

정형체의 투사 선분의 오차 최소화에 의한 영상기반 모델링

박 종 승[†]

요 약

다중영상으로부터 투사 선분을 이용하여 3D 모델을 생성하고 각 면의 텍스처를 획득하는 구조 복원 기법을 제안한다. 사용자는 매우 단순한 절차를 통해 정확한 3D 모델 데이터를 획득할 수 있다. 모델 파라미터 추정을 위해 내재된 비선형 최적화 방법은 사용자 지정 영상 선분과 모델의 투사 선분의 거리를 최소화하는 방법에 기반하고 있다. 모델링 기법의 기능적 주요 목표는 형상이 포함된 다중 영상으로부터 그 형상의 3차원 구조를 복원하고 각 면의 텍스처를 생성하는 것이다. 본 연구에서는 3D 정형체를 사용하여 사용의 편리성을 증대시킬 수 있고 정형체의 파라미터의 오차를 최소화하여 복원된 구조의 정확성을 높이는 방법을 제시한다. 제안된 방법은 유한 선분에 기반한 오차 함수를 도입하여 무한 직선에 기반한 방법보다 정확한 모델링이 가능하다. 제안된 방법을 다양한 실제 영상에 적용한 실험 결과를 제시하고 다중 영상기반 모델링 도구의 개발 과정에서의 기술적인 문제점과 해결책을 기술한다.

키워드: 영상기반 모델링, 3차원 복원, 다중 영상 분석

Image-based Modeling by Minimizing Projection Error of Primitive Edges

Jong-Seung Park[†]

ABSTRACT

This paper proposes an image-based modeling method which recovers 3D models using projected line segments in multiple images. Using the method, a user obtains accurate 3D model data via several steps of simple manual works. The embedded nonlinear minimization technique in the model parameter estimation stage is based on the distances between the user provided image line segments and the projected line segments of primitives. We define an error using a finite line segment and thus increase accuracy in the model parameter estimation. The error is defined as the sum of differences between the observed image line segments provided by the user and the predicted image line segments which are computed using the current model parameters and camera parameters. The method is robust in a sense that it recovers 3D structures even from partially occluded objects and it does not be seriously affected by small measurement errors in the reconstruction process. This paper also describes experimental results from real images and difficulties and tricks that are found while implementing the image-based modeler.

Key Words : Image-based Modeling, 3D Reconstruction, Multi-view Analysis

1. 소 개

컴퓨터 그래픽스에서의 3D 모델링과는 달리 영상기반 모델링은 영상에 보여지는 물체의 모습에 기반하여 물체의 기하정보를 얻는 기법이다. 일반적으로 두 장 이상의 다중 영상이 기하정보 획득을 위해서 사용된다. 단일 영상만으로도 기하정보를 얻는 방법이 연구되고 있으나 평행육면체(parallelepiped)의 사용[1]이나 소실점(vanishing point)의 사용[2]과 같은 강력한 가정을 도입해야 한다. 다중 영상으로부터 3차원 구조 생성 방법에는 자동으로 특징(feature)을 검출하고 이들의 대응관계(correspondence relation)를 분석하는 방법이 있

다[3]. 특징으로서는 점(point)[4], 직선(straight line) [5, 6] [7, 8]이나 선분(line segment)[9, 10], 점과 직선[11] 등이 흔히 사용된다. 또한 거리 데이터를 직접 사용하여 형상을 복원할 수도 있으나[12] 특수한 장치를 사용해야 한다는 문제점이 있다. 장면의 물체, 배경, 카메라, 조명 등의 많은 가변적인 환경 요인을 고려하여 전체 과정을 완전히 자동화하는 것은 아직까지 성공적이지 못하다[13, 14]. 최근 적절한 사용자의 상호작용을 통해서 3차원 구조를 생성하는 방법들이 제시되고 있다[10, 11]. 이들은 다중 영상에 대해 사용자 입력을 통한 구조 복원 방식인 영상기반 모델링(image-based modeling) 기법들이다. 기존의 영상기반 모델링 방법들은 사용의 편리성과 모델의 정확성의 trade-off에 따라서 특징이 조금씩 다르다. 사용의 편리성과 모델의 정확성의 두 가지 요소를 동시에 만족시키는 정도가 복원 방법의 유용성을 결

※ 본 연구는 2004년도 인천대학교 교내연구비 지원에 의하여 수행되었음.

† 종신회원: 인천대학교 컴퓨터공학과 전임강사

논문접수: 2005년 1월 30일, 심사완료: 2005년 7월 18일

정한다.

모델링 도구는 3D 구조의 정확성과 사용자의 편리성 두 요소를 동시에 만족시키도록 제작하여야 한다. 편리성의 개선을 위한 방법으로서 3D 정형체(primitive)의 기하 구조를 사용하여 수작업 시간을 단축함으로써 사용의 편리성을 높일 수 있다. 대표적인 방법으로서 Debevec 등[5]은 기하구조를 블록 프리미티브들의 집합으로 정의하고 블록들의 기하 파라미터들을 추정하는 방법을 사용하였다. 이러한 프리미티브를 사용한 사용자 대화 방법은 기존의 작업을 매우 단순화시켜서 빠른 모델링이 가능하도록 하였고 영상기반 모델링 기법이 상용제품들[15, 16]로 발전하는 계기가 되었다. 이러한 프리미티브에 기반한 영상기반 모델링 방법은 기하구조에 기반한 모델링 방법에 비해서 큰 개선효과를 얻을 수 있었지만 모든 장면을 표현하기 위해서 장면을 이루는 모든 블록들의 관계를 기술해야 하는 단점이 있다.

모델링의 전체 작업 흐름은 여러 장의 영상들에 대한 모서리점(corner point) 입력, 3D 구조 복원, 텍스처(texture) 추출의 3단계로 이루어진다. 먼저 촬영한 이미지들을 준비하고 모델러 사용자 그래픽 인터페이스(Modeler Graphic User Interface)로 이들 이미지들에 대해서 3D 정형체들을 배치하는 작업을 수행한다. 원하는 3D 정형체들을 선택하고 정형체의 선분에 해당하는 영상의 투사선분의 일치 관계를 입력한다. 입력 결과로 3D 정형체와 각 모서리점의 해당 2D 좌표점들을 얻는다. 모든 모서리점을 입력할 필요가 없이 보이는 면(face)에 대해서만 입력한다. 3D 모델 복원기는 얻어진 데이터를 가지고 3D 복원을 실시한다. 그 결과로 3차원 모델을 얻는다. 모델 면의 텍스처는 입력된 2D 좌표점들을 사용하여 추출된다. 복원된 3D 구조는 모델러 내부에서 관찰할 수 있다.

단순 정형체를 사용한 복원 방식에는 복원 대상이 단순 정형체가 아닌 경우에 심각한 오차를 유발할 수 있는 문제가 있다. 점이나 선분을 사용한 복원 방식은 일반적인 형태의 형상도 복원이 가능하나 모델링 작업의 편리성이 떨어지게 된다. 이러한 문제의 극복을 위해서 본 논문에서는 입력 시에만 정형체를 이용하여 빠른 입력이 가능하도록 하고 구조 복원 시에는 선분을 기반으로 한 복원을 통해서, 보다 일반적인 형상의 복원이 가능하도록 한다. 편리성을 위해서 기존의 방법들과 유사한 3D 정형체들을 이용하여 쉽고 빠르게 3D 모델을 구축하도록 한다. 사용자는 3D 정형체를 선택한 후, 주어진 다중 영상에서 해당하는 선분을 지정한다. 3D 모델 복원기에서는 주어진 영상 선분으로부터 실제의 3D 선분을 추정한다. 이 과정에서는 계산량이 매우 많은 수치적 연산이 요구된다. 복원 과정은 모델의 정확성에 중점을 둔다. 정확한 모델의 계산을 위해서 입력된 정형체의 선분들을 독립적인 선분들로 간주하여 파라미터를 계산한다. 이러한 조치는 입력 영상이 충분히 원근 효과(perspective effect)를 포함하지 않거나 렌즈 왜곡(lens distortion)이 심한 경우 등과 같이 열악한 환경에서도 만족스러운 구조를 얻기 위함이다.

본 논문의 구성은 다음과 같다. 2장에서는 사용자로부터 주어지는 영상과 영상 선분으로부터 3차원 모델을 생성하는 과정에 대하여 설명한다. 3장에서는 모델의 각 면에 대한 텍스처를 획득하는 방법을 설명한다. 4장에서 실험 결과를 제시하고 5장에서 결론을 맺는다.

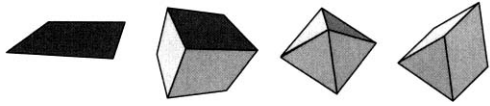
2. 선분에 기반한 3D 구조의 복원

다중 영상에서 3D 형상을 복원하는 방법으로는 점, 직선, 선분 등의 특징을 선정하고 카메라 기하에 대해 특징이 영상에 나타나는 기하 방정식의 해를 구하는 방법이 일반적이다. 2D 영상으로부터의 3D 물체 재구성에는 이제까지 상당한 연구 결과들이 제시되었다. 캘리브레이션된 두 대 이상의 카메라를 사용하는 스테레오 시스템이 대표적인 복원 방법이다. 스테레오 방법은 모든 영상 점의 조밀 형상을 제공하는 방법으로서 기하공간에서의 물체의 인식을 위해서는 특징(feature)에 기반한 접근이 필요하다. 물체의 주요 지점을 특징으로 지정하고 각 특징의 3차원 공간 구성을 통해서 물체의 기하적 재구성이 가능하다.

특징으로서 점, 직선, 선분 등이 사용될 수 있으나 직선이나 선분이 물체의 기하적 복원에 유리하다. 재구성 대상이 인공물, 건축물, 도시 환경 등인 경우에 영상 상에서 물체의 각 변(edge)에 대응되는 선분을 지정함으로써 3차원 물체에 대한 정보를 찾을 수 있다. 직선에 기반한 복원[5, 6, 7, 8]이나 선분에 기반한 복원[9, 10]에 대해서 이미 상당한 이론적 연구가 발표되었다. 본 논문에서는 정형체의 표현에 가장 적합한 선분에 기반한 복원 방법을 사용하였다. 선분에 기반한 복원 방법은 영상의 선분과 물체의 한 변의 투사 직선과의 거리를 계산한다. 본 논문에서의 방법은 기하 조건식의 유도나 미지의 파라미터 추정을 위한 최적화 방법은 기존의 방법[5, 6]과 유사하나 파라미터의 계산 시에 영상의 선분과 물체의 한 변의 투사 선분과의 거리를 계산하여 오차를 줄이도록 하였다. 정형체의 각 선분을 영상에 투사시켜서 투사선분을 구하고 투사선분과 영상의 입력 선분과의 거리로 두 선분간의 오차를 계산하였다. 투사선분을 두 끝점을 가지는 유한선분으로 정의하고 이에 기반하여 오차를 계산한다는 면에서 기존의 방법과 다르다.

2.1 편리한 선분 입력을 위한 정형체의 사용

먼저 기하 구조를 가정하는 3D 정형체들에 대해서 설명한다. 편리한 사용자 입력을 위해서 기하 구조가 미리 정의된 3D 정형체들을 사용한다. 정형체는 (그림 1)과 같은 단순한 모양의 *Plate*, *Cube*, *Pyramid*, *Wedge*로서 3D에서의 기본적인 공간 도형이다. 이들 도형의 각 좌표축으로의 이동 및 스케일링 연산과 도형간의 조합 연산을 통해서 복잡한 구조의 표현이 가능하다. 추가적인 정형체의 등록은 쉽게 이루어질 수 있다. 정형체의 사전 등록은 빠른 입력을 위한 수단일 뿐으로서 모델링의 기하구조를 한정시키지 않는다.



(그림 1) 모델링에서 사용되는 정형체들의 예

새로운 정형체가 시스템에 등록될 때마다 해당하는 정형체의 기하 표현 파라미터들을 설정하고 이로부터 정형체의 각 모서리점의 위치를 계산하는 함수를 등록한다. (그림 1)에서의 각 정형체에 대해서 다음과 같은 미지의 파라미터들로 그 정형체의 모든 모서리점들을 결정할 수 있다 :

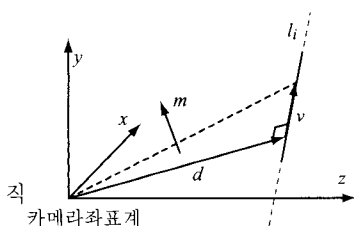
- **Plate** 정형체 (5개의 파라미터) : 각 정형체에 대해서 정형체의 기준점의 위치를 위한 3개의 파라미터와 가로 길이 및 세로 길이의 2개의 파라미터가 있다.
- **Cube** 정형체 (6개의 파라미터) : 각 정형체에 대해서 정형체의 기준점의 위치를 위한 3개의 파라미터와 가로 길이, 세로 길이, 높이의 3개의 파라미터가 있다.
- **Pyramid** 정형체 (6개의 파라미터) : 각 정형체에 대해서 정형체의 기준점의 위치를 위한 3개의 파라미터와 가로 길이, 세로 길이, 높이의 3개의 파라미터가 있다.
- **Wedge** 정형체 (6개의 파라미터) : 각 정형체에 대해서 정형체의 기준점의 위치를 위한 3개의 파라미터와 가로 길이, 세로 길이, 높이의 3개의 파라미터가 있다.

미리 정의된 정형체로부터 사용자가 선택한 정형체들의 선분에 해당하는 영상 선분들을 사용자의 입력으로 얻는다. 먼저 단순 정형체를 작업 판넬에서 선택하여 작업 창에 추가하고 이들의 모서리점을 영상에서 일치시킨다. 이러한 입력작업이 쉽고 정확히 이루어질 수 있도록 사용자가 간단한 마우스 드래깅 조작을 통해서 모든 입력을 제공할 수 있는 GUI 모듈을 구현하였다.

모델링 시스템은 입력 정보로부터 3D 물체를 재구성하게 된다. 3D 좌표계에서는 영상이 카메라로부터 초점거리만큼 떨어진 평면 상에 위치한다고 볼 수 있다. 시스템은 3D 물체의 위상 표현은 이미 알고 있으므로, 물체의 각 변들이 영상 상에 지정된 선분들에 그대로 투영될 수 있는, 최적의 물체 크기, 위치, 방향, 그리고 카메라의 위치와 방향을 계산하여 원래 물체를 공간 상에 재구성할 수 있다.

2.2 기하 구조의 복원

기하구조의 복원 알고리즘을 설명한다. M 개의 영상이 있고 이들이 각각 N 개의 직선을 포함하고 있다고 하자. 영상 또는 카메라를 위한 인덱스로 j ($1 \leq j \leq M$)를 사용하고, 선을



(그림 2) 카메라 좌표계에서의 직선의 표현

위한 인덱스로 i ($1 \leq i \leq N$)를 사용하여 영상을 I_j 로 표시하고 직선을 l_i 로 표시한다.

수치 해석 방법으로 근사 해를 구하기 위해서는 우선 사용자가 영상 상에 지정한 하나의 선분에 대한 오차를 함수의 형태로 나타낼 필요가 있다. 사용자가 지정한 선분의 끝점들을 각각 (x_1, y_1) , (x_2, y_2) 라 하고, 3D 물체에서 이 선분에 대응되는 변을 지나는 직선을 l_i 라 하자. 세계 좌표계(world coordinate system)에서 카메라 좌표계(camera coordinate system)로의 변환 과정은 회전 행렬 R_i 와 평행 이동 벡터 t_j 로 각각 표현할 수 있다. 즉, 세계 좌표계에서의 점 p^W 는 카메라 좌표계에서 $p^C = R_i p^W + t_j$ 의 좌표를 가진다. (그림 2)에서와 같이, 카메라 설정 I_j 에 대응되는 카메라 좌표계에서 직선 l_i 의 방향 벡터를 v 라고 하고 원점에서 l_i 상의 최단 거리의 점을 가리키는 벡터를 d 라 하면, 카메라 좌표계의 원점과 직선 l_i 를 동시에 포함하는 평면 P 의 법선 벡터는 $m = v \times d$ 로 계산된다.

카메라의 초점거리를 f 라 하면, 사용자가 사용한 영상은 카메라 좌표계에서 $z = f$ 평면 상에 위치한다. 따라서, 평면 P 와 평면 $z = f$ 이 교차하여 생기는 직선은 물체의 한 변을 지나는 직선 l_i 가 투사되어 영상 상에서의 궤적을 나타내고, 평면 P 의 법선 벡터를 $m = [m_x, m_y, m_z]^T$ 라 하면, 이 직선은 $m_x x + m_y y + m_z f = 0$ 으로 계산된다. 카메라 설정과 물체의 크기, 위치, 방향이 모두 해당 영상을 촬영할 때의 상황과 동일하다면, 사용자가 지정한 선분의 끝점들 (x_1, y_1) , (x_2, y_2) 는 이 직선 상에 위치하여야 한다.

영상에서의 끝점 (x_1, y_1) , (x_2, y_2) 로 지정된 선분과, 직선 l_i 로부터 계산된 직선의 방정식 $m_x x + m_y y + m_z f = 0$ 간의 오차는 선분 상의 모든 점에서 직선 $m_x x + m_y y + m_z f = 0$ 까지의 거리를 모두 적분함으로써 정의할 수 있다. 즉, 카메라 설정 I_j 에 대한 직선 l_i 의 오차는 (x_1, y_1) 로부터의 거리를 매개변수 s 로 표현하면, 다음과 같이 계산된다 :

$$E_i^j = \int_{s=0}^S h^2(s) ds \tag{1}$$

여기서, 상수 S 는 (x_1, y_1) 에서 (x_2, y_2) 까지의 거리를 나타내고, $h(s)$ 는 매개변수 값 s 에 대응되는 선분 상의 점에서 직선 $m_x^i x + m_y^i y + m_z^i f = 0$ 까지의 최단 거리를 나타내는 함수이다. 식 (1)은 $s=0$ 일 때의 선분 상의 점, $s=S$ 일 때의 선분 상의 점, $s=0$ 일 때의 선분 상의 점에서 최단거리에 있는 직선 상의 점, $s=S$ 일 때의 선분 상의 점에서 최단거리에 있는 직선 상의 점의 네 점이 이루는 면적과 같고 이는 다음과 같이 계산된다 :

$$E_i^j = \frac{S}{3} (h_1^2 + h_1 h_2 + h_2^2)$$

여기서 h_1 과 h_2 는 각각 (x_1, y_1) 과 (x_2, y_2) 에서의 $h(s)$ 값이다. 위에서 계산한 E_i^j 는 해당 물체의 변들 중에서 오직 하

나를 지나는 직선 l_i 에 대한 오차이므로, 물체 전체로 보아서는 영상 상에서 지정된 모든 변들의 오차를 합하여 물체 전체에 대한 오차를 구할 수 있다. 또, 물체 하나에 대해서 여러 개의 영상을 동시에 적용할 수도 있으므로, 카메라 I_j 역시 하나가 아니라 여러 개일 수 있다. 따라서, 최종적인 전체 오차는 다음과 같이 계산된다 :

$$err = \sum_j \sum_i E_i^j$$

올바른 초기 해를 구하기 위한 방법으로 Taylor와 Kriegman[9]는 다양한 회전행렬을 임의로 생성하고 이들을 시도해보는 방법을 사용하였다. Debevec 등[5]은 건물 등의 구조물의 알려진 수직 및 수평 방향 직선을 이용하여 직선의 방향을 고정시킨 뒤에 회전행렬만을 먼저 고려하였다. Taylor와 Kriegman[9]의 방법에서는 수많은 임의의 회전행렬을 초기값으로 생성함에도 불구하고 올바른 해를 찾지 못하는 경우가 발생할 수 있다. Debevec 등[5]의 경우에는 알려진 방향의 직선을 가정하는 것은 일반적인 영상에 대해서 적용될 수 없다.

2.3 선분을 이용한 복원

다중 영상에서 선분을 이용하여 구조를 얻는 방법으로서 투사선분과 입력선분의 차이를 최소화하는 투사선분의 파라미터를 구한다. 영상 또는 카메라를 위한 인덱스가 j ($1 \leq j \leq M$) 이고 직선의 인덱스가 i ($1 \leq i \leq N$) 일 때 투사선분은 직선 l_i 의 위치와 영상 I_j 의 카메라 위치 및 방향으로 결정된다. 3D 직선 l_i 의 위치를 p_i 라고 하고 영상 I_j 의 카메라 위치와 방향을 q_j 라고 하면 직선의 투사식은 $F(p_i, q_j)$ 로 표현할 수 있다. u_{ij} 를 직선 l_i 를 영상 I_j 로 투사한 것의 측정치라고 하자. 목적함수는 $F(p_i, q_j)$ 와 u_{ij} 의 시각차(disparity)의 합으로 둘 수 있다. 3차원 복원의 목표는 주어진 영상 측정치로부터 목적함수를 최소화하는 p_i 와 q_j 를 찾는 것이다. 직선 l_i 의 측정된 선분 $\langle(x_1, y_1), (x_2, y_2)\rangle$ 과 예측된 선분 m 의 여러 E_i 는 두 선분의 시각차로 정의된다. 여러 E_i 는 다음과 같은 계산식으로 구할 수 있다 :

$$E_i = [m_x^2(x_1^2 + x_1x_2 + x_2^2) + 3m_y^2(y_1^2 + y_1y_2 + y_2^2)] / 3 + m_x^2 + m_y^2 + m_x m_y (2x_1y_1 + 2x_2y_2 + x_1y_2 + x_2y_1) / 3 + m_x m_z (x_1 + x_2) + m_y m_z (y_1 + y_2) / (m_x^2 + m_y^2)$$

즉 목적함수는

$$f_{obj} = \sum_{i=1}^N E_i \tag{2}$$

이다. 위의 목적함수 (2)는 미지수에 대한 비선형으로서, 초기값에 의존되는 국부최소치가 해가 될 수 있다. 올바른 해를 얻기 위해서는 적당한 초기치를 구해야 한다.

카메라 위치 q_i 는 이동 벡터 t_i 와 회전 행렬 R_i 로 표현된

다. 문제의 간결성을 위해서 기준좌표를 첫번째 카메라 위치로 또한 모든 카메라의 초점거리를 1로 둔다. 직선은 직선의 방향을 나타내는 단위 벡터 v 와 원점과 가장 가까이 위치한 직선상의 점에 대한 위치 벡터 d 의 쌍 $\langle v, d \rangle$ 으로 표현한다. 3D 직선 l_i 와 영상 I_j 의 카메라 중심이 이루는 평면의 노말을 m_{ij} 으로 하자. 실제로 세계 좌표를 영상 I_j 의 카메라 좌표로 변환은 R_j 과 t_j 로 이루어 진다. 직선 l_i 는 카메라 좌표로 $\langle v_{ij}^c, d_{ij}^c \rangle$ 로 표현되고 세계 좌표로는 $\langle v_i, d_i \rangle$ 로 표현된다고 하면 카메라 좌표계로 표현되는 m_{ij} 은 $m_{ij} = v_{ij}^c \times d_{ij}^c$ 이다. 세계좌표로 표현되는 직선의 벡터를 이용하면 $m_{ij} = R_j (v_i \times (d_i - t_j))$ 이다. 따라서,

$$m_{ij}^T R_j v_i = 0 \tag{3}$$

$$m_{ij}^T R_j (d_i - t_j) = 0 \tag{4}$$

의 두 제약식을 얻을 수 있다.

영상의 측정치 u_{ij} 를 $\langle(x_1, y_1), (x_2, y_2)\rangle$ 라고 하자. 카메라 중심과 측정 에지를 지나는 평면의 노말은 $m_{ij}' = (x_1, y_1, 1)^T \times (x_2, y_2, 1)^T$ 이다. 이제 식 (3)으로부터 다음의 목적함수를 고안한다 :

$$f_{obj1} = \sum_{j=1}^m \sum_{i=1}^n (m_{ij}^T R_j v_i)^2 \tag{5}$$

즉, 카메라 위치와 직선의 위치에 무관하게 카메라 방향과 직선의 방향을 추정할 수 있다. 일단 R_j, v_i 가 얻어지면, 식 (4)로부터 다음의 목적함수를 얻는다 :

$$f_{obj2} = \sum_{j=1}^m \sum_{i=1}^n (m_{ij}^T R_j (d_i - t_j))^2 \tag{6}$$

위의 목적함수를 사용하여 d_i 와 t_j 를 추정한다.

f_{obj1} 와 f_{obj2} 의 최소화는 일반적인 비선형 최적화 방법을 사용한다. 전체적인 파라미터 추정 절차는 다음과 같다 :

- 과정 1: 식 (5)의 목적함수 f_{obj1} 를 사용하여 카메라 방향과 직선의 방향을 추정한다.
- 과정 2: 식 (6)의 목적함수 f_{obj2} 를 사용하여 카메라 위치와 직선의 위치를 추정한다.
- 과정 3: 식 (2)의 목적함수 f_{obj} 를 사용하여 비선형 최적화한다.

위의 파라미터 추정 방법은 Taylor와 Kriegman[9]의 방법과 유사하다. 이 방법은 초기치가 해와 유사한 경우에 대해서는 적절히 수렴하지만 어떤 초기치의 경우에 전혀 수렴하지 않거나 해와 거리가 먼 수렴 결과를 보여주는 경우가 있다.

초기치에 민감한 문제의 주요한 한 원인은 목적식이 투사 직선의 방정식에 기반하는데 있다. 투사직선과 입력선분의

거리를 계산함에 따라서 애매성이 존재하고 초기치에 민감하게 된다. 본 논문에서의 방법은 이 문제의 해결을 위해서 두 끝점이 정의되는 투사선분과 입력선분의 오차를 정의하고 이에 기반하여 최소화를 진행하도록 한다. 현재의 파라미터 값으로부터 3D 선분의 좌표를 알 수 있고 이를 투사시켜 길이가 유한한 투사선분을 얻을 수 있다.

최종 오차 에러는 각 물체의 크기, 위치, 방향, 그리고 각 카메라의 위치, 방향을 입력으로 하는 함수이다. 입력으로 사용되는 모든 매개변수의 개수를 k 이라 하면, 사용자가 지정한 영상 좌표 값들로부터 3차원 물체를 재구성하는 문제는 이제 k 차원 상에서 최종 오차를 의미하는 함수 에러를 최소화하는 최적화 문제가 된다. 일반적인 차원에서의 최적화 방법에는 여러가지가 있으나, 본 논문에서는 LM (Levenberg-Marquardt) 최적화 방법[17, 18]을 사용하였다. 비선형 LM 최적화 방법은 일반적으로 편미분 함수를 필요로 하나 편미분 함수의 계산이 매우 복잡하거나 불가능한 경우에는 각 파라미터의 $\pm \epsilon$ 만큼의 전후에서의 오차 함수값을 계산하는 방식으로 편미분 함수를 대신할 수 있다.

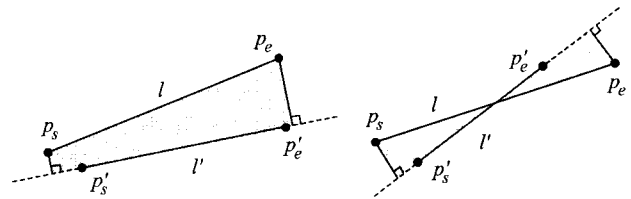
구하고자 하는 미지의 파라미터의 수는 추가된 정형체의 수와 입력 영상의 수에 의해서 정해진다. 각 입력 영상과 정형체에 대해서 다음과 같은 미지의 파라미터들이 정해진다 :

- 정형체의 기하 파라미터 : 사용된 정형체의 해당하는 기하 파라미터의 수만큼 추가된다. *Plate*는 5개, *Cube*, *Pyramid*, *Wedge*는 각각 6개의 파라미터가 사용된다.
- 카메라 파라미터 : 한 입력 영상에 대해 6개의 카메라 파라미터가 추가된다. 각 카메라 위치에 대해서 3개의 회전 파라미터들과 3개의 이동 파라미터들인 총 6개의 카메라 파라미터가 있다.

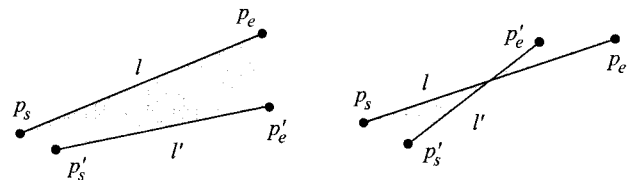
추정해야 할 파라미터들의 수를 결정하고 이들에 대한 초기치를 설정한다. 각각의 영상에 대한 카메라 설정 값들, 각각의 3D 정형체에 대한 크기, 위치, 방향을 모두 찾아야 되므로, 이들 전체가 k 차원 공간을 형성하고, 찾고자 하는 근사 해는 이 k 차원 공간에서 에러 함수의 값을 최소로 하는 점이 된다.

파라미터들은 정형체에 대해서 정의되어 있지만 오차 계산을 위한 목적함수는 정형체의 각 선분에 대해서 계산된다. 따라서 현재 상태의 파라미터 값들로부터 선분의 좌표를 얻고 각 선분에 대한 오차를 계산하도록 하여야 한다. 정형체의 파라미터 값들로부터 3D 선분을 계산하고 이 3D 선분을 카메라 파라미터를 사용하여 영상에 투사하고 투사된 2D 선분과 사용자가 입력한 입력 선분과의 오차를 계산한다. 총 오차는 모든 정형체들의 선분의 오차들을 합한 값이다. 만약 정형체의 일정 선분들이 어떤 영상에서 보이지 않는 경우나 사용자가 해당 영상 선분을 입력하지 않은 경우에는 그 선분을 오차 계산에서 제외시킨다. 계산된 총 오차는 카메라 파라미터들과 정형체의 파라미터들을 오차가 줄어드는 방향으로 수정하도록 하여 오차가 점차적으로 감소하게 된다.

식 (2)에 표시한 오차 함수는 빠른 계산이 가능하지만 초기치에 민감하다는 문제점을 가지고 있다. (그림 3)는 식 (2)



(그림 3) 무한 직선에 기반한 오차의 정의



(그림 4) 유한 선분에 기반한 오차의 정의

이 표현하는 오차를 나타낸다. 한 3D 정형체의 3D 에지 L 에 대해서 수동으로 입력된 영상에서의 선분을 l 이라고 하자. 선분 l 은 수동 입력한 모서리점에 해당하는 영상 좌표인 $\langle p_s, p_e \rangle$ 이다. 현재의 카메라 파라미터 값과 정형체의 좌표 값을 이용하여 3D 에지 L 를 영상에 투사하여 투사 선분 l' 을 얻는다. 식 (2)의 오차는 추정된 선분 l 이 투사 선분 l' 을 포함하는 직선에 대해서 정의되는 면적에 해당한다.

이러한 오차계산은 추정 선분의 끝점 좌표를 알 수 없는 경우에 유용하다. 그러나 이 오차 정의는 선분 l 과 선분 l' 사이의 오차가 아니라, 선분 l 과 선분 l' 이 놓여있는 직선에 투사된 선분과의 오차를 의미한다. 선분 l 의 두 끝점과 이들 점에서 선분 l' 이 놓여 있는 직선 상으로의 최단 거리 위치의 네 점들이 이루는 면적이 오차가 된다. 따라서 투사 선분의 방향과 선분 상의 한 점만 주어지면 오차를 구할 수 있다.

투사 선분 l' 의 두 끝점의 영상 좌표를 계산할 수 있다면 오차를 더 정확히 정의할 수 있다. 본 논문에서의 제안 방법은 투사 선분의 방향 성분과 더불어 끝점의 위치 정보도 사용하여 투사 선분을 유한 선분으로 구하고 이로부터 에러를 정의하였다. 투사 선분의 끝점 좌표는 모델의 3D 좌표가 결정된 경우에만 얻을 수 있다. 그러나 정형체의 기하 구조를 알 수 있으므로 정형체의 모서리 점을 영상에 투사하면 추정 선분의 끝점 좌표를 알 수 있다. 즉 현재의 모델 파라미터 값으로부터 정형체의 모서리 점의 3D 좌표를 계산하고 이들 점을 투사시켜서 영상 좌표를 얻는다.

추정 선분의 끝점 좌표를 알 수 있으면 (그림 4)와 같이 더 정확한 오차를 계산할 수 있다. (그림 4)에서 정의된 오차는 두 선분의 네 끝점들이 이루는 면적에 해당한다. 선분 l 의 끝점을 $\langle p_s, p_e \rangle$ 라고 하고 선분 l' 의 끝점을 $\langle p'_s, p'_e \rangle$ 라고 하자. 두 선분의 오차는 다음과 같이 정의한다 :

- 두 선분이 교차하지 않는 경우: 두 선분의 네 끝점이 이루는 사변형(quadrilateral)의 면적
- 두 선분이 교차하는 경우: 두 교차 선분이 이루는 두 삼각형의 면적의 합

먼저 두 선분이 교차하는지의 여부를 결정해야 한다. 두 선분이 교차하는 경우는 점 p_s 와 점 p_e 가 서로 선분 l' 의 한

대쪽에 위치하고 또한 점 p'_s 와 점 p'_e 가 서로 선분 l 의 반대쪽에 위치하는 경우이다. 점 p_s 와 점 p_e 가 서로 선분 l' 의 반대쪽에 위치하는지의 여부는 세 점 (p_s, p'_s, p'_e) 와 세 점 (p_e, p'_s, p'_e) 가 하나는 시계방향이고 다른 하나는 반시계 방향인지를 확인하면 된다. 따라서 두 선분의 교차 여부는 다음과 같이 표현된다 :

$$l \text{ and } l' \text{ intersect iff } (ccw(p_s, p'_s, p'_e) \neq ccw(p_e, p'_s, p'_e)) \\ \text{and } (ccw(p_s, p_e, p'_s) \neq ccw(p_s, p_e, p'_e))$$

여기서 ccw 는 세 점이 반시계 방향이면 참인 함수이다. 두 선분이 교차하지 않는 경우에서 두 선분이 이루는 면적은 다음과 같이 계산한다 :

$$e(l, l') = \begin{cases} \|(p'_s - p_s) \times (p'_e - p_e)\|/2 & \text{if } ccw(p_s, p'_s, p'_e) \neq ccw(p_s, p'_s, p_e) \\ \|(p'_e - p_s) \times (p'_s - p_e)\|/2 & \text{otherwise} \end{cases}$$

두 선분이 교차하는 경우에는 두 선분의 교점 p_m 을 계산하고 두 선분이 이루는 두 삼각형의 면적의 합을 다음과 같이 계산한다 :

$$e(l, l') = \|(p_s - p_m) \times (p'_s - p_m)\|/2 + \|(p_e - p_m) \times (p'_e - p_m)\|/2$$

전체적인 에러는 형상의 모든 선분들에 대해서 각 영상에 투사된 모든 투사선분들의 에러의 합으로 나타낸다. 3D 선분 L_i 의 영상 I_j 에서의 선분을 $l_{i,j}$ 라고 표시하면 전체 에러는 다음과 같다 :

$$f_{obj3} = \sum_i \sum_j e(l_{i,j}, l'_{i,j}) \tag{7}$$

이제 목적함수 f_{obj3} 를 사용하여 비선형 최적화를 한다. 여기서 만약 3D 선분 L_i 가 영상 I_j 에서 보이지 않는 경우 또는 입력 선분이 없는 경우의 (i,j) 에 해당하는 항은 0으로 둔다. 즉 가려져서 보이지 않는 선분들은 전체적인 오차에 영향을 미치지 않도록 하고 보이는 선분들을 사용하여 파라미터를 추정한다. 보이는 선분들이 복원된 후 정형체의 기하 정보를 사용하여 보이지 않는 선분의 좌표들을 계산할 수 있다. 정형체의 기하 정보를 사용함은 직각 성질 및 대칭 성질 등의 가정을 이용하는 것이므로 형상의 정확성 측면에서 바람직하지 않으나 해당 선분이 영상에 보여지지 않은 상황에서만 사용된다.

3. 텍스처의 획득

3차원 모델이 구해지면 각 면에 대한 고품질 텍스처를 추출한다. 각 면의 텍스처는 영상에서의 모서리 투사점 좌표로부터 바로 얻을 수 있다. 텍스처 추출 단계는 일반적인 영상 기반 모델링 시스템의 필수 부분이므로 반드시 구현되

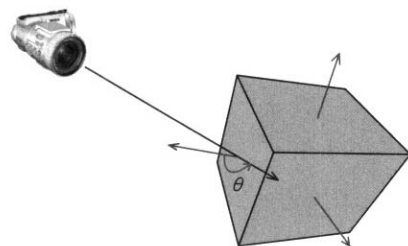
어야 한다. 본 장에서는 텍스처 추출을 위한 고려 사항들에 대해서 설명한다. 먼저 복원된 기하 데이터로부터 3차원 모델의 각 면에 해당하는 부분의 영상에서의 영역을 계산한다. 다중 영상을 이용하므로 영상의 해당 영역이 여러 영상에서 중복해서 존재할 수 있다. 따라서 다중 영상으로부터 중복되는 텍스처를 처리하기 위해서 다음과 같은 작업이 필요하다 :

- 텍스처의 품질을 결정하는 기준 : 주어진 텍스처에 대해서 얼마나 왜곡이 적고 넓게 모델의 면을 덮을 수 있는가의 기준으로서, 면의 2차원적 모양의 뒤틀림 정도에 반비례하고, 내부에 포함하는 화소수에 비례하는 척도이다.
- 칼라 정규화 기능을 포함한 텍스처 블렌딩 기능 : 각 텍스처의 칼라 히스토그램을 이용한다. 품질이 최상인 텍스처의 인접 영역의 히스토그램을 기준으로 변환 대상 텍스처의 히스토그램을 변환한다.
- 텍스처가 부재한 면에 대한 가상 텍스처 생성 기능 : 정형체의 대칭성을 이용하여 대칭면의 텍스처를 사용한다.

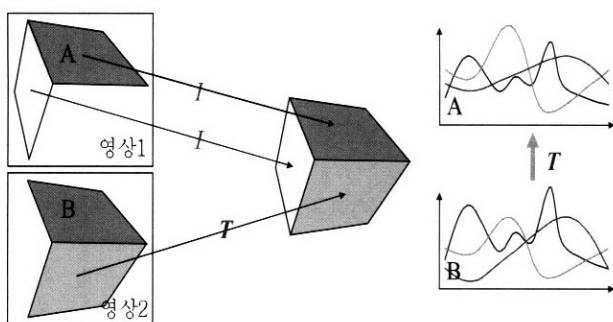
텍스처의 품질을 결정하여 가장 우수한 텍스처를 선정하는 방법을 고안하였다. 모델의 각 면에 대하여 면의 왜곡율(face distortion rate)를 정의한다. 왜곡율은 면의 2차원 모양에 대하여 원근 투사(perspective projection)로 인해 변형된 모양의 정도이다. 카메라 중심축과 면의 노말 벡터(normal vector)와의 각도 차이를 사용하여 면의 왜곡율을 정의할 수 있다. 각도 차이 θ 는 (그림 5)에 표시된 바와 같이 카메라 방향의 중심축(optical axis)과 표면 노말 방향의 각도 차이로 정의한다.

왜곡율 δ 은 $\delta = -\sin\theta$ 로 정의한다. δ 은 0에서 1의 값을 가지며 0은 왜곡이 없음을 의미하고 1은 왜곡이 매우 큼을 의미한다. θ 가 90° 이하이거나 270° 이상에서는 면이 보이지 않는 경우이므로 0으로 둔다. 또 하나의 텍스처 품질 결정 기준은 면의 투사 영역의 화소 단위의 면적으로서 r 로 표시한다. 많은 수의 화소를 포함하는 면의 투사 영역이 고품질이기 때문이다. 이 두 가지 선정기준을 이용하여 δ 가 작고 영상에서의 면적 r 가 큰 영역의 텍스처를 선정한다. 각 면에 대해서 왜곡율 δ 와 면적 r 을 구하고 r/δ 가 최소가 되는 면을 선정한다.

한 모델의 각 면에 대한 텍스처가 서로 다른 영상으로부터 추출될 경우에 대한 고려가 필요하다. 동일 물체에 대한 영상이라도 조명 등의 환경적 요인에 따라서 영상의 색 분포가 다르게 된다. 동일 물체의 각 면에 대한 텍스처를 추



(그림 5) 텍스처의 품질 결정을 위한 카메라 각도



(그림 6) 텍스처 칼라 정규화

출할 때에 이를 고려하여 색 분포를 정규화하여 주어야 한다. 정규화의 개념을 (그림 6)에서 보여준다. 공통으로 포함하는 면에 대한 칼라 히스토그램을 생성한다. 공통 면에 대한 텍스처의 영상 1에서의 품질이 영상 2에서의 품질보다 우수하다고 할 때, 공통면의 영상 2에서의 투사 영역 B에 대한 히스토그램을 영상 1에서의 투사 영역 A에 대한 히스토그램으로의 변환을 구한다. 이 히스토그램 변환을 사용하여 영상 2에만 존재하는 텍스처들의 변환 텍스처를 모델의 면에 대한 텍스처로 지정한다.

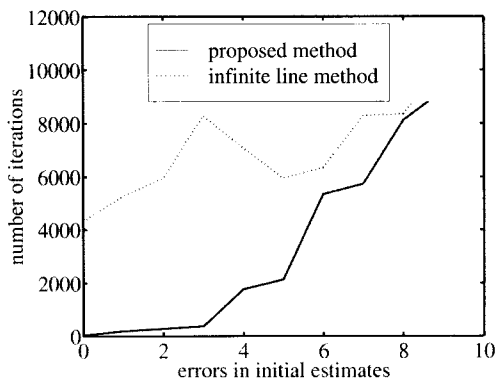
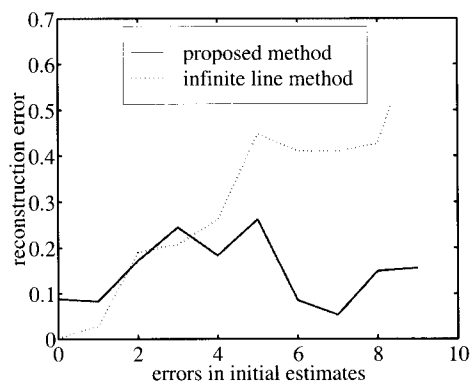
4. 실험 결과

형상 복원의 정확도를 측정하기 위해서 (그림 7)에서와 같은 정사각형 패턴의 영상 네 장으로부터 사각형의 복원을 실험하였다. 영상에서의 왜곡이나 입력선분의 부정확한 모서리점 위치로 인해서 복원된 좌표는 오차를 포함하게 된다. 오차의 정도를 측정하기 위해서 정사각형 패턴의 영상을 사용하였다. (그림 8)은 복원된 정사각형의 오차를 표시한다. 복원된 사각형 형상의 수평길이의 차이로써 오차의 정도를 표현하였다. 예러는 작을수록 복원이 정확함을 의미하며, 두 길이가 동일하면 예러가 0에 해당하고 큰 길이가 작은 길이의 두 배만큼 차이가 나면 예러가 0.5에 해당하도록 오차를 표현하였다. 오차의 수준의 증가할수록 제안된 방법이 기존의 무한 직선 방법에 비해서 더 정확도가 높은 결과를 제공함을 보여준다. 또한 동일 수렴 조건에서 파라미터 추정을 위한 반복 횟수가 감소되었다. 실제적인 계산시간은 어떤 최적화 방법을 사용하는지와 최적화에서의 수렴 기준 설정에 의존되지만 본 실험 결과는 정확도를 높이기 위해 최적화에서의 부가적인 반복 연산이 필요가 없음을 의미하므로 상대적인 속도 향상으로 볼 수 있다.

건축물과 같이 단순 정형체들로 구성된 구조에 대해서 구조 복원 실험을 수행하였다. 입력 영상이 두 장 이상이면 복원이 가능하다. 그러나 영상의 수가 작은 경우 보이지 않는 면이 많이 발생하게 되어서 복원에 어려움이 있을 수 있다. 텍스처 또한 획득하기 불가능하므로 대칭면에 대한 텍스처를 복사하여 사용하는 방법을 이용해야 한다. 일반적으로 세 장 이상의 영상이 적합하다.



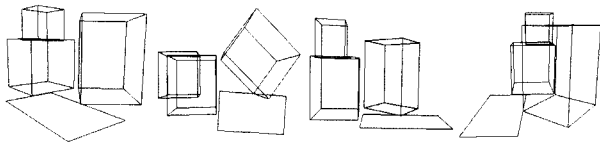
(그림 7) 네 장의 정사각형 패턴 영상



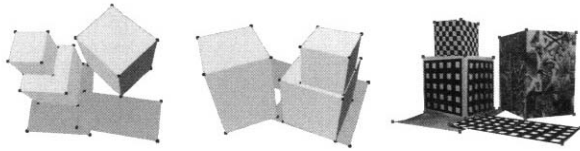
(그림 8) 정사각형 패턴의 복원 시의 정확도 및 수렴 속도 비교: 복원된 형상의 오차 (왼쪽), 최적화 수행에서의 반복 횟수 (오른쪽)



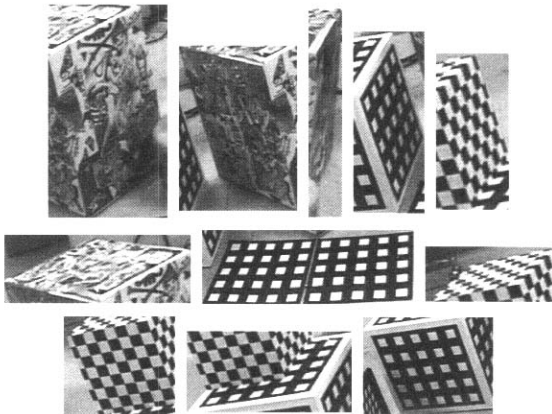
(그림 9) 모델러 GUI 화면 (위), 세 장의 영상에서 모서리 점을 일치시킨 모습 (아래)



(그림 10) 세 장의 영상으로부터 복원된 3D 구조의 여러 방향에서의 모습



(그림 11) 세 장의 영상으로부터 복원된 3D 구조의 solid view (왼쪽)와 texture-mapped view (오른쪽)



(그림 12) 텍스처 추출 과정에서 사용된 각 영상에서의 패치들

세 장의 입력 영상에 대하여 단순 정형체들을 추가하고 이들의 모서리점들을 입력시킨 결과가 (그림 9)에 있다. 정형체로서 세 개의 *Cube*와 두 개의 *Plate*가 사용되었다. 모서리 점들의 수동 입력이 완료되면 최적화 과정을 통해서 바로 3D 구조를 얻을 수 있다. 또한 복원된 구조를 내장된 3D 뷰어를 통해서 즉시 확인할 수 있다. (그림 9)에 보여진 세 장의 입력 영상에 대해서 구조를 복원한 모습이 (그림 10)에 있고, 복원된 3D 구조의 solid view와 texture-mapped view가 (그림 11)에 있다.

3D 구조가 복원되면 각 면의 텍스처들이 자동 추출된다. 각 면에 대한 텍스처들이 중복하여 보여지는 경우에는 텍스처 품질 결정 기준에 의하여 가장 우수한 텍스처가 선정된다. (그림 11)의 텍스처 매핑에 사용된 각 면의 텍스처들은 영상들로부터 자동 획득된 것이다. 가장 고품질로 결정된 영상의 패치들이 (그림 12)에 있다. 동일한 해상도일 때 영상에서 보이는 물체의 면이 영상이 맺히는 영상 평면과 평행이 경우에 텍스처 왜곡이 가장 적고 평행과 멀어질수록 텍스처 왜곡이 심해짐을 알 수 있다.

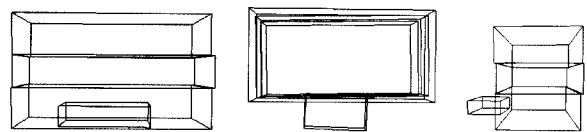
구조가 대형 건물이거나 촬영 시 접근이 어려운 경우에서와 같이 여러 측면에서의 영상 획득이 어려운 경우에서도 3D 복원이 가능하였다. 전면에서의 두 영상만 주어지는 경우에 대한 실험 영상이 (그림 13)에 있다. 단지 건물의 전면

과 좌측면만 보이고 나머지 네 개의 면은 영상에서 전혀 보이지 않는 경우이다. 이러한 경우에는 영상에서 보이는 면에 대해서만 모서리 점들을 지정한다. 일부의 모서리 점에 대해서 구조를 복원한 모습이 (그림 14)에 있다. 복원된 3D 구조의 솔리드 모습과 텍스처 매핑 모습이 (그림 15)에 있다. 이렇게 일부만 보이더라도 복원이 가능한 이유는 미리 정의된 단순 정형체의 기하 정보를 사용하여 보이지 않는 모서리 점의 위치를 추정할 수 있기 때문이다. 보이는 면에 대한 모서리 점이 적절한 위치로 움직이면서 동시에 보이지 않는 면에 대한 모서리 점도 단순 정형체의 기하 구조에 일관되도록 적절히 움직여 준다. 추출된 텍스처에 해당하는 영상 패치들이 (그림 16)에 있다. 한 방향으로 긴 면에 대한 텍스처의 경우 면의 방향이 정면에 가깝더라도 텍스처의 품질은 상당히 떨어진다. 약간의 오차로 인한 영상에서의 모서리 점의 불일치나 렌즈 왜곡으로 인한 영상의 일그러짐이 품질 저하의 원인이 된다.

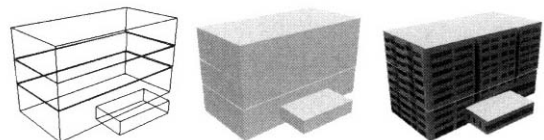
복원된 3D 구조에는 일정 수준의 오차가 포함되어 있다. 오차의 요인으로서 입력 영상들에 충분한 원근 효과를 가지고 있지 않는 경우, 영상 자체의 렌즈 왜곡, 사용자가 수동 입력한 모서리 점의 좌표가 정확하지 않은 경우 등이 있다. 원근 효과가 미비한 경우나 렌즈 왜곡으로부터 발생하는 오차는 3D 구조에 미치는 영향이 크지 않았다. 사용자의 수동 입력 좌표의 부정확성은 3D 구조의 복원에 매우 직접적인



(그림 13) 한 건물에 대한 다른 각도에서 촬영된 두 입력 영상



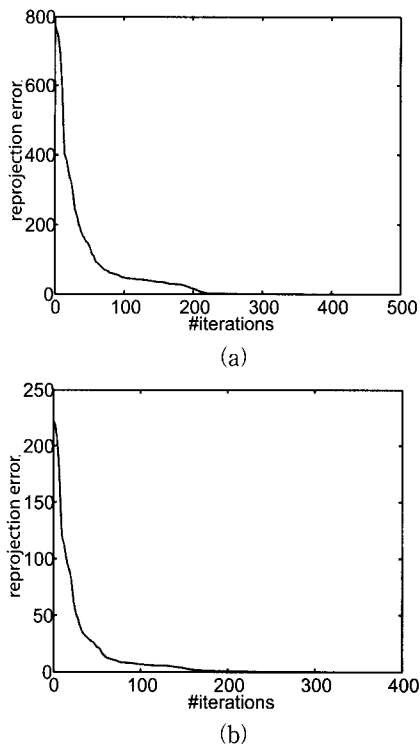
(그림 14) 복원된 3D 구조의 여러 방향에서의 모습



(그림 15) 복원된 3D 구조의 솔리드 모습과 텍스처 매핑 모습



(그림 16) 건물 영상의 텍스처 추출 과정에서 사용된 영상 패치들



(그림 17) 파라미터 추정 과정에서 반복 횟수에 따른 오차 감소:
(a) 박스 영상의 구조 복원시의 오차 감소, (b) 빌딩 영상의 구조 복원시의 오차 감소

영향을 주었다. 그러나 사용자가 대략 3 화소 이내의 오차로 입력한다는 조건에 대해서는 대부분 만족스러운 구조 복원이 가능하였다. 사용자의 손떨림이나 영상의 저해상도로 인한 화소 오차는 대략 3 화소 이내의 범위에 있었다.

파라미터 추정의 수렴 속도는 추정할 파라미터의 개수에 직접적으로 의존된다. (그림 9)의 경우는 입력 영상이 세 개, *Cube* 정형체가 세 개, *Plate* 정형체가 두 개이므로 총 46개의 파라미터를 추정해야 한다. 이때의 수렴 속도는 Pentium 4 2.6GHz 컴퓨터에서 약 2초가 걸렸다. 비선형 최적화 반복 추정의 오차 감소가 (그림 17)에 있다. (그림 17)(a)는 (그림 9)의 세 장의 상자 영상에 대한 수렴을 보여주고 (그림 17)(b)는 (그림 13)의 두 장의 건물 영상에 대한 수렴을 보여준다. 두 경우 모두 약 200번 반복 후에 추정치는 해와 유사한 범위로 수렴함을 알 수 있다. 수렴 속도는 초기치에 의존되는 경향이 있다. 초기치가 해 가까이 주어진 경우에 수렴 속도는 크게 빨라진다.

5. 결 론

본 논문에서는 다중영상으로부터 3D 모델을 생성하고 모델의 면에 대한 텍스처를 생성하는 영상기반 모델러에 대한 기술적인 측면을 다루었다. 개발된 모델러는 3D 정형체를 사용하여 수작업 시간을 단축함으로써 사용의 편리성을 높였다. 또한 3D 선분을 이용한 오차의 최소화 방법으로 모델의 정확도를 증가시켰다.

유한 선분에 기반한 오차 함수를 정의하고 이를 사용하여 3D 구조의 정확도를 높일 수 있었다. 사용자의 수동 입력 영상 좌표값에 특이한 오류가 없는 대부분의 경우에서 정확한 3D 구조를 빠르게 얻을 수 있었다. 정확한 기하 정보를 알 수 있는 실 영상을 사용한 실험 결과, 제안된 방법은 기존의 방법에 비해 초기치의 오차에 매우 강건한 특성을 보였다. 초기치로부터의 투사점 오차가 약 3 화소 미만으로 작은 경우에는 기존의 방법과 유사한 정확도의 모델을 생성하였고, 투사점 오차가 3 화소 이상으로 증가함에 따라서 기존의 방법에 비해 그 정확도가 매우 크게 개선되었다. 기존의 방법은 복원 에러가 투사점 오차에 비례하여 증가하였으나 제안된 방법은 투사점 오차가 약 10 화소 미만에서는 일정하게 유지되었다. 최적화 과정의 수행 속도는 반복 횟수에 비례한다. 반복 횟수는 대부분의 경우 약 50회까지는 매우 급격히 최적 해로 접근하고, 이후는 완만하게 접근하면서 약 200번의 반복 후에는 최적 해로 수렴하였다.

3D 모델 생성기의 최적 해를 찾는 과정은 여전히 개선의 여지가 있다. 해의 초기치가 최적값과 너무 먼 경우에 최적화 시간이 매우 오래 걸리거나 수렴하지 않는 경우가 발생한다. 이 문제의 해결책으로서 전체적인 카메라 기하정보를 분석하고 이로부터 적절한 가이드 특징점들을 구하고 최적값에 근접한 초기해를 알 수 있도록 구현하였다.

향후 연구 과제로서 단일 영상으로부터 모델의 생성을 위해 계층학적 접근의 모델링 알고리즘의 개발이 필요하다. 또한 텍스처 획득 과정에서 에지 매칭을 통한 여러 텍스처의 블렌딩 연구가 필요하다.

참 고 문 헌

- [1] M. Wilczkowiak, E. Boyer, P. Sturm, "Camera Calibration and 3D Reconstruction from Single Images Using Parallelepipeds," 8th International Conference on Computer Vision, pp.142-148, 2001.
- [2] A. Criminisi, I. Reid, A. Zisserman, "Single View Metrology," International Journal of Computer Vision, Vol.40, No.2, pp. 123-148, 2000.
- [3] A. Fusiello, "Uncalibrated Euclidean reconstruction: A review," Image and Vision Computing, Vol.18, No.67, pp. 555-563, 2000.
- [4] R. I. Hartley and A. Zisserman, 'Multiple View Geometry in Computer Vision', 2nd Ed., Cambridge University Press, 2004.
- [5] E. Debevec, Camillo J. Taylor, and Jitendra Malik, "Modeling and Rendering Architecture from Photographs: A Hybrid Geometry- and Image-Based Approach," Proceedings of SIGGRAPH 96, pp.11-20, 1996.
- [6] Adrien Bartoli and Peter Sturm, "Multiple-View Structure and Motion from Line Correspondences," Proceedings of the IEEE International Conference on Computer Vision, Nice,

France, IEEE CSP, pp.207-212, 2003.

[7] Adrien Bartoli, Richard Hartley, and Fredrik Kahl, "Motion from 3D Line Correspondences: Linear and Non-Linear Solutions," IEEE International Conference on Computer Vision and Pattern Recognition, IEEE CSP, pp.477-484, 2003.

[8] L. Quan, and T. Kanade, "Affine Structure from Line Correspondences With Uncalibrated Affine Cameras," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, pp.834-845, 1997.

[9] Camillo J. Taylor and David J. Kriegman, "Structure and Motion from Line Segments in Multiple Images," IEEE Trans. PAMI, Vol.17, No.11, pp.1021-1032, 1995.

[10] Zhengyou Zhang, "Estimating Motion and Structure from Correspondences of Line Segments between Two Perspective Images," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.17, No.12, pp.1129-1139, 1995.

[11] J. Oliensis and Mike Werman, "Structure from Motion using Points, Lines, and Intensities," IEEE International Conference on Computer Vision and Pattern Recognition, IEEE CSP, pp. 599-606, 2000.

[12] 조동욱, 김지영, 배영래, 고일석, "복잡한 물체의 기술을 위한 원뿔 표면의 분류 및 임계치 선정", 정보처리학회논문지 제 11-B권, 제3호, pp.297-302, 2004.

[13] John Oliensis, "A Critique of Structure-from-Motion Algorithms," Computer Vision and Image Understanding: CVIU, Vol.80, No.2, pp.172-214, 2000.

[14] S. El-Hakim, J.-A. Beraldin, and J.-F. Lapointe, "Towards

Automatic Modeling of Monuments and Towers," IEEE Proceedings of the International Symposium on 3D Data Processing Visualization and Transmission, pp.526-531, 2002.

[15] Canoma. <http://www.canoma.com/>.

[16] Realviz ImageModeler. <http://www.realviz.com/>.

[17] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, "Numerical Recipes in C", 2nd Ed, Cambridge University Press, 1992.

[18] S. Jacoby, J. Kowalik, and J. Pizzo, "Iterative Methods for Nonlinear Optimization Problems", Prentice-Hall, 1972.



박종승

e-mail : jong@incheon.ac.kr

1992년 경북대학교 전자계산학과(이학사)

1994년 포항공과대학교 컴퓨터공학과
(공학석사)

1999년 포항공과대학교 컴퓨터공학과
(공학박사)

1999년~2001년 한국전자통신연구원 컴퓨터소프트웨어연구소
선임연구원

2001년~2003년 디지털아리아 시각정보처리연구소 책임연구원

2004년~현재 인천대학교 컴퓨터공학과 전임강사

관심분야: 영상처리, 3D 비전, 증강현실, 게임공학