

# 순차 이미지를 이용한 3차원 물체의 사실적인 복원방법

김승권\*, 김강현\*\*, 전희성\*  
\*울산대학교 대학원 컴퓨터정보통신공학과  
\*\*부산 기능대학 정보통신시스템과

## Realistic 3D Reconstruction Method from an Image Sequence

Seung-kwon Kim\*, Kang-hyun Kim\*\*, Heesung Jun\*

Graduate School of Computer Engineering & Information Technology, University of Ulsan  
Dept. of Information & Communication System, Busan Polytechnic College

### 요 약

영상이나 이미지로부터 3차원 정보를 복원해내기 위한 기술은 컴퓨터 비전과 그래픽스 분야  
에서 많은 연구의 대상이 되어 왔다. 본 연구에서는 연속적으로 획득된 여러 장의 이미지로부터  
특징점을 추출한 후 그 점들로부터 프로젝티브 복원을 하고 카메라 파라미터를 계산하여 유클  
리디언 공간으로 변환시켜 3차원 데이터를 계산하는 방법을 구현하였다. 계산된 3차원 데이터와  
가장 적합한 폴리곤을 선택하고 텍스처 매핑을 하는 방법을 결합하여 사실적인 3차원 모델을  
생성할 수 있었다.

### 1. 서론

실 세계의 물체를 컴퓨터 상에 표현하기 위해서는  
먼저 모델링 과정을 통하여 물체의 3차원 정보를 저  
장하여야 한다. 기존의 모델링 방법인 Factorization방  
법은 camera calibration이 필요하다는 단점이 있어  
이러한 문제점들을 해결하기 위해 컴퓨터 그래픽스  
분야에서는 사전에 보정되지 아니한 카메라를 사용하  
는 기법을 사용한다[1,2]. 이 방법은 카메라의 특성을  
사전에 파악하지 않고도 calibration을 수행 할 수 있  
다. 여러대의 카메라로부터의 영상점들로부터 Eucli-  
dean복원과 Projective복원을 통해 3차원 특징점들을  
복 원하고 그 점들을 이용하여 최적의 polygon들을  
구하고 텍스처 매핑[5]과정을 거쳐 보다 사실적인 모  
델을 구현하였다.

본 논문의 구성은 다음과 같다. 2장에는 3차원 데이  
터의 복원 방법인 사영복원과 유클리디언 복원 방법  
에 대해 기술하고 3장에서는 본 시스템의 구현절차  
및 결과에 대해 서술하였으며 4장에 본 시스템의 장  
점과 향후 연구과제에 대해 언급하였다.

### 2. 3차원 데이터의 복원

#### 2.1 사영복원(Projective Reconstruction)

m개의 카메라(또는 관측점)에서 모두 관측 가능한  
n개의 3차원 점이 있을 때, 이러한 영상좌표의 측정  
값  $(u_{ij}, v_{ij})$ 만을 이용하여, 각 카메라의 원근투영행  
렬  $P_i$ 와 각 영상점의 3차원 위치  $X_j$ 를 복원하려는  
것이 사영복원 문제이다. 이상의 내용을 동차좌표를  
사용하여 식으로 표현하면 다음과 같다.

$$\lambda_{ij} \begin{bmatrix} u_{ij} \\ v_{ij} \\ 1 \end{bmatrix} = P_i X_j \quad (1)$$

$$W_s = \begin{bmatrix} \lambda_{11} \begin{bmatrix} u_{11} \\ v_{11} \\ 1 \end{bmatrix} & \cdots & \lambda_{1n} \begin{bmatrix} u_{1n} \\ v_{1n} \\ 1 \end{bmatrix} \\ \vdots & \ddots & \vdots \\ \lambda_{m1} \begin{bmatrix} u_{m1} \\ v_{m1} \\ 1 \end{bmatrix} & \cdots & \lambda_{mn} \begin{bmatrix} u_{mn} \\ v_{mn} \\ 1 \end{bmatrix} \end{bmatrix} \quad (2)$$

$$= \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_m \end{bmatrix} [X_1 \ X_2 \ \cdots \ X_n]$$

$\lambda_{ij}$  는 사영깊이(projective depth) 또는 배율(scale factor)이라 부른다. 식(1)을 모든 카메라와 모든 점에 대해 종합하면 식(2)와 같다.

### 2.2 반복 사영복원

본 연구에서는 Chen과 Medioni의 사영복원 알고리즘 [10,11]을 바탕으로 다음과 같이 일부 수정한 방법을 사용하였다.

1. 영상좌표를 정규화하여 행렬  $W$ 를 계산
2. 식 (2)을 이용하여 초기의 행렬  $W$ 에 Scale Factor  $\lambda_{ij}$ 를 곱하여 측정 행렬  $W_s$ 를 계산(초기값  $\lambda_{ij} = 1$ )
3. 특이값 분해(SVD - singular value decomposition) 방법을 이용하여  $W_s$ 의 Rank4 Factorization을 행함
4.  $W_s$ 를 특이값 분해했을 때 4번째 특이값과 5번째 특이값의 차이가 미리 정한 범위에 들어올 때까지 단계 2로 돌아가 계산을 반복함.
5.  $W_s$ 로부터 특이값 분해를 하여  $P$ 와  $X$ 를 계산

사영복원의 주목적은 식 (2)을 Rank4가 되게 만드는  $\lambda_{ij}$  를 구하는 것이라 할 수 있다.

$$W_s = UDV^T \quad (3)$$

위와 같이 분리한 행렬로부터 사영 운동행렬과 형상벡터를 구한다.

### 2.3 유클리디언 복원(Euclidean Reconstruction)

사영복원(Projective Reconstruction)을 통해 운동(motion)과 형상(shape)에 대한  $P$ 와  $X$ 를 구하였다. 여기서 우리가 원하는 실질적인 유클리디언 형태의  $P$ 와  $X$ 를 구하기 위해  $P = PH$ ,  $X = H^{-1}X$  형태로 식을 놓고 특정 제약조건에 의해  $H$ 를 구함으로써 유클리디언 복원을 수행한다[1].

$$W_s = PX = PHH^{-1}X = PX \quad (4)$$

사영 운동 행렬(Projective motion matrix)  $P$ 는

$$P = [M | T] \quad (5)$$

$$M = [m_{x1} \ m_{y1} \ m_{z1} \ \dots \ m_{xn} \ m_{yn} \ m_{zn}]^T$$

$$T = [T_{x1} \ T_{y1} \ T_{z1} \ \dots \ T_{xn} \ T_{yn} \ T_{zn}]^T$$

$$m_{xi} = \mu f_i \hat{i}_i + \mu_i \mu_{0i} \hat{k}_i$$

$$m_{yi} = \mu_i \alpha f_i \hat{j}_i + \mu_i \nu_{0i} \hat{k}_i$$

$$m_{zi} = \mu_i \hat{k}_i$$

$$T_{xi} = \mu f_i t_{xi} + \mu_i \mu_{0i} t_{zi}$$

$$T_{yi} = \mu_i \alpha f_i t_{yi} + \mu_i \nu_{0i} t_{zi}$$

$$T_{zi} = \mu_i t_{zi} \quad (6)$$

이다. 이를 다시 정리하면,

$$P = PH = K_i [R_i | t_i] = P[A|B] = [M|T]$$

$$PA = M, \quad PB = T \quad (7)$$

$$T_{xi} = \hat{p}_{xi} B \quad T_{yi} = \hat{p}_{yi} B \quad T_{zi} = \hat{p}_{zi} B \quad (8)$$

가 된다. 식 (7)에서  $B$ 를 구하기 위해서는 먼저 복원될 3차원 좌표점의 기준점을 정해야 한다.

복원될 형상행렬(Shape Matrix)은

$$X \sim \begin{bmatrix} S \\ 1 \end{bmatrix}$$

$$S = [s_1 \ s_2 \ \dots \ s_m], \quad s_j = [x_j \ y_j \ z_j]^T \quad (9)$$

$$x_j = [v_j \ s_j^T \ v_j]^T$$

이다. 여기서 월드좌표계(world coordinate system)의 원점이 특징점들의 무게중심으로 되도록 기준점을 정하면,

$$\sum_{j=1}^m v_j \ s_j = 0 \quad (10)$$

이 되고, 이것으로 다음과 같은 결과를 얻을 수 있다.

$$\begin{aligned} \sum_{j=1}^m \lambda_{ij} m_{ij} &= \sum_{j=1}^m (m_{xi} \cdot v_j \ s_j + v_j \ T_{xi}) \\ &= m_{xi} \cdot \sum_{j=1}^m v_j \ s_j + T_{xi} \sum_{j=1}^m v_j = T_{xi} \sum_{j=1}^m v_j \\ \sum_{j=1}^m \lambda_{ij} v_{ij} &= T_{yi} \sum_{j=1}^m v_j \quad \sum_{j=1}^m \lambda_{ij} = T_{zi} \sum_{j=1}^m v_j \end{aligned} \quad (11)$$

가 된다. 식(11)에서

$$\frac{T_{xi}}{T_{zi}} = \frac{\sum_{j=1}^m \lambda_{ij} m_{ij}}{\sum_{j=1}^m \lambda_{ij}}, \quad \frac{T_{yi}}{T_{zi}} = \frac{\sum_{j=1}^m \lambda_{ij} v_{ij}}{\sum_{j=1}^m \lambda_{ij}} \quad (12)$$

이고, 식 (8)과 (12)에서 미지수가 4개인  $2n$ 개의 선형 방정식을 만들 수 있으며, 선형 최소사승법[2]을 통해 미지수인  $B$ 값을 비교적 쉽게 찾을 수 있다.

이제 식 (7)의  $PA = M$ 에서  $A$ 를 구하기 위해 다음과 같이 놓는다.

$$MM^T = PAA^T P^T \quad MM^T = PQ P^T \quad (13)$$

$$\begin{bmatrix} m_{xi} \\ m_{yi} \\ m_{zi} \end{bmatrix} \begin{bmatrix} m_{xi} & m_{yi} & m_{zi} \end{bmatrix} = P \begin{bmatrix} a & b & c & d \\ b & e & f & g \\ c & f & h & i \\ d & g & i & j \end{bmatrix} P^T$$

$$= \begin{bmatrix} | & m_{xi}|^2 & m_{xi} \cdot m_{yi} & m_{xi} \cdot m_{zi} \\ m_{yi} \cdot m_{xi} & | & m_{yi}|^2 & m_{yi} \cdot m_{zi} \\ m_{zi} \cdot m_{xi} & m_{zi} \cdot m_{yi} & | & m_{zi}|^2 \end{bmatrix} \quad (14)$$

식 (14)에서 우리는 각 프레임당 10개의 선형방정식을 만들 수 있다. 여기서 이미지의 중점을 (0,0)으로, 스케일 벡터  $\mu_i$ 를 1이라 가정하면,

$$|m_{xi}|^2 = |m_{yi}|^2$$

$$m_{xi} \cdot m_{yi} = 0, m_{xi} \cdot m_{zi} = 0, m_{yi} \cdot m_{zi} = 0 \quad (15)$$

가 된다. 식 (14)과 식 (15)에서 각 프레임마다 4개의 방정식을 만들 수 있으며, 선형 최소자승법[2]을 적용하면 식(13)의 Q를 구할 수 있다. 그 후 Q를 랭크3 행렬분해(Rank3 Matrix decomposition)를 하여 A를 구할 수 있다. 단 A값이 존재하는 Q값을 구해야 하는데, 이것은 최소자승법으로 구해진 여러 해를 조합하여 조건에 해당되는 답을 선택하여야 한다.

### 3. 시스템 구현 및 실험

#### 3.1 시스템 구현

시스템 구현은 C++언어를 사용하였으며, 프로그램 인터페이스 부분(그림3)은 Qt 라이브러리[6]를 이용했으며, 관측 프로그램(그림4)은 MFC 와 OpenGL[7,8]을 이용하였다.

3차원 복원 절차는 그림1 에 표시한 것과 같으며 자세한 내용은 다음과 같다.

1. 데이터 준비단계 - 연속적인 이미지에 KLT 트래커[3,4]를 사용하여 특징점 데이터를 추출하고 조정하여 관측행렬 W를 생성한다.
2. 3차원 복원 - W행렬로부터 사영복원과 유클리디안 복원을 수행하여 3차원 좌표표를 얻는다.
3. 모델링 - 최초 입력된 이미지, 특징점 데이터, 폴리곤데이터, 그리고 복원된 3차원 좌표점을 사용하여 모델을 생성한다.

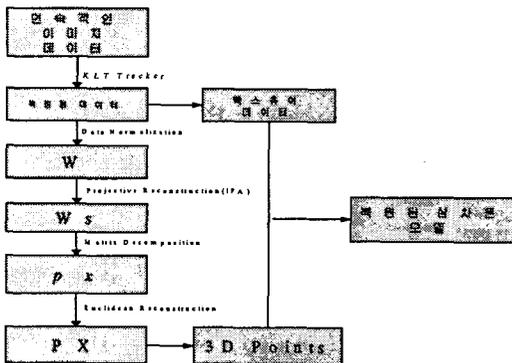


그림 1. 구현된 모델링 절차

#### 3.2 시스템 실험

시스템의 효용성을 검증하기 위해 여러 논문에서 사용되는 집모양의 데이터[9]를 사용하였다.

Movi House Data 중 본 실험에서 7개의 프레임만을 사용하였다. 최초 프레임에서 400개의 특징점을 검출하여, 7번

째 프레임까지 추적된 312개의 특징점을 데이터로 사용하였다.

그림 3에서 보는대로 시스템 실행시 KLT트래커[3,4]에 의해 특징점들이 추적된다. 표1은 추적된 특징점의 2차원 좌표점을 도시화 한 것이다.

frame feature	0	1	...	6
0	139.0, 407.0	143.3, 408.5	...	171.3, 416.1
1	178.0, 466.0	184.3, 466.9	...	221.9, 470.1
2	459.0, 285.0	456.7, 282.4	...	442.9, 268.1
...	...	...	...	...
311	210.0, 167.0	210.7, 167.9	...	217.3, 172.1

표 1. 각 프레임의 특징점들에 대응되는 2차원 좌표

산출된 각 프레임의 특징점 좌표들이 삼각형 폴리곤으로 구성되며 이렇게 구성된 폴리곤에 포함된 특징점들이 3차원 복원에 필요한 데이터로써 사용된다.

프레임당 가장 넓은 면적을 형성하는 좌표점들로서 이루어진 폴리곤을 본 시스템의 연산을 통해 검출하여 최종적으로 이를 선택한다. 모든 프레임에 대한 결과는 표3에 나타나 있다.

1st polygon	...	10th polygon
Area[1]: 4652.000000		Area[1]: 3431.000000
Area[2]: 4971.500000		Area[2]: 3636.000000
Area[3]: 5217.000000		Area[3]: 3966.000000
Area[4]: 5436.000000	...	Area[4]: 4395.500000
Area[5]: 5756.500000		Area[5]: 4740.000000
Area[6]: 5978.000000		Area[6]: 4952.000000
Area[7]: 6479.000000		Area[7]: 5479.500000
Best frame no: 7	...	Best frame no: 7

표 2. 각 프레임의 폴리곤 면적산출과 최적프레임 선택

선택된 데이터는 Movi 집 모양에서 총 10개의 특징점과 10개의 삼각형 폴리곤으로 구성된다.

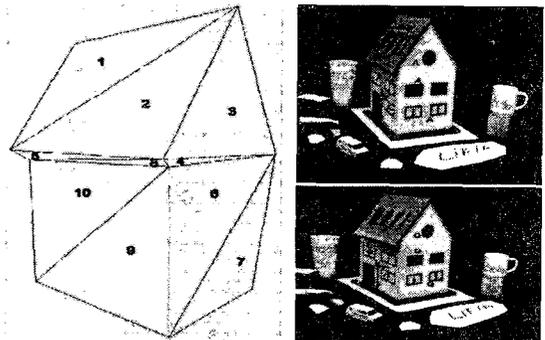


그림 2. 데이터의 폴리곤 번호와 영문표기

위의 그림에서 보는것처럼 각 프레임에서 가장 최적의 프레임이 선택되는것을 확인하기 위해 최초 프레

임은 영문 A로 설정하고 최종프레임 G까지 총 7개의 프레임을 표시하였다.

폴리곤No.	프레임No.	영문표기	폴리곤No.	폴리곤No.	영문표기
1	7	G	6	1	A
2	7	G	7	1	A
3	1	A	8	7	G
4	4	D	9	7	G
5	7	G	10	7	G

표 3. 폴리곤에 대응되는 최적 프레임과 영문표기

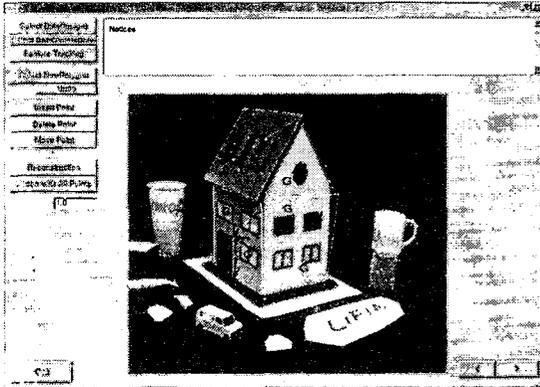


그림 3. 복원을 위한 특징점과 면 구조의 설정

시스템에서 최적의 프레임이 선택되고 그에 대응되는 폴리곤과 텍스처매핑[5]이 결합되어 아래 그림과 같은 실사에 가까운 결과를 얻을 수 있다.



그림 4. 관측프로그램을 통한 데이터 복원의 결과

그림에서 알 수 있듯이 지붕 및 같은 단면의 벽면과 다른 방향의 벽면에 표시된 영문으로 최적의 폴리곤을 성공적으로 선택했음을 확인할 수 있다.

#### 4. 결론

본 연구에서는 3차원 물체를 촬영한 연속적인 영상 데이터로부터 3차원 물체의 형상을 복원하는 유클리디언 복원방법을 구현하였으며, 텍스처 매핑을 통하여

사실적인 3차원 복원을 수행하였다. 구현된 시스템은 다음과 같은 장점을 가지고 있다.

1. 특징점들을 KLT를 통해 자동으로 추출하고, 사용자가 한 프레임에서 특징점을 선택하면 모든 프레임에서 최적의 폴리곤을 자동적으로 검출하여 가장나은 복원결과를 도출해 낼 수 있다.
2. 텍스처 매핑이 시스템 내부에서 자동으로 이루어짐으로서, 모델링시 수작업이 거의 필요가 없으며, 복원과정에서 저장된 모델에 관한 매핑소스와 각종 데이터를 사용하여 다른 시스템에서 복원된 모델을 쉽게 사용할 수 있다.

향후 연구 과제는 여러개의 복원된 결과를 종합하여 완전하고 사실감 있는 모델을 획득하는 것이다.

#### [참고문헌]

- [1] M. Han and T. Kanade, "Creating 3D Models with Uncalibrated Cameras", Proc. of IEEE Computer Society Workshop on the Application of Computer Vision, Dec. 2000
- [2] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2000
- [3] Bruce D. Lucas and Takeo Kanade. "An Iterative Image Registration Technique with an Application to Stereo Vision". International Joint Conference on Artificial Intelligence, pages 674-679, 1981.
- [4] Jianbo Shi and Carlo Tomasi. "Good Features to Track". IEEE Conference on Computer Vision and Pattern Recognition, pages 593-600, 1994.
- [5] Alan Watt and Fabio Policarpo, The Computer Image, Addison-Wesley, 1998.
- [6] Qt Library (<http://www.trolltech.com/>)
- [7] M. Woo, J. Neider, T. Davis, D. Shreiner, OpenGL Programming Guide, 3rd edition, Addison Wesley, 1999
- [8] Richard S. Wright and Jr. Michael Sweet, OpenGL SuperBible, WaiteGroup, 2001
- [9] MOVI house data ([http://www.inrialpes.fr/movi/pub/Images/sequences/g3\\_vp\\_ra\\_s1/index.html](http://www.inrialpes.fr/movi/pub/Images/sequences/g3_vp_ra_s1/index.html))
- [10] Q. Chen, "Multi-view Image-Based Rendering and Modeling", Ph. D. Dissertation, USC, 2000
- [11] Q. Chen and G. Medioni, "Efficient iterative solution to M-view projective reconstruction problem", Proc. IEEE CVPR 1999