

스테레오 영상과 준원근 카메라 모델을 이용한 객체의 3차원 형태 및 움직임 복원

김 상 훈[†]

요 약

본 논문은 영상내 객체정보의 정확한 복원을 위하여, 연속된 2차원 영상으로부터 특정 객체의 특징점을 추출하고, 특징점의 위치 데이터들로부터 원형의 3차원 모양 및 모션 정보를 복원하는 알고리즘과 결과를 제시하였다. 2차원 영상의 특징점 검출을 위해서는 물체와 배경이 명확히 구분되는 실험영상 환경에서 색상변환을 통한 자동 추출 방법을 사용하였다. 추출된 2차원 객체의 특징점들로부터 3차원 모양, 움직임 정보를 복원하기 위하여 스테레오 카메라와 준원근 카메라 모델을 적용하고 SVD(Singular Value Decomposition)에 의한 인수분해연산을 수행하였다. 준원근 카메라 모델의 근본적인 문제인 깊이정보의 복원 에러가, 스테레오 영상 분석에 의해 최소화 되었다. 본 논문에서 제시된 방법들의 성능을 객관적으로 평가하기 위하여 크기와 위치가 알려진 3차원 물체에 대해 실험을 행하였으며, 영상의 21개 특징점 위치와 공간상에서의 3개 방향으로의 움직임 각도를 연산에 의해 복원한 후 원형의 데이터와 비교하여 본 알고리즘의 정확성을 증명하였다.

3D Object's shape and motion recovery using stereo image and Paraperspective Camera Model

Sang-Hoon Kim[†]

ABSTRACT

Robust extraction of 3D object's features, shape and global motion information from 2D image sequence is described. The object's 21 feature points on the pyramid type synthetic object are extracted automatically using color transform technique. The extracted features are used to recover the 3D shape and global motion of the object using stereo paraperspective camera model and sequential SVD (Singular Value Decomposition) factorization method. An inherent error of depth recovery due to the paraperspective camera model was removed by using the stereo image analysis. A 3D synthetic object with 21 features reflecting various position was designed and tested to show the performance of proposed algorithm by comparing the recovered shape and motion data with the measured values.

키워드 : 객체 검출(Object Detection), 스테레오 영상(Stereo Image), 준원근 카메라 모델(Paraperspective Camera Model), SVD, 3차원 움직임 복원(3D Motion Recovery)

1. 서 론

컴퓨터가 연속된 영상으로부터 임의의 객체에 대한 3차원적인 모양과, 카메라를 기준으로 한 물체의 움직임 정보를 함께 인식하는 일은 사람-컴퓨터간의 상호작용 분야에서 매우 중요한 영역을 차지하고 있으며 멀티미디어환경에서의 다양한 응용분야가 예상된다. 이는 컴퓨터에 의한 대상의 정확한 인식의 결과가 사람과 컴퓨터간의 자연스러운 대화방법을 제공해 주고, 컴퓨터를 포함한 다양한 기계장치의 조작을 좀 더 편리하게 해줄 수 있기 때문이다. 최근 이러한 형태의 수요가 급속히 증가하고 있는 활용분야를 살

펴보면, 화자인식 카메라, 무인운반 시스템, 로봇비전, 거리감지 시스템, 가상공간을 이용한 원격조정 및 원격 회의시스템등을 들 수 있다[6, 7]. 최근 많이 활용되는 특징점을 기반으로 한 연구방법중 주목할 만한 연구분야는 행렬인수분해(matrix factorization) 기반의 방법이 있다. SVD(Singular Vector Decomposition)연산을 사용하여 물체의 3차원 움직임과 모양을 추정하는 최근의 연구로서는 원근 투영 카메라 모델을 선형적으로 근사화 한, 준원근 투영(paraperspective projection) 카메라 모델을 사용한 일괄처리(batch-type) 알고리즘[1]과 정사투영 카메라 모델을 사용한 순차형 알고리즘이 있다[2]. 그러나 전자의 경우는 원근 효과를 고려한 카메라 모델을 사용하였으나, 프레임마다의 처리가 고려되지 않았고, 후자의 경우에는 정사투영 카메라 모델을 사용하여 카메라의 축 방향의 움직임이 큰 경우, 많은 오

* 본 연구는 한경대학교 2002년도 학술연구조성비의 지원에 의한 것임.
[†] 정 회 원 : 국립한경대학교 정보제어공학과 교수
논문접수 : 2002년 11월 20일, 심사완료 : 2003년 3월 25일

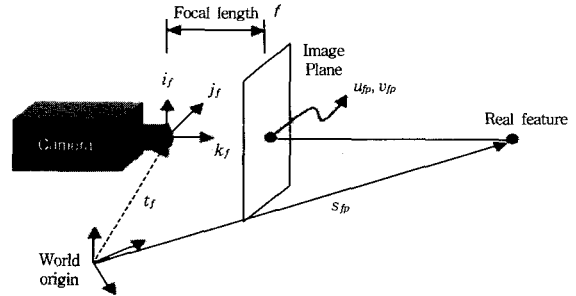
차를 수반하는 문제점이 있다. 또한 순차처리 형식의 SVD 연산을 이용한 연구[3]는 모양 및 움직임 정보 추출의 정확성이 개선되었다. 그러나 이 역시 동영상 입력에 대한 자동 특징점 추출에 어려움이 있고, 물체의 큰 움직임을 기반으로 하거나, 또는 연산처리의 오류로 인해 입력 특징점의 일부가 손실되어 개수가 변하는 경우는 정상적인 모양, 움직임 복원 연산이 불가능하다. 또한 2차원 영상면에서 획득한 객체의 특징점들은 원근투영 카메라모델(perspective projection camera model)의 장치를 통해서 얻어진 데이터들이나, 이를 이용하여 3차원적인 모양, 움직임 정보를 예측하는 연산 과정에서는 준원근 투영 카메라 모델(paraperspective projection camera model)을 사용하게 된다. 즉 두 카메라 모델의 차이로 인해 생기는 실제 영상면에서의 좌표 값 차이는 대상 물체의 깊이 값이 클수록, 또 물체의 크기가 클수록 근본적인 오차의 범위를 증가시킨다[8]. 한편 최근에 스테레오 영상 시퀀스를 이용한 방법들은 단일시점의 영상 시퀀스를 이용한 방법에 비해 더 많은 3차원 공간적 정보를 얻을 수 있음으로 인해, 물체와 배경과의 완전한 분리 및 3차원 객체의 복원측면에서의 효과를 기대할 수 있다[9]. 본 논문에서는 준원근 단일 카메라 사용에 의한 근본적 문제점을 극복하기 위해 스테레오 카메라를 이용한 방법을 제시한다. 본 논문에서는 가상의 실험용 영상으로부터 2차원적인 특징점들을 추출하고 이들의 이동궤적(trajectory)을 입력으로 하고, 준원근 카메라모델과 SVD에 의한 행렬인수분해 알고리즘, 스테레오 영상에 의한 깊이정보의 오차보상 등을 적용함으로써 객체의 3차원적인 위치정보와 움직임을 계산하고 그 결과를 평가한다.

2. 카메라 모델

2.1 카메라 좌표와 world 좌표

본 논문에서는 (그림 1)과 같이 물체는 정지하여 있고 카메라가 움직인다고 가정하여 필요한 정보를 계산한 후, 후반부에서 카메라에 대한 물체의 모양 및 움직임 변화로 역변환하는 방법을 취한다. 물체의 모양을 나타내는 물체표면의 특징점들의 좌표 \vec{s}_p 와 카메라의 움직임을 나타내기 위하여 전체 좌표계에서의 카메라 축의 방향에 대한 회전값을 표시한 $\vec{i}_f, \vec{j}_f, \vec{k}_f$ 평행 이동량 \vec{t}_f 을 매 프레임마다 계산한다. 이 경우 구하여진 카메라의 물체에 대한 상대적인 움직임을 카메라에 대한 물체의 움직임으로 전환하면 물체의 움직임과 모양을 계산해 낼 수 있다 SVD 연산을 이용한 순차형 알고리즘은 기존의 일괄처리 알고리즘[1]의 문제점인 메모리 증가문제를 해결할 수 있으며, 이로 인하여 실행 속도가 빨라지므로 실시간 적용에 유리하다는 장점이 있다. 카메라 모델로서는 비선형적인 원근 투영을 선형화한 준원근 투영 카메라 모델을 사용함으로써 기존의 알고리즘의 문

제점인 카메라의 축 방향, 즉 깊이방향의 움직임이 큰 경우에도 좋은 예측결과가 나오도록 하였다.

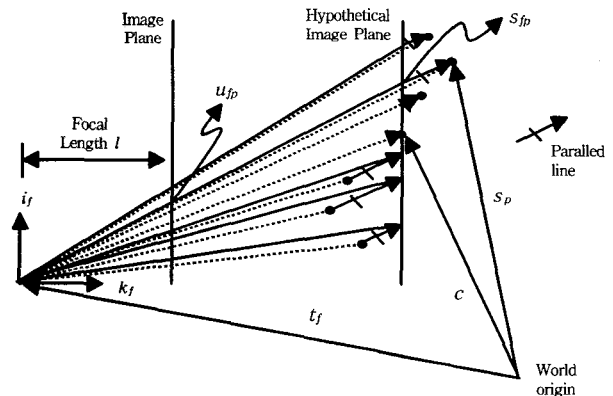


(그림 1) 카메라 좌표와 World 좌표

2.2 준원근 투영 카메라 모델

원근 투영 카메라 모델의 비선형적인 특성을 선형적인 상태로 근사화한 준원근 투영 카메라 모델은 (그림 2)와 같이 구성된다. 준원근 투영 카메라 모델에 의한 좌표는 교점이 있는 평면과 화면이 평행하기 때문에 가상화면과 영상 좌표를 단지 카메라의 초점 거리와 두 화면 사이의 거리에 비례하여 선형적으로 축소하는 효과를 갖는다. 3차원 물체상의 임의의 점 \vec{s}_p 를 카메라의 초점에서 물체의 중심(\vec{c})으로의 방향, $\vec{c} - \vec{t}_f$ 으로 투영한 가상화면에서의 좌표 \vec{s}_{fp} 는 다음과 같다.

$$\vec{s}_{fp} = \vec{s}_p - \frac{(\vec{s}_p \cdot \vec{k}_f) - (\vec{c} - \vec{t}_f) \cdot \vec{k}_f}{(\vec{c} - \vec{t}_f) \cdot \vec{k}_f} (\vec{c} - \vec{t}_f) \quad (1)$$



(그림 2) 준원근 투영 카메라 모델

간소화를 위해서 단위 초점거리 $l=1$ 과 전체 좌표계의 원점을 물체의 무게중심으로 취하면, \vec{s}_{fp} 를 화면상에 원근 투영하여 얻은 좌표 (u_{fp}, v_{fp}) 는 다음과 같다.

$$\begin{aligned} u_{fp} &= \frac{1}{z_f} \left\{ \left[\vec{i}_f + \frac{\vec{i}_f \cdot \vec{t}_f}{z_f} \right] \cdot \vec{s}_f - \vec{t}_f \cdot \vec{i}_f \right\} \\ v_{fp} &= \frac{1}{z_f} \left\{ \left[\vec{j}_f + \frac{\vec{j}_f \cdot \vec{t}_f}{z_f} \right] \cdot \vec{s}_f - \vec{t}_f \cdot \vec{j}_f \right\} \end{aligned} \quad (2)$$

식 (2)에서

$$z_f = -\vec{i}_f \cdot \vec{k}_f, x_f = -\frac{\vec{i}_f \cdot \vec{i}_f}{z_f}, \quad (3)$$

$$y_f = -\frac{\vec{i}_f \cdot \vec{j}_f}{z_f}$$

$$\vec{m}_f = \frac{\vec{i}_f - x_f \vec{k}}{z_f}, \quad \vec{n}_f = \frac{\vec{j}_f - y_f \vec{k}}{z_f} \quad (4)$$

가 성립되면,

$$u_{fp} = \vec{m}_f \cdot \vec{s}_p + x_f, \quad v_{fp} = \vec{n}_f \cdot \vec{s}_p + y_f \quad (5)$$

이 된다. 식 (5)에서 \vec{m}_f, \vec{n}_f 은 전체 좌표계에서의 카메라 축의 방향 $\vec{i}_f, \vec{j}_f, \vec{k}_f$ 를 포함하고 있으므로 카메라의 움직임을 나타내는 부분임을 알 수 있으며, 모양정보를 나타내는 부분인 \vec{s}_{fp} 와 분리되어 표현됨을 보여준다.

3. 3차원 형태 및 움직임 복원 연산

3.1 일괄 계산 형태의 알고리즘

식 (5)와 같이 나타나는 데이터를 모든 프레임에 대하여 취합한 후에 물체의 모양과 각 프레임에서의 움직임을 측정하는 일괄처리 알고리즘은 다음과 같다. 임의의 한 점에 대한 식 (2)를 P 개의 특징점 ($\vec{s}_1, \vec{s}_2, \dots, \vec{s}_p$)과 F 개의 프레임에 대하여 행렬식으로 표현하면 다음과 같다.

$$\begin{bmatrix} u_{11} & \dots & u_{1P} \\ \vdots & \vdots & \vdots \\ u_{F1} & \dots & u_{FP} \\ v_{11} & \dots & v_{1P} \\ \vdots & \vdots & \vdots \\ v_{F1} & \dots & v_{FP} \end{bmatrix} = \begin{bmatrix} \vec{m}_1 \\ \vdots \\ \vec{m}_F \\ \vec{n}_1 \\ \vdots \\ \vec{n}_F \end{bmatrix} \begin{bmatrix} \vec{s}_1 & \dots & \vec{s}_p \end{bmatrix} + \begin{bmatrix} x_1 \\ \vdots \\ x_F \\ y_1 \\ \vdots \\ y_F \end{bmatrix} \begin{bmatrix} 1 & \dots & 1 \end{bmatrix} \quad (6)$$

식 (6)을 간단히 표현하면 식 (7)과 같이 요약된다.

$$W = MS + T[1 \dots 1] \quad (7)$$

여기서 $W(2F \times P)$ 는 입력 행렬, $M(2F \times 3)$ 은 움직임 행렬, $S(3 \times P)$ 는 모양행렬, $T(2F \times 1)$ 는 평행이동 벡터이다. 식 (2)와 전체 좌표계의 원점을 물체의 중심에 옮겨 놓음으로써 평행이동 T 의 원소인 x_f 와 y_f 는 특징점들의 좌표로부터 다음과 같이 직접 구할 수 있다.

$$x_f = \frac{1}{P} \sum_{p=1}^P u_{fp}, \quad y_f = \frac{1}{P} \sum_{p=1}^P v_{fp} \quad (8)$$

식 (7)의 $T[1 \dots 1]$ 벡터를 좌변으로 이항하면 rank가 3인 두 개의 행렬 M 과 S 의 곱으로 나타나는 W^* 을 얻을 수 있다.

$$W^* = W - T[1 \dots 1] = MS \quad (9)$$

이때 W^* 는 rank가 3보다 클 수 없으나 입력 영상에 존재하는 잡음 등에 의해 보다 큰 경우가 발생할 수도 있으므로 잡음을 제거하고 M 과 S 를 분리하기 위하여 SVD 연산을 사용한다. 입력행렬 W^* 을 SVD 연산하면, 상대적으로 값이 큰 3개의 단일 값들(singular value)과 잡음에 의한 단일 값들이 함께 발생하므로 가장 큰 3개만을 채택한다. 이 경우 $W^* \in R^{2F \times P}$ 의 SVD 연산 결과인 W^*_{SVD} 는 두 개의 orthogonal matrix인 $Q_1 \in R^{2F \times P}, Q_2 \in R^{2F \times P}$ 와 대각행렬 $\Sigma \in R^{3 \times 3}$ 로 표현할 수 있으며 식 (10)과 같다.

$$W^*_{SVD} = Q_1 \Sigma Q_2^T \quad (10)$$

이때, $\Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3)$ 이고 $\sigma_1 \geq \sigma_2 \geq \sigma_3 > 0$ 이며 Q_1 은 W^*_{SVD} 의 왼쪽 단일 행렬(singular matrix)의 열 중에서 가장 큰 3개의 단일 값들에 대응하는 열로 이루어진 행렬이고, Q_2 는 W^*_{SVD} 의 오른쪽 단일행렬의 열 중에서 가장 큰 3개의 단일 값들에 대응하는 열로 이루어진 행렬이다. $\hat{M} = Q_1 \Sigma, \hat{S} = Q_2^T$ 로 정의함으로써 W^*_{SVD} 는 $W^*_{SVD} = \hat{M} \hat{S}$ 로 분리될 수 있다. 이 경우 $\hat{M} = [\hat{m}_1 \dots \hat{m}_f \hat{n}_1 \dots \hat{n}_f]^T, \hat{S} = [\hat{s}_1 \dots \hat{s}_p]$ 이다. \hat{M} 행렬에 의해서 전개(span)된 열 공간은 움직임을 나타내는 공간이므로 움직임 공간이라 하고, \hat{S} 에 의해서 전개된 행 공간은 물체의 모양을 나타내는 공간이므로 모양공간이라고 한다. 이러한 공간의 차원은 rank이론에 의해서 3이라는 것을 알 수 있으므로 SVD를 사용한 행렬인수분해 과정은 결국 높은 차원을 갖는 입력공간(input space)에서 차원이 3인 움직임을 나타내는 부분공간 M 과 모양을 나타내는 부분 공간 S 를 찾는 것이다.

3.2 실시간 적용을 위한 순차형 알고리즘

3.2.1 매 프레임에서의 입력 처리

매 프레임에서 시스템에 입력되는 데이터는 화면에서의 특징점의 좌표로 이루어진 다음의 두 개의 벡터 \vec{U}_f 와 \vec{V}_f 이다.

$$\begin{aligned} \vec{U}_f &= [u_{f1}, u_{f2}, \dots, u_{fp}]^T, \\ \vec{V}_f &= [v_{f1}, v_{f2}, \dots, v_{fp}]^T \end{aligned} \quad (11)$$

여기서 u_{fi} 와 v_{fi} 는 f 번째 프레임에서의 i 번째 특징점의 이미지 평면에서 x 와 y 좌표를 각각 나타낸다. 실시간 적용을 위한 순차형 알고리즘은 매 프레임마다 (\vec{U}_f, \vec{V}_f) 를 입력하여 f 번째 프레임에 해당되는 움직임 ($\vec{i}_f, \vec{j}_f, \vec{k}_f$)과 모양정보 (S_f)를 계산하고, 다음 프레임 $f+1$ 번째에 대한

입력 ($\vec{U}_{f+1}, \vec{V}_{f+1}$)을 받고 새로운 카메라의 움직임 ($\vec{i}_{f+1}, \vec{j}_{f+1}, \vec{k}_{f+1}$)과 더 정확해진 모양정보 (S_{f+1})을 계산한다. 여기서 새로운 matrix time series $Z_f \in R^{p \times p}$ 을 다음과 같이 정의하여,

$$Z_f = Z_{f-1} + U_f U_f^T + V_f V_f^T = W_f^T W_f \quad (12)$$

를 입력 행렬로 사용한다. 입력행렬 Z_f 는 프레임 수가 계속 증가하여도 메모리 크기가 증가하지 않는 장점을 갖는다.

3.2.2 순차적인 모양공간과 움직임 공간 연산

W_f 의 rank는 3이므로 Z_f 의 rank도 3이고 W_f 의 SVD가 다음과 같다면

$$W_f = Q_{1f} \Sigma_f Q_{2f}^T \quad (13)$$

여기서 $Q_{1f} \in R^{2F \times 3}, Q_{2f} \in R^{p \times 3}$ 이고 $\Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3), \sigma_1 \geq \sigma_2 \geq \sigma_3 > 0$ 이므로 Z_f 는 식 (14)와 같이 나타낼 수 있다.

$$Z_f = (Q_{1f} \Sigma_f Q_{2f}^T)^T (Q_{1f} \Sigma_f Q_{2f}^T) = (Q_{1f} \Sigma_f^2 Q_{2f}^T) \quad (14)$$

Z_f 의 고유벡터(eigenvector)들이 W_f 의 오른쪽 단일 벡터 들인 Z_f 의 고유값들을 구함으로써 모양공간(shape space)을 나타내는 \hat{S} 를 구할 수 있다. 단일 값 행렬인 Σ_f 는 Z_f 의 고유값(eigenvalue)들의 제곱근에 해당된다. 모양공간을 알면 식에 의해서 현재 (f 번째 프레임) 움직임 공간을 나타 내는 \vec{m}_f, \vec{n}_f 는 다음과 같이 구할 수 있다.

$$\widehat{m}_f = U_f^T Q_{2f}, \quad \widehat{n}_f = V_f^T Q_{2f} \quad (15)$$

3.2.3 유일한 해를 찾기 위한 제한조건

하나의 행렬을 두 개 행렬의 곱으로 나타내는 것은 유일하지 않으므로 위의 식을 두 개의 행렬로 분리하는 것도 유일하게 결정되지 않는다. 이의 물리적인 의미는 움직임 공간과 모양공간을 구하였지만 현재의 정확한 움직임과 물체의 모양을 알 수는 없고 단지 정확한 값이 들어있는 공간만을 알고 있을 뿐이다. 따라서 정확한 값을 찾기 위해서 일정한 제한조건을 포함하는 비-단일 행렬(non-singular matrix) $A \in R^{3 \times 3}$ 을 삽입하여 움직임 공간 M 에서 정확한 움직임 M 과 모양공간 \hat{S} 에서 정확한 모양 s 를 구한다.

$$W^*_{SVD} = \widehat{M} A A^{-1} \widehat{S} = M S, \quad (16)$$

$$M = \widehat{M} A, \quad S = A^{-1} \widehat{S}$$

여기서 A 를 구하기 위해 다음의 조건들을 이용한다. 즉

식 (5)로부터

$$|\vec{m}_f|^2 = \frac{1 + x_f^2}{z_f^2}, \quad |\vec{n}_f|^2 = \frac{1 + y_f^2}{z_f^2} \quad (17)$$

이고 다음의 두 제한조건을 만족해야 한다.

[제한조건 1]

$$\frac{|\vec{m}_f|^2}{1 + x_f^2}, \quad \frac{|\vec{n}_f|^2}{1 + y_f^2} (= \frac{1}{z_f^2}) \quad (18)$$

[제한조건 2]

$$\vec{m}_f \cdot \vec{n}_f = \frac{x_f^2 y_f^2}{z_f^2} = x_f y_f \frac{1}{2} \left(\frac{|\vec{m}_f|^2}{1 + x_f^2} + \frac{|\vec{n}_f|^2}{1 + y_f^2} \right) \quad (19)$$

[제한조건 2]을 보면 z_f^2 에 대한 값을 식 (17)에 있는 두 개 중 하나를 사용할 수 있지만 잡음이 섞인 입력으로 두 개의 값은 정확하게 일치하지는 않을 수 있기 때문에 두 값의 평균을 취한다.

대칭행렬 $L_f \in R^{3 \times 3}$ 을 다음과 같이 정의한다.

$$L = A A^T = \begin{bmatrix} l_1 & l_2 & l_3 \\ l_2 & l_4 & l_5 \\ l_3 & l_5 & l_6 \end{bmatrix},$$

$$\vec{m}_f \cdot \vec{n}_f = (\widehat{m}_f A) (\widehat{n}_f A)^T = \widehat{m}_f A A^T \widehat{n}_f^T = \widehat{m}_f L \widehat{n}_f^T \quad (20)$$

또한 식 (20)에서 L 을 $\vec{l} = [l_1, l_2, l_3, l_4, l_5, l_6] \neq 0$ 이라 면 다음과 같이 표현할 수 있다.

$$\vec{m}_f \cdot \vec{n}_f = g(\widehat{m}_f, \widehat{n}_f) \vec{l},$$

$$\vec{m}_f = (m_{f1} m_{f2} m_{f3}), \quad \vec{n}_f = (n_{f1} n_{f2} n_{f3}),$$

$$g(\vec{m}_f, \vec{n}_f) = [m_{f1} n_{f1} \quad 2m_{f1} n_{f2} \quad 2m_{f1} n_{f3} \quad m_{f2} n_{f2} \quad 2m_{f2} n_{f3} \quad m_{f3} n_{f3}]$$

따라서 식 (18), 식 (19)는 행렬 형태로 다음과 같이 다시 표현할 수 있다.

[제한조건 1]

$$\vec{\alpha}_f \vec{l} = 0, \quad \vec{\alpha}_f = \frac{g(\widehat{m}_f, \widehat{m}_f)}{1 + x_f^2} - \frac{g(\widehat{n}_f, \widehat{n}_f)}{1 + y_f^2}$$

[제한조건 2]

$$\vec{\beta}_f \vec{l} = 0, \quad \vec{\beta}_f = g(\widehat{m}_f, \widehat{n}_f) - x_f y_f \frac{1}{2} \left(\frac{g(\widehat{m}_f, \widehat{m}_f)}{1 + x_f^2} - \frac{g(\widehat{n}_f, \widehat{n}_f)}{1 + y_f^2} \right)$$

제한조건을 f 번째 프레임까지 적용하면

$$G_f \vec{l} = \vec{0} \quad (21)$$

여기서 $G_f = [\vec{\alpha}_1 \ \vec{\beta}_1 \ \dots \ \vec{\alpha}_f \ \vec{\beta}_f]$ 이다. 식 (21)이 $\vec{l} \neq \vec{0}$ ($\|\vec{l}\| = 1$)이 존재하려면 $\text{nullity}(G_f) \geq 1$ 이어야 하지만 잡음의 영향 때문에 G_f 는 full rank(즉, nullity가 0)를 갖는 시스템이므로 $\vec{0}$ 이 아닌 \vec{l} 은 존재하지 않는다. 잡음의 영향을 고려하고 $\vec{l} \neq \vec{0}$ 인 \vec{l} 을 구하기 위해 에러 $\|G_f \vec{l} = \vec{0}\|$ 을 최소로 하는 최소 자승해인 \vec{l} 을 구하여야 한다. 최소 자승해를 구하기 위해 식 (21)의 양변에 G_f^T 을 곱하면

$$G_f^T G_f \vec{l} = D_f \vec{l} = \vec{0} \quad (22)$$

이고 $D_f \in R^{6 \times 6}$ 을 eigendecomposition을 하면 6개의 고유값들과 고유 벡터들이 각각 존재한다. f 프레임에서의 D_f 는 다음과 같이 구한다.

$$D_f = D_{f-1} + \alpha_f \alpha_f^T + \beta_f \beta_f^T$$

고유값들 중에서 가장 작은 값에 해당하는 고유벡터를 법선 벡터로 갖는 평면은 G_f 의 행(row)벡터 들과의 오차를 최소로 갖는 평면이 된다. 그러므로 D_f 의 고유값 중에서 가장 작은 값에 해당하는 고유벡터를 \vec{l} 로 정합함으로써 $\|G_f \vec{l} = \vec{0}\|$ 를 최소화할 수 있다. 이렇게 구한 L 이 유한하다면, cholesky decomposition을 사용하면 L 에서 선형 변환 행렬인 A 를 구할 수 있다.

3.2.4 단일 카메라에서의 물체의 모양과 움직임 복원

위와 같이 계산하여 얻은 행렬 A 를 식 (16)에 대입하여 모양공간 \hat{S} 에서 물체의 모양을 나타내는 S_f 를 추정할 수 있고 움직임 공간 \hat{M} 에서 움직임을 나타내는 $\vec{m}_f = \widehat{m}_f A$ 와 $\vec{n}_f = \widehat{n}_f A$ 를 구할 수 있다. 우선, 식 (4)로부터 다음 식들을 얻을 수 있다.

$$\vec{i}_f = z_f \vec{m}_f + x_f \vec{k}_f, \quad \vec{j}_f = z_f \vec{n}_f + y_f \vec{k}_f \quad (23)$$

실질적인 카메라의 움직임을 나타내는 $\vec{i}_f, \vec{j}_f, \vec{k}_f$ (즉, f 번째 프레임에서 전체 좌표계에서의 카메라의 방향)들이 서로 직교하고 크기가 1이라는 사실과 식 (23)을 이용하면 다음 식이 성립된다.

$$\vec{i}_f \times \vec{j}_f = (z_f \vec{m}_f + x_f \vec{k}_f) \times (z_f \vec{n}_f + y_f \vec{k}_f) = \vec{k}_f \quad (24)$$

$$|\vec{i}_f| = |z_f \vec{m}_f + x_f \vec{k}_f| = 1 \quad (25)$$

$$|\vec{j}_f| = |z_f \vec{n}_f + y_f \vec{k}_f| = 1 \quad (26)$$

위 식에서 z_f 을 알지 못하지만 식 (18)을 적용하면 다음과 같이 식을 얻을 수 있다.

$$E_f \vec{k}_f = H_f \quad (27)$$

여기서

$$E_f = \begin{bmatrix} (\widehat{m}_f \times \widehat{n}_f) \\ \widehat{m}_f \\ \widehat{n}_f \end{bmatrix}, \quad H_f = \begin{bmatrix} 1 \\ -x_f \\ -y_f \end{bmatrix},$$

$$\widehat{m}_f = \sqrt{1+x_f^2} \frac{\vec{m}_f}{|\vec{m}_f|}, \quad \widehat{n}_f = \sqrt{1+y_f^2} \frac{\vec{n}_f}{|\vec{n}_f|}$$

그러므로 카메라의 움직임을 나타내는 $(\vec{i}_f, \vec{j}_f, \vec{k}_f)$ 는 다음과 같이 구해진다.

$$\vec{k}_f = E_f^{-1} H_f, \quad \vec{i}_f = \widehat{n}_f \times \vec{k}_f, \quad \vec{j}_f = \widehat{m}_f \times \vec{k}_f \quad (28)$$

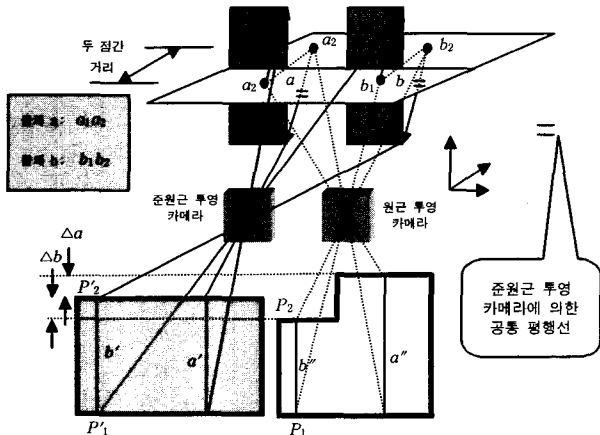
식 (28)로 부터 $\vec{i}_f, \vec{j}_f, \vec{k}_f$ 을 구할 수 있고 식 (23)으로부터 깊이 정보인 z_f 을 계산할 수 있다. $\vec{i}_f, \vec{j}_f, \vec{k}_f, x_f, y_f, z_f$ 를 알고 있기 때문에 식 (3)을 이용하여 t_f 도 구하여 진다.

4. 스테레오 영상을 이용한 물체의 모양과 움직임 복원

4.1 준원근 단일 카메라 모델의 문제점

단일 카메라를 이용한 본 연구의 처리과정은 준원근 카메라 모델을 이용하여, 2차원 영상으로부터 근본적으로 찾아내기 힘든 3차원 정보를 복원해 냄으로써, 강력한 성능을 보여준다. 하지만 단일 카메라를 입력으로 사용하는 대상물체의 모든 특징점들은 물체의 다양한 움직임으로 인하여, 항상 안정적으로 추출될 수는 없으며, 또 특징점의 분포를 카메라의 좁은 시야 내로 한정해야 하는 문제점을 가지고 있다. 특히 3차원 정보의 복원을 위하여 입력 특징점의 위치는 물체의 다양한 깊이정보를 반영하는 넓은 위치에 분포할 수록 정확한 예측결과를 기대할 수 있으므로, 단일 카메라에 의한 시각적 제한성은 물체의 움직임을 제한하는 요인이 된다. 이에 본 연구에서는 2대의 카메라를 사용하여, 물체의 모양에 대한 시각의 범위를 넓힘으로써, 더 많은 3차원 정보를 취득할 수가 있고 또한 그 결과를 단일 카메라의 경우와 비교한다. 단일 카메라에 의한 준원근 모델을 사용하는 가장 큰 문제점은 입력으로 사용하는 2차원 영상면에서의 특징점들이 원근투영 카메라(perspective projection camera)에 의해 얻어진 좌표인 반면, 실제의 연산을 위한 카메라 모델은 준원근 투영 카메라 모델이라는 점이다. 각각, 좌표 값 변화에 대한 비선형성과 선형성을 가지고 있는 두 모델을 같은 모델로 가정하고 사용하면서 근본

적인 모양정보 예측의 오차가 발생되며, 깊이 방향의 오차는 원 측정값의 30%에 달한다. 이는 움직임 예측을 위한 연산에 그대로 반영되므로 같은 오차를 유발한다. 따라서 본 논문에서는 스테레오 카메라를 사용함으로써, 3차원적인 물체의 측정값에 가깝게 근사하여 오차가 보상되는 결과를 보여준다. 본 논문의 연구과정에서 사용하는 두 대의 카메라는 에피폴라 구속조건(epipolar constraint)을 만족하도록 한다.



(그림 3) 카메라 모델에 따른 깊이 값 오차 해석

4.2 연산모델의 깊이 오차 해석 및 보상

본 논문에서 물체의 3차원적인 움직임을 추적하기 위하여 사용한 카메라 모델은 준원근 카메라 모델이다. 그런데 이러한 해를 구하기 위한 과정에서 입력으로 사용한 물체의 2차원 특징점들은 대부분의 경우처럼 원근 카메라 모델을 사용하여 얻어진 결과이다. 이는 근본적으로 입력 데이터의 비선형성과 연산 모델의 선형성의 차이로 인한 오차를 발생시킨다.

(그림 3)에서 보듯이 동일한 평면상에 있는 각각 2점으로 이루어진 2개의 성분 a, b 의 특징점에 대해, 두 카메라 모델의 실제 영상면에 비추어지는 위치 값을 비교해 보면 다음과 같다. 먼저 Z축 방향으로 a 만큼 떨어져 있는 두 점 a_1, a_2 와 b 만큼 떨어져 있는 b_1, b_2 는 a, b 가 서로 같다면 준원근 투영 카메라 모델의 평행선을 이용한 투영관계를 이용하면, 카메라와 특징점 간의 위치와 상관없이 실제 영상면에서 항상 같은 크기의 거리로 해석된다. 그러나 원근 투영법에 의한 각각의 두 점 사이의 거리 a, b 는 같은 거리 값을 가지고 있더라도 카메라와의 거리 관계에 따라, 서로 다른 깊이 정보를 가지고 있는 것으로 해석된다. 오차가 발생하는 과정을 좀 더 상세히 해석하면 다음과 같다. 평면상의 임의의 점 b_1, b_2 가 원근 투영 카메라에 의해 실제 영상면에 만들어진 점이 P_1, P_2 라고 할 때,

$$b' = P_2 - P_1 \tag{29}$$

이 되며, 이는 가상 영상면을 중심으로 z 방향으로 떨어진 b 의 거리가 2차원적인 영상면에 반영된 거리이다. 또한 그

림의 준원근 투영 카메라 모델에 의한 동일한 투영이 이루어졌다고 했을 때의 거리값은

$$b' = P_2' - P_1 \tag{30}$$

으로 표현된다. 본 연구에서 SVD를 이용한 행렬인수분해 방법은 원근 투영법에 의해 얻어진 입력을 준원근 투영법에 의해 연산하므로 식 (29)에 의해 구해진, Z축상의 특징점간의 변화된 거리가 식 (30)에 의해 해석된다. 이는 $\Delta a = |a' - a''|, \Delta b = |b' - b''|$ 만큼의 거리차가 실제 3차원상의 특징점 거리에 대한 오차로 발생된다. 이 과정에서 보상되어야 할 오차는 두 카메라간의 거리가 충분히 클 때, $\Delta a \approx \Delta b$ 가 되므로 다음과 같이 나타난다.

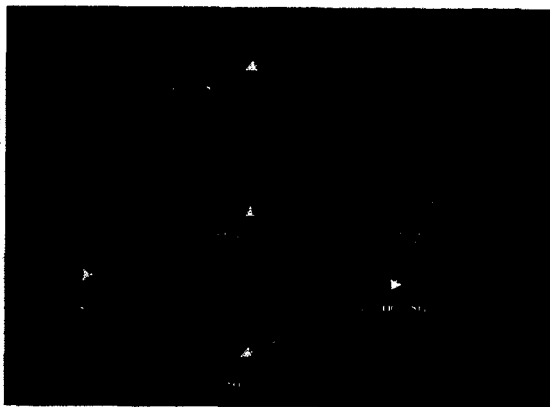
$$S_{fa} = S_{fb} = \frac{S_{fa} + S_{fb}}{2} \approx \frac{S_{fa'} + S_{fb'}}{2} \tag{31}$$

$$= \frac{(S_{fa} + \Delta a) + (S_{fb} - \Delta b)}{2} \approx \frac{S_{fa} + S_{fb}}{2}$$

여기서 S_{fa}, S_{fb} 는 a, b 의 길이를 장축으로 갖는 임의의 물체에 대한 모양으로 가정한다. 위 식과 같이 실제 물체 모양의 측정값에 근사하기 위해서는 a 와 b 의 계산결과에 대한 평균값을 구하면 정답에 근사하는 효과를 얻을 수 있다. 즉 카메라의 초점으로부터 가까운 거리에 있는 점의 깊이 값 변화는 상대적으로 적게 계산되며, 초점으로부터 먼 거리에 있는 특징점의 깊이 값의 변화는 변화량이 동일하더라도 상대적으로 먼 거리에 있는 점으로 해석된다. 이러한 두 카메라 모델의 차이로 인한 오차는 실험의 결과 중 (그림 9)에서 보여주는 Sf[3]의 값 중 11~20번까지의 특징점이 갖는 오차를 정확히 설명해 준다. 즉 11~20까지의 특징점들은 카메라의 초점과 같은 평면상에 위치하면서, 입력되는 특징점 들은 모두 원근 투영 카메라 모델에 의해 얻어진 경우이다. 카메라의 위치가 물체의 움직임에 대해 정면에 위치하지 않으므로 좌우의 특징점들의 동일한 양의 움직임 변화도 영상면에서는 서로 다른 변화량을 제공할 것이며 이러한 결과는 (그림 2)의 준원근 투영 카메라 모델을 가정한 해석을 하였을 때, a 와 b 의 거리 값이 서로 다른 것으로 해석이 된다. 따라서 본 논문에서는 위의 해석과 같이 오차 보상효과가 있는 스테레오 영상을 사용한 순차형 SVD 연산 방법을 제안한다. 본 절에서 제기한 위의 오차는 단일 카메라가 물체의 깊이정보를 완벽하게 복원할 수 없음을 보여주는 이유가 되며, 카메라가 물체의 대칭적인 움직임이 있는 물체의 중심에 위치할수록 이러한 두 카메라 모델의 차이로 인한 오차는 감소될 수가 있다. 따라서 2차원 특징점의 복원을 위한 방법으로, 물체의 움직임을 대칭적인 시각에서 충분히 반영할 수 있을 만큼 거리를 두고 카메라를 배치한 후, 두 카메라의 준원근 투영법에 의한 각각의 결과에 대해 평균적 보상을 취해 주고 그 결과를 움직임 예측의 연산에 활용함으로써, 측정값에 근접하는 결과를 얻을 수 있다.

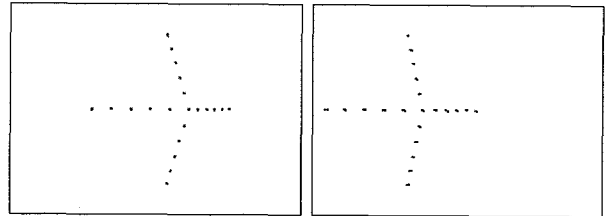
5. 실험 및 분석

스테레오 영상의 실험 환경을 위해서 본 연구에서는 우선 2대의 카메라를 이용하며, 3D의 특징점을 잘 반영하고 특징점 측정이 가능한 실험영상을 제작하여 적용하였다. (그림 4)의 영상은 피라미드의 4개 모서리 부분에 특징점을 배치함으로써 3차원의 각 축 방향의 움직임이 발생할 때, 3차원적인 모양과 움직임의 정보가 선형적이고 고른 영역에서 제공될 수 있도록 제작되었다. 이는 곧 본 논문에서의 알고리즘에 대한 성능평가를 위한 정답영상이 된다. (그림 4)에서 제작된 실험영상의 정면상과 3차원상의 주요 위치 값들이 나타나 있으며 모든 위치 값들은 피라미드의 꼭지점(0,0,0)을 기준으로 설정되었다. 카메라의 위치는 왼쪽의 경우 (0, -40, -400)이며, 오른쪽은 (0, 40, -400)이다. (그림 5), (그림 6)은 실험을 위한 60개의 프레임 중 20번째, 40번째에 해당하는 왼쪽과 오른쪽에서 획득한 움직임 영상이다. 피라미드 물체는 60프레임에 걸쳐 x 축을 중심으로 좌·우 30도의 회전과 y 축을 중심으로 한 위, 아래 30도의 회전을 하도록 구성되었다. 또한 특징점의 간격은 x, y 축 방향으로 20씩이며, z 축의 방향으로 5씩의 간격으로 구성되어 있다. 알고리즘을 통해 물체의 모양을 측정된 결과는 중심점을 제외한 20개 특징점들이 중심점인 (0, 0, 0)으로부터의 갖는 3차원적인 상대거리로서 표현되며, (그림 7), (그림 8), (그림 9)에서와 같이 단일 카메라 사용시의 영상(왼쪽, 오른쪽), multi view 사용시의 영상에 대하여 상대 비교하였으며, 그래프의 종축에 있는 Sf[1], Sf[2], Sf[3]는 객체의 무게중심으로 부터의 x, y, z 방향의 특징점 위치를 나타낸다. 그림의 모양정보는 마지막 프레임까지 알고리즘이 수행된 결과이다. 그림의 결과와 같이 단일 카메라를 사용한 경우는 카메라를 포함한 평면에 존재하는 특징점들에 대해 3차원적인 움직임의 정보를 충분히 얻지 못하고 카메라 모델에 의한 z 축 방향의 예측오차가 발생함을 알 수 있으며, 두 대의 카메라를 이용한 경우, 각각의 단일 카메라로부터의 오차부분을 평균적으로 보상함으로써 실제 값에 정확하게 근접함을 알 수 있다.

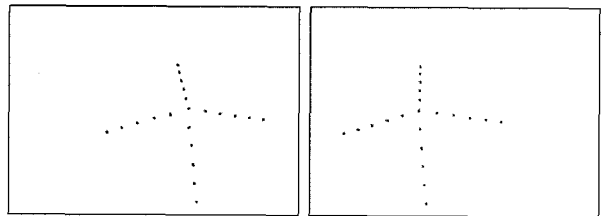


(그림 4) 3차원 정보를 포함하는 피라미드형 실험영상의 위치정보(정면상)

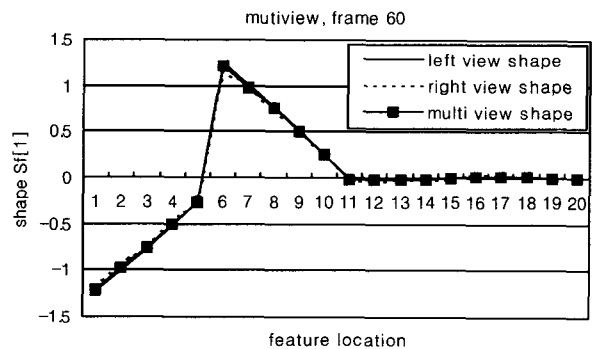
또한 x, y, z 축을 중심으로 한 회전량을 각각 3차원에서의 회전축인 α, β, γ 라고 정의하고 물체의 움직임에 대한 정보로 추출한다. 그 결과는 (그림 10)에 나와 있으며, 물체의 모양 정보가 정확히 계산되었을 때, 움직임 정보 또한 상대적으로 정확한 값을 가질 수 있으므로, 스테레오 영상에 의해 보상된 경우가 가장 실제 상황과 일치하는 결과를 가져온 것으로 나타났다.



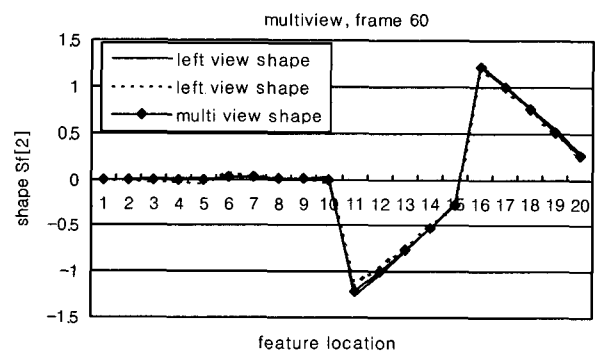
(그림 5) 스테레오 카메라에서 본 움직임이 있는 실험영상의 20번째 프레임 모습



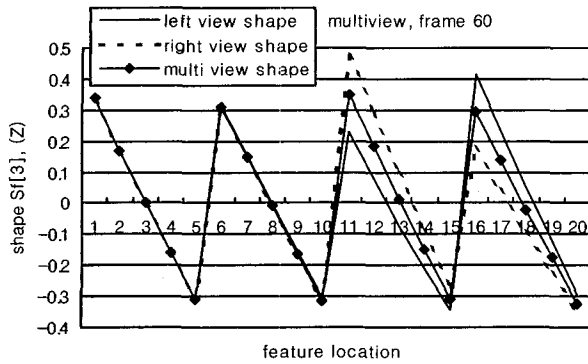
(그림 6) 스테레오 카메라에서 본 움직임이 있는 실험영상의 40번째 프레임 모습



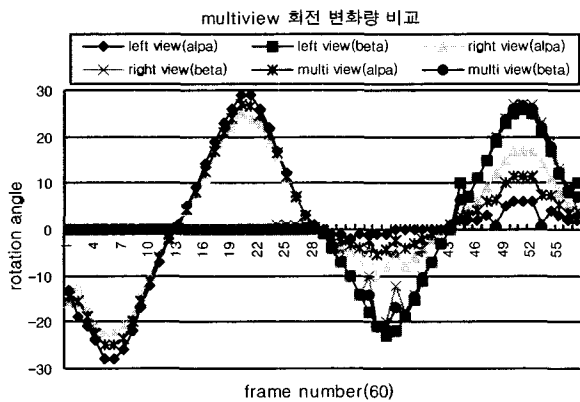
(그림 7) 모양 정보의 복원 결과 비교(Sf[1])



(그림 8) 모양 정보의 복원 결과 비교(Sf[2])



(그림 9) 모양 정보의 복원 결과 비교(Sf[3])



(그림 10) 물체의 움직임 정보(α, β, γ)의 변화량 비교

6. 결 론

본 논문에서는 스테레오 원근 카메라 모델을 이용한 순차적인 SVD 연산을 취하여 2차원 영상 시퀀스로부터 3차원적인 객체의 특징점(모양정보)과 움직임 정보를 프레임단위로 복원하는 방법을 제시하였다. 또한 두 대의 카메라를 사용하여 multi-view에 의한 깊이 정보 분석방법을 통해 준 원근 카메라 모델의 근본적 오차를 보상하였다. 일반적으로 2차원 입력 영상을 얻기 위해 사용하는 카메라는 원근 투영 카메라 모델인 반면, 3차원적인 모양과 움직임을 계산하기 위해 설정한 카메라 모델은, 원근 투영 카메라 모델의 비선형성을 보완하는 준원근 투영 카메라 모델이다. 두 모델간의 차이는 2차원의 영상으로부터 3차원 정보를 예측하기 위한 상황의 근본적인 문제점이며, 이는 본 연구의 오차 해석에 의해 두 대의 카메라에 의한 정보의 평균값을 적용해 줌으로써, 측정값에 근사하는 방법을 제시한다. 단일 카메라 사용시, 깊이 방향의 모양정보는 측정값의 20%에 이르며, 본 연구의 평균 보상법에 의해 오차의 범위는 측정값의 3%이내로 개선된다.

향후에는 본 연구에서의 연구결과를 토대로 저 전송률 동화상 통신의 규격을 최대한 만족할 수 있는 3차원 모양 특징점의 완전한 복원과 자동 특징점 추출 오류 방지, 움직임에 대한 실시간 처리 시스템 구성 등이 요구된다.

참 고 문 헌

- [1] C. J. Poelman and T. Kanade, A Paraperspective Factorization Method for Shape and Motion Recovery, *Technical Report CMU-CS-93-219*, Carnegie Mellon University, 1993.
- [2] T. Morita and T. Kanade, A Sequential Factorization Method for Shape and Motion from Image Stream, *Proc. the 1994 ARPA Image Understanding Workshop*, Vol.2, pp. 1177-1188, Nov., 1994.
- [3] B. O. Jung, A Sequential Algorithm for 3-D Shape and Motion Recovery from Image Sequences, *Thesis for the degree of master*, Korea University, Jun., 1997.
- [4] J. M. Chung and N. Ohnishi, Cue Circles : Image Feature for Measuring 3D Motion of Articulated Objects using Sequential Image Pair, *Proc. Intl Conf. Face and Gesture Recognition*, Nara(Japan), pp.474-479, April, 1998.
- [5] A. Azarbayejani and A. Pentland, Recursive Estimation of Motion, Structure and Focal Length, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.7, No.6, pp.562-575, Jun., 1995.
- [6] Rama Chellappa, C. L. Wilson and S. Sirohey, Human and Machine Recognition of Faces : A Survey, *Proc. IEEE*, Vol.83, No.5, pp.705-740, May, 1995.
- [7] J. Weng, N. Ahuja and T. S. Huang, Optimal Motion and Structure Estimation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.15, No.9, Sept., 1993.
- [8] T. S. Huang and O. D. Faugeras, Some Properties of the E-matrix in Two-view Motion Estimation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.11, No.12, pp.1310-1312, Dec., 1989.
- [9] A. Pentland and B. Horowitz, Recovery of Non-rigid Motion and Structure, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.13, No.7, pp.730-742, 1991.



김 상 훈

e-mail : kimsh@hnu.hankyong.ac.kr

1987년 고려대학교 전자공학과 학사

1989년 고려대학교 대학원 전자공학과 석사

1999년 고려대학교 대학원 전자공학과 박사

2001년 UWA 방문연구원(Australia)

1989년~1994년 LG반도체 연구원

1999년~2001년 KIST 위촉연구원

1999년~현재 국립한경대학교 정보제어공학과 조교수

관심분야 : 3D 영상처리, face detection, real-time object tracking