

# 유비쿼터스 환경의 디스플레이를 위한 실시간 기하보정

경동욱<sup>0</sup> 임한규 정기철  
 송실대학교, 정보과학대학, 미디어학과, HCI Lab.  
 {kiki227<sup>0</sup>, hglim, kcjung}@ssu.ac.kr

## Real-time Geometric Calibration for Omni-Display in Ubiquitous Computing

Dongwuk Kyoung<sup>0</sup>, Hungyu Lim, Keechul Jung  
 HCI Lab., School of Media, College of Information Science, Soongsil University

### 요 약

최근 프로젝터를 기반으로 사용자가 원하는 위치에 영상을 제공하기 위한 많은 연구가 진행중에 있다. 프로젝트는 투사 방향이 정확하지 않으면 영상왜곡이 발생되며, 영상왜곡은 일반적으로 기하보정 변환행렬을 사용하여 보정한다. 기존의 연구는 선행작업으로 그물형태의 점을 투사하거나 또는 특정 마커를 설치해서 기하보정 변환행렬을 계산한다. 이 방법들은 투사방향이 변화될 때 마다 선행작업을 요구하므로 실시간 기하보정을 수행할 수 없다. 본 논문은 투사방향이 변화될 때에도 기하보정 변환행렬을 동시에 계산하여 보정된 영상을 제공하는 실시간 기하보정 시스템을 제안한다. 우리의 시스템은 웹카메라를 사용하여 실시간으로 기하보정 변환행렬을 계산하며, 다이렉트쇼(DirectShow) 기술을 사용하여 동영상 프레임의 추출하고, 추출된 프레임이미지를 기하보정 변환행렬을 사용하여 워핑(warping) 함으로써 수행한다.

### 1. 서 론

지금까지는 사용자가 정보를 획득하기 위하여 특정장치(예, 컴퓨터 모니터, TV, 프로젝션되는 스크린 등)가 있는 장소로 이동하였으나, 미래의 유비쿼터스(ubiquitous) 시대에서는 사용자가 언제 어디서든지 필요한 정보를 제공받을 수 있다.

현재 진행되고 있는 관련된 연구로써 EO-projector(Everywhere Display projector), 인터랙티브 프로젝션(Interactive Projection) 그리고 타일드 디스플레이(Tiled Display)가 있다. IBM에서 진행하고 있는 EO-projector는 사용자가 원하는 표면공간에 상,하,좌,우로 조절이 가능한 거울과 카메라를 이용하여 기하보정 변환행렬을 사전에 계산해야만 보정된 영상을 제공할 수 있다[1-2]. 그리고 인터랙티브 프로젝션은 웹카메라 일체형인 휴대용 프로젝터를 사용하여 다양한 물체에 필요한 정보를 제공한다. 이때 영상보정에 필요한 위치와 방향을 계산하기 위해서 'piecodes'라는 원모양의 컬러 마커를 사용한다. 하지만 보정된 영상을 제공하기 위해서 사전에 특정 마커를 설치해야만 한다[3]. 또한 타일드 디스플레이는 여러대의 프로젝터를 이용하여 고해상도의 영상을 제공하는 연구이다[4]. 이때 보정된 영상을 제공하기 위해서 그물형태의 점을 프로젝션하여 기하보정 변환행렬을 구한다. 이와 같은 관련 연구들은 투사방향이 변화되는 환경에서 쉽게 적용할 수 없다.

본 논문에서는 프로젝션 중에도 기하보정 변환행렬을 동시에 계산하여, 투사방향이 변화될 때에도 보정된 영상을 끊임없이 제공하는 실시간 기하보정 시스템을 제안한다. 제안된 시스템은 고정된 웹카메라를 사용하여 기하보정 변환행렬을 실시간으로 계산하며, 계산된 변환행렬을 다이렉트쇼의 기술을 사용하여 영상을 보정한다. 우리의

시스템은 미래의 가상전화기나 가상회의 그리고 영화 등과 같은 보정된 영상을 어디에서든지 사용자에게 제공하는 기술로써, 유비쿼터스 환경(ubiquitous computing)에 사용될 수 있다.

### 2. 프로젝터 기반의 실시간 기하보정 시스템

본 시스템은 기하보정 변환행렬을 계산하는 기하보정 단계와 기하보정 변환행렬을 이용하여 영상을 보정하는 영상보정 단계로 구분되며, 각 단계는 독립적인 처리과정을 수행한다(그림 1). 기하보정 단계에서는 캡처이미지를 이용하여 외곽점을 구하고, 변동이 있을 때에만 기하보정 변환행렬을 계산한다. 그때 변동을 검사하기 위해서 이전의 외곽점과 비교하여 임의의 웹카메라의 오차 X(외부환경의 영향으로 같은 공간이 다르게 캡처되어 발생하는 외곽점 좌표의 오차) 보다 클 때만 계산되어진다. 영상보정 단계에서는 새로운 기하보정 변환행렬이 전달되기 전까지는 이전 기하보정 변환행렬을 사용하여 동영상의 매 프레임의 워핑한다.

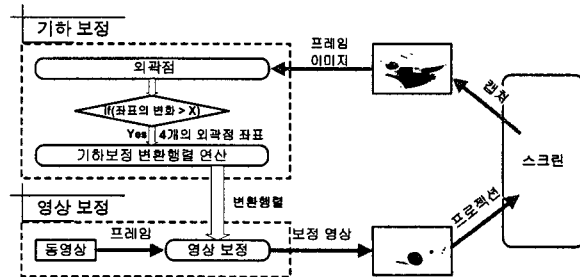


그림 1. 실시간 기하보정 시스템의 처리과정

2.1 외곽점 구하기

고정된 웹 카메라를 이용한 캡처 영상은 외곽점과 일반 영상이 함께 캡처된다(그림 2(b)). 이때 기하보정 변환행렬을 계산하기 위한 외곽점 선정방식이 중요하다. 외곽점을 구하기 위한 선행작업으로써 이진화영상과 모폴로지연산을 사용하여 각 컴포넌트를 구한다(그림 2(c)).

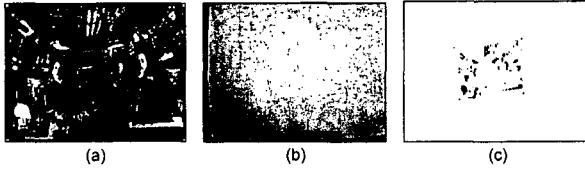


그림 2. 캡처된 영상처리 : (a) 초기 프로젝션 영상(모서리에 외곽점 표시), (b) 웹카메라를 이용한 캡처 영상, (c) 이진화 및 모폴로지연산 수행

모폴로지연산 이후 각 컴포넌트들의 중심값(컴포넌트를 구성하는 픽셀들 좌표의 평균값)을 편의상 컴포넌트를 대표하는 좌표로써 사용한다. 각 컴포넌트들의 좌표를 X값으로 정렬하여 최소값, 최대값을 구하고, Y값으로도 정렬하여 최소값과 최대값을 구했을 때 그림 4(a)처럼 각 컴포넌트들을 포함하는 외부사각형을 그릴 수 있다. 이때 외부사각형과 만나는 컴포넌트들을 외곽점이라 하며, 그림 3에서처럼 외곽점만을 연결한 내부사각형의 형태에 따라 2개에서 4개의 외곽점을 구할 수 있다. 단 내부사각형이 직사각형(정사각형 포함)의 형태를 가질 때 2개의 외곽점을 가지는 형태와 같은 결과를 가지므로 그림 3에서 제외하였다.

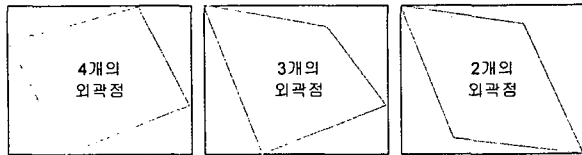


그림 3. 사각형의 형태에 따른 외곽점

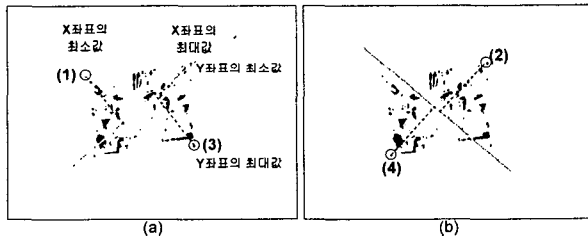


그림 4. 좌표 값 정렬을 이용한 사각형 및 외곽점 찾기

그림 4(a), 4(b)는 외부사각형에서 대각선을 이용하여 모든 외곽점과 위치를 결정하는 방법으로써, 각 대각선에서 모든 컴포넌트에 수직선을 그었을 때 가장 긴 컴포넌트를 가지는 4개의 컴포넌트를 구할 수 있다. 이때 기하보정 단계는 기하보정 변환행렬을 계산하기 위해서 각 외곽점의 위치 정보도 필요하기 때문에 그림 4에서처럼 각 위치 정보도 함께 결정할 수 있는 방법이다.

2.2 기하보정 변환행렬 구하기

외곽점은 기하보정 변환행렬을 구하기 위해서 사용되며, 이때 변환행렬을 계산하는 기술로써 키스톤 보정기술(keystone correction)이 사용된다[5]. 기하보정 변환행렬은 첫 번째로 프로젝터 좌표와 카메라 좌표와 관계( $T$ )를 정의하고, 두 번째는 스크린 좌표와 카메라 좌표의 관계( $C$ )를 정의한다. 첫 번째와 두 번째를 식 1로 나타내고 있다. 이 두 가지 처리로 세 번째인 스크린 좌표와 프로젝터 좌표의 관계( $P$ )를 식 3으로 정의할 수 있다.

$$T \times \begin{pmatrix} x_{proj} \\ y_{proj} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} x_{cam} \\ y_{cam} \\ 1 \end{pmatrix}, C \times \begin{pmatrix} x_{world} \\ y_{world} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} x_{cam} \\ y_{cam} \\ 1 \end{pmatrix} \quad (1)$$

$$C^{-1} \times T \times \begin{pmatrix} x_{proj} \\ y_{proj} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} x_{world} \\ y_{world} \\ 1 \end{pmatrix}, P = C^{-1} \times T \quad (2)$$

여기서 외곽점은  $T$ 를 구하기 위한 좌표로써  $(x_{cam}, y_{cam})$ 을 이용한다. 지금까지의 처리과정으로 스크린 좌표와 프로젝터 좌표 그리고 카메라 좌표의 관계를 알 수 있다.

$$P \times \begin{pmatrix} x_{proj} \\ y_{proj} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} x_{world} \\ y_{world} \\ 1 \end{pmatrix}, S \times \begin{pmatrix} x_{image} \\ y_{image} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} x_{world} \\ y_{world} \\ 1 \end{pmatrix} \quad (3)$$

$$S^{-1} \times P \times \begin{pmatrix} x_{proj} \\ y_{proj} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} x_{image} \\ y_{image} \\ 1 \end{pmatrix}, W = S^{-1} \times P \quad (4)$$

프로젝터 버퍼 프레임 영상을 보정하기 위한 변환행렬( $W$ )은  $S, P$  행렬을 이용해서 구한다. 변환행렬  $S, P$ 는 스크린상의 프로젝트 영역에서 가로, 세로의 비율이 같고, 최대 큰 사각형의 꼭 지점을 식 3, 4에 대입하여 구한다. 최종 기하보정 변환행렬( $W$ )은  $S^{-1} \times P$ 의 연산으로써 결정된다.

2.3 영상보정

본 시스템은 마이크로소프트에서 제공하는 다이렉트쇼(DirectShow)를 이용하여 동영상 또는 웹카메라 영상에서 매 프레임 추출하여, 추출된 프레임 이미지를 별도의 변환 필터를 생성 및 추가하여 동영상 프레임을 보정한다. 그림 5는 동영상에서 프레임을 추출하고 추출된 이미지를 보정하기 위한 다이렉트쇼의 필터 구조이며, 제안된 시스템에서는 각 프레임을 보정하기 위해서 별도의 영상보정 변환필터를 생성하여 이를 다이렉트쇼에 적용하였다. 적용된 변환필터는 프레임을 보정하기 위해서 기하보정 변환행렬이 새롭게 계산될 때 마다 업데이트되며 업데이트 되기 전에는 이전 기하보정 변환행렬을 적용하여 프레임을 보정한다. 이때 외곽점을 표현하기 위해서 각 모서리에 RGB(255, 255, 255)의 값의  $8 \times 8$  볼록을 생성한다.

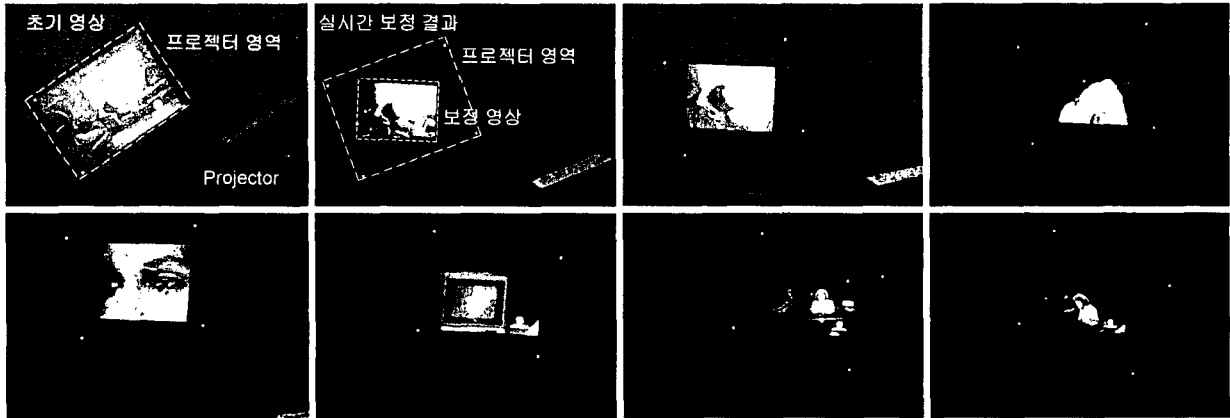


그림 7. 실시간 기하보정 시스템의 결과 영상

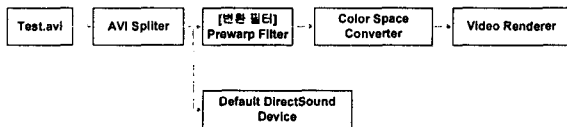


그림 5. 동영상 기하보정을 위한 DirectShow의 구조

그림 6은 변환 필터를 통해서 보정된 프레임버퍼 영상이다.



그림 6. 기하보정된 프레임 버퍼 영상

### 3 실험 및 결과

본 시스템을 위한 장치로써 웹카메라 1대와 프로젝터 1대 그리고 컴퓨터 1대가 사용되며, 프로그램 구현을 위해서 VC++ 6.0과 DirectShow가 사용되었다. 표 1은 제안된 시스템의 성능 분석 결과이다. 이때 기하보정 단계에서 기하보정 변환행렬이 계산되는 시간은 외관점 처리와 변환행렬 처리 시간의 통합으로써 0.3238(s)의 시간 간격으로 계산되며, 동영상의 프레임이 보정하여 출력될 때 약 1초에 10프레임 정도의 속도를 가진다.

표 1. 실시간 기하보정 시스템의 성능 분석

처리시간 (s)	기하보정 단계		영상보정 단계 (10.5 fps)
	외관점 처리	변환행렬 처리	
	0.1748	0.1490	
	기하보정 단계의 전체 수행시간 0.3238		

### 4 결론

본 시스템은 투사방향이 계속 변화하더라도 실시간으로 보정하는 시스템이다. 즉 프로젝터의 움직임을 고정된 웹카메라를 통해서 실시간으로 기하보정 변환행렬을 계산하고 동시에 영상을 워핑하는 시스템이다. 여기서 몇 가지 문제점으로, 우선 외관점을 구하기 위해서 프로젝터의 투사각도가 좌우 각각 45도 이상으로 기울어 질때 외관점을 구하지 못하는 문제점과 영상보정시 보정 변환행렬을 매 프레임에 적용함으로써 느린 수행을 보여주는 문제점을 가지고 있다. 앞으로는 위의 두 가지 문제를 해결하기 위해서 연구할 것이다.

### 참고문헌

- [1]Claudio Pinhancz, "The Everywhere Displays Projector: A Device to Create Ubiquitous Graphical Interfaces," LNCS, Vol. 2201, pp. 315-331, 2001.
- [2]CS Pinhancz, FC Kjeldsen et al., "Ubiquitous Interactive Graphics," IBM. Research Report, TJ Watson Research Center, New York, USA, 2002.
- [3]Beardsley, P., et al., "Interaction Using a Handheld Projector," IEEE Computer Graphics and Applications, Vol. 25, pp. 39-43, 2005.
- [4]R. Yang., et al., "PixelFlex: A Reconfigurable Multi-Projector Display System," In Proceedings of IEEE Visualization, 2001.
- [5]Sukthankar R., et al., "Smarter Presentations: Exploiting Homography in Camera-Projector Systems," In Proceedings of International Conference on Computer Vision, Vol. 1, pp. 247-253, 2001.