



Reconstruction 3D

École Nationale Supérieure
de **Techniques Avancées**

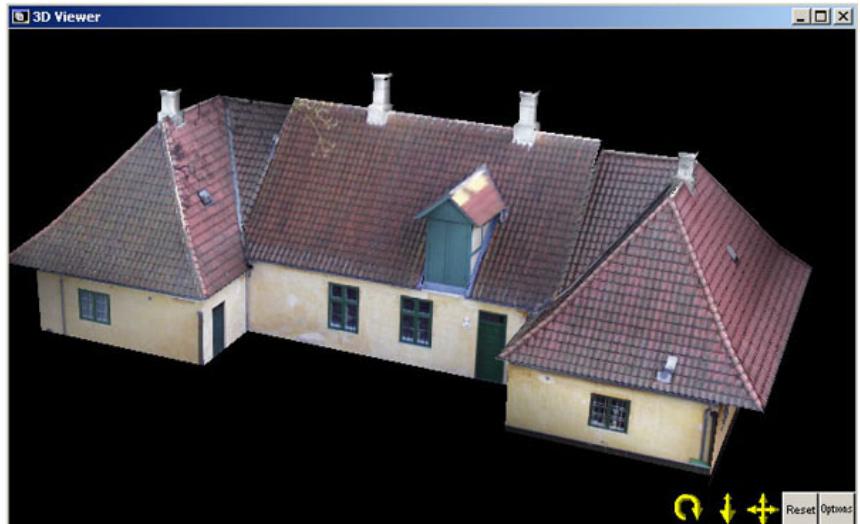
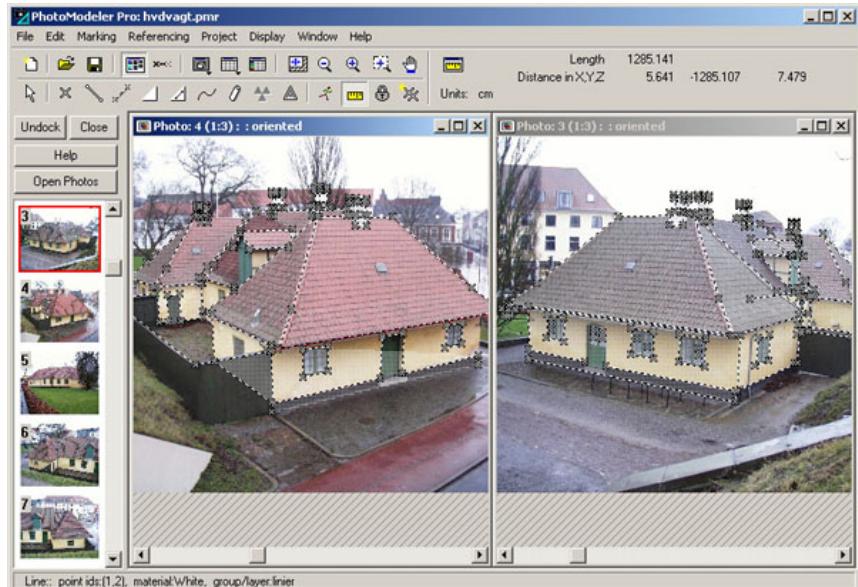
Cours ENSTA Paris ROB313 – 2019/2020
David Filliat
david.filliat@ensta-paris.fr



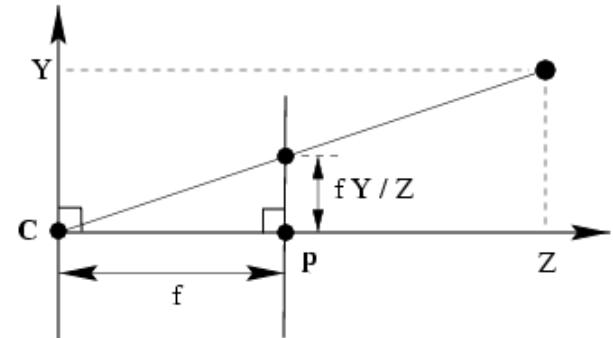
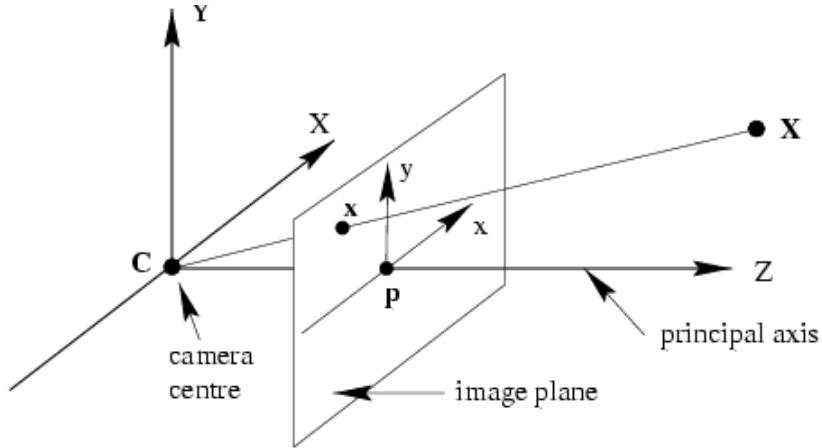
Objectifs

Reconstruction 3D

- Estimation de la structure 3D d'une scène
- A partir de $N \geq 2$ images de la scène



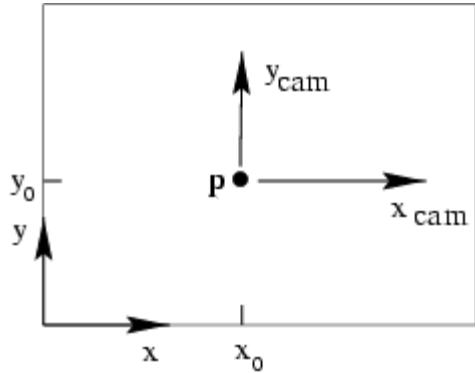
Modèle de caméra sténopé (pinhole) – paramètres intrinsèques



$$(X, Y, Z)^T \mapsto (fX/Z, fY/Z)^T$$

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & & & \\ & f & & \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Modèle de caméra sténopé (pinhole) – paramètres intrinsèques



$$(X, Y, Z)^T \mapsto (fX/Z + p_x, fY/Z + p_y)^T$$

$(p_x, p_y)^T$ principal point

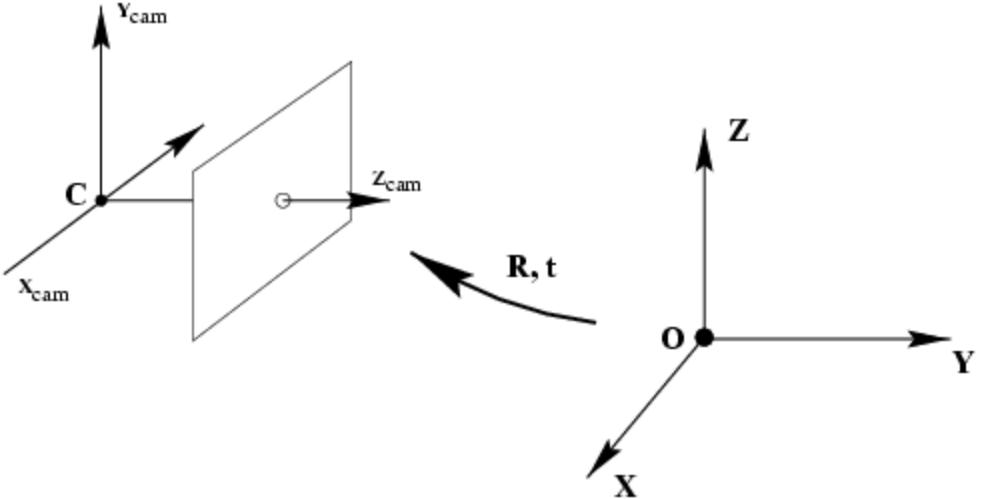
$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{pmatrix} = \begin{bmatrix} f & p_x & 0 \\ f & p_y & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$x = K[I|0]X_{\text{cam}}$$

$$K = \begin{bmatrix} f & p_x \\ f & p_y \\ 1 & 1 \end{bmatrix}$$

Matrice de calibrage (intrinsèque)

Modèle de caméra sténopé (pinhole) – paramètres extrinsèques



$$\tilde{X}_{\text{cam}} = R(\tilde{X} - \tilde{C})$$

$$X_{\text{cam}} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \underbrace{\begin{bmatrix} R & -RC \\ 0 & 1 \end{bmatrix}}_{\text{Paramètres extinsèques}} X$$

$$x = K[I | 0]X_{\text{cam}}$$

$$x = KR[I | -\tilde{C}]X$$

$$x = PX$$

$$P = K[R | t]$$

$$t = -R\tilde{C}$$

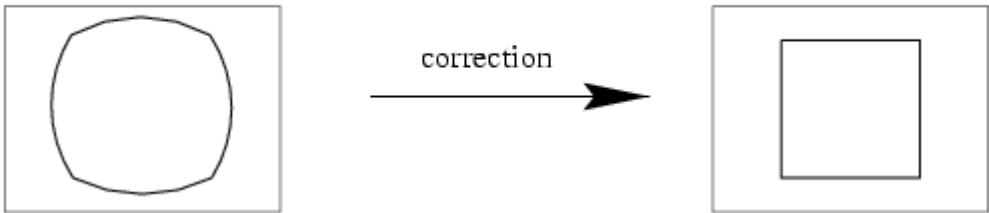
Modèle de caméra sténopé (pinhole) – distortion



radial distortion



linear image



Correction de la distortion : $\hat{x} = x_c + L(r)(x - x_c)$ $\hat{y} = y_c + L(r)(y - y_c)$

Choix de la fonction et du centre de distorsion :

$$x = x_o + (x_o - c_x)(K_1 r^2 + K_2 r^4 + \dots)$$

$$y = y_o + (y_o - c_y)(K_1 r^2 + K_2 r^4 + \dots)$$

$$r = (x_o - c_x)^2 + (y_o - c_y)^2 .$$

Distortion radiale

$$(\tilde{x}, \tilde{y}, 1)^\top = [\mathbf{I} \mid \mathbf{0}] \mathbf{X}_{\text{cam}}$$

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} = L(\tilde{r}) \begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix}$$

Géométrie de 2 vues : problèmes

Correspondances

- Etant donné un point X dans la première image, comment constraint-il la position du point correspondant x' dans la seconde ?

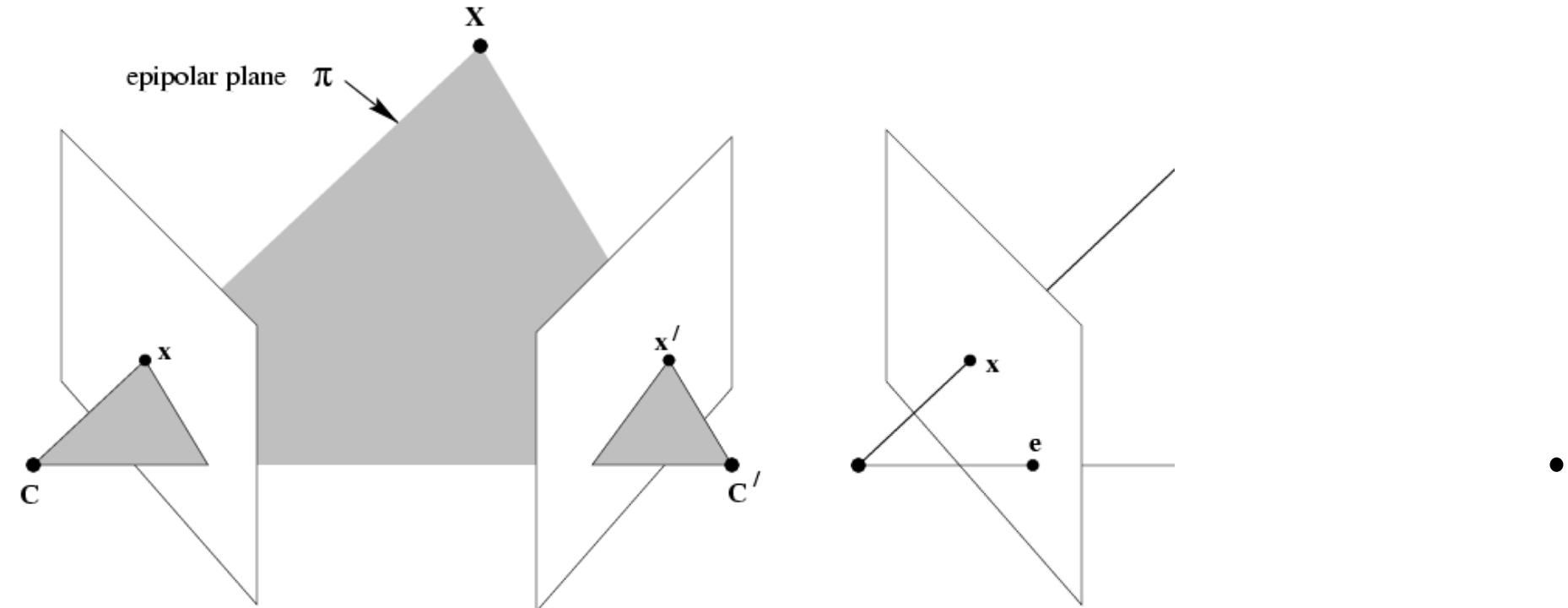
Géométrie des camera (mouvement)

- Etant donné un ensemble de points dans une image et leurs correspondants dans la seconde $\{x_i \leftrightarrow x'_i\}$, $i=1,\dots,n$, quelles sont les caméras P et P' ?

Géométrie de la scène (structure)

- Etant donné un ensemble de points dans une image et leurs correspondants dans la seconde $\{x_i \leftrightarrow x'_i\}$, $i=1,\dots,n$, et les caméras P et P' quelle est la position des points X dans l'espace ?

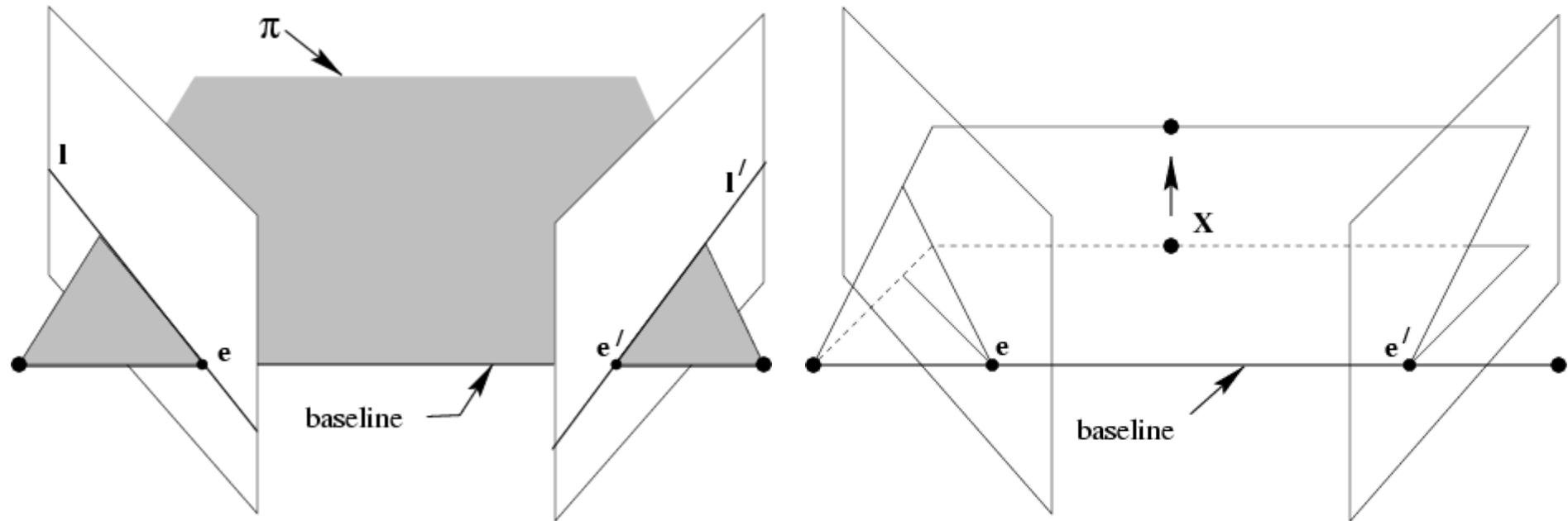
Géométrie épipolaire



C, C', x, x' et X sont coplanaires

Que dire de x' si seuls C, C', x sont connus ?

Géométrie épipolaire

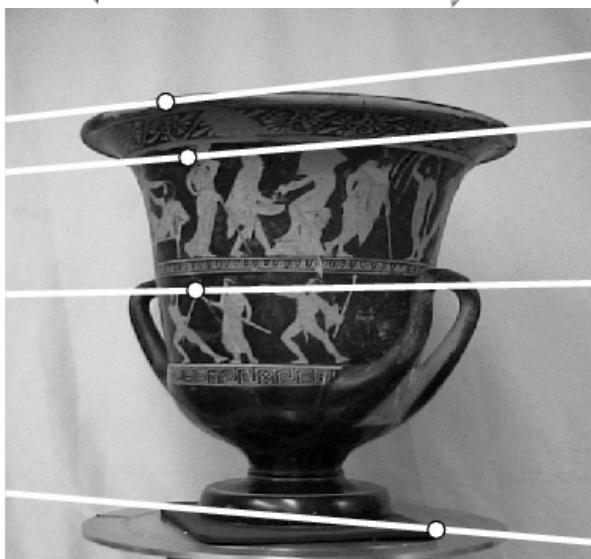
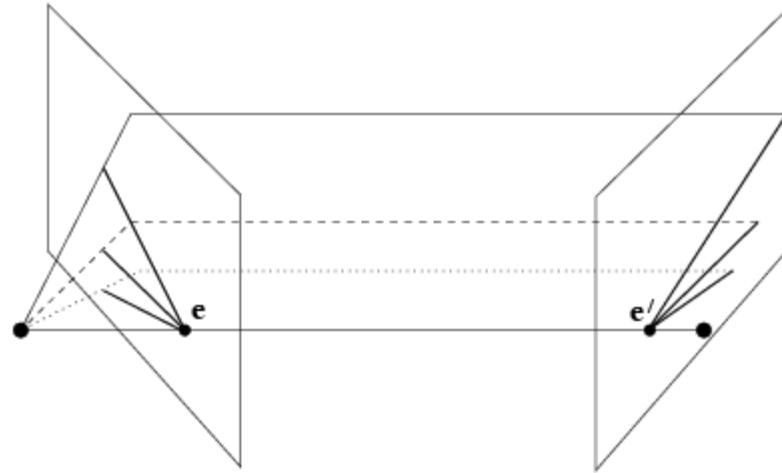


Tous les points sur π se projettent
sur l et l'

Famille de plans π et lignes l et l'
Se coupent en e et e'

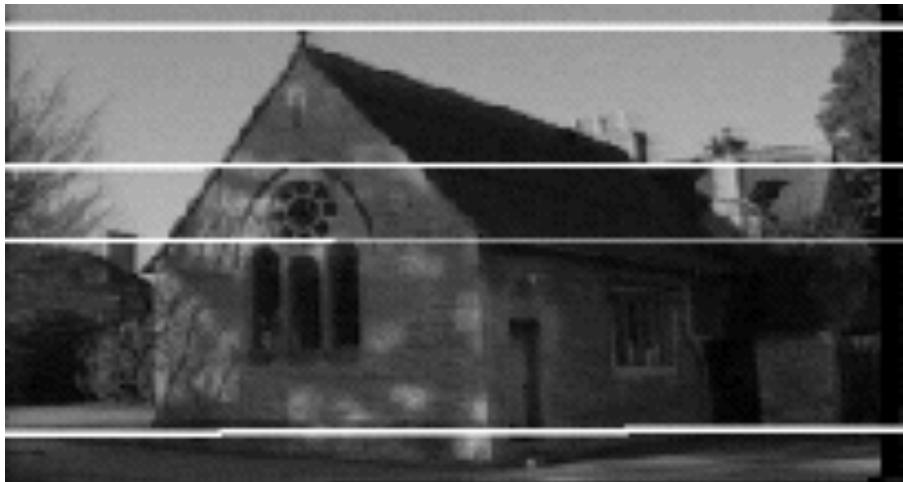
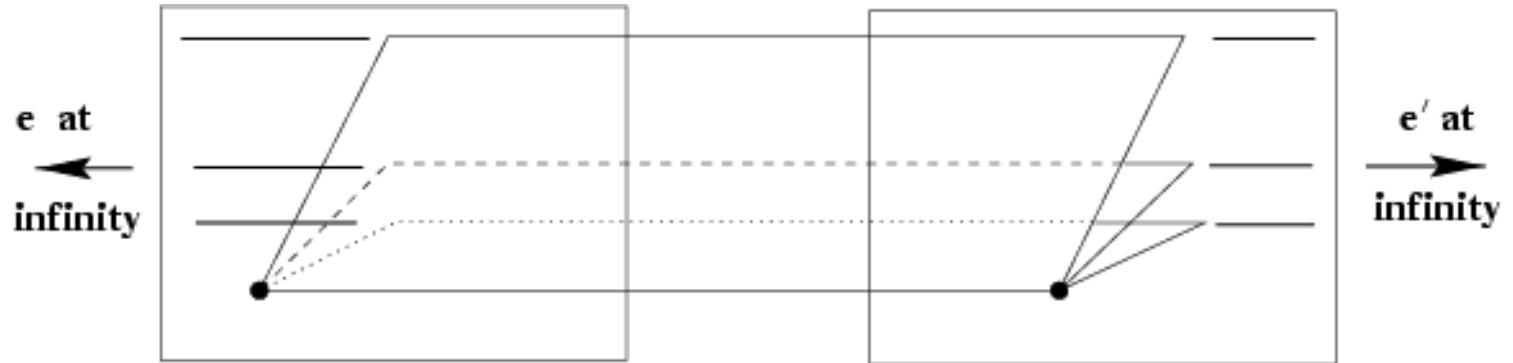
Géométrie épipolaire

Ex: Caméras convergentes



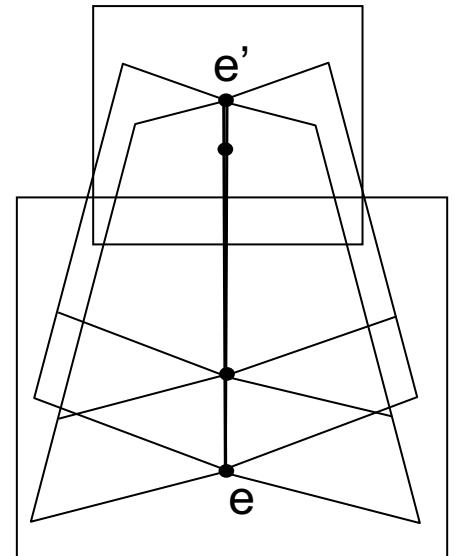
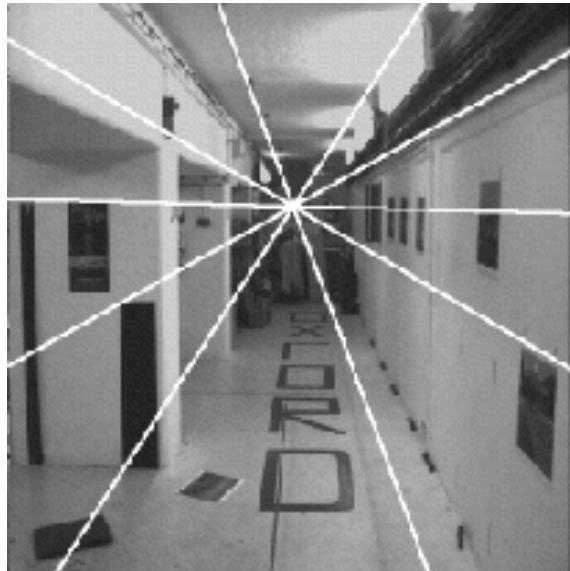
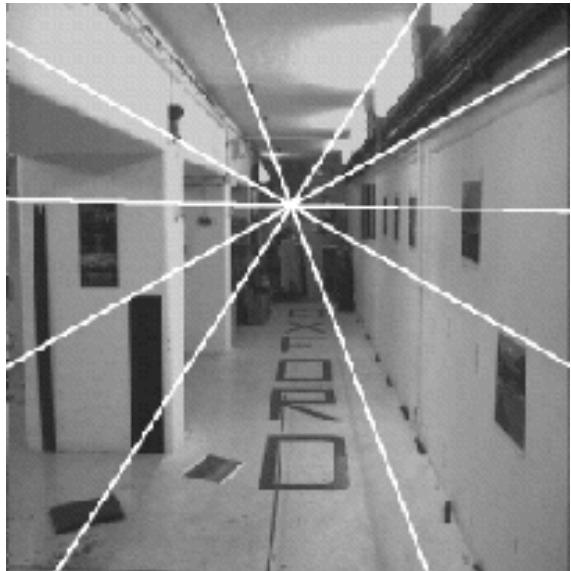
Géométrie épipolaire

Ex: Caméra en translation dans le plan image



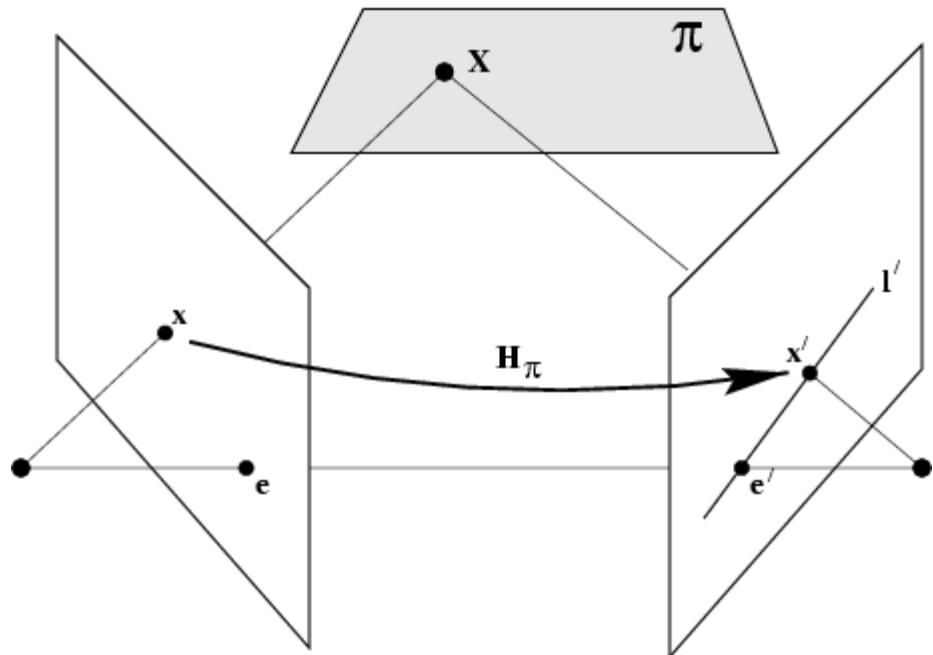
Géométrie épipolaire

Ex: Caméra avançant



Matrice fondamentale F

Représentation algébrique de la géométrie épipolaire



$$x \mapsto l'$$

$$x' = H_\pi x$$

$$l' = e' \times x'$$

$$= [e']_x H_\pi x = Fx$$

$$l' = Fx$$

$$x'^T l' = 0$$

$$x'^T Fx = 0$$

Matrice Fondamentale F

Calcul à partir des matrices des caméras

$$x = Px$$

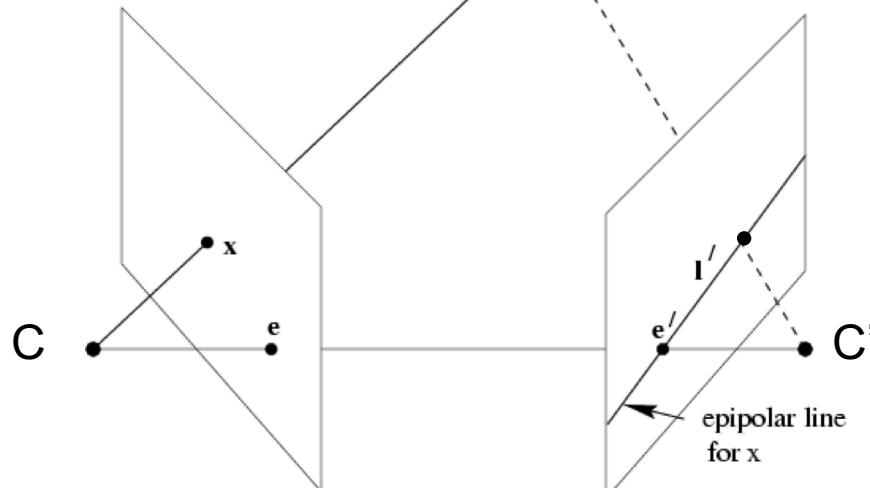
$$l' = e' \times x'$$

$$l' = [e']_x P' P^+ x$$

$$l' = F x$$

$$\Rightarrow F = [e']_x P' P^+$$

$$(P^+ P = I)$$



(rq: si \$C=C' \Rightarrow F=0\$)

Matrice fondamentale F

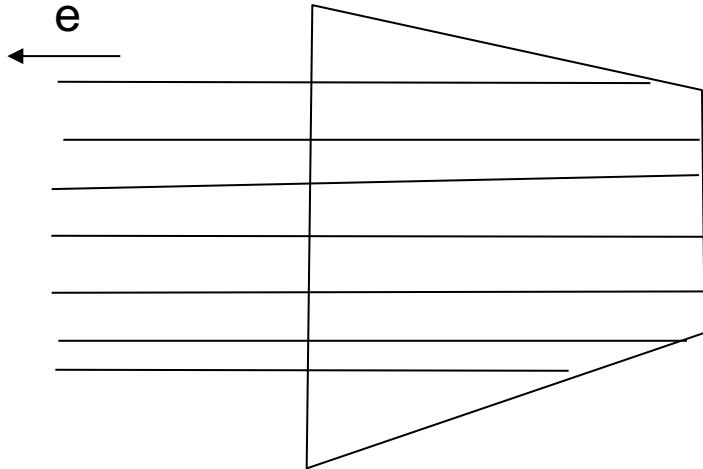
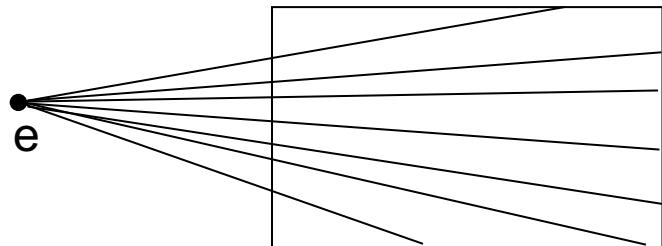
F est l'unique matrice 3×3 de rang 2
 qui vérifie $x'^T F x = 0$ pour tout $x \leftrightarrow x'$

- **Transposée:** si F est la fondamentale de (P,P'), alors F^T est la fondamentale pour (P',P)
- **Lignes épipolaires :** $l' = Fx$ & $l = F^T x'$
- **Eipoles:** sur toutes les lignes épipolaires, donc
 $e'^T F x = 0, \forall x \Rightarrow e'^T F = 0$, de même $F e = 0$
- **F** a 7 ddl. , i.e. $3 \times 3 - 1(\text{homogene}) - 1(\text{rang}2)$
- **F** est une projection, transformant un point x en une ligne $l' = Fx$ (non inversible)

Rectification stéréo

Simplification de la mise en correspondance

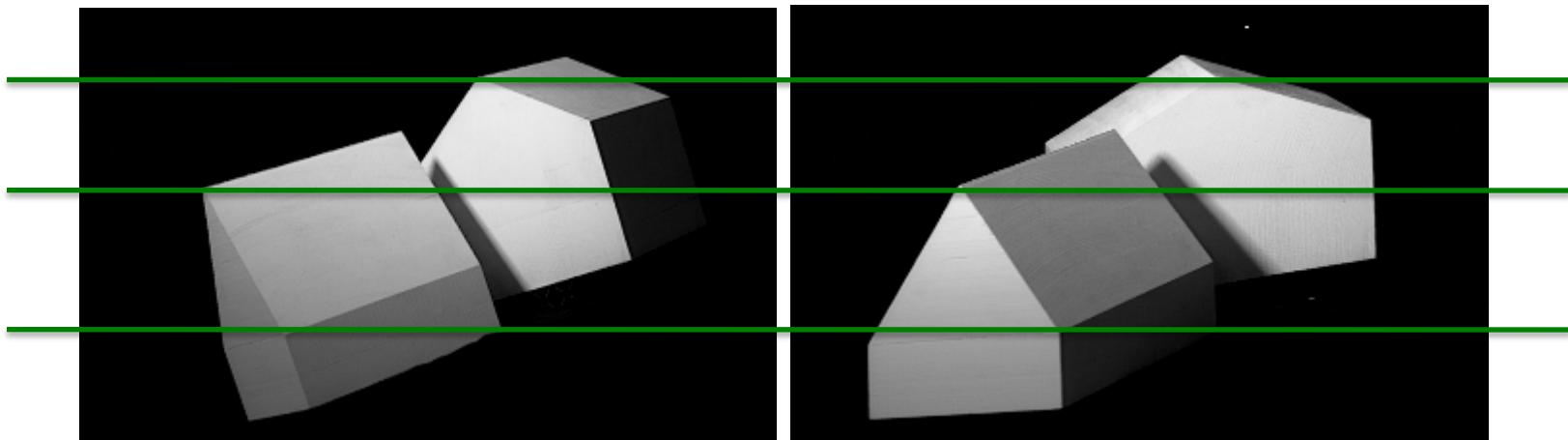
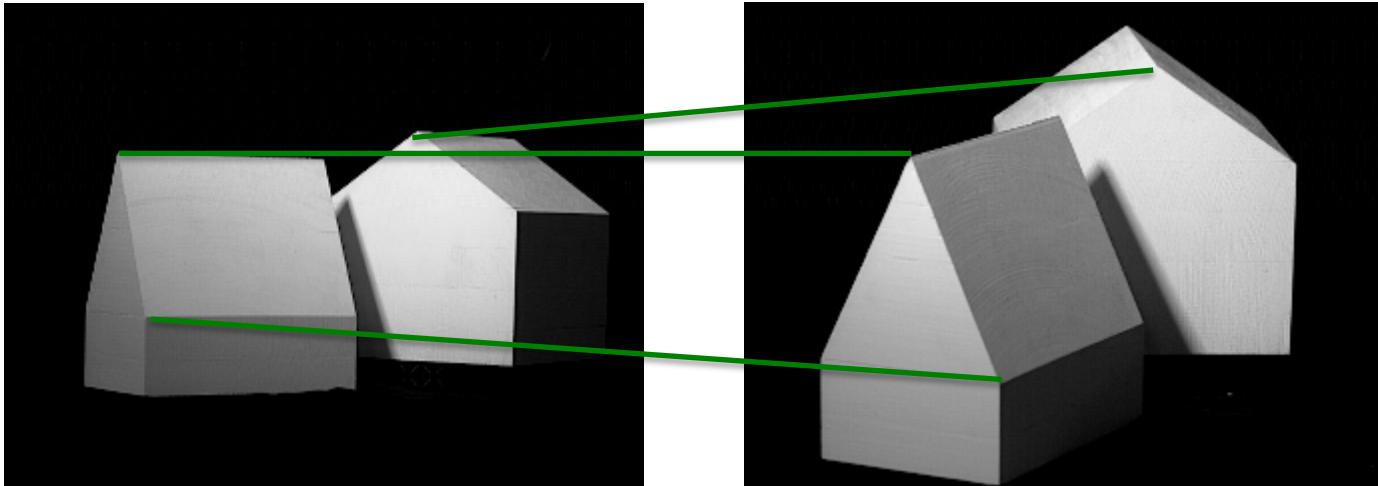
- Pour des images stéréo, en déformant les images
- Trouver homographie tq épipoles soient horizontales



$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = H\mathbf{e}$$

- Transférer épipole en $(1,0;0)$
- (épipoles définies par $\mathbf{e}'^T F = 0$, et $F\mathbf{e} = 0$)
- (problèmes quand épipole proche ou dans l'image)

Rectification stéréo



Rectification stéréo

Pipeline stéréo

- Rectification des deux images
- Calcul de la disparité (décalage en pixels entre images)
- Profondeur = 1/Disparité

image $I(x,y)$



Disparity map $D(x,y)$

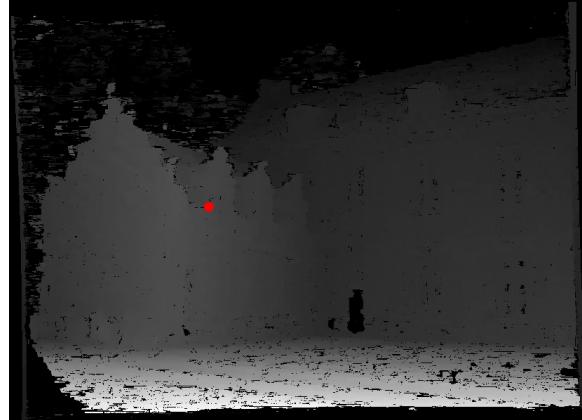


image $I'(x',y')$



$$(x', y') = (x + D(x, y), y)$$

Transformation projectives et invariances de F

Transformation projective de l'image

$$\hat{x} = Hx, \hat{x}' = H'x' \Rightarrow \hat{F} = H'^{-T} FH^{-1}$$

F invariante aux transformations projectives de l'espace

$$x = Px = (PH)(H^{-1}X) = \hat{P}\hat{X}$$

$$x' = P'X = (P'H)(H^{-1}X) = \hat{P}'\hat{X}$$

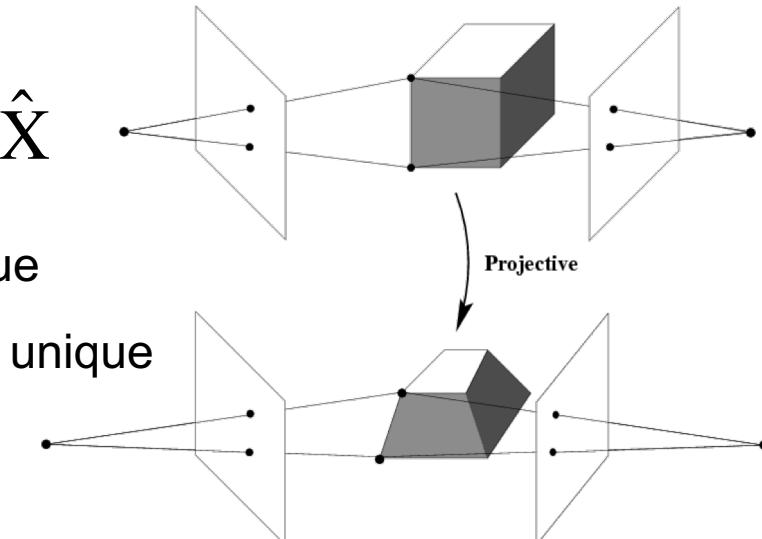
$$(P, P') \mapsto F \quad \text{unique}$$

$$F \mapsto (P, P') \quad \text{non unique}$$

- Choix d'une caméra canonique

$$\begin{aligned} P &= [I \mid 0] \\ P' &= [M \mid m] \end{aligned}$$

$$F = [e']_x P' P^+ = [m]_x M$$



Cameras canoniques à partir de F

Théorème : F correspond à P, P' ssi $P'^T F P$ est anti-symétrique

$$(X^T P'^T F P X = 0, \forall X)$$

Et donc, si F est une matrice fond., S matrice anti-symétrique

$$P = [I | 0] \quad P' = [SF | e'] \quad \text{Correspondent à } F = \text{mat. Fond.}$$

$$\left([SF | e']^T F [I | 0] = \begin{bmatrix} F^T S^T F & 0 \\ e'^T F & 0 \end{bmatrix} = \begin{bmatrix} F^T S^T F & 0 \\ 0 & 0 \end{bmatrix} \right)$$

Choix possible pour S :

$$P = [I | 0] \quad P' = [[e']]_x F | e']$$

Représentation canonique (cf Hartley/Zisserman pour démo):

$$P = [I | 0] \quad P' = [[e']]_x F + e' v^T | \lambda e'] \quad \forall \lambda, v$$

Rq : Matrice essentielle E

~Matrice fondamentale pour caméra calibrée (sans K)

- Caméra: $P' = K'[R \mid t]$
- Caméra normalisées : $\hat{P} = [I \mid 0] \quad \hat{P}' = [R \mid t]$

$$\begin{aligned}\hat{x} &= \hat{P} X \\ \hat{x}'^T E \hat{x} &= 0 \\ E &= K'^T F K\end{aligned}\quad \left(\hat{x} = K^{-1}x; \hat{x}' = K^{-1}x' \right)$$

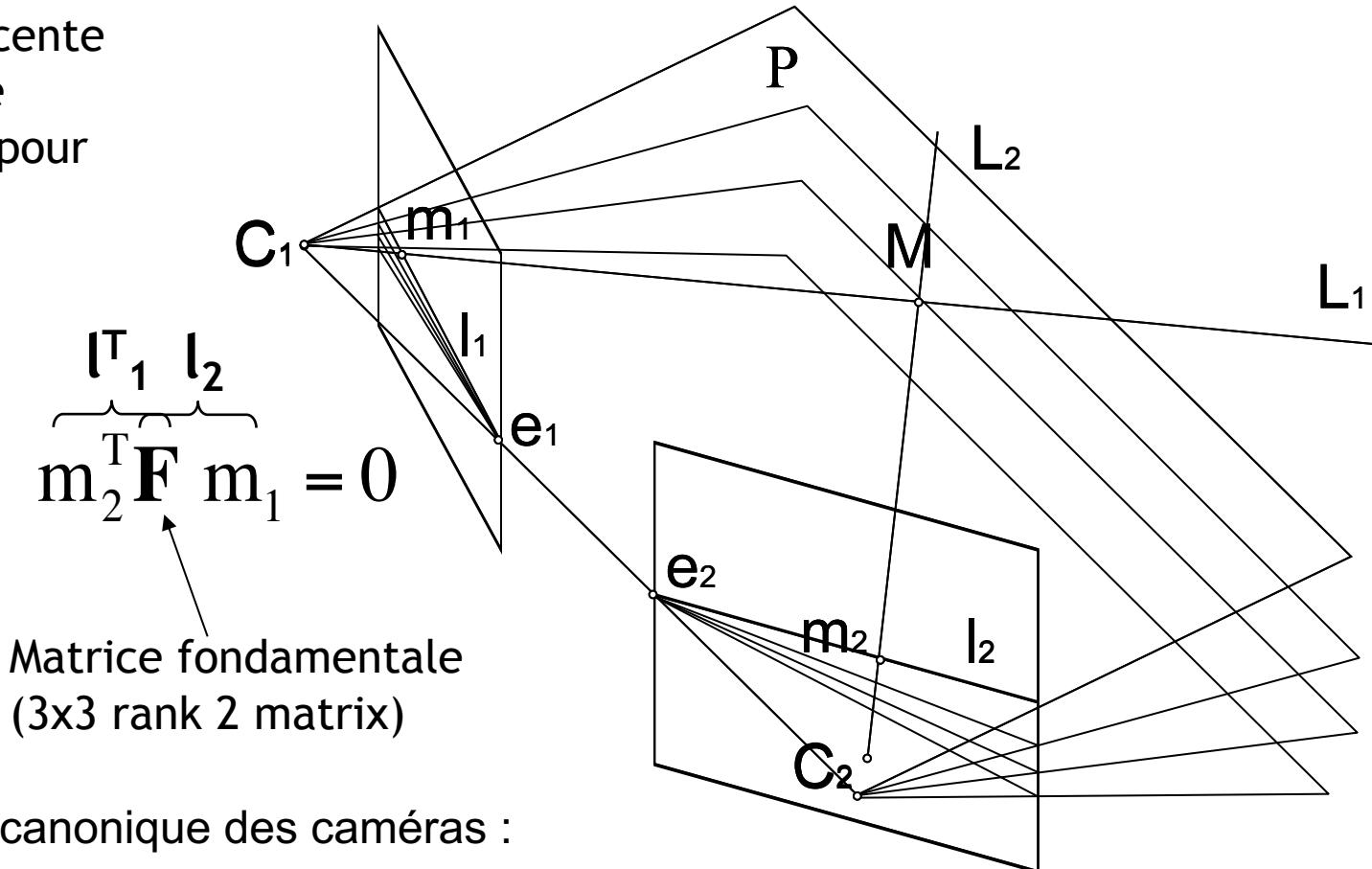
5 ddl (3 pour R; 2 pour t à l'échelle près)

E est essentielle si et seulement si ses deux valeurs singulières sont égales (et troisième=0)

$$E = U \text{diag}(1,1,0) V^T$$

Géométrie épipolaire - résumé

Structure sous-jacente
d'un ensemble de
correspondances pour
une scène rigide



Representation canonique des caméras :

$$P = [I \mid 0] \quad P' = [[e']^T F + e' v^T \mid \lambda e']$$

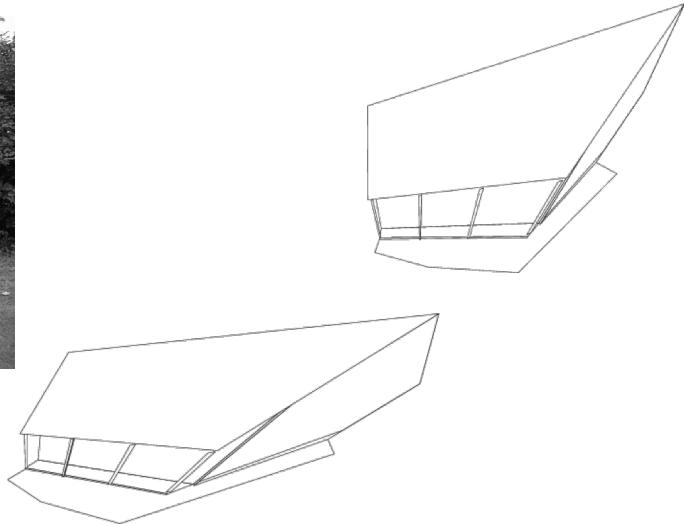
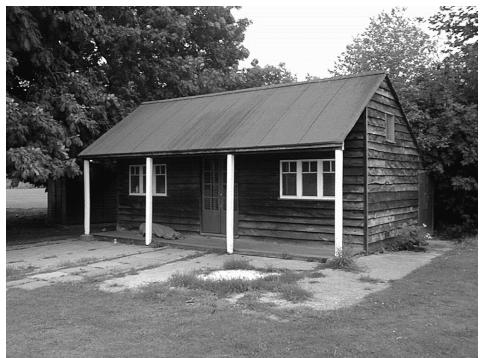
Reconstruction 3D

Problème de reconstruction:

- Étant donné $x_i \leftrightarrow x'_i$, calculer P, P' et X_i

$$x_i = PX_i \quad x'_i = P'X_i \quad \text{Pour tout } i$$

Possible à une transformation projective près sans autre information (sans calibrage, ni info sur la scène)



Reconstruction 3D

Méthode

- (i) Calculer F à partir des correspondances
- (ii) Calculer les matrices des cameras à partir de F
- (iii) Calculer les points 3D pour chaque paire de points

(i) Calcul de F

- utiliser $x'_i F x_i = 0$, linéaire en les coeff de F
- Résolution linéaire (7 pt), 8pt+ (moindres carrés)

(ii) Calcul des cameras à partir de F

$$P = [I \mid 0] \quad P' = [[e']]_x F + e' v^T \mid \lambda e'$$

(iii) Points 3D par triangulation

- Calcul de l'intersection de 2 rayons

(i) Calcul de F

$$\mathbf{x}'^T \mathbf{F} \mathbf{x} = 0$$

$$x' x f_{11} + x' y f_{12} + x' f_{13} + y' x f_{21} + y' y f_{22} + y' f_{23} + x f_{31} + y f_{32} + f_{33} = 0$$

Séparation des inconnues

$$[x' x, x' y, x', y' x, y' y, y', x, y, 1] [f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33}]^T = 0$$

(données)

(inconnues)

(linéaire)

$$\begin{bmatrix} x'_1 x_1 & x'_1 y_1 & x'_1 & y'_1 x_1 & y'_1 y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots \\ x'_n x_n & x'_n y_n & x'_n & y'_n x_n & y'_n y_n & y'_n & x_n & y_n & 1 \end{bmatrix} \mathbf{f} = 0$$

$$\mathbf{A}\mathbf{f} = 0 \quad \Rightarrow \mathbf{SVD}$$

(i) Calcul de F – contraintes de singularité

Pour que F soit bien la matrice fondamentale, reste à imposer :

$$e'^T F = 0 \quad Fe = 0 \quad \det F = 0 \quad \text{rank } F = 2$$

SVD pour matrice F calculée linéairement (rang 3)

$$F = U \begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \sigma_3 \end{bmatrix} V^T = U_1 \sigma_1 V_1^T + U_2 \sigma_2 V_2^T + U_3 \sigma_3 V_3^T$$

Calcul de la meilleure approximation de rang 2 $\min \|F - F'\|_F$

$$F' = U \begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & 0 \end{bmatrix} V^T = U_1 \sigma_1 V_1^T + U_2 \sigma_2 V_2^T$$

(i) Calcul de F – contraintes de singularité



Rang 3



Rang 2

Cas d'une caméra en translation

(i) Calcul de F > 8 points

$$\begin{bmatrix} x_1 x_1' & y_1 x_1' & x_1' & x_1 y_1' & y_1 y_1' & y_1' & x_1 & y_1 & 1 \\ x_2 x_2' & y_2 x_2' & x_2' & x_2 y_2' & y_2 y_2' & y_2' & x_2 & y_2 & 1 \\ \vdots & \vdots \\ x_n x_n' & y_n x_n' & x_n' & x_n y_n' & y_n y_n' & y_n' & x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = 0$$

SVD ou utiliser Levenberg-Marquardt pour minimiser $\|Af\|^2$

(i) Calcul de F > 8 points

$$\begin{bmatrix} x_1x_1' & y_1x_1' & x_1' & x_1y_1' & y_1y_1' & y_1' & x_1 & y_1 & 1 \\ x_2x_2' & y_2x_2' & x_2' & x_2y_2' & y_2y_2' & y_2' & x_2 & y_2 & 1 \\ \vdots & \vdots \\ x_nx_n' & y_nx_n' & x_n' & x_ny_n' & y_ny_n' & y_n' & x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = 0$$

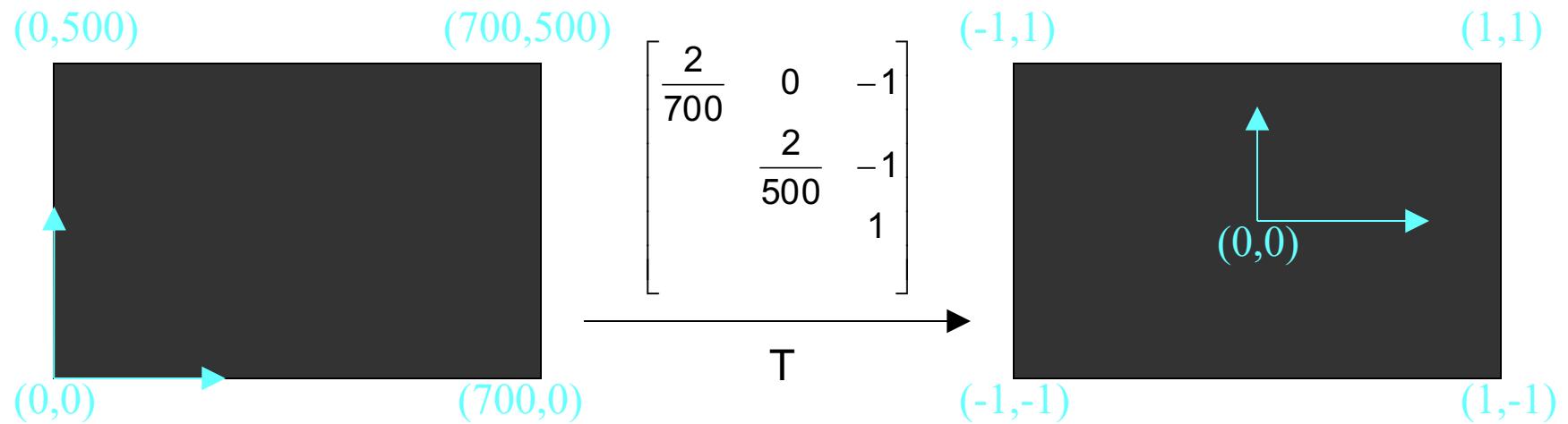
~10000 ~10000 ~100 ~10000 ~10000 ~100 ~100 ~100 1

Ordres de grandeur différents pour les colonnes
 → Mauvais résultat des moindres carrés



(i) Calcul de F > 8 points

Transform image to $\sim[-1,1] \times [-1,1]$



Bon résultats des moindres carrés (Hartley, PAMI '97)

$$\text{Puis } F = T^T F^T T$$

NOMBREUSES AUTRES MÉTHODES DE CALCUL (cf Hartley and Zisserman)

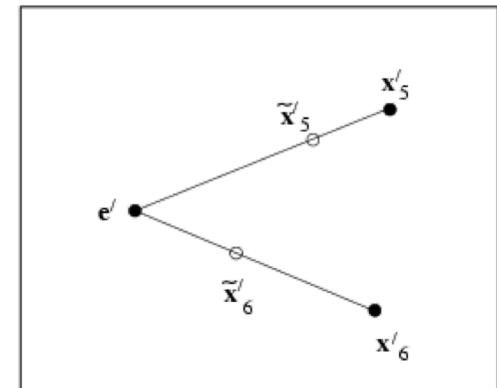
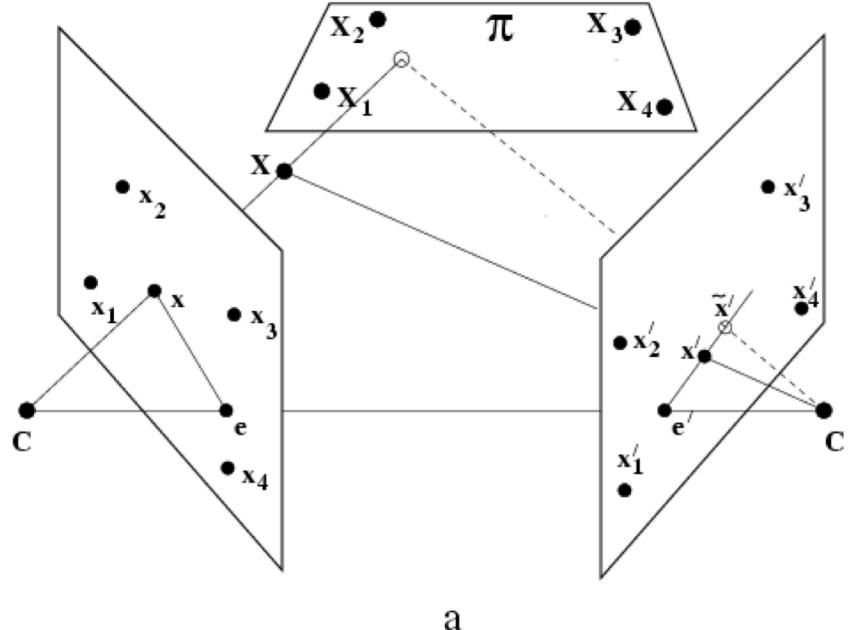
(i) Calcul de F – 6 points

Cas particulier avec seulement 6 points

- x_1, x_2, x_3, x_4 dans un plan, x_5, x_6 hors du plan
- Calcul de H à partir de x_1, x_2, x_3, x_4

$$e' = (x_5' \times H x_5) \times (x_6' \times H x_6)$$

$$F = [e']_x H$$



Reconstruction 3D

Méthode

- (i) Calculer F à partir des correspondances
- (ii) Calculer les matrices des cameras à partir de F
- (iii) Calculer les points 3D pour chaque paire de points

(i) Calcul de F

- utiliser $x'_i F x_i = 0$, linéaire en les coeff de F
- Résolution linéaire (7 pt), 8pt+ (moindres carrés)

(ii) Calcul des cameras à partir de F

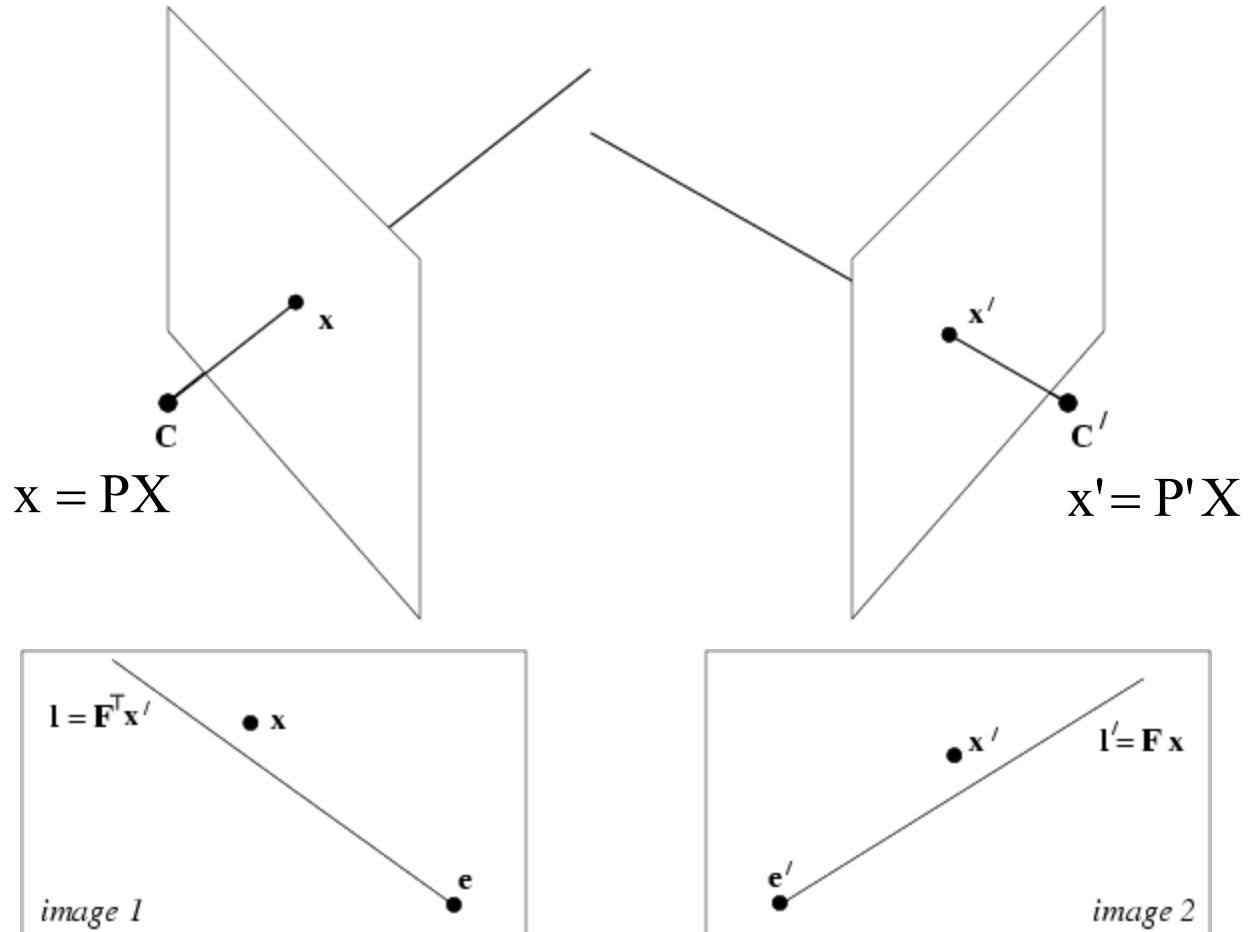
$$P = [I \mid 0] \quad P' = [[e']]_x F + e' v^T \mid \lambda e'$$

(iii) Points 3D par triangulation

- Calcul de l'intersection de 2 rayons

(iii) Reconstruction des points

Bruit de mesure



(iii) Reconstruction des points

Méthode linéaire

- Pour une image

$$x \times P X = 0$$

$$x(p^{3T}X) - (p^{1T}X) = 0$$

$$y(p^{3T}X) - (p^{2T}X) = 0$$

$$x(p^{2T}X) - y(p^{1T}X) = 0$$

Seulement 2 eq indépendantes

- Pour les deux images

$$AX = 0$$

$$A = \begin{bmatrix} xp^{3T} - p^{1T} \\ yp^{3T} - p^{2T} \\ x'p'^{3T} - p'^{1T} \\ y'p'^{3T} - p'^{2T} \end{bmatrix}$$

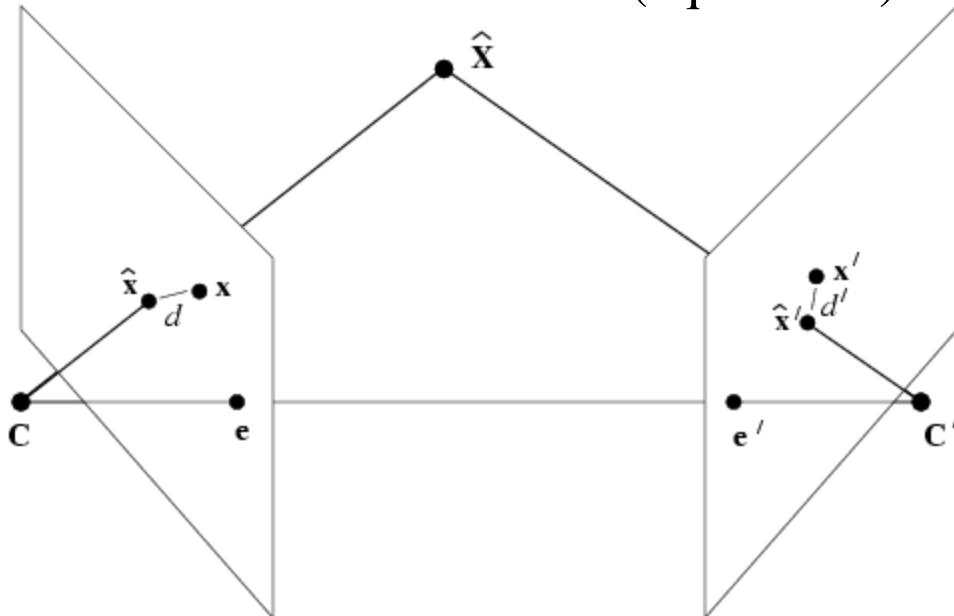
Résolution linéaire
(SVD par ex)

(iii) Reconstruction des points

Méthode non linéaire

– Minimiser $d(x, \hat{x})^2 + d(x', \hat{x}')^2$ avec $\hat{x}'^T F \hat{x} = 0$

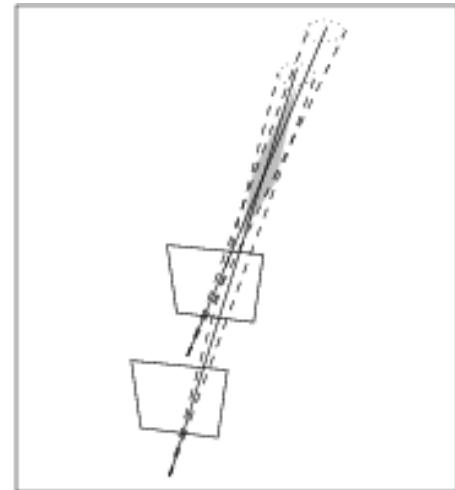
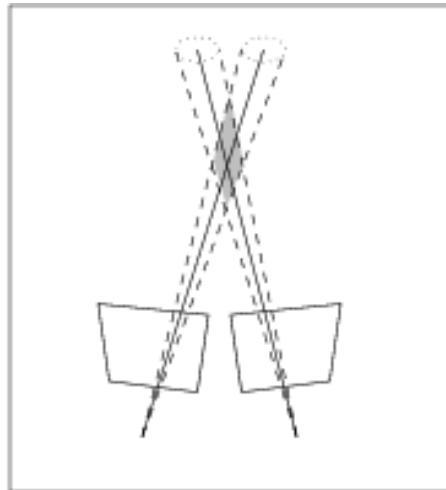
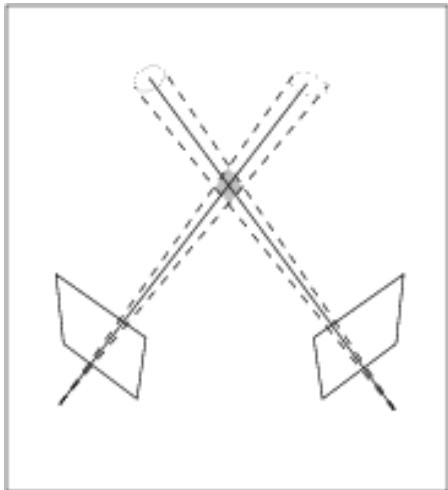
ou (équivalent) $\hat{x} = P \hat{X}$ and $\hat{x}' = P' \hat{X}$



Peut se calculer avec LM

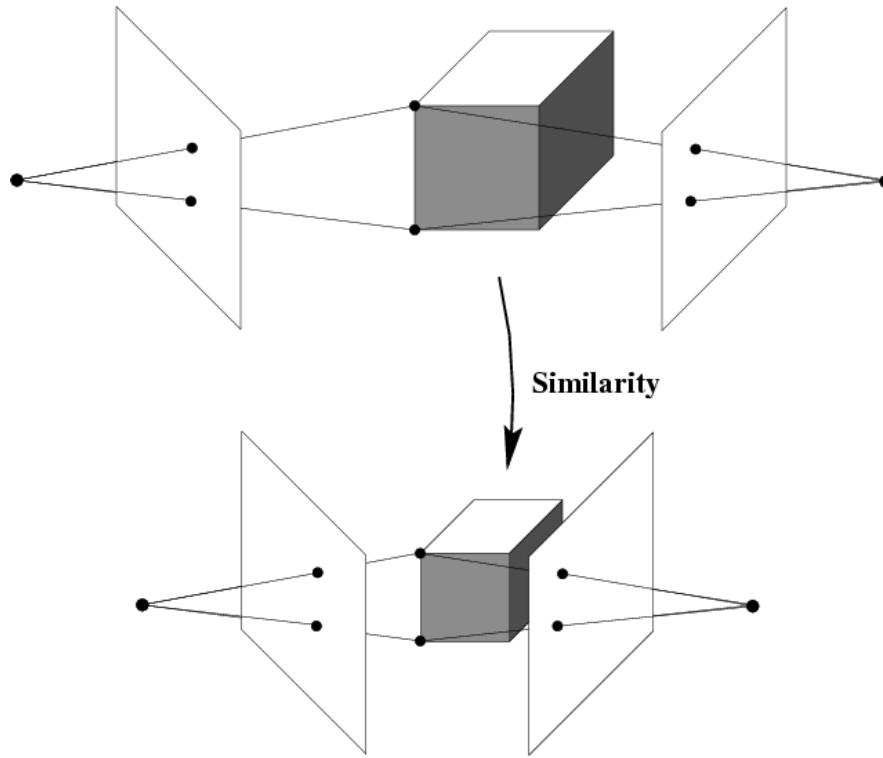
(iii) Reconstruction des points

Erreur de reconstruction



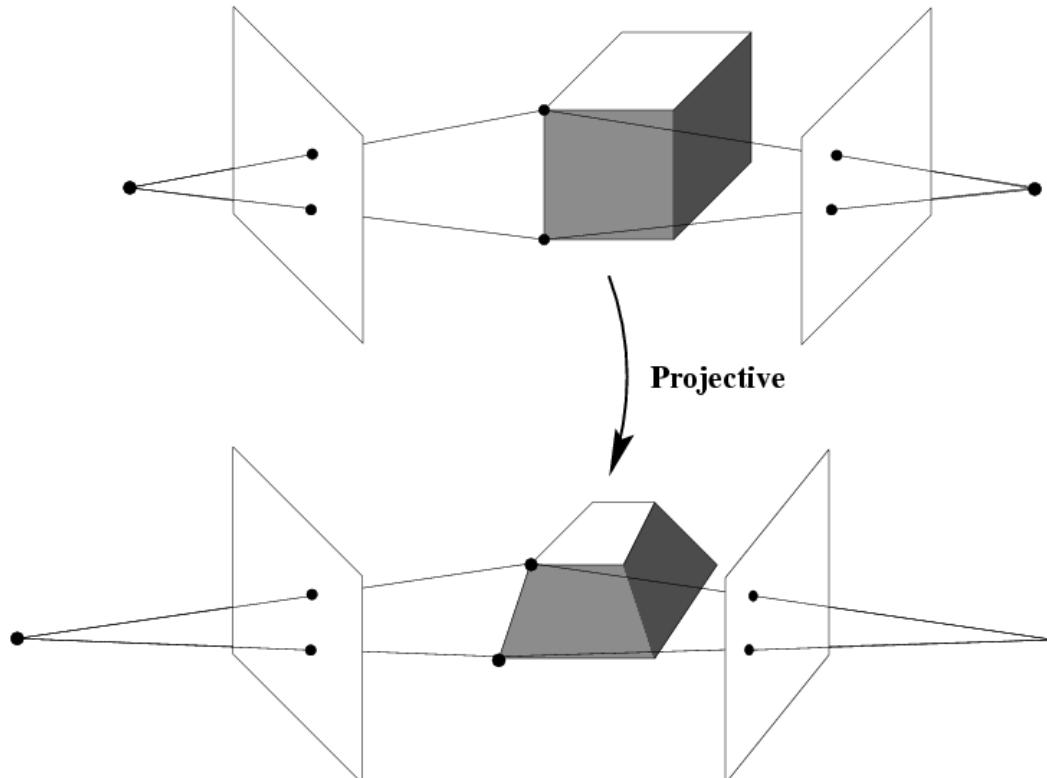
dépend des angles entre rayons

(iii) Reconstruction 3D : Ambiguités



$$\mathbf{x}_i = \mathbf{P}\mathbf{X}_i = (\mathbf{P}\mathbf{H}_S^{-1})(\mathbf{H}_S\mathbf{X}_i)$$

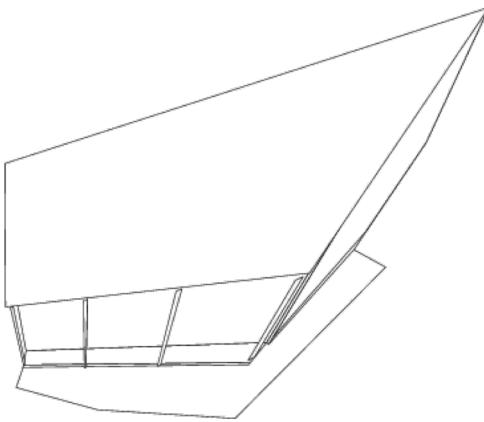
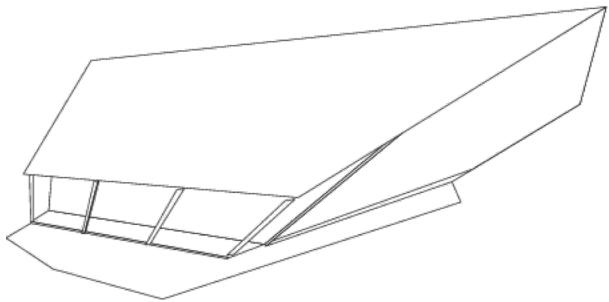
(iii) Reconstruction 3D : Ambiguités



$$\mathbf{x}_i = \mathbf{P}\mathbf{X}_i = (\mathbf{P}\mathbf{H}_P^{-1})(\mathbf{H}_P \mathbf{X}_i)$$

Reconstruction possible avec des caméras calibrées à une transformation projective près

(iii) Reconstruction projective



(iii) Reconstruction métrique

Approche directe

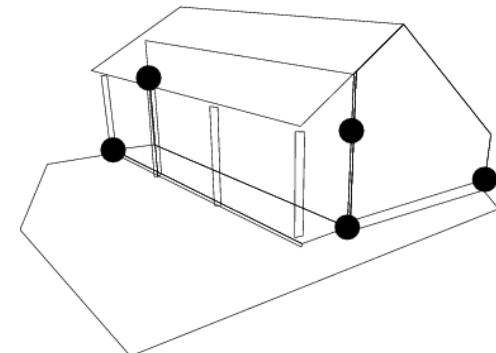
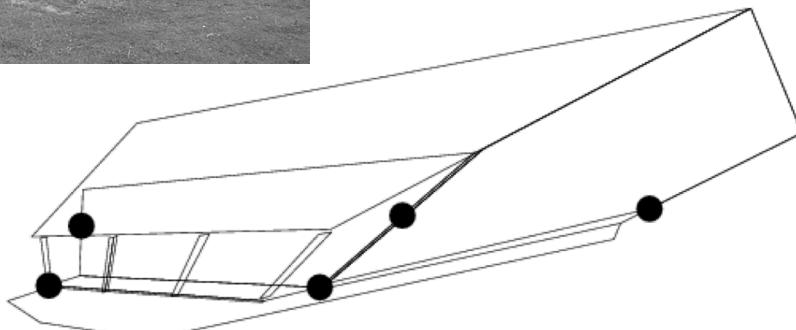
- Utiliser des points X_{Ei} connus pour passer du projectif au métrique



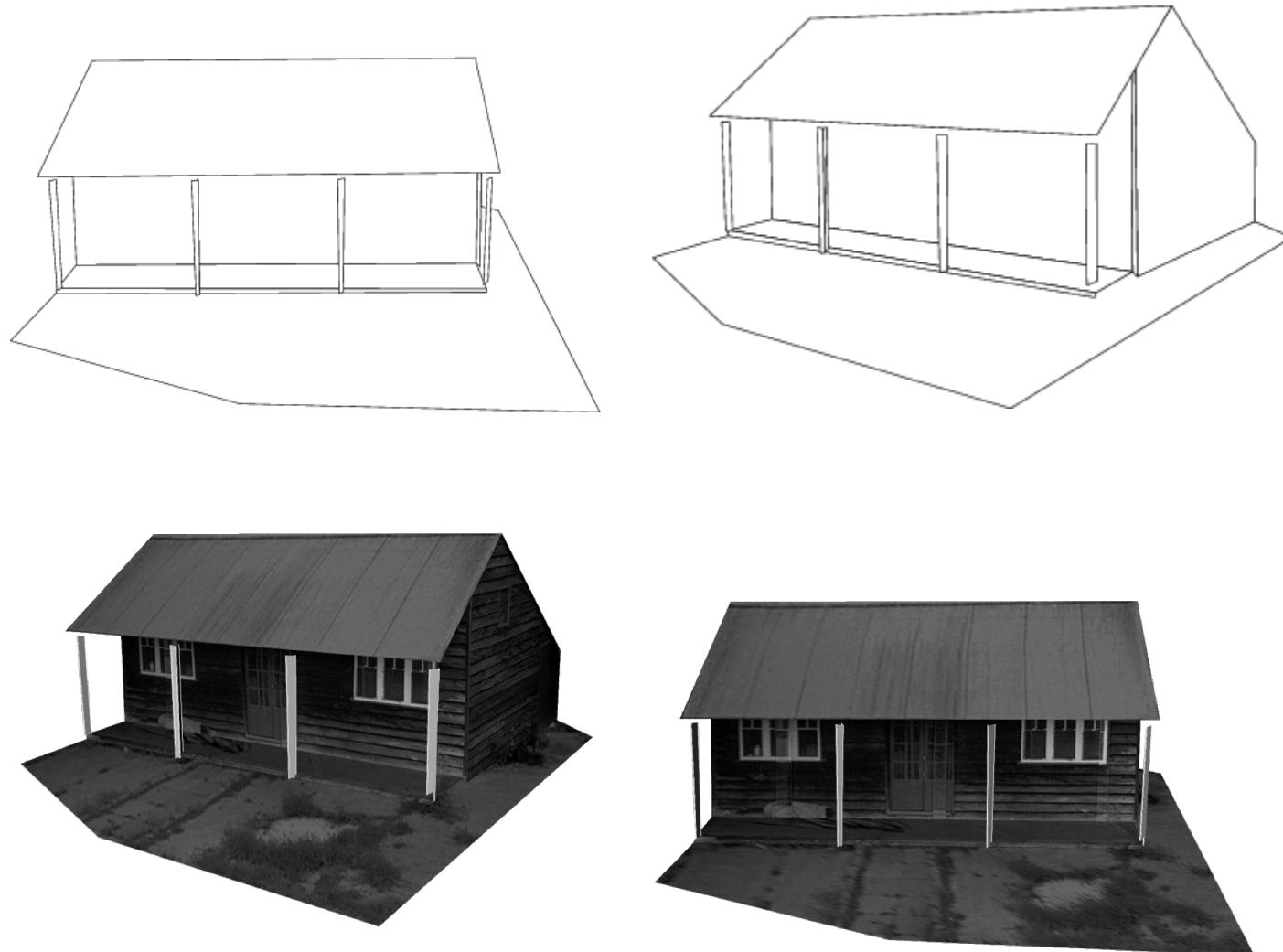
$$X_{Ei} = H X_i$$

$$x_i = P H^{-1} X_{Ei}$$

(2 éq. Lin. en H^{-1} par vue,
3 pour deux vues)



(iii) Reconstruction métrique



Reconstruction 3D - résumé

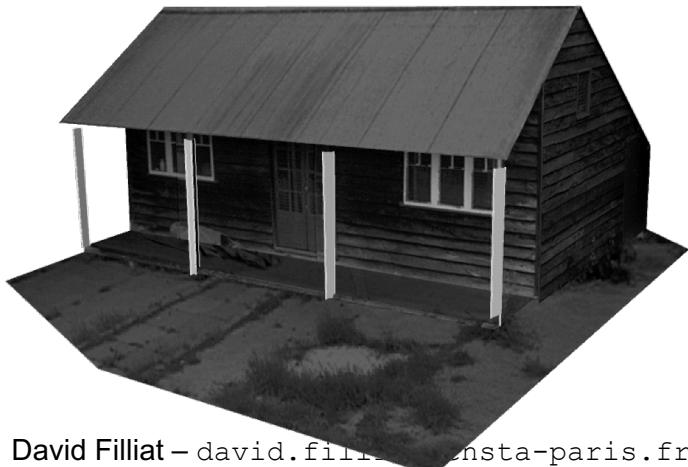
Algorithme

Calcul d'une reconstruction projective ($P, P', \{X_i\}$)

- (i) Calcul F à partir de $x_i \leftrightarrow x'_i$
- (ii) Calcul P, P' à partir de F
- (iii) Trianguler X_i à partir de $x_i \leftrightarrow x'_i$

Corriger reconstruction de projectif à métrique

- **Méthode directe:** calc H depuis points connus



$$X_{Ei} = H X_i$$

$$P_M = P H^{-1} \quad P'_M = P' H^{-1} \quad X_{Mi} = H X_i$$

Reconstruction robuste

Calcul de F automatique / robuste

1. Détection de points d'intérêt
2. Estimation des correspondances
3. RANSAC
4. Ré-estimation de F
5. Correspondances guidées
6. Estimation finale de F

Reconstruction robuste

Trouver les correspondances



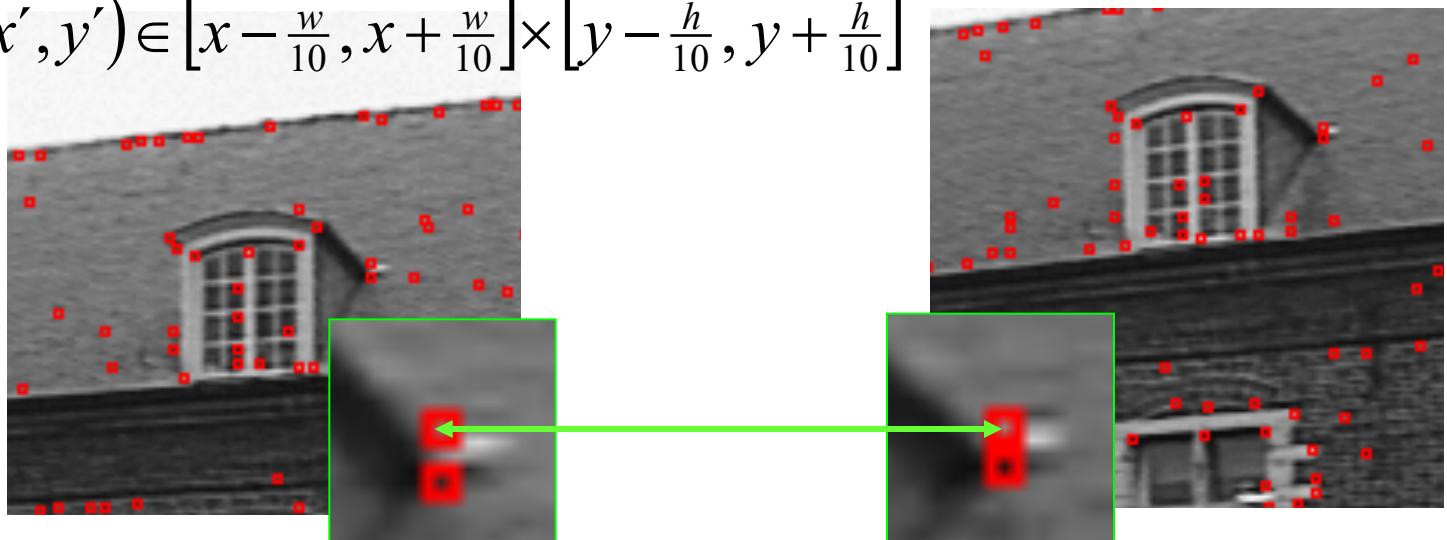
Choisir les points les plus « visibles » (e.g. 1000/image)
Harris, SIFT, SURF, Kaze ...

Reconstruction robuste

Trouver les correspondances

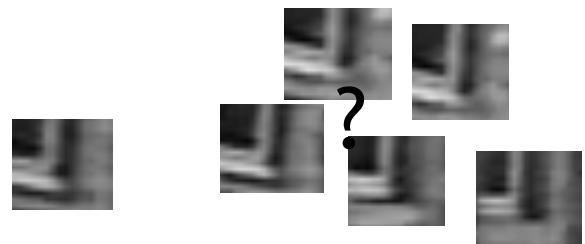
Calculer ZNCC, SSD, SAD pour tous les points avec des coordonnées proches

$$\text{e.g. } (x', y') \in \left[x - \frac{w}{10}, x + \frac{w}{10} \right] \times \left[y - \frac{h}{10}, y + \frac{h}{10} \right]$$



Garder les meilleurs match réciproques

-> peut contenir encore bcp erreurs



Reconstruction robuste

Calcul de F automatique / robuste

Step 1. extraire features

Step 2. Calculer corresp. potentielles

Step 3. do

Step 3.1 choisir échantillon Min (i.e. 7 matches)

Step 3.2 calcule solution pour F

Step 3.3 calculer les inliers

jusqu'à $\Gamma(\#inliers, \#samples) < 95\%$

RANSAC

(génération hypothèse)

(vérifie hypothèse)

Step 4. Calcule F basé sur tous les inliers

Step 5. Recherche de corresp. supplémentaires

Step 6. Raffiner F avec toutes les corresp

#inliers	90%	80%	70%	60%	50%
#samples	5	13	35	106	382

$$\Gamma = 1 - \left(1 - \left(\frac{\#inliers}{\#matches}\right)^7\right)^{\#samples}$$

Reconstruction robuste

Trouver plus de correspondances

- A partir de F estimée par RANSAC

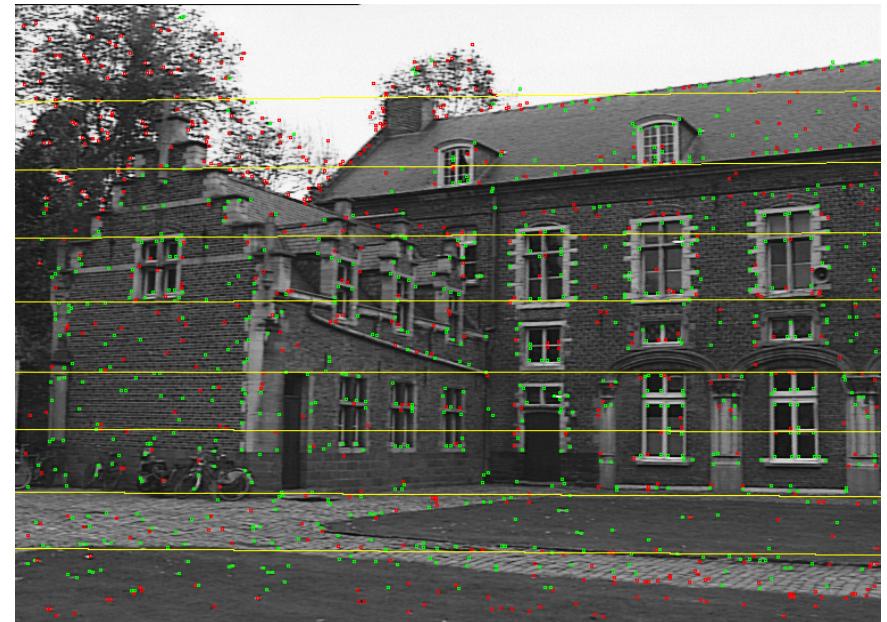


restreindre la recherche au voisinage des lignes epipolaires
(± 1.5 pixels)

Reconstruction robuste

Estimation finale de F

- Ré-estimer F à partir de toutes les correspondances



Géométrie épipolaire entre les deux vues déterminée par F

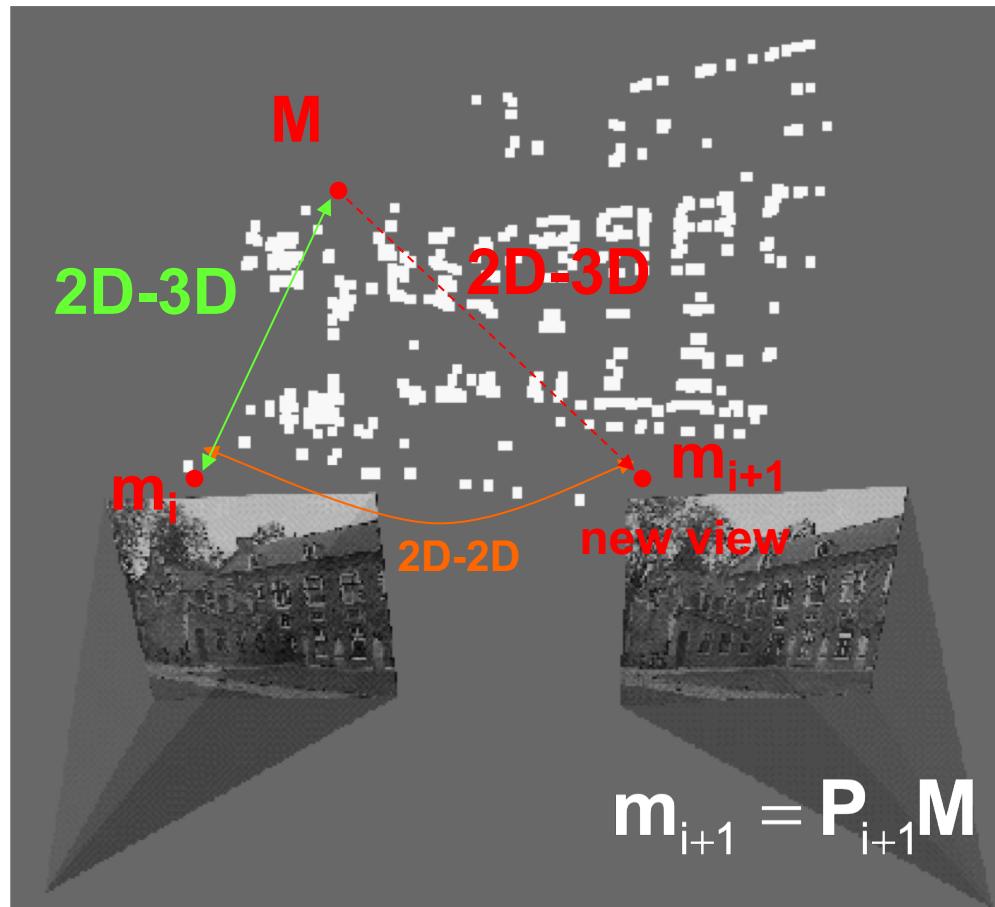
Reconstruction N vues

Approche séquentielle

Initialise la structure à partir de 2 vues

Pour chaque nouvelle vue

- Déterminer la pose P_{i+1} de la caméra par rapport à la caméra précédente (Calcul F - RANSAC)
- Raffiner les points 3D reconnus (Triangulation)
- Ajouter les nouveaux points 3D

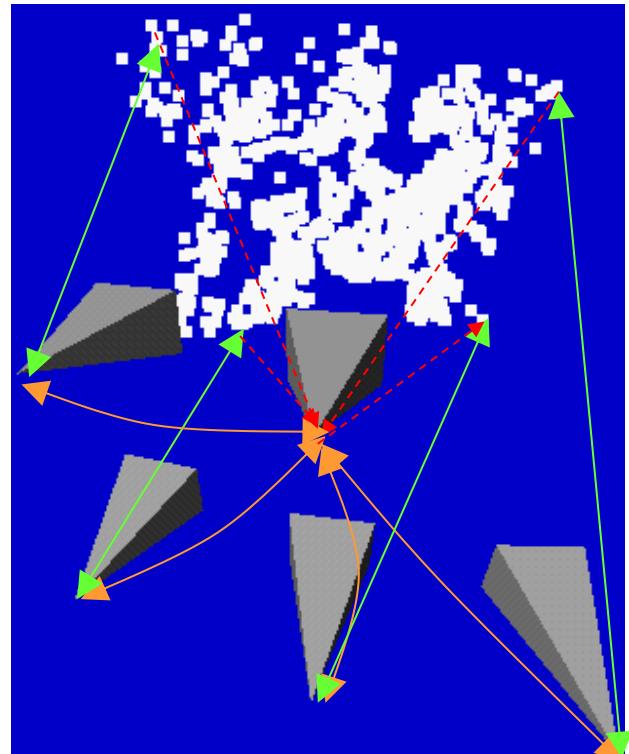


Reconstruction N vues Ajustement de faisceau

Initialise la structure à partir de 2 vues

Pour chaque nouvelle vue

- Déterminer P_{i+1} / img i
- Pour toutes les vues proches k
 - Determiner P_{i+1} / img k
 - Estimer matching 2D/3D
- Raffiner P_{i+1} à partir de tous les match
- Raffiner et étendre les points 3D à partir de tous les matching



Reconstruction 3D – résumé

Reconstruction robuste 2 vues

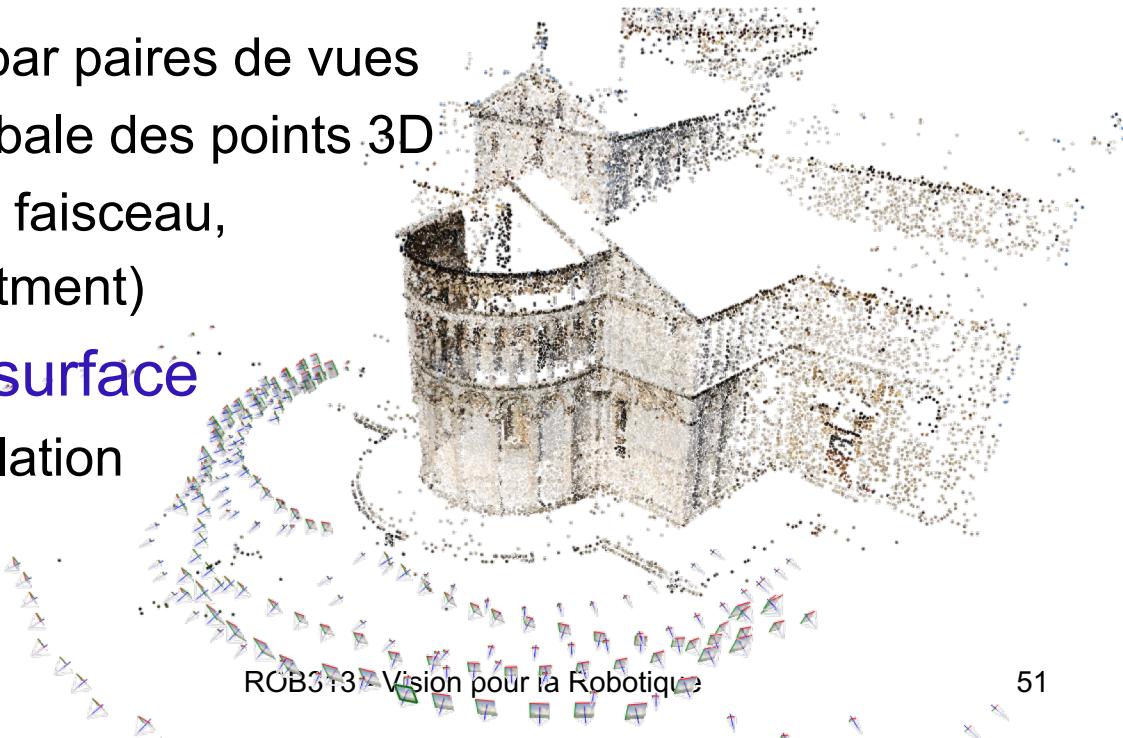
- Matching de points
- Estimation de F par RANSAC
- Raffinement

Reconstruction multi-vues

- Reconstruction par paires de vues
- Optimisation globale des points 3D
(ajustement de faisceau,
bundle adjustment)

Reconstruction de surface

- Filtrage / triangulation



Exemple



Logiciels

Plusieurs logiciels open-source

- Ex : COLMAP : <https://colmap.github.io/index.html>



Sparse model of central Rome using 21K photos produced by COLMAP's SfM pipeline.

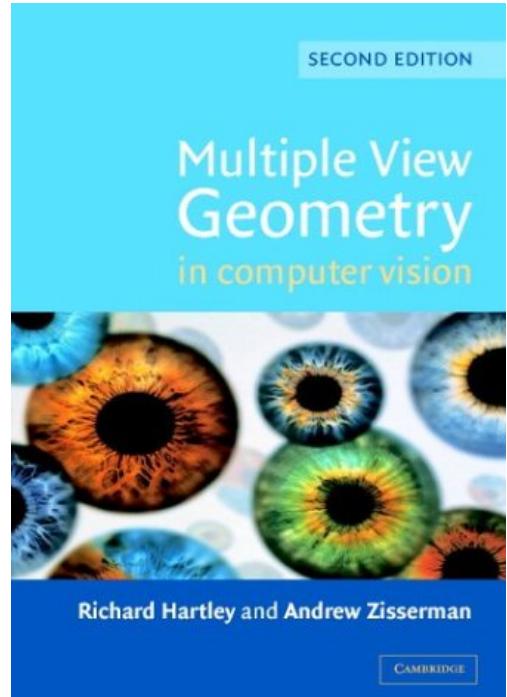


Dense models of several landmarks produced by COLMAP's MVS pipeline.

Références

Multiple view geometry in computer vision

- Hartley and Zisserman



Slides de Marc Pollefeys

- <http://www.cs.unc.edu/~marc/mvg/slides.html>