2D-2D Matching & Triangulation

Feb. 24, 2019

이재민

Epipolar Geometry - Epipolar constraint

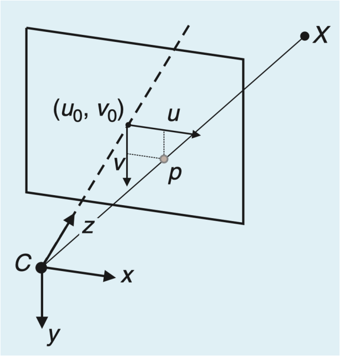


Fig1. Perspective Camera Model

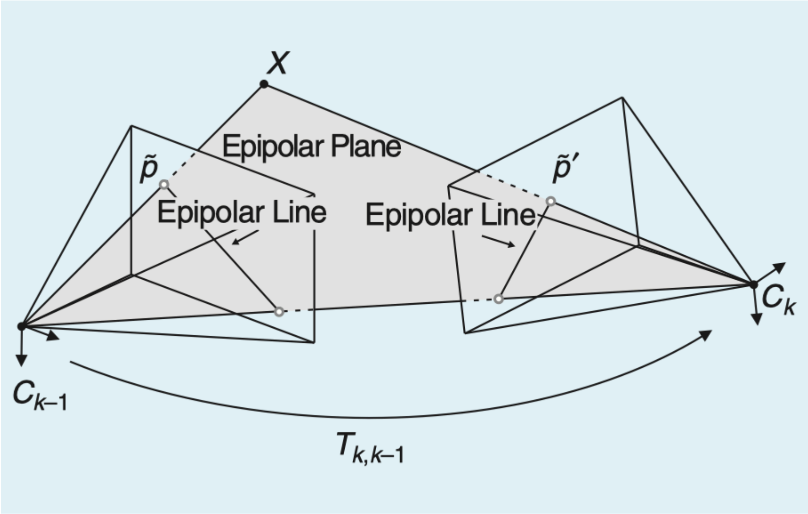


Fig2. Epipolar Geometry

- 표기
  - X : 실 공간 상 점의 위치
  - $\tilde{p}, \tilde{p}'$: 이미지 평면 투영점
  - C_{k-1}, C_k : 카메라 포즈
  - $T_{k,k-1}$: 두 카메라 간 변환

$$p = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad X = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$

- Perspective Camera Model

$$\lambda p = K \left[R | t \right] \begin{bmatrix} X \\ 1 \end{bmatrix}$$

- 2D-2D 자세 추정 방법 : 특징점 매칭 -> Essential Matrix 복원 -> Essential Matrix 분해

Epipolar Geometry - Epipolar constraint

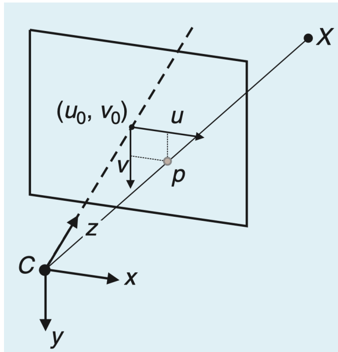


Fig1. Perspective Camera Model

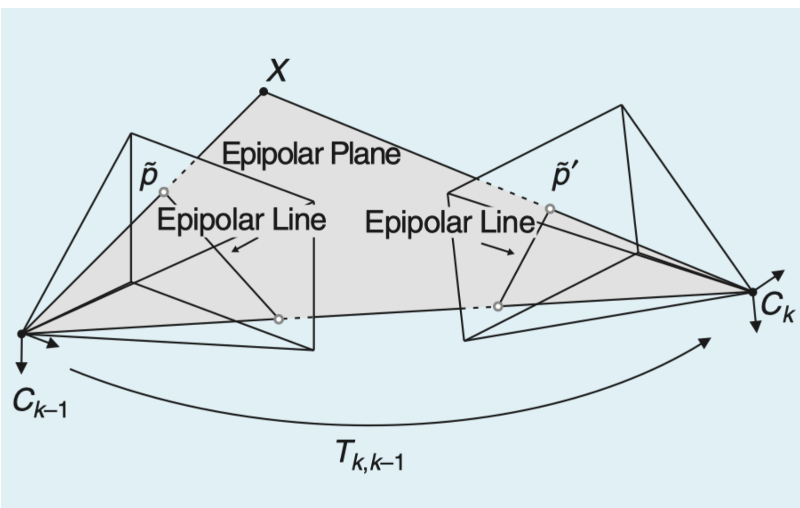


Fig2. Epipolar Geometry

- 유도
  - 두 이미지 투영점은 아래와 같이 계산 됨

    $$\lambda\tilde{p} = KX, \lambda'\tilde{p}' = K(RX + t)$$

  - 두 식을 결합하면 아래와 같음.

    $$K^{-1}\tilde{p}' = RK^{-1}\tilde{p} + t$$

  - 양변에 $t^\wedge$ 을 곱하고 정리

    $$t^\wedge K^{-1}\tilde{p}' = t^\wedge RK^{-1}\tilde{p}$$

    $$\tilde{p}'^T K^{-T} t^\wedge K^{-1}\tilde{p}' = \tilde{p}'^T K^{-T} t^\wedge RK^{-1}\tilde{p}$$

    $$\tilde{p}'^T K^{-T} t^\wedge RK^{-1}\tilde{p} = 0$$

  - $E = t^\wedge R, F = K^{-T}EK^{-1}$라고 하면

    $$\tilde{p}'^T F\tilde{p} = 0$$

- Note.

$$t = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}, t^\wedge = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & -t_x & 0 \end{bmatrix}$$

Epipolar Geometry - Essential Matrix

- 카메라 내부 파라미터를 정확히 알고있을 경우

$$\tilde{p'}^T E \tilde{p} = 0, \quad where\ E = t^\wedge R$$

  -> 행렬 분해(SVD, QR)로 두 카메라 간 카메라 자세 추정 가능

- Essential Matrix의 제약조건
  - 역행렬 존재하지 않음.

$$det(E) = 0$$

  - 5자유도 : 이동(3) + 회전(3) - 스케일(1)

$$rank(E) = 5$$

  - 3개의 특이값은 아래와 같은 형태로만 존재함.

$$\sigma_1 = \sigma_2, \sigma_3 = 0$$

  - 기타 대수적 제약조건

$$2EE^T E = tr(EE^T)E = 0$$

- Essential Matrix 복원: 8, 5-point algorithm 사용, 추가 정보를 RANSAC통해 활용함
  1-point algorithm도 존재(로봇 운동제약 적용)

Epipolar Geometry - Essential Matrix

- Essential Matrix 복원 (8-point algorithm by Longuet-Higgins)
  - Normalized image coordinate에서 하나의 매칭 쌍에 대해 Epipolar constraint성립

$$(u_1, v_1, 1)\begin{pmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{pmatrix}\begin{pmatrix} u_2 \\ v_2 \\ 1 \end{pmatrix} = 0.$$

  - 8개의 매칭쌍에 대해 아래와 같은 선형방정식 형태로 변환

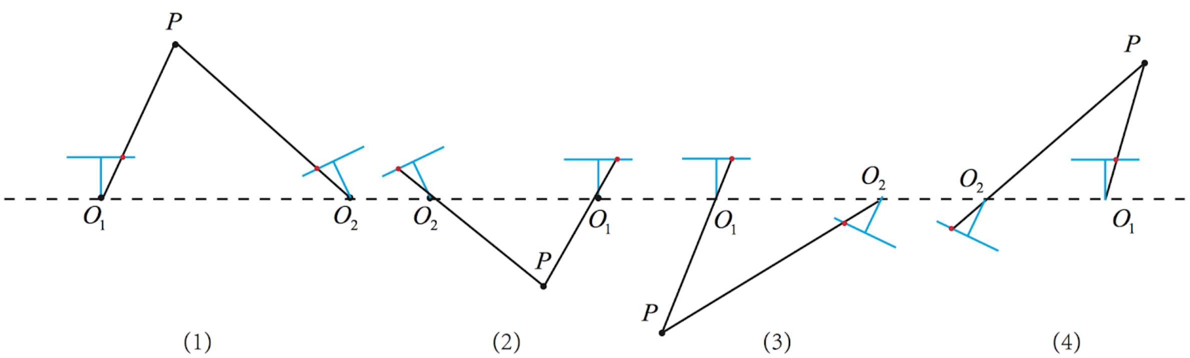
$$\begin{pmatrix} u_1^1 u_2^1 & u_1^1 v_2^1 & u_1^1 & v_1^1 u_2^1 & v_1^1 v_2^1 & v_1^1 & u_2^1 & v_2^1 & 1 \\ u_1^2 u_2^2 & u_1^2 v_2^2 & u_1^2 & v_1^2 u_2^2 & v_1^2 v_2^2 & v_1^2 & u_2^2 & v_2^2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ u_1^8 u_2^8 & u_1^8 v_2^8 & u_1^8 & v_1^8 u_2^8 & v_1^8 v_2^8 & v_1^8 & u_2^8 & v_2^8 & 1 \end{pmatrix}\begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \\ e_8 \\ e_9 \end{pmatrix} = 0.$$

  - 좌측의 Coefficient의 rank가 8이어야 함
  - E is degenerated for a planar scene.

Epipolar Geometry - Essential Matrix

- Essential Matrix 분해
  - 특이값 분해(SVD) 이용, $E = U\Sigma V^T$ 로 분해
  - $\Sigma = diag(\sigma_1, \sigma_2, \sigma_3)$ 일 때 $E = U\ diag(\frac{\sigma_1+\sigma_2}{2}, \frac{\sigma_1+\sigma_2}{2}, 0)\, V^T$ 로 조정
  - 4개의 가능한 해 존재

$$t_1^{\wedge} = UR_Z(\tfrac{\pi}{2})\Sigma U^T, \quad R_1 = UR_Z^T(\tfrac{\pi}{2})V^T$$
$$t_2^{\wedge} = UR_Z(-\tfrac{\pi}{2})\Sigma U^T, \quad R_2 = UR_Z^T(-\tfrac{\pi}{2})V^T$$



  - (1)의 해 만이 물리적으로 가능한 해로 양의 깊이를 가짐.

Triangularization

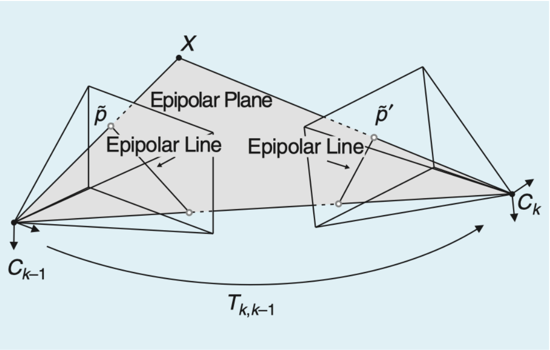
- Epipolar contraint에서 구한 R,t를 이용하여 스케일 추정
  - Epipolar contraint(스케일 고려)

$$\lambda' \tilde{p}' = \lambda R \tilde{p} + t$$

  - 양변에 $\tilde{p}'^{\wedge}$곱하여 식 정리

$$\lambda' \tilde{p}'^{\wedge} \tilde{p}' = \lambda \tilde{p}'^{\wedge} R \tilde{p} + \tilde{p}'^{\wedge} t$$

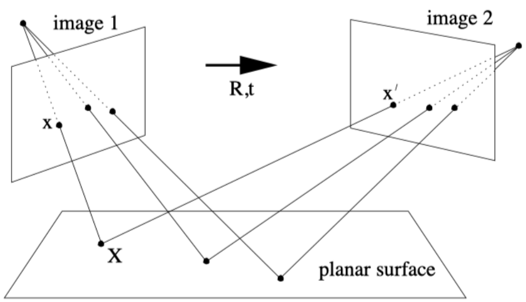
$$\lambda \tilde{p}'^{\wedge} R \tilde{p} + \tilde{p}'^{\wedge} t = 0$$



- 복수의 매칭쌍에 대해 최소제곱법 사용하여 최종 스케일 추정

Homography

- 호모그래피: 평면 간의 매핑을 표현하는 행렬



- Recall: 점 X가 지나는 평면의 방정식

$$-\frac{n^T X}{d} = 1$$

- Projective Camera model에서 유도

$$\begin{aligned} x' &= K(RX + t) \\ &= K(RX + t(-\frac{n^T X}{d})) \\ &= K(R - \frac{tn^T}{d})K^{-1}x \\ &= Hx \end{aligned}$$

- Direct Linear Transform 알고리즘
  - $\begin{pmatrix} u_2 \\ v_2 \\ 1 \end{pmatrix} = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix} \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix}$ 꼴로 표현할 때

$$\begin{pmatrix} u_1^1 & v_1^1 & 1 & 0 & 0 & 0 & -u_1^1 u_2^1 & -v_1^1 u_2^1 \\ 0 & 0 & 0 & u_1^1 & v_1^1 & 1 & -u_1^1 v_2^1 & -v_1^1 v_2^1 \\ u_1^2 & v_1^2 & 1 & 0 & 0 & 0 & -u_1^2 u_2^2 & -v_1^2 u_2^2 \\ 0 & 0 & 0 & u_1^2 & v_1^2 & 1 & -u_1^2 v_2^2 & -v_1^2 v_2^2 \\ u_1^3 & v_1^3 & 1 & 0 & 0 & 0 & -u_1^3 u_2^3 & -v_1^3 u_2^3 \\ 0 & 0 & 0 & u_1^3 & v_1^3 & 1 & -u_1^3 v_2^3 & -v_1^3 v_2^3 \\ u_1^4 & v_1^4 & 1 & 0 & 0 & 0 & -u_1^4 u_2^4 & -v_1^4 u_2^4 \\ 0 & 0 & 0 & u_1^4 & v_1^4 & 1 & -u_1^4 v_2^4 & -v_1^4 v_2^4 \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \end{pmatrix} = \begin{pmatrix} u_2^1 \\ v_2^1 \\ u_2^2 \\ v_2^2 \\ u_2^3 \\ v_2^3 \\ u_2^4 \\ v_2^4 \end{pmatrix}$$

  형태로 호모그래피를 구할 수 있음.
  (3개 이상의 점이 한 직선에 있을 경우 degenerate.)

  - rank(H)= 8이므로 4개 매칭쌍이 필요

- 자세 추정 방법 ($\boldsymbol{K}^{-1}\boldsymbol{H} = \lambda[\boldsymbol{r}_1\boldsymbol{r}_2\boldsymbol{t}]$ 활용)

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{P}\boldsymbol{X} \\ &= \boldsymbol{K}[\boldsymbol{r}_1\boldsymbol{r}_2\boldsymbol{r}_3\boldsymbol{t}] \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} \\ &= \boldsymbol{K}[\boldsymbol{r}_1\boldsymbol{r}_2\boldsymbol{t}] \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \end{aligned}$$

$$\boldsymbol{P} = \boldsymbol{K}\begin{bmatrix} \boldsymbol{r}_1 & \boldsymbol{r}_2 & (\boldsymbol{r}_1 \times \boldsymbol{r}_2) & \boldsymbol{t} \end{bmatrix}$$

Q&A

Thank you