

# 스케일된 직교분해 기법을 이용한 모양 및 카메라 움직임 추정

윤중현<sup>o</sup> 박종승 성미영 노성렬<sup>\*</sup>  
 인천대학교 컴퓨터공학과, 시리우스<sup>\*</sup>  
 {jhyoon<sup>o</sup>, park, mysung}@incheon.ac.kr, noh@sirius.co.kr

## Structure and Camera Motion Estimation using Scaled Orthographic Factorization

Jong-Hyun Yoon<sup>o</sup>, Jong-Seung Park, Mee Young Sung, Sung-Ryul Noh<sup>\*</sup>  
 Department of Computer Science and Engineering, University of Incheon, SIRIUS<sup>\*</sup>

### 요약

본 논문에서는 비디오 영상 및 이미지 시퀀스의 특징점 추적을 통해 얻은 2D 좌표를 분해기법을 사용하여 특징점에 대응되는 3D 좌표를 추정하는 방법을 제안한다. 3D 좌표를 복원함과 동시에 카메라의 위치와 방향을 계산하였다. 분해 기법에는 직교분해기법, 스케일된 직교분해기법, 근접 원근분해기법이 있다. 본 논문에서는 형상과 카메라 움직임을 계산하는 개선된 직교 분해 기법을 제안한다. 제안된 방법을 실 영상에 대해 실험한 결과 구조 및 카메라 파라미터 추정의 정확성이 개선되었다.

### 1. 서론

3차원 형상 추정 방법으로 분해 기법을 들 수 있다. 분해 기법은 직교분해기법 (orthographic factorization), 스케일된 직교분해기법 (scaled orthographic factorization), 근접 원근분해기법 (paraperspective factorization)이 있다[1][2]. 본 논문에서는 비디오 영상 및 이미지 시퀀스의 2차원 정보를 이용한 3차원 형상 추정 방법 중 스케일된 직교분해기법의 개선된 방법을 제시한다. 첫 번째 이미지로부터 특징점을 추출하고 이를 추적함으로써 첫 번째 이미지에 대응되는 특징점을 찾아낸다. 특징점 좌표 정보를 이용한 분해기법을 통해 특징점의 3차원 모델의 형상과 카메라의 이동 경로를 추정한다[3]. 그림 1을 통해 형상 복원의 절차를 볼 수 있다.

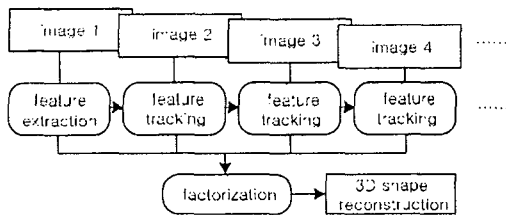


그림 1. 형상 복원을 위한 절차

### 2. 3D 구조 계산과 카메라 움직임 추정을 위한 분해기법

2D 이미지 시퀀스나 비디오 프레임으로부터 물체를 복원하는 방법으로 다양한 접근이 있다. 대표적인 방법으로 분해기법을 들 수 있다[4]. 그 종류로 직교분해기법, 스케일된 직교분해기법, 근접 원근분해기법이 있다. 본 논문에서는 직교분해기법과 스케일된 직교분해기법을 개선하여 빠르고 정확한 구조계산 및 카메라 움직임 추정이 가능한 방법을 연구하였다. 분해기법을 사용하기 전

에 복원하려는 물체에 대한 특징점 추적을 가정한다.

#### 2.1 분해 기법의 비교

본 논문에서 직교분해기법, 스케일된 직교분해기법의 두 가지 분해기법을 비교하였다. 보다 정확한 분해기법으로 근접 원근분해기법이 알려져 있다. 근접 원근분해기법의 경우 상당히 정확한 형상 추정이 가능하다. 그러나 노이즈에 민감하여 형상 추정이 불안정적이며, 형상 추정을 위한 계산이 어렵다는 단점이 있다. 이런 단점으로 인하여 근접원근분해기법의 적용이 실제적으로 곤란한 경우가 많다. 직교분해기법 중에서 근접원근분해기법과 유사한 결과를 낼 수 있는 것이 스케일된 직교분해기법이다. 스케일된 직교분해기법을 통해 근접한 형상을 추정할 수 있고, 카메라의 깊이값 계산에 의해 직교분해기법보다 정확한 카메라의 이동 경로 또한 추정이 가능하다[5][6].

#### 2.2 개선된 스케일된 직교분해기법을 통한 모양 복원

$F$  개의 이미지에 대하여  $N$  개의 특징점이 주어져 있다고 가정한다. 이때  $\{(x_{fp}, y_{fp}) | f=1, \dots, F, p=1, \dots, N\}$  는  $p$  번째 이미지에서  $f$  번째 특징점을 나타낸다. 특징점들로부터  $2F \times N$  인  $W$  행렬을 구성한다.  $W$  행렬은  $W = [X^T | Y^T]^T$  같은 형태로 이루어진다.  $X$  는 특징점들의  $x$  좌표로 수평 하게 구성된  $F \times N$  의 행렬 이고,  $Y$  는 특징점들의  $y$  좌표로 수평 하게 구성된  $F \times N$  의 행렬이다. 직교분해기법을 위한 과정으로  $W' = [X' | Y']$  의 형태로 변형한다.  $X'$  와  $Y'$  는  $X' = x'_{fp}$ ,  $Y' = y'_{fp}$ ,  $x'$  와  $y'$  는

$$x'_{fp} = x_{fp} - (1/N) \sum_{p=1}^N x_{fp}, y'_{fp} = y_{fp} - (1/N) \sum_{p=1}^N y_{fp}$$

로 계산된다.

Singular value decomposition(SVD)을 사용하여  $W' = U^* D^* V$ 와 같이 분해한다[7].  $U$ 의 첫 번째 세 개의 열로부터  $U'$ 를,  $D$ 의 첫 번째 세 개의 열 중 첫 번째 세 개의 행으로부터  $3 \times 3$ 인  $D'$ 를,  $V$ 의 첫 번째

<sup>\*</sup> 본 연구는 산업자원부 (MOCIE)의 지원으로 인천정보산업진흥원 (IIT)을 통하여 수행되었음.

제 세 개의 행으로  $V$ 를 만든다. 이렇게  $W=U^*D^*V'$ 로 변형된다. 여기서  $R$ 은  $R=U^*[D]^{1/2}$ 인  $2F \times 3$ 의 카메라 모션 행렬이고,  $S$ 는  $S=[S]^{1/2} \cdot V'$ 인  $3 \times N$ 의 형상 행렬이다[3].  $R \cdot S$ 을 구하는 방법으로 역행렬이 존재하는  $Q$ 를 사용한다.  $W'=R \cdot S=S^*Q^{-1} \cdot S'$ 를 통해  $R \cdot S$ 의  $R$ 과  $S$ 를 구한다 [1].

2.3 개선된 스케일된 직교분해기법에 의한 카메라 모션 정보

카메라의 이동 경로와 방향을 표시함으로써 모션정보를 나타낼 수 있다. 이동경로는 각 이미지 프레임의 특징점 으로부터 쉽게 얻어낼 수 있다. 각 이미지 프레임에서 특징점들의 평균값은

$$((I/N) \cdot \sum_{p=1}^N x_{fp}, (I/N) \cdot \sum_{p=1}^N y_{fp})$$

으로 나타낸다. 이들은 카메라 이동경로의  $x, y$  좌표를 표시한다. 직교분해기법의 경우 카메라의 깊이값을 계산할 수 없기에 깊이값을 상수값으로 고정한다.  $R$ 에서 카메라의 방향을 알 수 있다.  $R=[i_i \ i_n \ j_i \ j_n]^T$ 의 형태이다. 여기서  $i_i$ 과  $j_i, i_n$ 과  $j_n$ 과 같이 대응되는 요소들을 서로 외적 함으로써 각 프레임에서의 카메라 방향을 계산할 수 있다.

스케일된 직교분해기법에서는 직교분해기법과 달리 카메라의 깊이 값을 계산할 수 있다. 개선된 스케일된 직교분해기법에서는 스케일된 직교분해기법과 같은 방법으로 카메라 이동좌표와 방향을 계산한다. 하지만 스케일된 직교분해기법에서 계산된 카메라의 이동경로의  $z$  좌표값은 단위가 조절 (normalize)되어 있는 상태이므로 카메라 위치의  $x, y$  좌표도 단위를 조절해야 한다. 각 프레임에서 특징점의 평균으로 계산된 카메라 이동 경로는 서로 방위 (orientation)가 다르다. 방위를 맞춰야 올바른 카메라의 이동 경로는 계산할 수 있다. 프레임 별 이동 경로에 각각의 회전 행렬을 적용시킴으로써 방위 맞춘다. 단위를 조절한  $x, y$  좌표와 계산된  $z$  좌표를 함께 각 프레임 별 회전행렬에 적용시킴으로써 올바른 카메라의 모션 정보를 계산할 수 있다.

3. 실험결과

3.1 큐브 데이터를 사용한 실험

바둑판 패턴을 붙인 큐브의 다중 이미지를 사용하여 실험을 하였다. 큐브 데이터를 복원하는데 각 프레임 별로 58개의 특징점을 가지는 11개의 프레임을 사용하였다. 큐브의 표면에 붙인 패턴의 모서리 점들을 특징점으로 사용하였다. 그림 2는 큐브 데이터의 형상을 추정한 모습과 카메라의 이동 경로, 방향을 보여주고 있다. 그림 3과 그림 4를 통해 실 데이터와 형상이 추정된 데이터 사이의 오차를 보여주고 있다. 원본 이미지의 특징점과 투영된 좌표점의

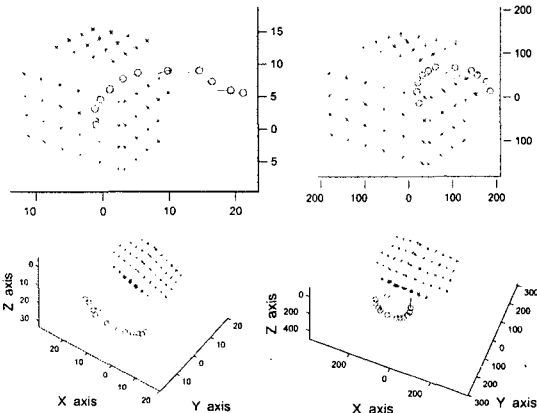


그림 2. 큐브 데이터에 대한 직교분해기법(왼쪽)과 스케일된 직교분해기법(오른쪽)을 사용해 형상을 추정한 모습과 카메라위치

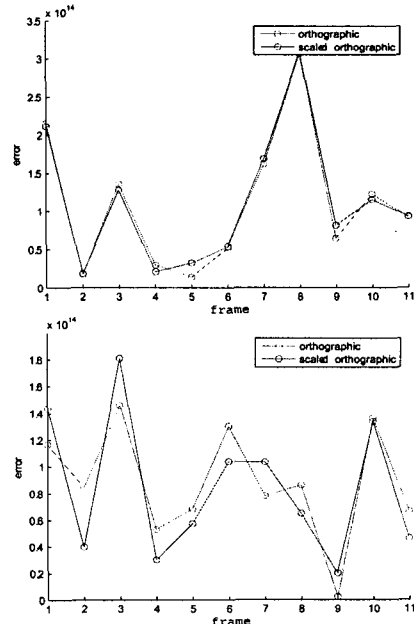


그림 3. 큐브 데이터에 대한 직교분해기법과 스케일된 직교분해기법으로 추정된 3D 형상의 x 좌표(위), y 좌표(아래)의 투영 오차

차이는 0에 가까운 것을 볼 수 있다. 큐브 표면에 패턴을 붙임으로써 성공적인 특징점 추적이 이루어졌기에 에러값이 적게 나왔음을 알 수 있다.

3.2 사람 얼굴 데이터를 사용한 실험

여러 방향에서 촬영한 사람의 얼굴의 이미지 프레임에 대해서 실험을 해보았다. 총 6개의 프레임에 대하여 각 프레임당 눈썹, 눈, 코, 입에 대해 91개의 추적된 특징점을 사용하였다. 그림 4는 사람의 얼굴형상을 추정한 모습이다. 그림 5를 통해 프레임 별 투영 오차의 평균을 특징점마다 보여주고 있다. 각 점에서  $x, y$ 좌표의 오차의 평균이 최대 0.15를 넘지 않는 작은 값이 나온 결과를 볼 수 있다. 특징점 추적의 정확도가 큐브 데이터보다 낮은 이유로 오차 값이 큐브 데이터보다 크게 나온 것을 볼 수 있다.

3.3 강아지 인형의 이미지를 사용한 실험

강아지 인형의 얼굴 형상을 추정하는데 6개의 프레임을 사용하였고, 각 프레임당 158개의 특징점을 사용하였다. 그림 6은 인형의 얼굴형상을 복원한 모습과 카메라의 이동 경로 모습이고, 그림 7을 통해 투영된 결과의 에러를 볼 수 있다. 직교분해기법과 스케일된 직교분해기법 모두 실 좌표와 추정된 형상이 투영된 좌표의 오차가  $x$  좌표는 0.25를 넘지 않고  $y$ 좌표는 0.03을 넘지 않는 결과를 볼 수 있다.

4. 결론

본 논문에서는 개선된 스케일된 직교분해기법을 통해 물체의 3D 형상을 추정하고, 카메라의 이동경로와 방향을 계산하는 방법을 제안하였다. 근접원근분해기법이 실제적인 적용이 어렵다는 단점을 보완하는 방법으로 직교분해기법과 스케일된 직교분해기법을 통한 실험을 해보았다. 직교분해기법의 경우 복원된 형상이  $x$ 축 방향으로 넓게 스케일링 (scaling)된 모습으로 형상 추정 결과를 얻을 수 있었다. 카메라의 이동경로의 경우 깊이값을 계산할 수

없기에 실제적인 카메라의 이동경로를 볼 수 없었다. 그러나, 개선된 스케일된 직교분해기법은 직교분해기법의 경우보다 더욱 실물과 비슷한 형상을 추정하는 결과를 얻을 수 있었다. 카메라의 모션 정보 또한 개선된 스케일된 직교분해기법에서의 깊이값 계산으로 직교분해기법보다 좀더 정확한 결과를 얻을 수 있었다. 하지만 스케일된 직교분해기법에서 카메라의 위치계산시  $x, y$ 좌표가 깊이값 계산에 의해 구해진  $z$  좌표와 단위가 맞지 않는 문제가 있었다. 이는 단위를 조절하는 개선된 방법을 통해서 그 문제를 해결하였다. 직교분해기법과 스케일된 직교분해기법 모두 공통적으로 카메라의 방향을 추정하는 문제점이 있었다. 이 문제는 각 카메라의 방향에 대응되는 프레임에서의 회전 행렬을 적용시킴으로써 방향을 맞춤으로써 해결할 수 있었다.

한가지 물체를 촬영한 카메라의 이동 외에 연속적으로 이동하는 카메라의 이동 경로를 계속적으로 보여주는 문제가 향후 연구 과제이다.

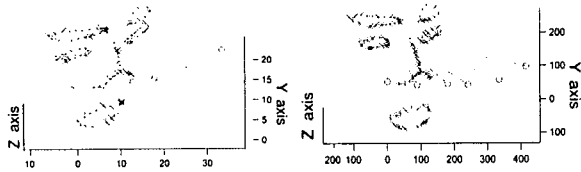


그림 4. 얼굴 형상에 대한 직교분해기법(왼쪽)과 스케일된 직교분해기법(오른쪽)을 사용해 형상을 추정한 모습과 카메라 위치

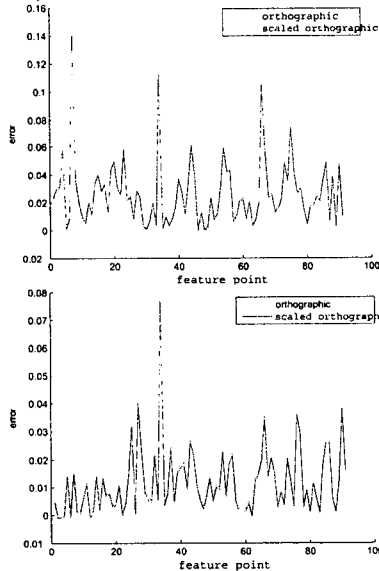


그림 5. 얼굴 형상에 대한 직교분해기법과 스케일된 직교분해기법을 사용해 추정된 3D 형상의 투영된  $x$ 좌표의 오차의 평균(위),  $y$ 좌표의 오차의 평균(아래)

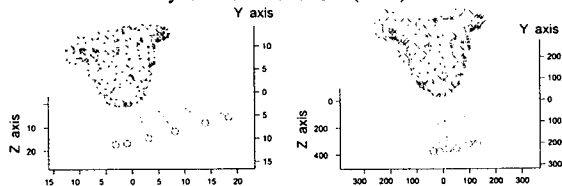


그림 6. 인형 형상에 대한 직교분해기법(왼쪽)과 스케일된 직교분해기법(오른쪽)을 사용해 형상을 추정한 모습과 카메라 위치

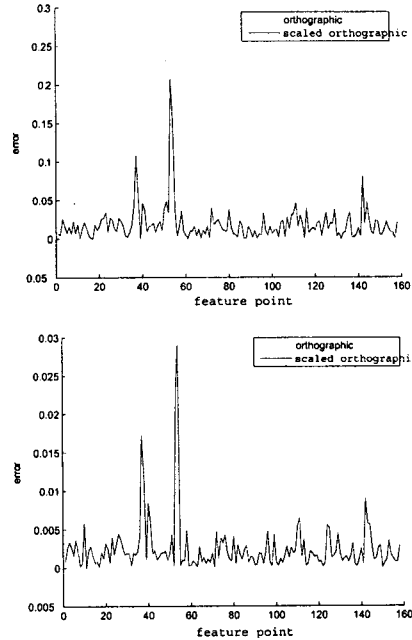


그림 7. 인형 형상에 대한 직교분해기법과 스케일된 직교분해기법을 사용해 추정된 3D 형상의 투영된  $x$ 좌표의 오차 평균(위),  $y$ 좌표의 오차 평균(아래)

Reference

- [1] Carlo Tomasi, Takeo Kanade, "Shape and Motion from Image Streams: a Factorization Method", Technical Report CMU-CS-91-172, Carnegie mellon university, Pittsburgh, PA, September 1991.
- [2] Conrad J, Poelman and Takeo Kanade, "A Paraperspective Factorization Method for Shape and Motion Recovery", Technical Report CMU-CS-93-219, Pittsburgh PA, Dec 1993.
- [3] Christoph Bregler, Aaron Hertzmann, Henning Biermann, "Recovering Non-Rigid 3D Shape from Image Streams", Proc. IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head, South Carolina, pp.190-696, June 2000.
- [4] Toshihiko Morita, Takeo Kanade, "A Sequential Factorization Method for Recovering Shape and Motion From Image Streams", IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 19, No. 8, August 1997.
- [5] L. Matthies, T. Kanade, R.Szeliski, "Kalman filter-based algorithms for estimating depth from image sequences", International Journal of computer vision, 3(3):209-236, September 1989.
- [6] C. Tomasi, T. Kanade, "Shape and Motion without depth", In Proceedings of the DARPA Image Understanding Workshop, pp.258-270, Pittsburgh, Pa, September 1990.
- [7] G. H. Golub, C. Reinisch, "Singular value decomposition and least squares solutions", In Handbook for Automatic Computation, volume 2, chapter I/10, pp.134-151. Springer Verlag, New York, NY, 1971.