준으로 거리가 멀면서 상대적으로 직사각형의 형태를 이룰 수 있는 대응점들을 선택했다.

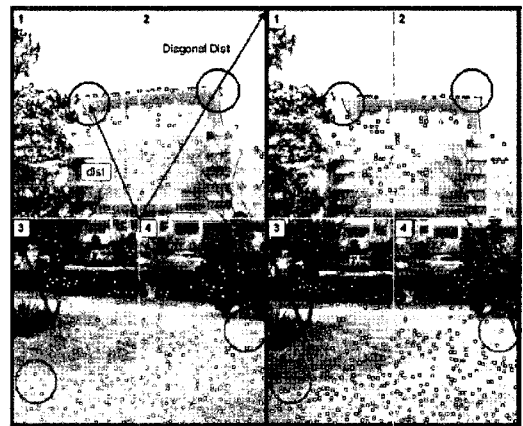


그림 6. 두 영상의 겹쳐진 영역에서 대응점을 찾은 결과. 원으로 표시한 지점이 대응점들이며, 작은 점으로 표시된 부분들은 코너점들이다
 Fig. 6. A result of detected matching points from overlapped area of two adjacent images. Detected points are marked with a circle, and the rest of detected corners are marked with small points

IV. 투시변환 및 색상 블렌딩

투시 변환은 2차원의 사변형을 다른 형태로 맵핑한다. 일반적으로 파노라마 영상을 만들기 위해서 재료 영상을 투시 변환으로 변형한 후에 모자이크 연산을 하게 된다. 투시 변환은 식 (4)와 같이 표현할 수 있다. 이때 I' 은 변환된 결과 영상 $(X, Y, 1)$ 이고, i 는 변환되기 전의 영상 $(x, y, 1)$ 이며, H 는 변환 행렬이다.

$$I' = Hi \tag{4}$$

이를 좀 더 자세히 표현하면 식 (5)와 같은 형태로 된다.

$$\begin{bmatrix} XW \\ YW \\ W \end{bmatrix} = \begin{bmatrix} h_0 & h_1 & h_2 \\ h_3 & h_4 & h_5 \\ h_6 & h_7 & h_8 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{5}$$

보통 $h_8 = 1$ 로 간주하며 $W = h_6x + h_7y + 1$ 이다. 위로부터 투영된 결과를 유도하면 식 (6) 및 식 (7)과 같이 된다.

$$X = \frac{h_0x + h_1y + h_2}{h_6x + h_7y + h_8} \tag{6}$$

$$Y = \frac{h_3x + h_4y + h_5}{h_6x + h_7y + h_8} \tag{7}$$

변환 행렬 H 의 변수들은 직교 좌표계에서 X, Y, Z 방향으로의 자유로운 변환을 가능하게 하며, 이들을 조합하면 크기변환, 회전변환, 어파인 변환 등이 모두 가능하다. 따라서 변환 행렬의 변수 중 1로 값이 정해져있는 h_8 을 제외한 8개의 변수를 추정하면 어떤 영상의 좌표 (x, y) 를 (X, Y) 로 변환할 수 있다. 식 (6) 및 식 (7)에 $h_8 = 1$ 을 이용하면 식 (8) 및 식 (9)와 같이 정리할 수 있다.

$$X = h_0x + h_1y + h_2 - h_6Xx - h_7Xy \tag{8}$$

$$Y = h_3x + h_4y + h_5 - h_6Yx - h_7Yy \tag{9}$$

이전 단계에서 찾은 대응점을 식 (8)과 식 (9)의

(X, Y) 및 (x, y) 에 적용하면 식 (10)처럼 확장할 수 있다.

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -X_1x_1 & -X_1y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -Y_1x_1 & -Y_1y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -X_2x_2 & -X_2y_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -Y_2x_2 & -Y_2y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & 1 & 0 & 0 & 0 & -X_nx_n & -X_ny_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -Y_nx_n & Y_ny_n \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \end{bmatrix} = \begin{bmatrix} X_1 \\ Y_1 \\ X_2 \\ Y_2 \\ \vdots \\ \vdots \\ \vdots \end{bmatrix} \tag{10}$$

n 은 주어진 대응점의 개수이고, 4개 이상의 대응점이 주어지면 최소 제곱 추측법(Least square estimation)을 이용하여 변환행렬 H 의 변수 값을 계산할 수 있다.

카메라 사이의 상대적인 위치가 변하지 않는 경우에는 그와 관련된 투시 변환의 변수들이 그대로 유지된다고 가정한다. 그렇지만 진동이나 충돌 등으로 인해 카메라 마운트가 흔들릴 수도 있으며, 정지 영상이 아닌 동영상에서는 카메라 사이의 위치가 바뀔 가능성이 있기 때문에 시간에 따른 보정이 필요하다. 이를 위해 변환행렬 H 를 식 (11)과 같이 순환적으로 갱신되도록 바꿀 수 있다.

$$H_n = H_{n-1}(I + D_n) \tag{11}$$

$$\begin{bmatrix} d_0 & d_1 & d_2 \\ d_3 & d_4 & d_5 \\ d_6 & d_7 & d_8 \end{bmatrix} \tag{12}$$

여기에서 H_{n-1} 은 갱신하기 전의 변환행렬이고, I 는 단위행렬이며 D_n 은 갱신을 위한 행렬 변화값으로 식(12)와 같이 표현할 수 있다. 주어진 영상을 갱신된 방정식 $X' = H_{n-1}(I + D_n)x$ 을 이용해 변환한 결과와 $X = H_{n-1}x$ 로 변환한 영상을 다시 $X'' = (I + D_n)x$ 으로 변환한 결과는 같게 되므로 식 (6)과 식 (7)은 갱신 과정에서 식 (13) 및 식 (14)와 같이 표현할 수 있다.

$$x'' = \frac{(1+d_0)x + (1+d_1)y + d_2}{d_6x + d_7y + (1+d_8)} \tag{13}$$

$$y'' = \frac{(1+d_3)x + (1+d_4) + d_5}{d_6x + d_7y + (1+d_8)} \quad (14)$$

파노라마 비디오를 만들려는 동영상에서 정해진 개수의 프레임이 지날 때마다 갱신 행렬 D_n 을 구하여 원래의 변환행렬을 갱신해주면, 카메라의 미세한 변화에도 좋은 품질의 결과영상을 얻을 수 있다.

기본적으로 투시 변환은 2차원의 사변형을 다른 형태로 변형시킨다. 또한 파노라마 영상을 만들기 위해서 재료 영상을 투시 변환으로 변형한 후에 모자이크 연산을 하게 된다. 그러나 투시 변환을 이용하여 영상을 변형시킨다하더라도 렌즈에 의해 필연적으로 발생하는 왜곡으로 인해서, 겹친 영역의 외곽선들이 일부분 일치하지 않는 현상이 발생한다. 이를 가리켜 제 2의 영상 또는 고스트 이미지라고 하며 그림 7(a)에 보이는 것처럼 동일한 외곽선이 작은 거리를 두고 겹쳐져 나타나게 된다.

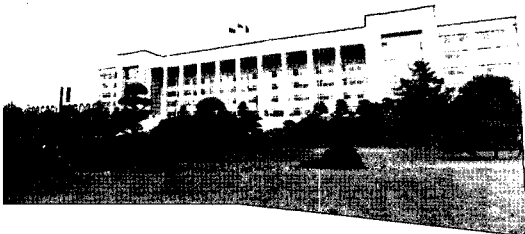
이를 제거하기 위해서 두 영상이 겹쳐지는 영역의 평균 화소값을 계산하여 대체하는 방법이 있지만, 이 방법은 간단한 대신 고스트를 완벽하게 제거하지 못한다. 본 논문에서는 겹친 영역 전반에 걸쳐

고스트 이미지를 줄이기 위해 쌍일차 색상 혼합법을 사용한다. 쌍일차 색상 혼합은 모자이크할 좌우 영상에서 겹쳐질 부분의 좌우경계의 좌표를 각각 0과 W로 놓고, 겹쳐질 화소의 수평적 거리에 따라 다른 가중치를 주어 해당 위치의 화소값으로 대체하는 방법이다. 그림 7(b)에 쌍일차 색상 혼합의 결과를 제시하였다.

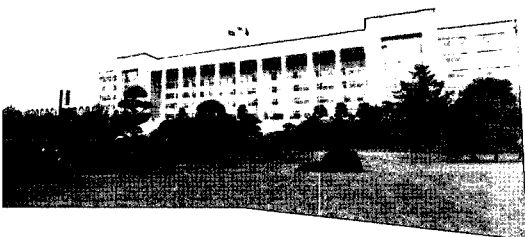
V. 실험결과 및 결론

본 논문에서는 다수의 동영상을 하나의 파노라마 동영상으로 만드는 방법을 제안했다. 이를 검증하기 위해 같은 지지대에 3대의 프로그래시브 CCD 카메라를 장착하여 실험영상을 취득했다. 실험에 사용된 동영상의 해상도는 704*480이며, 입력을 실시간으로 디지털 변환하기 위해 다채널 입력 MPEG-2 영상보드를 이용했다. 초당 프레임 레이트는 30fps이며, 실외 환경에서 조명의 변화를 줄이기 위해 CCD입력 신호 레벨을 이용한 자동 조리개를 렌즈 앞에 장착했다. 또한 카메라 간에 겹치는 영역을 조절할 수 있도록 했다.

그림 8(a)는 지지대에 장착된 모습이며, 그림 8(b)는 이를 자동차 지붕에 연결한 모습이다.



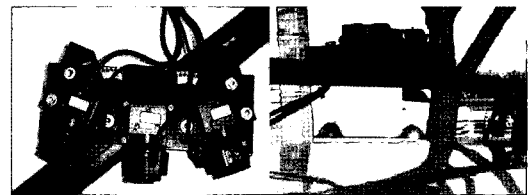
(a)



(b)

그림 7. 색상 블렌딩을 적용한 결과. (a) 모자이크된 영상으로 겹쳐진 부분에 고스트 이미지가 존재, (b) 색상 블렌딩을 적용한 결과로 세로선이 제거됨

Fig. 7. A result of color blending (a) a mosaic image that some vertical ghost image is exist at the overlapped region, (b) vertical ghost is removed after color blending



(a)

(b)

그림 8. 동영상 취득을 위한 카메라. (a) 각도 조절이 가능한 마운트, (b) 차량에 연결한 카메라 마운트

Fig. 8. A camera system for video acquisition. (a) the camera mount that could control the angel, (b) camera mount attached on a car

그림 9는 차량에 장착한 카메라로부터 얻은 동영상을 제안한 방법으로 모자이크한 결과이며, 그림 10은 디지털 카메라를 이용하여 촬영한 영상을 모자이크한 결과이다. 알고리즘은 Microsoft Visual C++ 6.0을 이용하여 구현하였고, 실험에 쓰인 컴퓨터는 2.6GHz로 동작하는 Intel Pentium 4 CPU이다.

표 1은 제안한 방법을 동일한 실험조건으로 템플릿 정합 및 SA(Simulated Annealing)를 이용한 방법과 비교한 결과이다. 표 1에 제시된 실험은 장면이 복잡한 영상, 중간정도의 영상, 복잡하지 않은 영상에 대해서 각각 무작위로 선택된 200쌍씩, 총

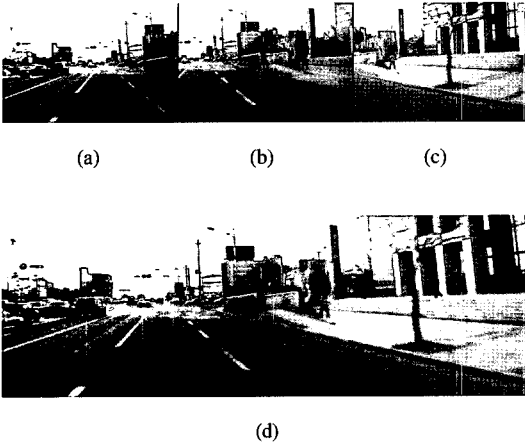


그림 9. 차량에서 촬영한 영상의 모자이크 결과 (a) 왼쪽 동영상에서 추출한 프레임, (b) 중앙 동영상에서 추출한 프레임, (c) 오른쪽 동영상에서 추출한 프레임, (d) 모자이크된 파노라마 영상
 Fig. 9. A result of mosaic captured from vehicle camera (a) a frame extracted from left video, (b) a frame extracted from center video, (c) a frame extracted from right video, (d) a mosaic panoramic image



그림 10. 카메라를 Panning하며 촬영한 영상의 파노라마 합성
 Fig. 10. A panoramic composition of several images which taken by panning camera

표 1. 제안한 방법과 템플릿 정합을 이용한 Simulated Annealing과의 실험결과 비교. 실험은 총 600쌍의 영상에 대해 수행되었고, 계산시간은 동일환경에서 밀리초(millisecond)단위로 측정했다.
 Table 1. Result comparison of proposed method and simulated annealing with template matching. 600 pairs of images are used for experiments, and processing time is measured with millisecond at the same environment

		제안한 방법	TM with SA
계산시간 (ms)	평균값	783.5	1407.25
	표준편차	339.99	31.58
	최소값	359	1312
	최대값	1906	1787
화질 (SAD)	평균값	40.45	4486
	표준편차	2.00	2.38
	최소값	26.26	33.21
	최대값	58.12	60.65

600장의 정지영상들을 사용했다. 이때, 결과물 화질의 비교를 위해 색상 블렌딩을 하기 전에 SAD(Sum

of Absolute Difference)를 구했으며, 계산시간은 파일 입력시점부터 투시 변환의 변수들을 구한 시점까지의 시간을 측정했다. 논문의 목적과 부합하기 위해서는 동영상 생성에 걸린 시간까지 측정해야하나 동영상은 목적으로 둔 기존의 방법이 없으며, 동영상 압축 방법에 따라 계산시간이 크게 변하기 때문에 정지영상에 대한 결과를 제시하였다.

실험결과 제안한 방법은 SA방법과 비교하여 약 44%정도 계산시간이 절약되었으며, SAD로 측정된 화질도 적은 양이지만 감소한 것을 확인할 수 있었다.

표 1에서 보면 계산시간에서 SA방법에 비해 상당히 큰 표준편차를 가지고 있는데, 이는 화면의 복잡도에 따라 영상에서 검출되는 코너점의 갯수가 변하기 때문이다.

우리는 두 장의 이웃한 영상의 상관관계를 추출하여, 빠르고 신뢰성있는 영상 모자이크 방법을 제안했다. 영상이 기본적으로 가지고 있는 코너점을 이용하여, 카메라 사이에 밝기가 차이나거나 명도, 색상이 일치하지 않는 등의 조건에서도 신뢰할만한 결과를 보였다. 그러나 코너점의 분포에 따라 계산시간이 일정하지 않기 때문에 각 영상에서 추출된 코너점의 숫자를 적절하게 조절하는 방법을 고안해야 할 것이다.

참고 문헌

- [1] Kenney, C.S., Manjunath B.S., "A condition number for point matching with application to registration and postregistration error estimation," IEEE Transaction on Pattern Analysis and Machine Intelligence(PAMI), Vol. 25, pp. 1437-1454, 2003.
- [2] M.Hansen, P.Anandan, K.Dana, G.van der Val, P.Burt, "Real-time scene stabilization and mosaic construction," IEEE Workshop on Application of Computer Vision, pp. 54-62, 1994.
- [3] A. John, R. Peter, "Electric Communication Development," Communications of the ACM, 40, pp. 71-79, May 1997.
- [4] M.Irani, S.Hsu, P.Anandan, "Video compression using mosaic representations," Signal Processing: Image Communication, Vol. 7, pp. 529-552, 1995.
- [5] S.Mann, R.W.Picard, "Virtual bellows: Constructing high quality images from video" International Conference on Image Process-

ing, Vol. 1, pp. 363-367, 1994.

- [6] J.Meehan, Panoramic Photography, Watson-Guptill, 1990.
- [7] Hui Chen, Wenping Wang, Ralph Martin, "Building Panoramas from Photographs Taken with an Uncalibrated Hand-held Camera," Proceedings of Vision, Modeling and Visualization 2000(VMV00), pp. 221-230, 2000.
- [8] Smolic A., Wiegand T., "High-resolution video mosaicing," Image Processing Proceedings International Conference on, Vol. 3, pp. 872-875, 2001.
- [9] Chiou-Ting Hsu, Tzu-Hung Cheng, Beuker, R.A., Jyh-Kuen Horng, "Feature-based video mosaic," Image Processing Proceedings International Conference on, Vol. 2, pp. 887-890, 2000.
- [10] Whichello A.P., Hong Yan, "Document image mosaicing," Pattern Recognition Proceedings Fourteenth International Conference on, Vol. 2. pp. 1081-1083, 1998.
- [11] Laganieri R., "Morphological corner detection," Computer Vision Sixth International Conference on, pp. 280-285, 1998.
- [12] Jun-Wei Hsieh, "Fast stitching algorithm for moving object detection and mosaic construction," Image and Vision Computing, Vol. 22, Issue 4, pp. 291-306, 2004.
- [13] H.Y. Shum, R. Szeliske, "Construction of Panoramic Image Mosaics with Global and Local Alignment," Panoramic Vision, Springer-Verlag New York, pp. 227-268, 2001.
- [14] Dong-Keun Kim, Byung-Tae Jang, Chi-Jung Hwang, "A planar perspective image matching using point correspondences and rectangle-to-quadrilateral mapping," Image Analysis and Interpretation 2002 Proceedings, pp. 87-91, 2002.

이 정 호 (Jungho Lee)

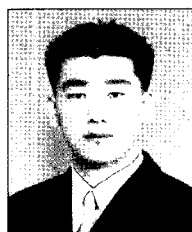
정회원



1999년 2월 인하대학교 전자공학과 졸업
 2001년 2월 인하대학교 전자공학과 석사
 2001년 3월~현재 인하대학교 전자공학과 박사과정
 <관심분야> 영상처리, 영상압축

이 관 수 (Kwansu Lee)

정회원



2004년 2월 인하대학교 전자공학과 졸업
 2006년 2월 인하대학교 전자공학과 석사
 <관심분야> 패턴인식, 영상처리

양 원 근 (Wonkeun Yang)

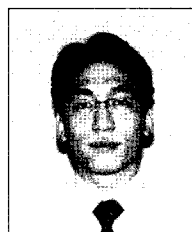
정회원



2004년 2월 인하대학교 전자공학과 졸업
 2006년 2월 인하대학교 전자공학과 석사
 2006년 3월~현재 인하대학교 전자공학과 박사과정
 <관심분야> 패턴인식, 영상처리

진 주 경 (Jukyung Jin)

정회원



2003년 2월 인하대학교 전자공학과 졸업
 2005년 2월 인하대학교 전자공학과 석사
 2005년 3월~현재 인하대학교 전자공학과 박사과정
 <관심분야> 패턴인식, 영상압축

정 등 석 (Dongseok Jeong)

중신회원



1977년 2월 서울대학교 전기공학과 졸업
 1985년 Virginia 주립 공과대학 전자공학과 공학석사
 1988년 Virginia 주립 공과대학 전자공학과 공학박사
 1988년 3월~현재 인하대학교 전자공학과 교수

<관심분야> 영상처리, 컴퓨터 비전, 멀티미디어 정보처리