

카메라 영상에 의한 물체와의 거리 측정에 관한 연구

김 승 태, 이 종 훈, 김 도 성, 이 명 호
연세대학교 전기공학과

A Study on Range Finding Using Camera Image

SeungTai Kim, JongHun Lee, DoSung Kim, MyoungHo Lee
Dept. of Electrical Engineering, Yonsei University

Abstract

This thesis deals with range finding using one camera and laser pointer.

Range finding will be used further recognition of the image, that is, range image which allows further segmentation of the scene.

In the first step, camera modeling is performed by camera calibration which executes least square fit. Least square fit uses the method of singular value decomposition. And perspective transform of camera is obtained.

Lastly range finding is performed by triangulation principle. The result of this algorithm are displayed.

비가 고가이다.

본 연구에서는 물체의 3차원 정보중의 하나인 camera로부터 물체까지의 거리 측정을 위하여 spot projector와 camera를 사용한다. 먼저 pinhole camera에 대한 modeling으로 부터 시작한다. camera model에 필요한 여러 parameter들을 구하기 위해서 camera calibration 작업을 행한다. 이 calibration 작업에 있어서 정확성을 높이기 위하여 singular value decomposition에 의한 Least square fit을 행하였다.

이 camera model로부터 perspective transform에 대한 식을 유도해 낸 다음 digitized picture와 imaging surface 위 점에서 행과 열사이의 대응 다음 triangulation에 의해서 range를 구한다.

제1장 서론

Robot vision에 있어서 대부분의 내용은 주로 물체의 상의 디지털 표현인 image를 얻고 분석하는 것에 관한 것이다. 그러나 robot의 경우 어떤 물체를 원하는 목적대로 조작하기 위해서는 물체의 3차원 위치를 양적으로 표시해야 한다. 컴퓨터 시각 장치를 통하여 입체적 구조가 상실된 2차원 화상으로부터 물체의 3차원 정보를 추출하려는 여러연구가 진행되었다. 이들 연구에서 논의된 레이저 투영(laser projection)방법이나 구조화된 빔(structured lighting)방법들은 높은 정확도를 얻을 수 있으나 실험이 복잡하고 장

제2장 카메라 모델링

물체가 렌즈를 통하여 plane의 어디에 부사되는지를 알아 내기 위해서는 먼저 camera를 수학적으로 적절히 묘사하는 것이 필요하다. 렌즈에 의한 복잡성을 피하기 위하여 가장 간단한 형태인 pinhole 카메라를 고려해보자(그림 1).

먼저 카메라 축점의 위치를 임의의 외부좌표계에서 측정된 vector C로 나타낸다.

image plane은 image plane에 직각인 방향벡터 A에 의해 나타내어진다. focal center로부터 vector A를 따라서 측정된 길이를 축점거리 f라 한다.

image plane에 점들을 위치시키기 위하여 A에 수직인 두 좌표축 벡터 H와 V를 정의해야 한다. 이들은 각각 수평, 수직방향으로의 image에 대응한다. focal center C에 원점을 둔 직교좌표계를 정의하는데 세 방향 vector H와 V를 사용함으로써 pinhole camera에 대한 model을 완전히 얻어진다 (그림 2). image plane은 다음과 같이 묘사된다.

$$C - fA + uH + vV$$

여기에서 u와 v는 image plane에서 점들의 coordinates로 사용되는 스칼라 변수이다.

앞에서 정의된 camera model로부터 perspective transform을 구해보자. 먼저 수직성분을 생각해보자. 수직성분을 구하기 위한 camera geometry의 측면은 그림3과 같다.

먼저 camera focal center C로부터 target point P까지의 변위 vector D를 만들자. 이 vector를 각각의 단위 vector와 내적함으로써 A, V의 투영성분을 구할수 있다. 이들은 그림에서 볼수 있는 바와 같이 삼각형의 두 변이다. V가 image plane과 평행하므로 삼각형의 닮음을 이용하여 다음과 같이 말할 수 있다.

$$v = f \frac{\bar{D} \cdot \bar{V}}{\bar{D} \cdot \bar{A}} \quad \bar{D} = \bar{P} - \bar{C} \quad (2-1)$$

마찬가지로 camera model을 위에서 내려다본 그림으로부터 점 P의 image의 수평 coordinate에 대해서 풀수 있다.

$$u = f \frac{\bar{D} \cdot \bar{H}}{\bar{D} \cdot \bar{A}} \quad (2-2)$$

이 두 식으로써 카메라 변수 f, C, A, H, V를 안다고 가정하면 image plane에서의 점의 위치를 알 수 있다.

그러나 digitized picture에 대해서는 image plane으로부터 digitizer에 의해 제공된 image 표현으로의 부가적인 mapping을 고려해서 한다. 전형적으로 image digitizer는 image plane을 pixel로 분할한다. digitizer에 의해 제공된 row, column number는 digitized pixel 정보가 저장된 메모리 주소를 표시하는데 전형적으로 사용된다.

pixel 주소로부터 물리적인 image 좌표로의 실제적인 mapping은 네가지 양에 의해 결정된다.

; pixel address of element (i, j)

; sampling resolution

; pixel 당 거리 n(수평), m(수직)

따라서 mapping은 다음과 같다.

$$u = n(j - j_0) \quad v = m(i - i_0) \quad (2-3)$$

n, m이 같은 값 즉 pixel이 square가 되는 것은 매우 바람직하다. 식 (2-1), (2-2), (2-3)으로부터 i와 j는 다음과 같이 구하여진다.

$$i = \frac{f D V}{m D A} + i_0, \quad V = \frac{f}{m} V + i_0 A \quad (2-4)$$

$$j = \frac{f D H}{n D A} + j_0, \quad D = \frac{f}{m} D + j_0 A \quad (2-5)$$

제3장 Camera Calibration

2장에서 카메라의 모델링을 행하였다. 그러나 카메라에 대한 모델을 정의하는 벡터들 C, A, H, V를 구하는 문제에는 논의하지 않았다. 이들값들의 최적치를 구하는 것이 카메라 Calibration이다. 카메라 Calibration은 least square fit를 통해서 얻어진다.

3.1 Least Square Fit

$$Y(x) = \sum_{k=1}^N a_k X_k(x) \quad (3-1)$$

을 생각하여 보자.

여기에서 $X_1(x), \dots, X_N(x)$ 는 basis function이라 불리는 임의 고정된 x의 함수이다. 식 (3-1)에서 chi square 값은 다음 식과 같이 주어진다.

$$\chi^2 = \sum_{k=1}^M \left[\frac{y_i - \sum_{k=1}^N a_k X_k(x_i)}{\sigma_i} \right]^2 \quad (3-2)$$

: 데이터 측정어러로 알려지지 않았을 경우에는 1로 한다.

least square fit는 chi square값을 최소로 하는 변수들의 값을 구하는 것이다.

이 변수들에 대한 식을 유도하기 전에 먼저 몇가지 notion을 정하자.

A : M X N matrix.

$$A_{ij} = \frac{X_j(x_i)}{\sigma_i} \quad (3-3)$$

벡터 b : M차원 벡터

벡터 a : N차원 벡터

3.2 Singular Value Decomposition

SVD방법은 다음의 정리에 기반을 두고 있다.

: 어떤 임의의 M by N matrix A (단 M의 값이 N보다 크거나 같아야 한다)는 다음과 같이 쓰여 질수 있다.

$$\begin{bmatrix} A \end{bmatrix} = \begin{bmatrix} U \end{bmatrix} \begin{bmatrix} W_1 & & \\ & \dots & \\ & & W_N \end{bmatrix} \begin{bmatrix} V^T \end{bmatrix}$$

U : M x N column orthogonal matrix

W : N x N diagonal matrix with positive or zero elements

V : N x N orthogonal matrix

$$Aa = b \quad (3-4)$$

$$r = |Aa - b| \quad (3-5)$$

위의식에서 r을 최소화하는 a의 값은 식 (3-6)과 같이 주어진다.

$$a = V [\text{diag}(1/w_j)] (V b) \quad (3-6)$$

3.3 Camera Calibration

그림 4에 보인 것처럼 바둑판 형태를 보는 카메라를 고려하자. 기준 좌표계로서 고정좌표구조를 사용하여 바둑판 위에 표시된 점의 위치를 정확하고도 쉽게 지정할 수 있다.

먼저 각 점에 대하여 image 좌표 (im, jm)과 3차원 고정좌표계에서의 점의 위치 pm을 기록하자.

perspective transform에 의하여 다음의 식을 만족하게 된다.

$$im = \frac{D_m \cdot V}{D_m \cdot A}, \quad jm = \frac{D_m \cdot H}{D_m \cdot A} \quad (3-7)$$

$$D_m = P_m - C, \quad m = 1, 2, \dots, n \quad (3-8)$$

위의 관계로부터 최적의 관계를 갖게 하는 C, A, H, V를 구하도록 하자. 먼저 im에 대하여 생각해보자.

$$im = \frac{D_m \cdot V}{D_m \cdot A} \quad (3-9)$$

$$D_m = P_m - C, \quad m = 1, 2, \dots, n$$

괄호안의 항을 em이라 하자.

$$em = (P_m V - im P_m A - C V - im C A) \quad (3-10)$$

$$\text{여기에서 } C V = C V, \quad C A = C A$$

이 시스템은 V와 A에 대하여 각각 3개의 미지수, Ca, Cv 합쳐서 8개의 미지수를 갖는 n개의 선형방정식으로 볼 수 있다. 다음에 단위벡터로서 A의 한 좌표 Ax를 l이라 하자. 따라서 미지수의 개수는 7개로 줄어든다. n차원 벡터 E의 성분으로 각 항 em을 생각한다면 다음과 같이 다시 쓸 수 있다.

$$Q : n \times 7$$

$$E = Q W - B, \quad W : 7 \times 1 \quad (3-11)$$

$$B : n \times 1$$

slop에 대한 식은 다음과 같이 다시 쓰여진다.

$$S = E E = |Q W - B| \quad (3-12)$$

이 식으로부터 S를 최소화하는 7차원 벡터 W를 구하고자 하는 것이다.

여기에서 normal equation에 의한다면

$$Q E = 0, \quad \text{또는 } Q Q W = Q B \quad (3-13)$$

가 된다.

이제 C, A, H, V를 안다.

Ca, Ch, Cv의 값을 안 후에 카메라의 3차원 위치 C는 3 x 3 선형 시스템을 풀으로써 얻어진다.

$$C A = C a \quad (3-14)$$

$$C H = C h \quad (3-15)$$

$$C V = C v \quad (3-16)$$

따라서 카메라의 모델은 완전하다.

제4장 Range Finding

제5장 실험 및 결과고찰

거리측정에 관한 기술포는 크게 삼각법의 원리를 이용한 것과 time of flight를 이용한 것의 두가지가 있다.

5.1 소프트웨어 알고리즘

4.1 Triangulation

Triangulation의 원리는 그림 5에 나타낸 바와 같다.

$$r_1 = \frac{d \sin \theta_2}{\sin [180^\circ - (\theta_1 + \theta_2)]} \quad (4-1)$$

$$r_2 = \frac{d \sin \theta_1}{\sin [180^\circ - (\theta_1 + \theta_2)]} \quad (4-2)$$

(1) Passive Triangulation

passive triangulation은 stereo 또는 binocular vision으로 불리는 것으로 두대의 카메라를 이용한다.

(2) Active Triangulation

1) Spot Sensing

한개의 projector와 한대의 카메라를 이용한 방법

$$r = \frac{D \sin \theta_1}{\sin [180^\circ - (\theta_1 + \theta_2)]} \quad (4-3)$$

2) Light-Stripe Sensing

Spot Sensing이 한점에 대하여 거리를 계산하나 이 방법은 Light-Stripe를 이용하여 카메라가 선을 봄으로써 거리를 측정하는 방법이다.

3) 카메라 모션

한대의 카메라를 가지고 이동을 시키면서 거리를 측정하는 방법이다.

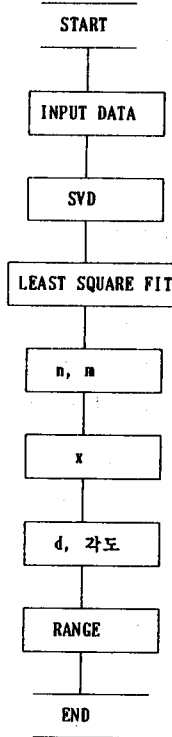
4.2 Time of flight

TOF방법은 어떤 신호가 물체까지 갔다가 오는 시간을 측정하므로써 거리를 아는 방법이다.

$$r = \frac{v \times t}{2} \quad (4-4)$$

v : 전달된 신호의 속도

t : 신호가 물체까지 갔다 오는 데 걸리는 시간



5.2 데이터 입력

- P1 = (-10, 15, 5), P2 = (-5, 15, 5)
- P3 = (0, 15, 5), P4 = (5, 15, 5)
- P5 = (10, 15, 5), P6 = (-10, 10, 5)
- P7 = (-5, 10, 5), P8 = (0, 10, 5)
- P9 = (5, 10, 5), P10 = (10, 10, 5)
- P11 = (-10, 5, 5), P12 = (-5, 5, 5)
- P13 = (0, 5, 5), P14 = (5, 5, 5)
- P15 = (10, 5, 5), P16 = (-10, 0, 5)
- P17 = (-5, 0, 5), P18 = (0, 0, 5)
- P19 = (5, 0, 5), P20 = (10, 0, 5)

5.3 Least Square Fit

(1) svdcmp

matrix A를 과 같은 형태로 나타냄

(2) svdksb

식 (3 - 6)로부터 a를 구해냄

(3) svdfit

구하고자 하는 변수의 값과 chi square값을 구함

5.4 n, m

$$u = \frac{n(j - j_0)}{f D V}, \quad v = \frac{m(i - i_0)}{f D H}$$

$$i = \frac{m D A}{n D A} + i_0, \quad j = \frac{n D A}{m D A} + j_0$$

위의 식들로부터 n, m을 구한다.

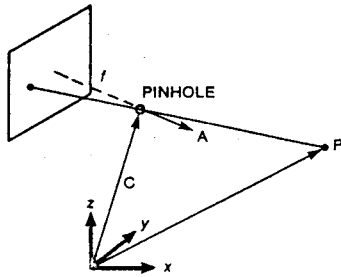


그림 1. Pinhole 카메라

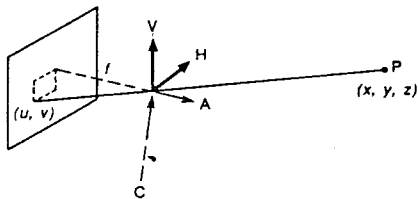


그림 2. Pinhole 카메라의 완전한 모델

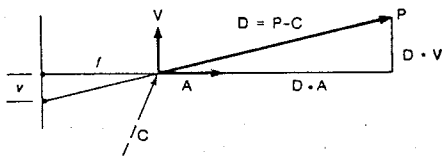


그림 3. 카메라 모델의 측면부

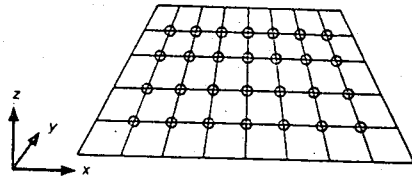


그림 4. Camera Calibration

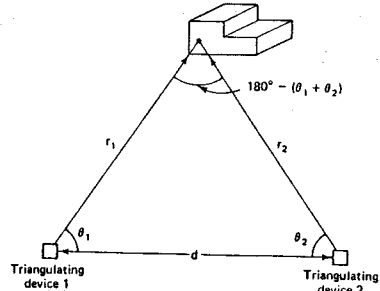


그림 5. 간단한 Triangulation

제6장 결론

이 논문은 카메라로부터 물체까지의 거리를 측정하는 소프트웨어를 설명한다. 본 논문에서는 least square fit를 수행하기 위하여 SVD 알고리즘을 이용했다. 이 알고리즘은 실행시간이 다른 알고리즘에 비하여 많이 소요되나 매우 큰 안정성과 정확도를 가지고 있다. 직접 실험에 의해 측정된 거리와 컴퓨터 프로그램으로부터 얻어진 결과 사이에는 약 3% 정도의 오차가 나왔다. 이것은 수작업의 오차를 고려할 때 상당히 좋은 결과라고 사려된다. 여기서 논의된 거리측정으로부터 range image를 구성할수 있으며 이로부터 아주 근접한 물체를 서로 구별할 수 있다.

제7장 참고문헌

- 1) Robotics age. In the beginning, Carl T. Helmers, Hayden, 1983
- 2) Extracting 3-D Measurements from Stereo TV cameras, Y. Yakimovsky and R. Cunningham, Computer Graphics and Image Processing. No. 7, 1978

- 3) Elementary Numerical Analysis, S.D. Conte, 1965
- 4) Robotics and AI, Andrew C. Staugaard, JR, 1987
- 5) Numerical Recipes in C, Cambridge, 1988
- 6) Matrix Computations, Gene H. Golub and Charles F. Van Loan, 1983, The Johns Hopkins University Press
- 7) "스테레오 비전을 이용한 3차원 물체의 구조분석", 박성일, 1988
- 8) "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision", Roger Y. Tsai, IEEE, 1986, pp 364-374