



UNIVERSITÉ DE
SHERBROOKE

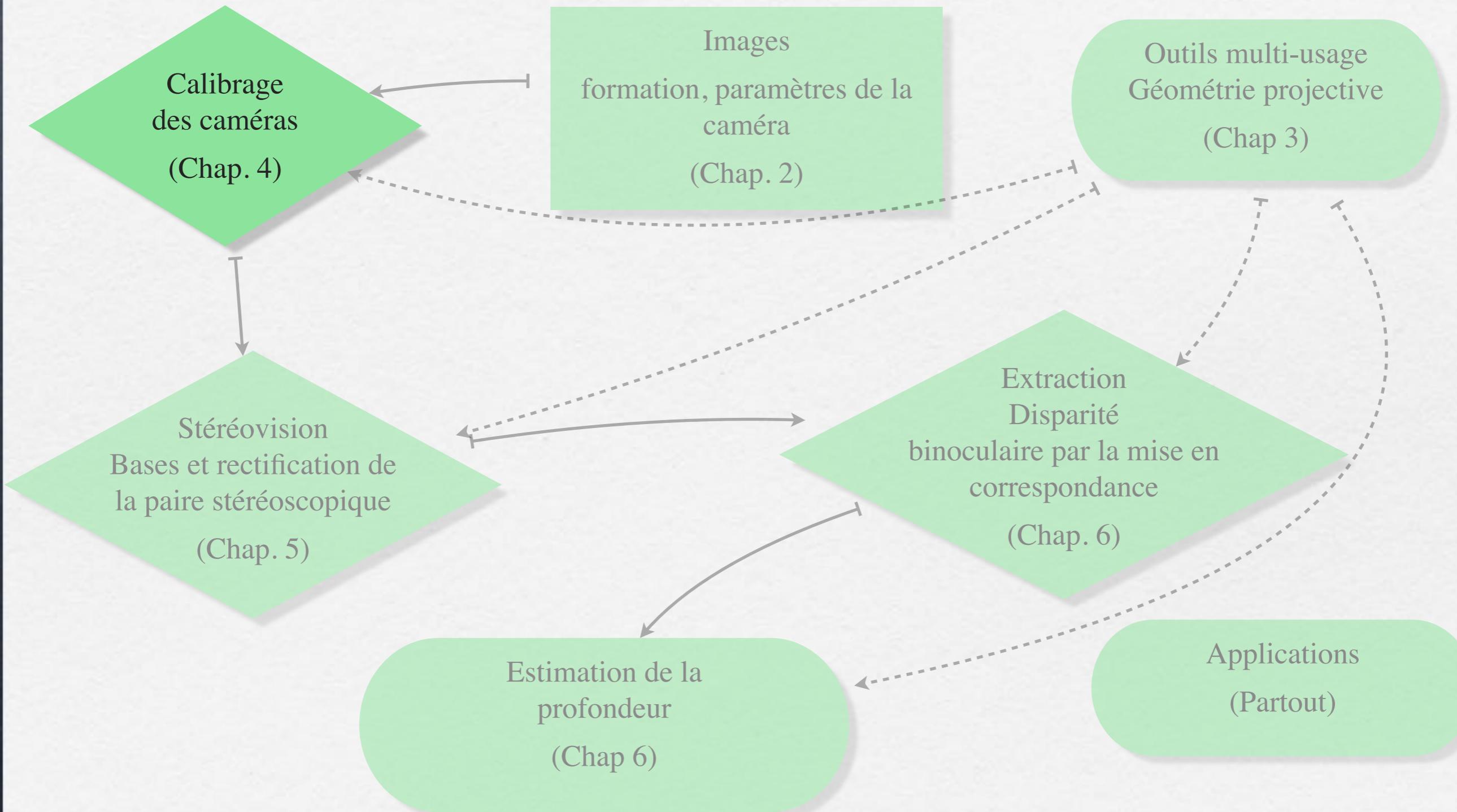
IMN269 - Vision tridimensionnelle

Chapitre 4

Calibrage géométrique des caméras

notes de cours de
Marie-Flavie Auclair-Fortier
feat. Olivier Godin

Cas de la vision stéréoscopique



Mise en contexte

- La reconstruction de la structure 3D d'une scène ou le calcul de la position des objets dans l'espace implique la capacité de déterminer les équations qui lient les coordonnées des points 3D aux coordonnées des points correspondants dans l'image

Objectif du chapitre

- ☒ Déterminer les paramètres (extrinsèques et intrinsèques) de la caméra qui influencent la géométrie de formation d'image, à partir des informations de l'image et de connaissances *a priori*

Mise en contexte

- Le lien entre la position d'un point de la scène (\mathbf{P}_s) et sa projection en pixels (\mathbf{p}) est déterminé par les équations de formation d'images dont les paramètres (de la caméra) sont inconnus.
✓ $\mathbf{p} = \text{fonction}(\text{paramètres}, \mathbf{P}_s)$
- Si on veut que les paramètres soient les seuls inconnues, on doit connaître les coordonnées de \mathbf{p} et \mathbf{P}_s .
✓ $(\mathbf{P}_s, \mathbf{p})$: paire de points correspondants
- *Problème* : Comme on a plusieurs paramètres à estimer, on doit connaître plusieurs paires correspondantes
➡ Utiliser une ou plusieurs images d'un modèle de calibrage



Plan du chapitre

1. Modèle de calibrage

2. Estimation sans distorsion radiale

3. Estimation avec distorsion radiale

4. Autocalibrage

1. Modèle de calibrage

Un modèle de calibrage est un modèle 3D dont la géométrie et la position des points caractéristiques sont connus.

Les caractéristiques d'un bon modèle sont :

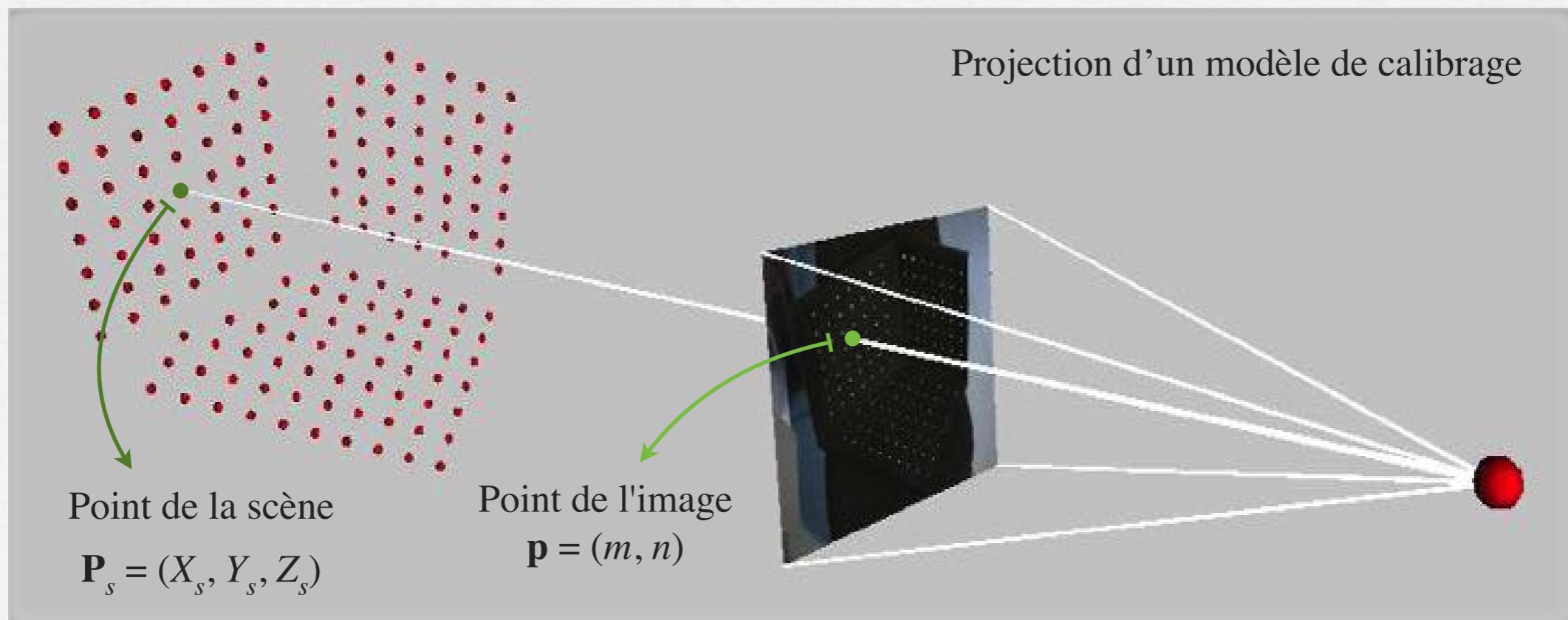
1. **Géométrie simple** : des points faciles à détecter précisément (coins, croix, etc.).
2. **Contraste élevé** : pour éviter les manques et les fausses détections.

1. Modèle de calibrage

- L'utilisation d'un modèle de calibrage suppose qu'on connaisse la position géométrique de plusieurs points caractéristiques dans la scène. Il faut pouvoir détecter facilement la position de leur projection dans l'image.

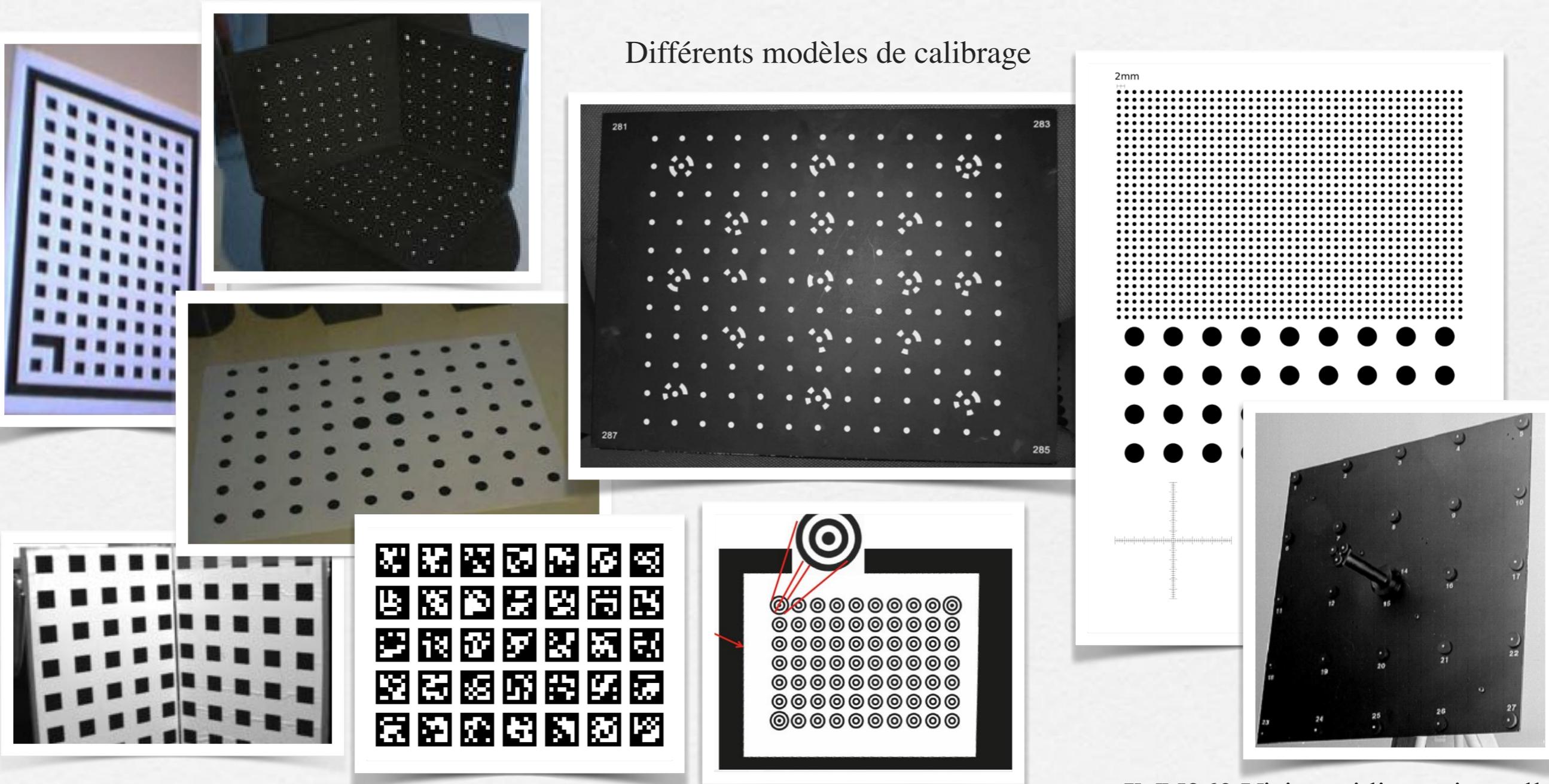
Paire correspondante

- paire (P_s, p) où p correspond à l'image du point P_s



1. Modèle de calibrage

- Les coins sont très fiables car ils sont indépendants du point de vue, de l'échelle, et de l'orientation



Sources diverses

1. Modèle de calibrage

Détection de jonctions (rappels)

- 1 détection de contours (ex. Canny)
- 2 transformée de Hough (trouve les lignes droites)
- 3 intersection des lignes : coins

OU

détecteur de coins basé sur le Laplacien (ex. détecteur de Tabbone)

OU

autre détecteur : Harris, Shi-Tomasi, SIFT, ...

Plan du chapitre

1. Modèle de calibrage

2. Estimation sans distorsion radiale

2.1. Méthode directe d'estimation

2.2. Méthode par la matrice de projection

2.3. Résolution de systèmes d'équations linéaires surdéterminés

3. Estimation avec distorsion radiale

4. Autocalibrage

2. Estimation sans distorsion radiale

- Rappel (c.f. chapitre 2, eq. (2.1))
 - ✓ Relation entre les coordonnées 3D d'un point dans la scène et les coordonnées pixels du point correspondant dans l'image

$$m = \left(\frac{z'}{S_x} \right) \left(\frac{R_{11}^c X_s + R_{12}^c Y_s + R_{13}^c Z_s + T_x^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right) + O_m \quad (4.1)$$

$$n = - \left(\frac{z'}{S_y} \right) \left(\frac{R_{21}^c X_s + R_{22}^c Y_s + R_{23}^c Z_s + T_y^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right) + O_n$$

- L'équation (4.1) tient compte de tous les paramètres de la caméra sauf la distorsion
 - ➡ $k_1 = k_2 = 0$
- Comme prévu, on laisse *aussi* tomber l'aspect échantillonnage : $\lfloor \bullet \rfloor$

2. Estimation sans distorsion radiale

1. Méthode directe d'estimation

- On veut résoudre l'équation (4.1)
- 2 équations, 17 inconnues
 - ✓ \mathbf{R}^c : 9 inconnues
 - ✓ \mathbf{T}^c : 3 inconnues
 - ✓ \mathbf{O} : 2 inconnues
 - ✓ \mathbf{S} : 2 inconnues
 - ✓ z'
- ➡ Ça prends au moins 9 paires correspondantes
- ➡ Les paramètres sont estimés explicitement
- ➡ Pas trivial à faire

2. Estimation sans distorsion radiale

On peut réécrire l'équation 4.1 sous la forme d'une transformation projective. Posons

$$M_{\text{int}} = \begin{bmatrix} \frac{z'}{S_x} & 0 & O_m \\ 0 & \frac{z'}{S_y} & O_n \\ 0 & 0 & 1 \end{bmatrix} \quad \text{et} \quad M_{\text{ext}} = \begin{bmatrix} R_{11}^c & R_{12}^c & R_{13}^c & T_x^c \\ R_{21}^c & R_{22}^c & R_{23}^c & T_y^c \\ R_{31}^c & R_{32}^c & R_{33}^c & T_z^c \end{bmatrix}$$

On peut donc écrire la relation entre les coordonnées dans la scène et les coordonnées en pixels en coordonnées homogènes :

$$\begin{bmatrix} m \\ n \\ 1 \end{bmatrix} \approx \begin{bmatrix} x \\ y \\ z \end{bmatrix} = M_{\text{int}} M_{\text{ext}} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix}$$

2. Estimation sans distorsion radiale

2. Méthode par la matrice de projection

■ Posons

$$\mathbf{M} = \mathbf{M}_{int}\mathbf{M}_{ext} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}$$

✓ L'équation (4.1) devient

$$m = \left(\frac{m_{11}X_s + m_{12}Y_s + m_{13}Z_s + m_{14}}{m_{31}X_s + m_{32}Y_s + m_{33}Z_s + m_{34}} \right)$$
$$n = \left(\frac{m_{21}X_s + m_{22}Y_s + m_{23}Z_s + m_{24}}{m_{31}X_s + m_{32}Y_s + m_{33}Z_s + m_{34}} \right)$$

■ Les paramètres sont estimés implicitement

➡ on ne les a pas directement, ils sont mélangés

✓ On réorganise (on passe le dénominateur à gauche)

$$m(m_{31}X_s + m_{32}Y_s + m_{33}Z_s + m_{34}) = m_{11}X_s + m_{12}Y_s + m_{13}Z_s + m_{14}$$

$$n(m_{31}X_s + m_{32}Y_s + m_{33}Z_s + m_{34}) = m_{21}X_s + m_{22}Y_s + m_{23}Z_s + m_{24}$$

2. Estimation sans distorsion radiale

2. Méthode par la matrice de projection

- On a mis toutes les inconnues dans un vecteur pour avoir un *système linéaire d'équations* à résoudre

2 équations à 12 inconnues (les m_{ij})

Forme $\mathbf{Ax} = \mathbf{0}$ sous déterminée

➡ il manque des informations

$$\begin{bmatrix} X_s & Y_s & Z_s & 1 & 0 & 0 & 0 & -mX_s & -mY_s & -mZ_s & -m \\ 0 & 0 & 0 & 0 & X_s & Y_s & Z_s & 1 & -nX_s & -nY_s & -nZ_s & -n \end{bmatrix} \begin{bmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

2. Estimation sans distorsion radiale

2. Méthode par la matrice de projection

- 2 équations, 12 inconnues : les 12 éléments de \mathbf{M}

✓ Ça prends au moins 6 paires correspondantes connues pour avoir assez d'équations et avoir une matrice \mathbf{A} qui soit carrée (et inversible)

$$\underbrace{\left(\begin{matrix} X_{s_i}, Y_{s_i}, Z_{s_i} \end{matrix} \right)}_{\substack{\text{point de la scène} \\ \text{ coordonnées scène}}} \rightarrow \underbrace{\left(\begin{matrix} m_i, n_i \end{matrix} \right)}_{\text{pixel de l'image}}, \quad i = 1, \dots, 6$$

$$\left[\begin{array}{ccccccccccccc}
 X_{s_1} & Y_{s_1} & Z_{s_1} & 1 & 0 & 0 & 0 & 0 & -m_1 X_{s_1} & -m_1 Y_{s_1} & -m_1 Z_{s_1} & -m_1 \\
 0 & 0 & 0 & 0 & X_{s_1} & Y_{s_1} & Z_{s_1} & 1 & -n_1 X_{s_1} & -n_1 Y_{s_1} & -n_1 Z_{s_1} & -n_1 \\
 X_{s_2} & Y_{s_2} & Z_{s_2} & 1 & 0 & 0 & 0 & 0 & -m_2 X_{s_2} & -m_2 Y_{s_2} & -m_2 Z_{s_2} & -m_2 \\
 0 & 0 & 0 & 0 & X_{s_2} & Y_{s_2} & Z_{s_2} & 1 & -n_2 X_{s_2} & -n_2 Y_{s_2} & -n_2 Z_{s_2} & -n_2 \\
 & & & & \vdots & & & & & & & \\
 X_{s_6} & Y_{s_6} & Z_{s_6} & 1 & 0 & 0 & 0 & 0 & -m_6 X_{s_6} & -m_6 Y_{s_6} & -m_6 Z_{s_6} & -m_6 \\
 0 & 0 & 0 & 0 & X_{s_6} & Y_{s_6} & Z_{s_6} & 1 & -n_6 X_{s_6} & -n_6 Y_{s_6} & -n_6 Z_{s_6} & -n_6
 \end{array} \right] = \left[\begin{array}{c}
 m_{11} \\
 m_{12} \\
 m_{13} \\
 m_{14} \\
 m_{21} \\
 m_{22} \\
 m_{23} \\
 m_{24} \\
 m_{31} \\
 m_{32} \\
 m_{33} \\
 m_{34}
 \end{array} \right] = \left[\begin{array}{c}
 0 \\
 0 \\
 0 \\
 0 \\
 0 \\
 0 \\
 0 \\
 0 \\
 0 \\
 0 \\
 0 \\
 0
 \end{array} \right]$$

2. Estimation sans distorsion radiale

2. Méthode par la matrice de projection

■ *Remarque*

- ✓ Dans les faits, on ne peut trouver \mathbf{M} qu'à un facteur d'échelle près

$$\begin{bmatrix} m \\ n \\ 1 \end{bmatrix} = \lambda \mathbf{M} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix} \equiv \mathbf{M} \begin{bmatrix} X_s \\ Y_s \\ Z_s \\ 1 \end{bmatrix}$$

Dans les faits, on utilisera BEAUCOUP plus que 6 paires correspondantes (on en discute section suivante)

2. Estimation sans distorsion radiale

3. Résolution de systèmes d'équations linéaires surdéterminés

- Souvent, on a des systèmes d'équations linéaires à résoudre
 - On pourrait résoudre avec exactement le bon nombre de paires de points
 - ✓ même nombre d'équations (N) que d'inconnues (M)

$$\boxed{\mathbf{A}} \quad \boxed{\mathbf{x}} = \boxed{\mathbf{b}}$$

\mathbf{A} carrée ($M \times M$) : $N = M$

- solution unique
- élimination Gaussienne

■ Problème

- ✓ Les données sont bruitées en niveaux de gris, mais aussi en positionnement (entre autres à cause de l'échantillonnage). Quelles paires de points choisir alors?
 - Il faut prendre plus de paires de points que nécessaire pour trouver une meilleure approximation
 - Idéalement des points répartis *dans toute l'image* pour compenser pour un défaut local du capteur ou de la lentille

2. Estimation sans distorsion radiale

3. Résolution de systèmes d'équations linéaires surdéterminés

- Le problème s'écrit sous la forme

$$\boxed{\begin{array}{|c|} \hline A \\ \hline \end{array}} \quad \boxed{x} = \boxed{\begin{array}{|c|} \hline b \\ \hline \end{array}}$$



A carrée inversible ($N \times M$, avec $N = M$)

solution unique



A rectangulaire sous déterminée ($N \times M$, avec $N < M$)

infinité de solutions



A rectangulaire surdéterminée ($N \times M$, avec $N > M$)

pas de solution : comment résoudre ?

- Il faut minimiser l'erreur quadratique $(Ax - b)^2$

→ Posons

$$e = (Ax - b) = \left[\begin{array}{cccc} A_1 & A_2 & \cdots & A_M \end{array} \right] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix} - b \quad \text{avec } A_i = \begin{bmatrix} A_{1i} \\ A_{2i} \\ \vdots \\ A_{Ni} \end{bmatrix} \quad \text{et } b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}$$

2. Estimation sans distorsion radiale

3. Résolution de systèmes d'équations linéaires surdéterminés

$$\mathbf{e} = \left(\sum_{i=1}^M \mathbf{A}_i x_i \right) - \mathbf{b}$$

→ vecteur $N \times 1$ qui représente l'accumulation des erreurs d'évaluation de \mathbf{b} par \mathbf{Ax}

- On veut minimiser l'erreur quadratique :

$$\mathbf{e}^2 = \mathbf{e} \cdot \mathbf{e} = \sum_{j=1}^N (e_j)^2 \quad \rightarrow \text{produit scalaire :} \quad (e_j)^2 = \left(\left(\sum_{i=1}^M \mathbf{A}_{ji} x_i \right) - b_j \right)^2$$

✓ on dérive \mathbf{e}^2 par rapport à chacun des x_i

$$\frac{\partial \mathbf{e}^2}{\partial x_i} = 2 \frac{\partial \mathbf{e}}{\partial x_i} \cdot \mathbf{e} = 2 \mathbf{A}_i \cdot \mathbf{e} = 2 \mathbf{A}_i^t (\mathbf{Ax} - \mathbf{b}) = \mathbf{0} \quad \rightarrow \text{vecteur } N \times 1$$

$$\Rightarrow \begin{bmatrix} \frac{\partial \mathbf{e}^2}{\partial x_1} \\ \frac{\partial \mathbf{e}^2}{\partial x_2} \\ \vdots \\ \frac{\partial \mathbf{e}^2}{\partial x_M} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1^t \\ \mathbf{A}_2^t \\ \vdots \\ \mathbf{A}_M^t \end{bmatrix} \begin{matrix} 1 \times N \\ N \times 1 \\ \downarrow \\ M \times N \end{matrix} (\mathbf{Ax} - \mathbf{b}) = \mathbf{A}^t (\mathbf{Ax} - \mathbf{b}) = \mathbf{A}^t \mathbf{Ax} - \mathbf{A}^t \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

2. Estimation sans distorsion radiale

3. Résolution de systèmes d'équations linéaires surdéterminés

$$\Rightarrow \mathbf{A}^t \mathbf{A} \mathbf{x} = \mathbf{A}^t \mathbf{b} \quad \text{avec } \mathbf{A}^t \mathbf{A}, \text{ une matrice } M \times M \text{ inversible}$$

$$\Rightarrow \mathbf{x} = (\mathbf{A}^t \mathbf{A})^{-1} \mathbf{A}^t \mathbf{b}$$

matrice pseudo-
inverse de \mathbf{A}

- Donc, pour résoudre un système surdéterminé, on n'a qu'à bâtir la pseudo-inverse de \mathbf{A} et multiplier par \mathbf{b}
- Autre application : retrait de l'effet de perspective (chapitre 3, section 2.4)
- ✓ utiliser plus de 4 points rend le résultat plus fiable





Plan du chapitre

1. Modèle de calibrage

2. Estimation sans distorsion radiale

3. Estimation avec distorsion radiale

3.1. Méthode de Tsai

3.2. Méthode d'optimisation Levenberg-Marquardt

4. Autocalibrage

3. Estimation avec distorsion radiale

- Quand on doit tenir compte de la distorsion, il y a 2 paramètres de plus à estimer : k_1 et k_2

$$m = \left(\frac{z'}{S_x} \right) \left(\frac{R_{11}^c X_s + R_{12}^c Y_s + R_{13}^c Z_s + T_x^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right) (1 + k_1 r^2 + k_2 r^4) + O_m \quad (4.2)$$
$$n = - \left(\frac{z'}{S_y} \right) \left(\frac{R_{21}^c X_s + R_{22}^c Y_s + R_{23}^c Z_s + T_y^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right) (1 + k_1 r^2 + k_2 r^4) + O_n$$

- ✓ Il faut ajouter une paire correspondante
- ✓ Difficile à trouver par une méthode explicite (directe) car on a des équations non linéaires
- ✓ Encore moins évident par la méthode implicite (matrice de projection), pas moyen d'isoler la distorsion

On peut alors utiliser la **méthode de Tsai** qui permet de calibrer un système avec distorsion radiale à l'aide de cinq paires de points seulement. Il est par contre nécessaire de faire certaines hypothèses...

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Hypothèses (contraintes)

$$\underbrace{(X_{s_i}, Y_{s_i}, Z_{s_i})}_{\begin{array}{l} \text{point de la scène} \\ \text{coordonnées scène} \end{array}} \rightarrow \underbrace{(m_i, n_i)}_{\text{pixel de l'image}}, \quad i = 1, \dots, N$$

On connaît N paires correspondantes ($N \geq 5$)

- Tous les points de la scène sont sur un seul et même plan
 - on veut avoir $Z_{s_i} = 0 \quad \forall i$
- L'axe des X de la scène est loin de l'axe optique
 - il faut que $T_y \neq 0$
- L'axe des Z de la scène ne doit pas être aligné avec l'axe des Z de la caméra
- $\mathbf{S} = (S_x, S_y)$ et $\mathbf{O} = (O_m, O_n)$ sont connus
 - données du fabricant ou calibrage préalable

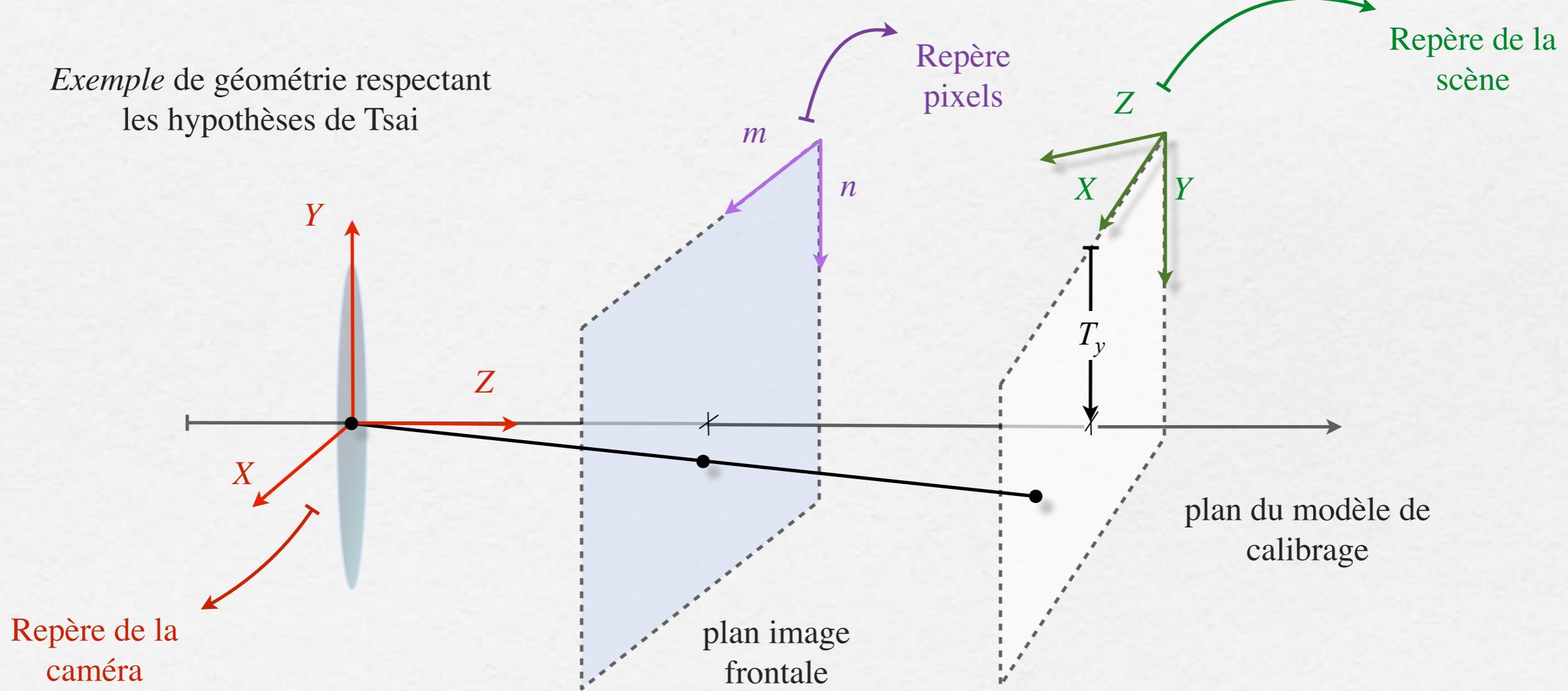
3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Remarque

- ✓ Le plan du modèle de calibrage ne doit pas être parallèle au plan image

Exemple de géométrie respectant les hypothèses de Tsai



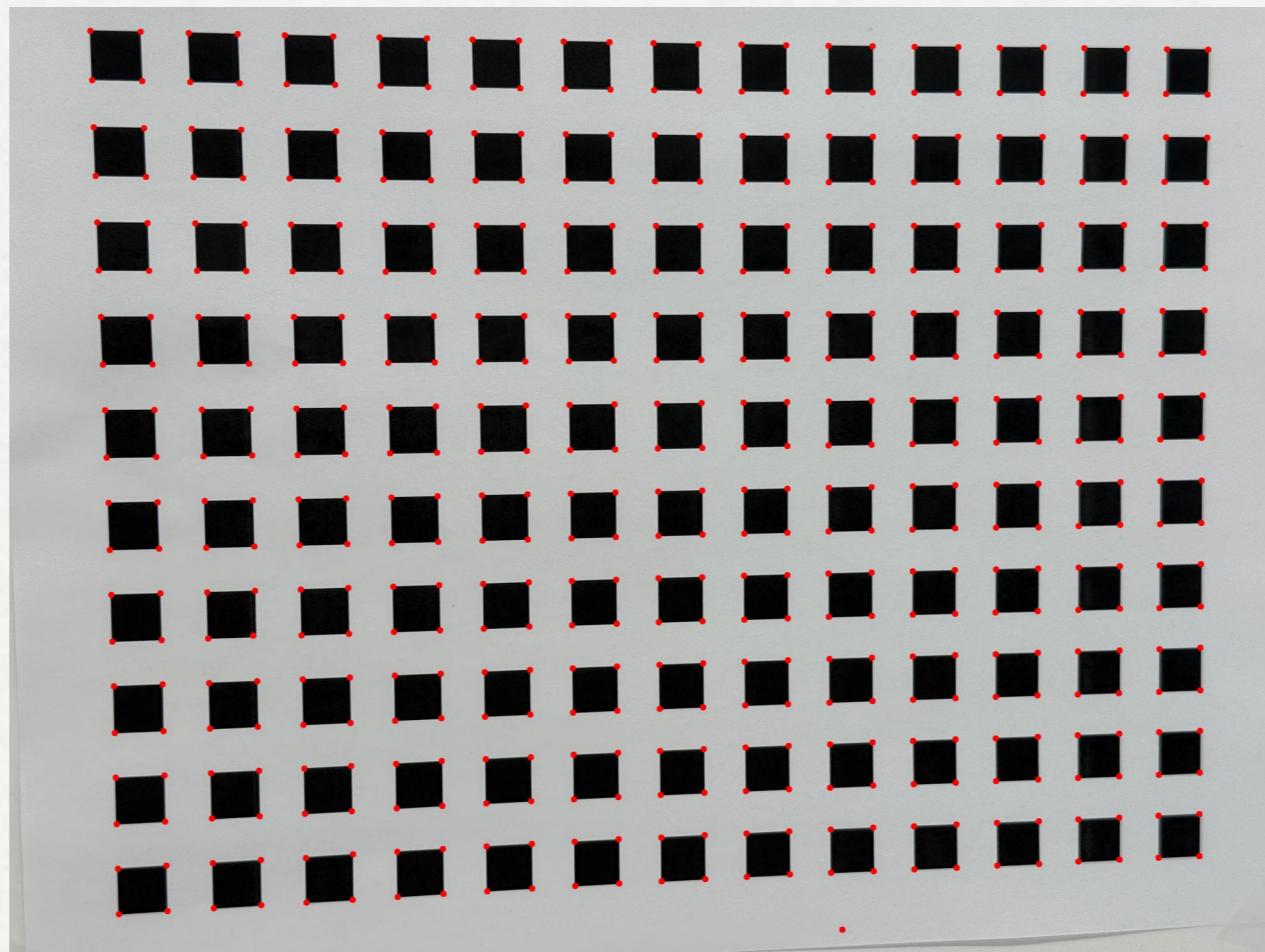
3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 0

Calculer la position des points distorsions dans le repère image réelle :

$$\begin{aligned}x_{di} &= (m_i - O_m) \cdot S_x \\y_{di} &= (n_i - O_n) \cdot S_y\end{aligned}\quad \text{pour } i = 1, \dots, N$$

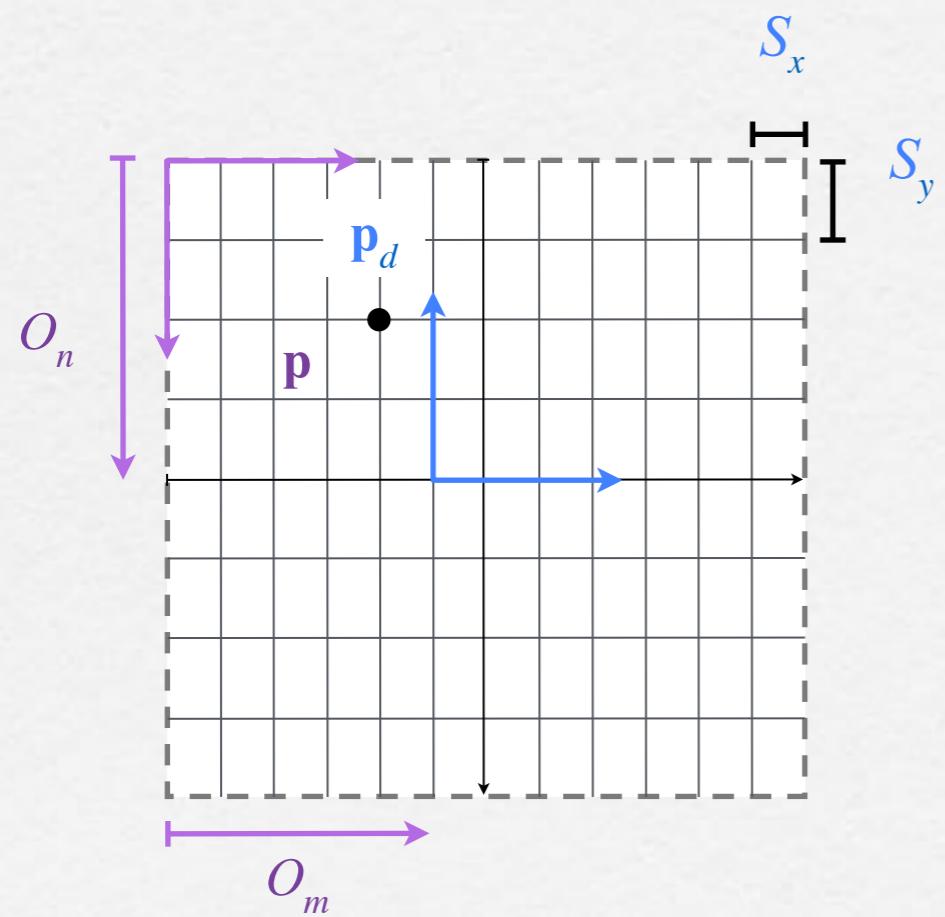


3. Estimation avec distorsion radiale

1. Méthode de Tsai

■ Étape 0

- ✓ On enlève l'échantillonnage et le changement de repère (équations inverses de la formation d'image)
 - ➡ On se replace dans le repère image : \mathbf{p} vs \mathbf{p}_d ➡ Ce qui nous permet de réécrire l'équation (4.1)



$$\begin{aligned}x_d &= z \left(\frac{R_{11}^c X_s + R_{12}^c Y_s + R_{13}^c Z_s + T_x^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right) \\y_d &= z \left(\frac{R_{21}^c X_s + R_{22}^c Y_s + R_{23}^c Z_s + T_y^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right)\end{aligned}\quad (4.3)$$

Remarques

- Il est évident qu'on ne pourra pas enlever l'erreur de positionnement dû à l'échantillonnage
- on prend pour acquis que la distorsion est négligeable pour l'instant
 - ➡ première approximation comme si il n'y avait pas de distorsion

3. Estimation avec distorsion radiale

1. Méthode de Tsai

■ Étape 1

i) Éliminer temporairement la 3e dimension

$$\frac{x_{d_i}}{y_{d_i}} = \begin{pmatrix} R_{11}^c X_{s_i} + R_{12}^c Y_{s_i} + R_{13}^c Z_{s_i} + T_x^c \\ R_{21}^c X_{s_i} + R_{22}^c Y_{s_i} + R_{23}^c Z_{s_i} + T_y^c \end{pmatrix}$$

$$\Rightarrow x_{d_i} (R_{21}^c X_{s_i} + R_{22}^c Y_{s_i} + T_y^c) = y_{d_i} (R_{11}^c X_{s_i} + R_{12}^c Y_{s_i} + T_x^c)$$

$$\text{car } Z_{s_i} = 0, \quad \forall i = 1 \cdots N$$

connues
inconnues
équation

4 : $X_{s_i}, Y_{s_i}, x_{d_i}, y_{d_i}$
6 : $R_{11}^c, R_{12}^c, R_{21}^c, R_{22}^c, T_x^c, T_y^c$
1 : par paire corresp.

✓ On divise à gauche et à droite par T_y^c et on isole x_{d_i}

$$\Rightarrow x_{d_i} = y_{d_i} \left(\frac{R_{11}^c}{T_y^c} X_{s_i} + \frac{R_{12}^c}{T_y^c} Y_{s_i} + \frac{T_x^c}{T_y^c} \right) - x_{d_i} \left(\frac{R_{21}^c}{T_y^c} X_{s_i} + \frac{R_{22}^c}{T_y^c} Y_{s_i} \right)$$

➡ On cherche à réduire le nombre d'inconnues

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 1

- i) Éliminer temporairement la 3e dimension

On peut alors bâtir le système suivant :

$$\begin{bmatrix} y_{di}X_{si} & y_{di}Y_{si} & y_{di} & -x_{di}X_{si} & -x_{di}Y_{si} \end{bmatrix} \begin{bmatrix} \frac{R_{11}^c}{T_y^c} \\ \frac{R_{12}^c}{T_y^c} \\ \frac{T_x^c}{T_y^c} \\ \frac{R_{21}^c}{T_y^c} \\ \frac{R_{22}^c}{T_y^c} \end{bmatrix} = x_{di}$$

Avec cinq paires de points (minimum), on peut retrouver les inconnues.

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 1

ii) Trouver $|T_y^c|$

$$(T_y^c)^2 = \begin{cases} \frac{S_r - \sqrt{S_r^2 - 4(r_{11}r_{22} - r_{12}r_{21})^2}}{2(r_{11}r_{22} - r_{12}r_{21})^2} & \text{si } (r_{11}r_{22} - r_{12}r_{21}) \neq 0 \\ \frac{1}{r_{1j}^2 + r_{2j}^2} & \text{si } (r_{1j}^2 + r_{2j}^2) \neq 0 \\ \frac{1}{r_{i1}^2 + r_{i2}^2} & \text{si } (r_{i1}^2 + r_{i2}^2) \neq 0 \end{cases}$$

avec $r_{ij} = \frac{R_{ij}^c}{T_y^c}$

et $S_r = r_{11}^2 + r_{12}^2 + r_{21}^2 + r_{22}^2$

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 1

- ii) Trouver $|T_y^c|$

- ✓ d'où ça vient ?

- ➡ \mathbf{R}^c est une matrice de rotation \Rightarrow orthogonale (orthonormale même)

- ➡ avec ces propriétés, on peut démontrer le résultat précédent

- ➡ On a donc

$$T_y^c = \pm \sqrt{(T_y^c)^2}$$

Remarque

- Pour l'instant, on ne sait pas de quel signe est T_y^c

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 1

- iii) Calculer les R_{ij}^c et T_x^c

$$R_{ij} = r_{ij} \cdot |T_y^c|$$

$$T_x^c = \frac{T_x}{T_y^c} \cdot |T_y^c|$$

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 1
 - iii) Calculer les R_{ij}^c et T_x^c

Remarque

- Le problème de signe de T_y^c se répercute sur les R_{ij}^c et T_x^c

Procédure pour déterminer le bon signe

- 1 Choisir une paire correspondante $(X_{s_i}, Y_{s_i}, 0) \rightarrow (x_{d_i}, y_{d_i})$ telle que (x_{d_i}, y_{d_i}) est loin du centre de l'image
- 2 Approximer la projection du point de la scène sur l'image :
$$\tilde{x}_i = R_{11}^c X_{s_i} + R_{12}^c Y_{s_i} + T_x^c$$
$$\tilde{y}_i = R_{21}^c X_{s_i} + R_{22}^c Y_{s_i} + |T_y^c|$$
- 3 Si $\text{sign}(\tilde{x}_i) = \text{sign}(x_{d_i})$ et $\text{sign}(\tilde{y}_i) = \text{sign}(y_{d_i})$
alors $T_y^c = +|T_y^c|$
Sinon $T_y^c = -|T_y^c|$ et corriger les signes de R_{ij}^c et T_x^c

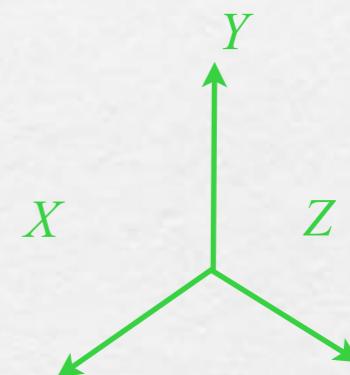
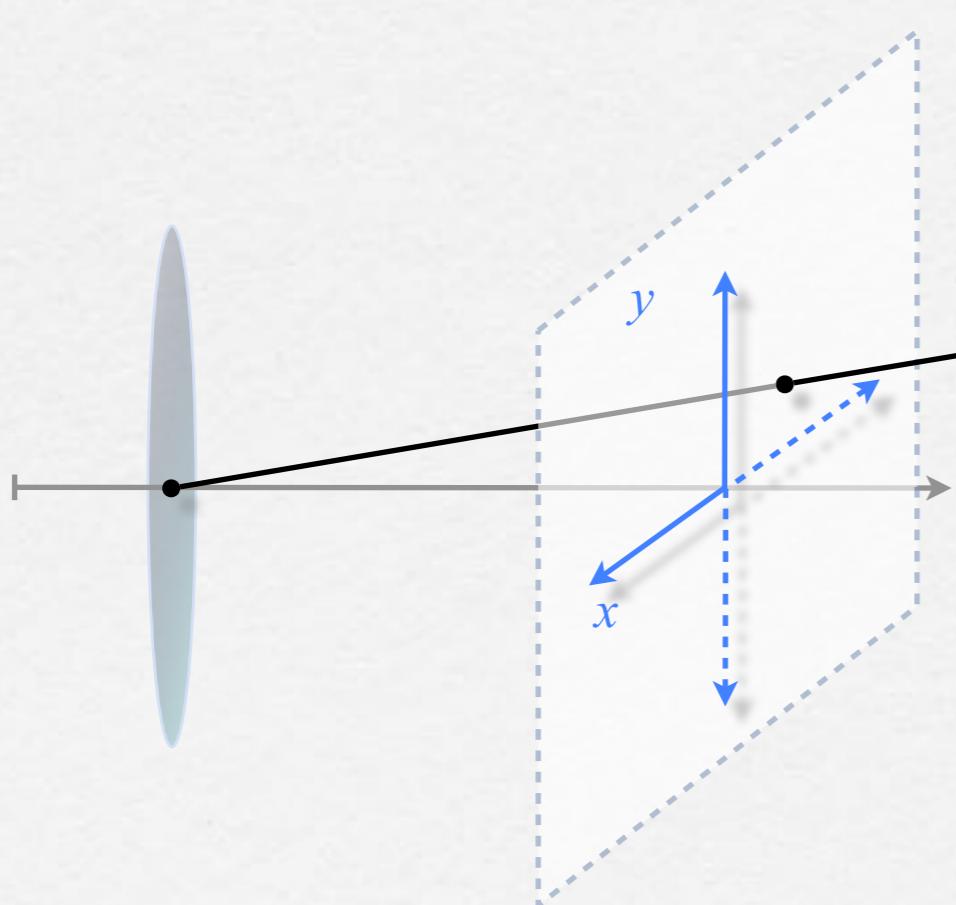
3. Estimation avec distorsion radiale

1. Méthode de Tsai

■ Étape 1

iii) Calculer les R_{ij}^c et T_x^c

Avoir le mauvais signe pour T_y équivaut à faire une rotation de 180° des axes x et y dans le plan de l'image



:-----> repère inversé dans le cas d'une erreur de signe
:-----> repère dans le cas où le signe est correct

3. Estimation avec distorsion radiale

1. Méthode de Tsai

■ Étape 1

iv) Calculer les éléments restants de \mathbf{R}^c

Rappel

- \mathbf{R}^c est une matrice de rotation
 - ✓ On s'arrange pour que les normes des lignes 1 et 2 soient = 1
 - ✓ La 3^e ligne doit être orthogonale aux deux autres lignes
 - ➡ produit vectoriel

$$R_{13}^c = \sqrt{1 - (R_{11}^c)^2 - (R_{12}^c)^2}$$

$$R_{23}^c = S \sqrt{1 - (R_{21}^c)^2 - (R_{22}^c)^2} \quad \text{avec } S = -\text{sign}(R_{11}^c R_{21}^c + R_{12}^c R_{22}^c)$$

$$R_{31}^c = \frac{1 - R_{11}^2 - R_{12} R_{21}}{R_{13}}$$

$$R_{32}^c = \frac{1 - R_{21} R_{12} - R_{22}^2}{R_{23}}$$

$$R_{33}^c = \sqrt{1 - R_{31}^2 R_{13}^2 - R_{32}^2 R_{23}^2}$$

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 2

- i) Calcul approximatif de z' et T_z^c

On utilise l'équation 4.3 avec $Z_s = 0$

$$y_d = z' \left(\frac{R_{21}^c X_s + R_{22}^c Y_s + T_y^c}{R_{31}^c X_s + R_{32}^c Y_s + T_z^c} \right)$$

pour bâtir le système

$$\begin{bmatrix} Y_i & -y_{di} \end{bmatrix} \begin{bmatrix} z' \\ T_z^c \end{bmatrix} = W_i y_{di}$$

avec $Y_i = R_{21}^c X_{si} + R_{22}^c Y_{si} + T_y^c$ et $W_i = R_{31}^c X_{si} + R_{32}^c Y_{si}$.

- ➡ avec au moins 2 paires de points, on résout le système et on trouve une approximation
- ➡ L'approximation est due au fait qu'on ne tient pas compte de k_1 (déformation radiale)

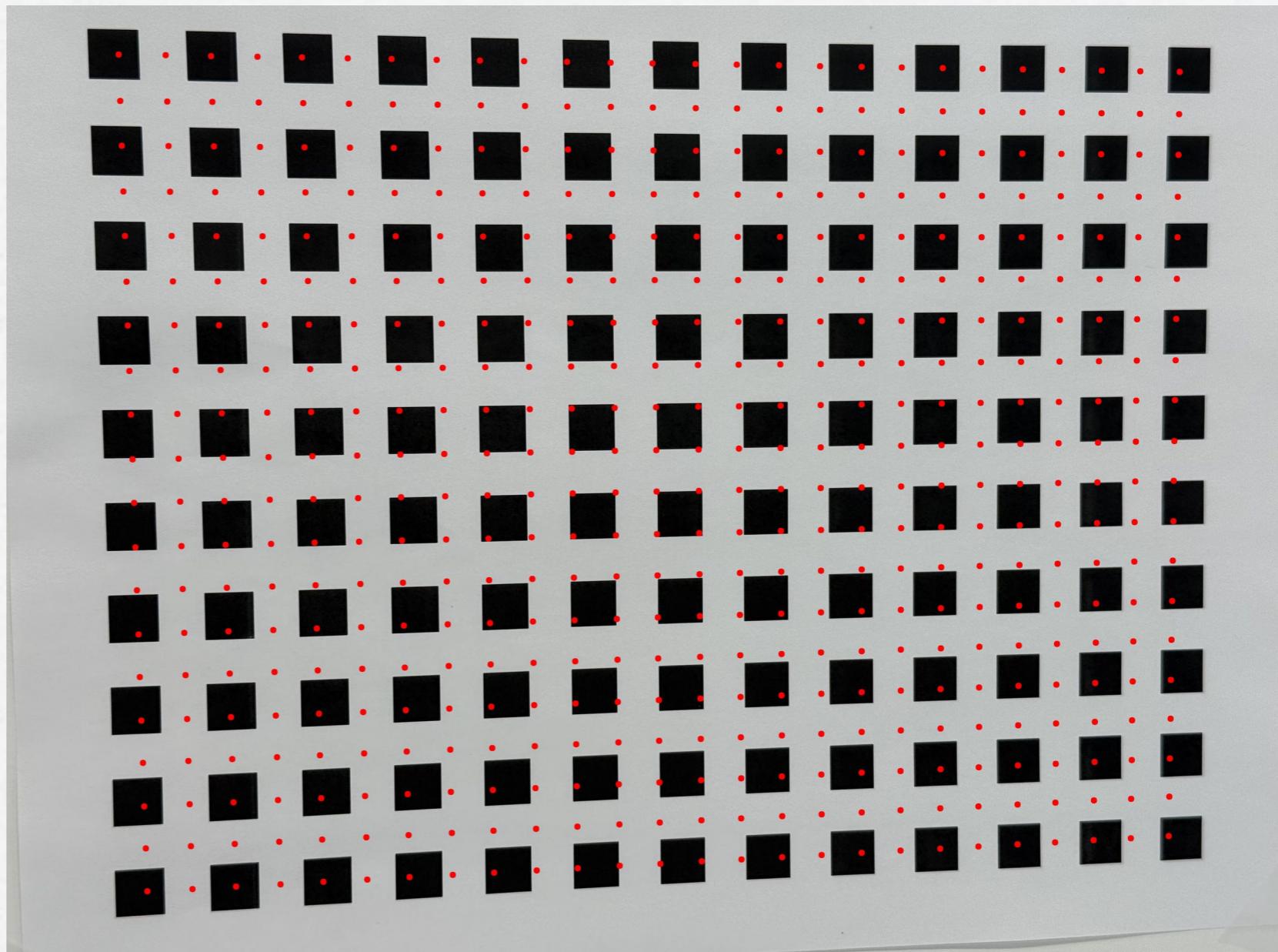
Important

- si $z' < 0$, il y a une incohérence, et il faut changer les signes de z' , T_z^c , R_{13}^c , R_{23}^c , R_{31}^c et R_{32}^c

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 2 : Vérification des résultats



3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 2

- ii) Optimisation de k_1 , z' et T_z^c

► *Rappel*

$$\mathbf{p}_c = \mathbf{p}_d(1 + k_1 r^2) \text{ avec } r^2 = x_d^2 + y_d^2 \quad \xrightarrow{\hspace{1cm}} \text{déformation radiale}$$

► La projection seule peut s'exprimer ainsi :

$$x_c = z' \left(\frac{R_{11}^c X_s + R_{12}^c Y_s + R_{13}^c Z_s + T_x^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right)$$

$$y_c = z' \left(\frac{R_{21}^c X_s + R_{22}^c Y_s + R_{23}^c Z_s + T_y^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right)$$

3. Estimation avec distorsion radiale

1. Méthode de Tsai

■ Étape 2

ii) Optimisation de k_1 , z' et T_z^c

➡ Ce qui nous mène à

$$x_d(1+k_1r^2) = z' \left(\frac{R_{11}^c X_s + R_{12}^c Y_s + R_{13}^c Z_s + T_x^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right)$$

$$y_d(1+k_1r^2) = z' \left(\frac{R_{21}^c X_s + R_{22}^c Y_s + R_{23}^c Z_s + T_y^c}{R_{31}^c X_s + R_{32}^c Y_s + R_{33}^c Z_s + T_z^c} \right)$$

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 2

- ii) Optimisation de k_1 , z' et T_z^c

Puisque $Z_s = 0$, on a les équations

$$x_{di} (1 + k_1 r_i^2) = \left(\frac{R_{11}^c X_{si} + R_{12}^c Y_{si} + T_x^c}{R_{31}^c X_{si} + R_{32}^c Y_{si} + T_z^c} \right) z' \quad \text{et} \quad y_{di} (1 + k_1 r_i^2) = \left(\frac{R_{21}^c X_{si} + R_{22}^c Y_{si} + T_y^c}{R_{31}^c X_{si} + R_{32}^c Y_{si} + T_z^c} \right) z'$$

Comme on n'a pas les vraies valeurs de k_1 , z' et T_z^c , il faut minimiser une fonction d'erreur quadratique :

$$E = \sum_i \left(x_{di} (1 + k_1 r_i^2) - \left(\frac{R_{11}^c X_{si} + R_{12}^c Y_{si} + T_x^c}{R_{31}^c X_{si} + R_{32}^c Y_{si} + T_z^c} \right) z' \right)^2$$

3. Estimation avec distorsion radiale

1. Méthode de Tsai

- Étape 2
 - ii) Optimisation de k_1 , z' et T_z^c
 - ✓ Comment optimiser ?
 - ➡ régression non-linéaire
 - ➡ méthodes itératives (ROP)
 - Descente du gradient
 - Levenberg-Marquardt (voir 3.2 dans ce chapitre)

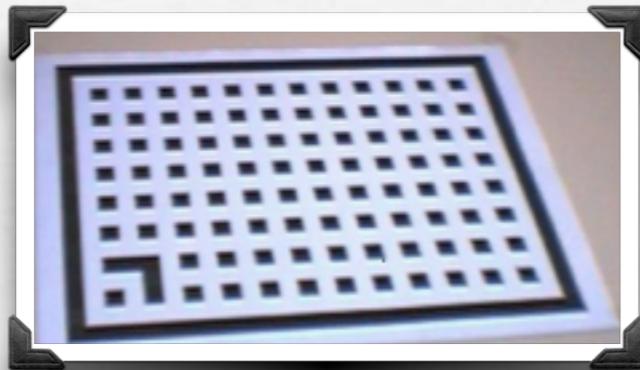
initialisation : z' et T_z^c trouvés en 2 i) et $k_1 = 0$

3. Estimation avec distorsion radiale

1. Méthode de Tsai

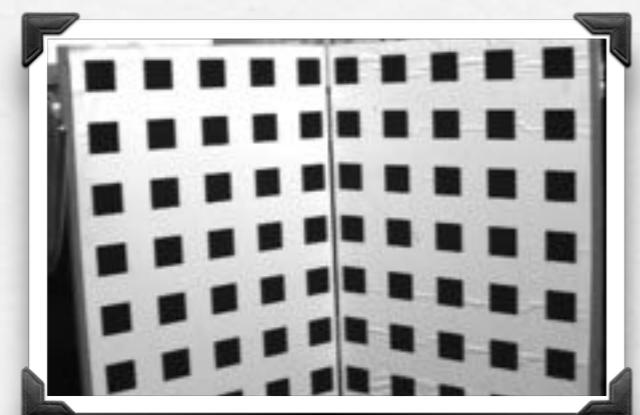
■ Remarque

- ✓ La méthode de Tsai existe en 2 versions
 - cible coplanaire



plus facile de fabriquer la cible et de déterminer (X_{s_i}, Y_{s_i})

- cible non-coplanaire



plus précise

3. Estimation avec distorsion radiale

2. Méthode d'optimisation Levenberg-Marquardt

Objectif

- ✓ Posons une fonction d'erreur $\mathbf{e}(\mathbf{p}) : \mathbb{R}^m \rightarrow \mathbb{R}^n$:
 - ➡ $\mathbf{e}(\mathbf{p}) = (e_1(\mathbf{p}), e_2(\mathbf{p}), \dots, e_n(\mathbf{p}))$.
- ✓ On veut trouver les *paramètres* $\mathbf{p} = (p_1, p_2, \dots, p_m)$ optimaux tels que
 - ➡ $Erreur(\mathbf{p}) = \|\mathbf{e}(\mathbf{p})\|^2$ est minimal

Algorithme : Levenberg-Marquardt

Entrées vecteur \mathbf{e}

première estimation des paramètres : \mathbf{p}_0

Sorties paramètres optimisés après k itérations : \mathbf{p}_k

3. Estimation avec distorsion radiale

2. Méthode d'optimisation Levenberg-Marquardt

Algorithme : Levenberg-Marquardt

// Initialisation

```
k = 0                                // itération
ν = 2                                // accélérer ou ralentir la convergence
p = p₀                                // paramètres à optimiser
A = Jt(p) J(p)                      // matrice Hessienne de Erreur(p)
g = Jt(p) e(p)                      // g correspond au gradient de e(p)
trouvé = (||g||∞ ≤ ε1)            // condition d'arrêt
μ = τ max{Aii}                      // paramètre d'amortissement
```

Précisions

- $J(\mathbf{p})$ est la matrice Jacobienne de $\mathbf{e}(\mathbf{p})$: $J_{ij}(\mathbf{p}) = \frac{\partial}{\partial p_j} e_i(\mathbf{p})$
- $\|g\|_\infty = \max\{g_1, g_2, \dots, g_n\}$



```
def calculer_jacobien(params):
    n_params = len(params)

    e0 = calculer_erreurs(params)

    J = np.zeros((len(e0), n_params))

    for j in range(n_params):
        delta = np.zeros_like(params)
        delta[j] = 0.000001
        e1 = calculer_erreurs(params + delta)
        J[:, j] = (e1 - e0) / 0.000001

    return J
```

3. Estimation avec distorsion radiale

2. Méthode d'optimisation Levenberg-Marquardt

Algorithme : Levenberg-Marquardt

// Corps

```
tant que (non trouvé) et ( $k < k_{\max}$ )           // jusqu'à conv. ou max d'itérations
     $k = k + 1$ 
     $\mathbf{h} = (\mathbf{A} + \mu \mathbf{I})^{-1} (-\mathbf{g})$           // descente du gradient
    si ( $\|\mathbf{h}\| \leq \varepsilon_2 \|\mathbf{p}\|$ )               // détermine si le changement est significatif
        trouvé = vrai
    sinon                                         // mise à jour si oui (le chang. est signif.)
         $\mathbf{p}_{\text{new}} = \mathbf{p} + \mathbf{h}$ 
         $\rho = (\text{Erreur}(\mathbf{p}) - \text{Erreur}(\mathbf{p}_{\text{new}})) / (\mathbf{h}^T (\mu \mathbf{h} - \mathbf{g}))$  // on vérifie que ça s'en va dans le bon sens
        si ( $\rho > 0$ )                                // on veut que l'erreur diminue et non augmente
             $\mathbf{p} = \mathbf{p}_{\text{new}}$ 
             $\mathbf{A} = \mathbf{J}^T(\mathbf{p}) \mathbf{J}(\mathbf{p})$ 
             $\mathbf{g} = \mathbf{J}^T(\mathbf{p}) \mathbf{e}(\mathbf{p})$ 
```

3. Estimation avec distorsion radiale

2. Méthode d'optimisation Levenberg-Marquardt

Algorithme : Levenberg-Marquardt

// Corps

// tant que (non trouvé) et ($k < k_{\max}$) (suite)

// sinon (suite)

// si ($\rho > 0$) (suite)

$$(\|\mathbf{g}\|_\infty \leq \varepsilon_1)$$

$$\mu = \mu \max\{1/3, 1 - (2\rho - 1)^3\}$$

$$\nu = 2$$

sinon

$$\mu = \mu \nu$$

$$\nu = 2 \nu$$

// paramètres différents pour la vitesse et la descente de gradient

fin si

fin si

fin tant que

3. Estimation avec distorsion radiale

2. Méthode d'optimisation Levenberg-Marquardt

- Utilisation dans le cadre du calibrage de Tsai

$$\mathbf{p} = (k_1, z', T_z)$$

$$e_i(\mathbf{p}) = \underbrace{x_{d_i} (1 + k_1 r_i^2) (R_{31} X_{s_i} + R_{32} Y_{s_i} + T_z)}_{\text{erreur pour chaque paire correspondante } (n)} - \underbrace{(R_{11} X_{s_i} + R_{12} Y_{s_i} + T_x) z'}_{\text{idéalement devrait } = 0}$$

$$\mathbf{e} = (e_1, e_2, \dots, e_n)$$

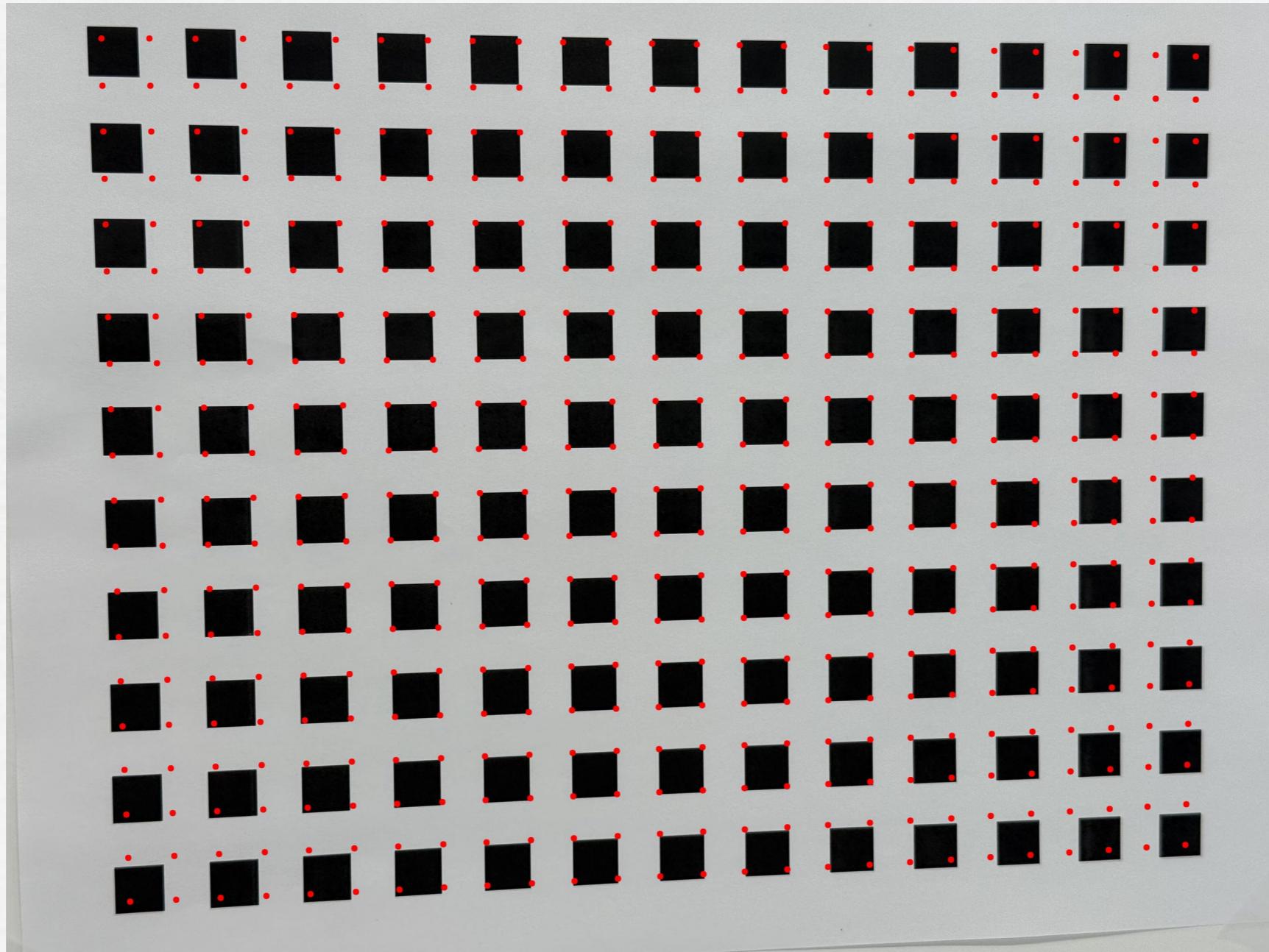
vecteur d'erreurs

erreur pour chaque paire correspondante (n)

idéalement devrait = 0

3. Estimation avec distorsion radiale

Vérifier la solution





Plan du chapitre

1. Modèle de calibrage

2. Estimation sans distorsion radiale

3. Estimation avec distorsion radiale

4. Autocalibrage

4.1. Principes

4.2. Extraction des lignes et points de fuite

4.3. Trouver la matrice de projection de la caméra

4.4. Position de la caméra

4. Autocalibrage

- *Problème* : Le calibrage avec modèle est limité
 - ✓ cadre contraignant
 - ✓ modification du montage plus difficile
 - ➡ il faut refaire le calibrage à chaque modification
 - ✓ on ne peut pas utiliser d'images provenant de sources non-calibrées

- L'autocalibrage est beaucoup basé sur la géométrie projective
 - ✓ Une vue
 - ✓ Deux vues
 - ✓ Plusieurs vues
 - ✓ Mouvement



Source : R. Cipolla et al. Camera calibration from vanishing points in images of architectural scenes. BMVC, 1999

Historique et faits à propos de ce bâtiment

Cet édifice est de style architectural Second empire. Il fut détruit en 1959 pour faire place à un agrandissement de la Banque de Montréal. Disons qu'à cette époque, l'on se ne souciait guère des édifices ayant une valeur patrimoniale.

Certains livres donnent 1876 pour l'année de construction. C'est celle du livre "Montréal perdu" qui a été retenue.

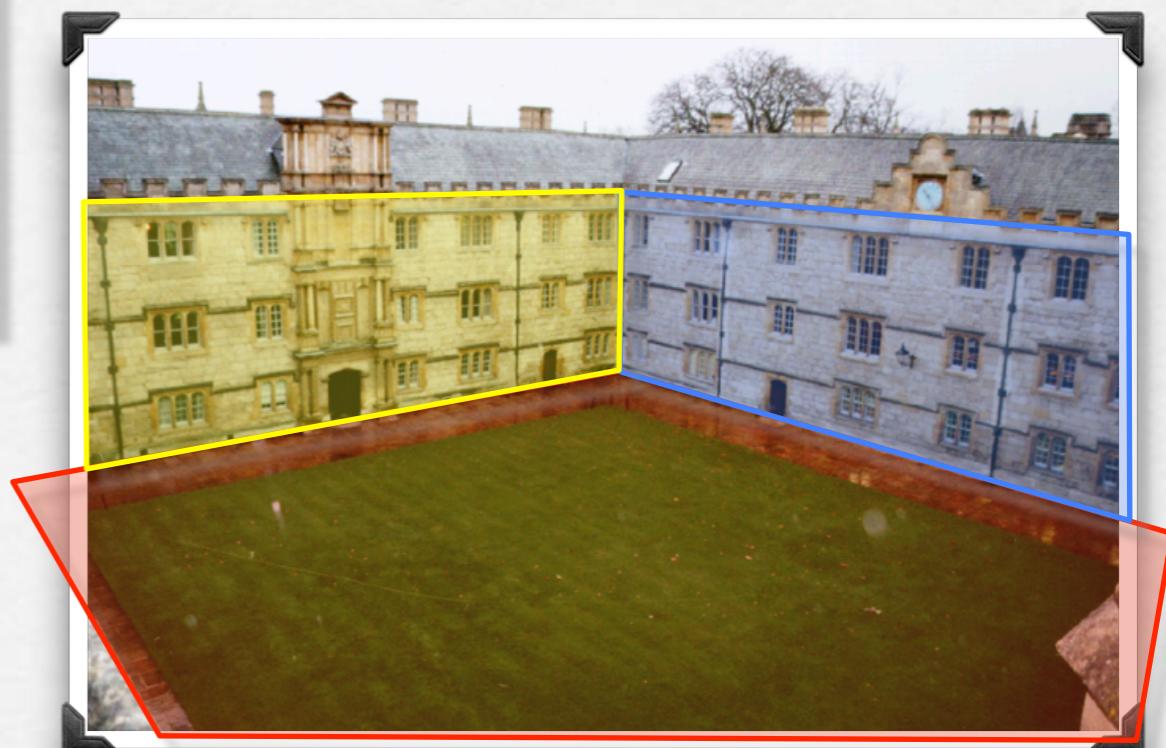
Il était situé sur le site du premier édifice de la Banque de Montréal qui avait été construit en 1817.

4. Autocalibrage

1. Principes

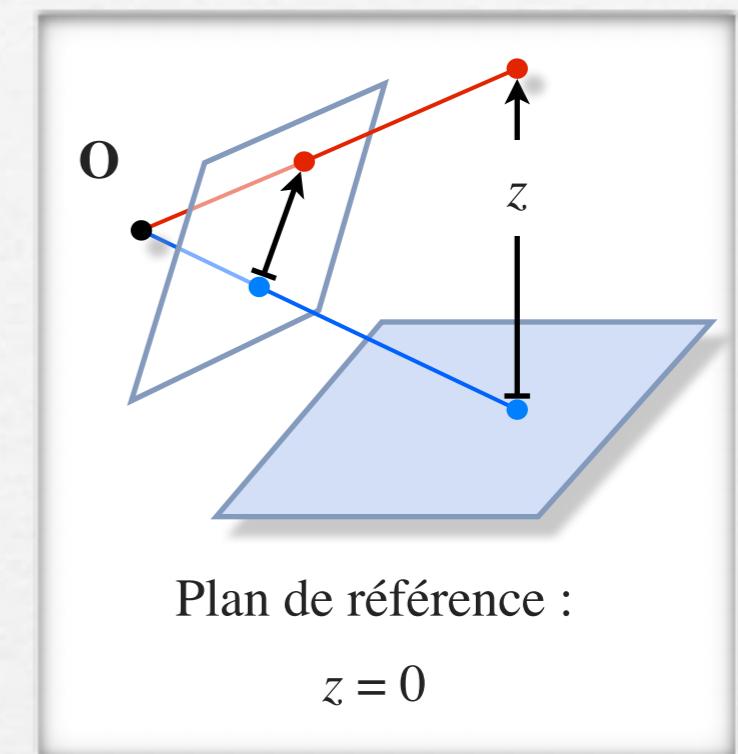
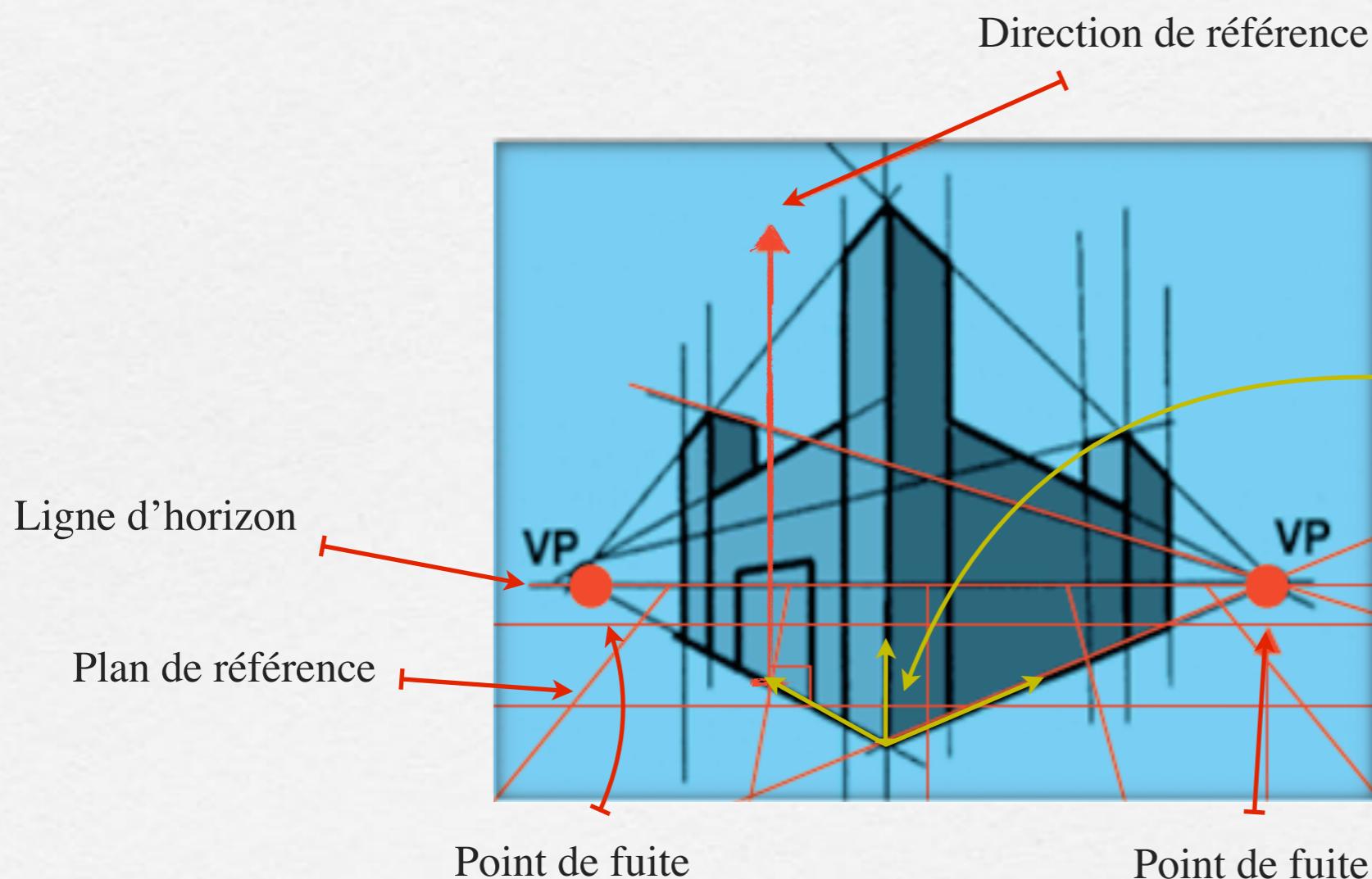
Objectifs

- Exploiter les informations présentes dans l'image pour éviter l'utilisation d'un modèle de calibrage
 - relations entre les points de fuite
 - axes principaux (plans perpendiculaires)
- Permet aussi de trouver
 - la hauteur des objets
 - calcul de la profondeur



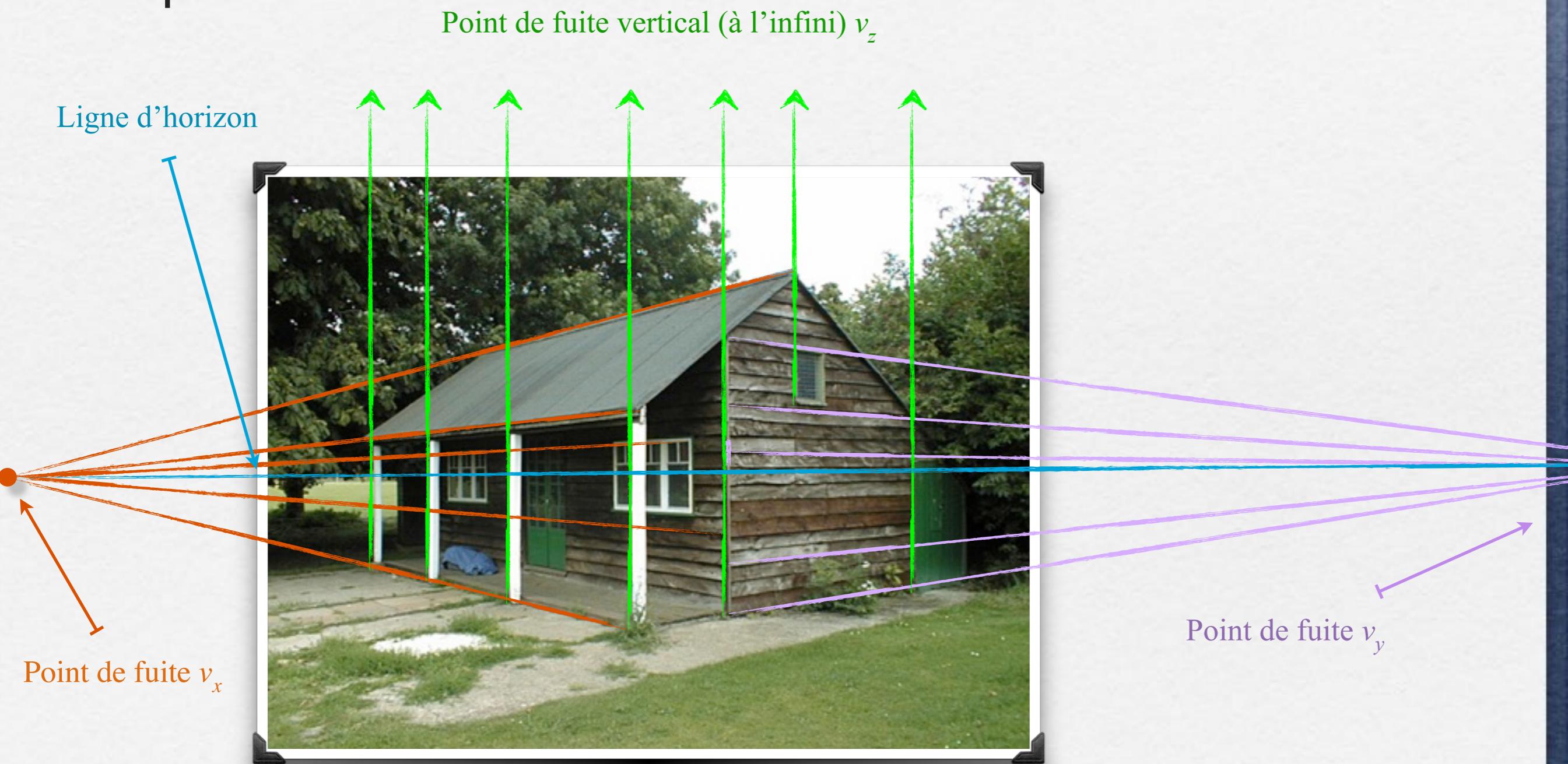
4. Autocalibrage

1. Principes



4. Autocalibrage

1. Principes



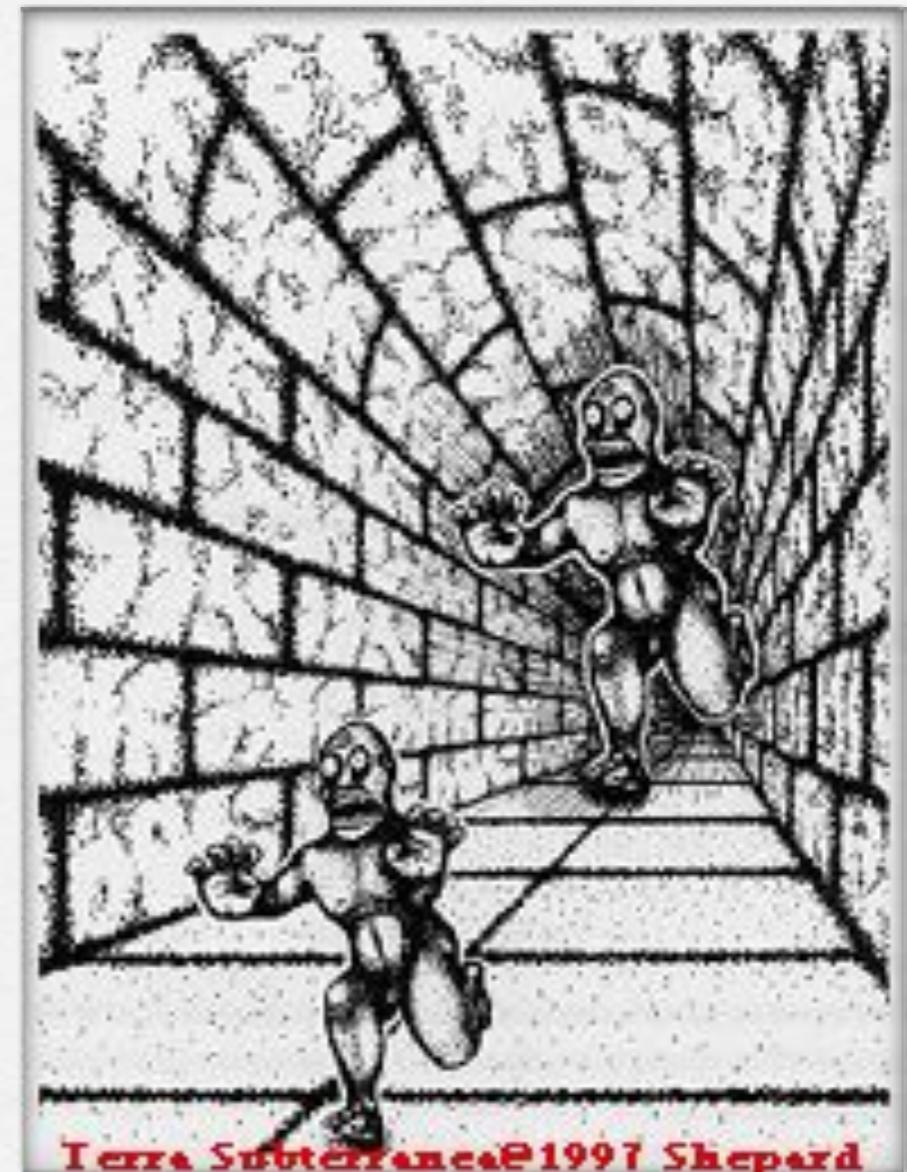
Courtoisie de Kenton Anderson

4. Autocalibrage

2. Extraction des lignes et points de fuite

- Algorithme type (points de fuite)

1. Détection de contours / lignes
2. Regroupement des points colinéaires (i.e. lignes de fuite) (e.g. par la transformée de Hough : c.f. IMN259 - Analyse d'images)
3. Calculer les points d'intersection des paires de lignes parallèles (i.e. points de fuite) par
 - a. Points d'accumulation OU
 - b. RaNSaC (plus de détails au chapitre 6)



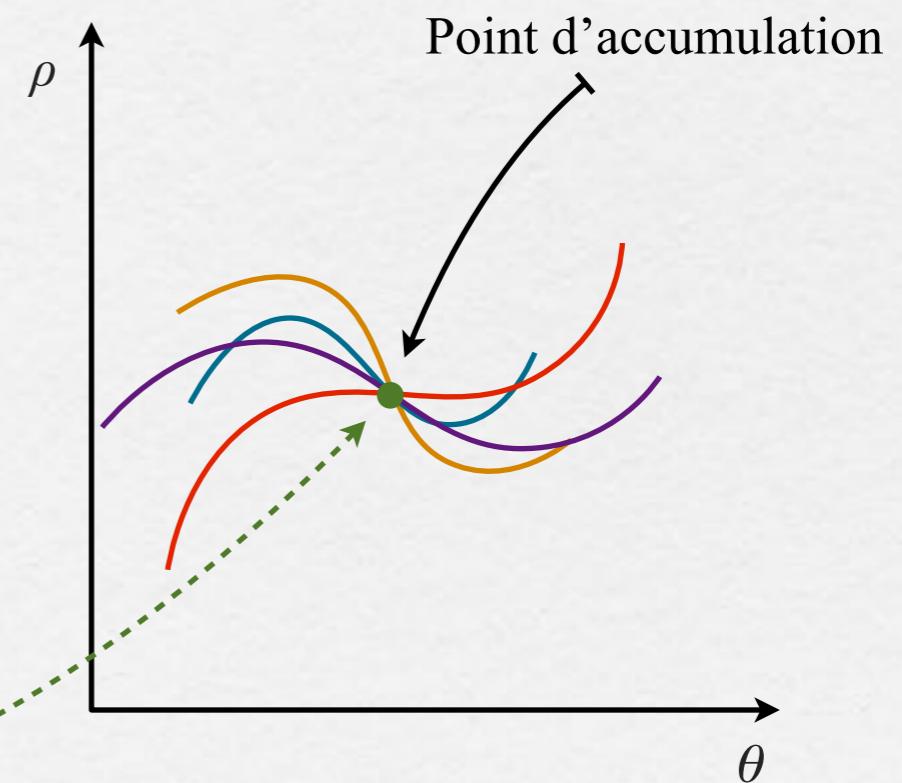
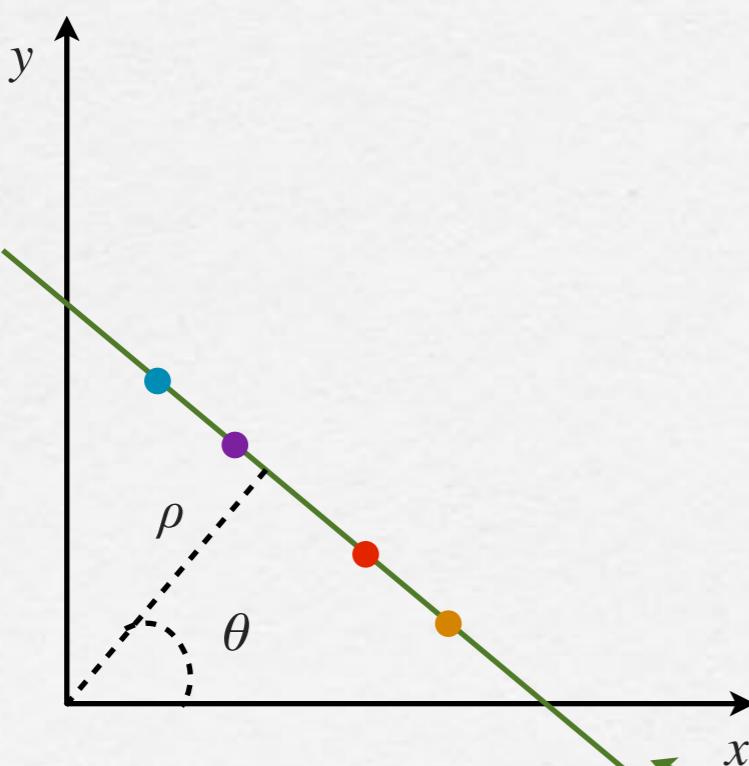
4. Autocalibrage

2. Extraction des lignes et points de fuite

- *Rappel* : Comment la transformée de Hough peut être utilisée pour trouver les paramètres des lignes

Idée

- Chaque courbe sinusoïdale dans l'espace $\theta \rho$ correspondant aux points (x, y) d'une même droite va se recouper en un point unique dans l'espace $\theta \rho$



4. Autocalibrage

2. Extraction des lignes et points de fuite

- Par points d'accumulation

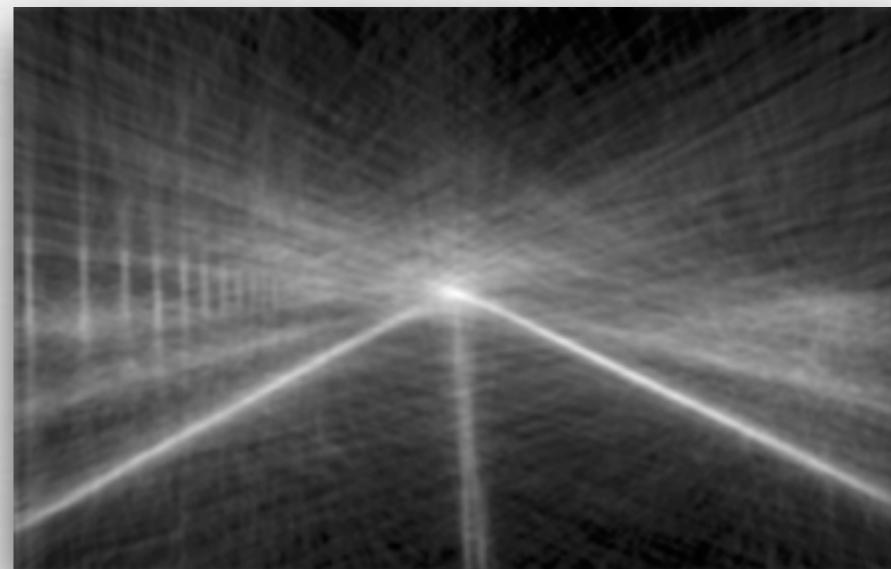
Image originale



Droites trouvées par la transformée de Hough



Points d'accumulation



4. Autocalibrage

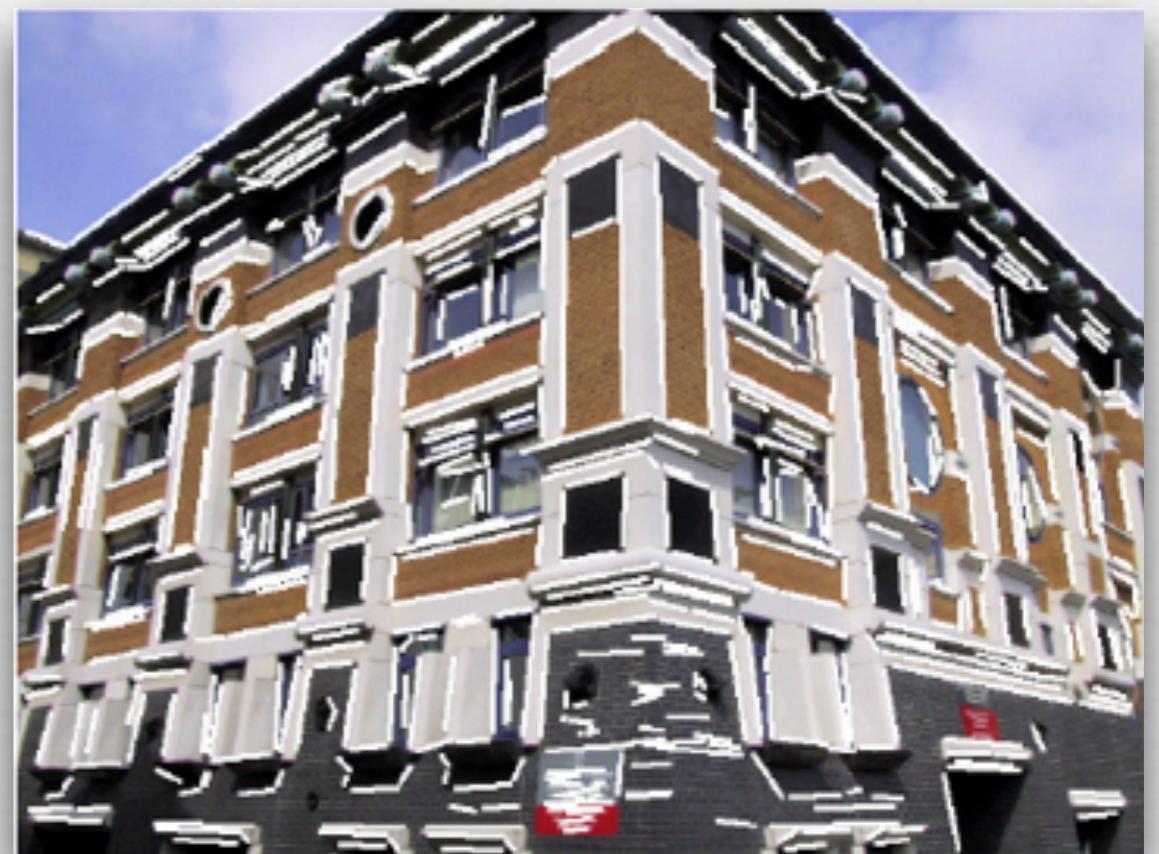
2. Extraction des lignes et points de fuite

- Par RaNSaC

RaNSaC sera vu au chap. 6



Image originale

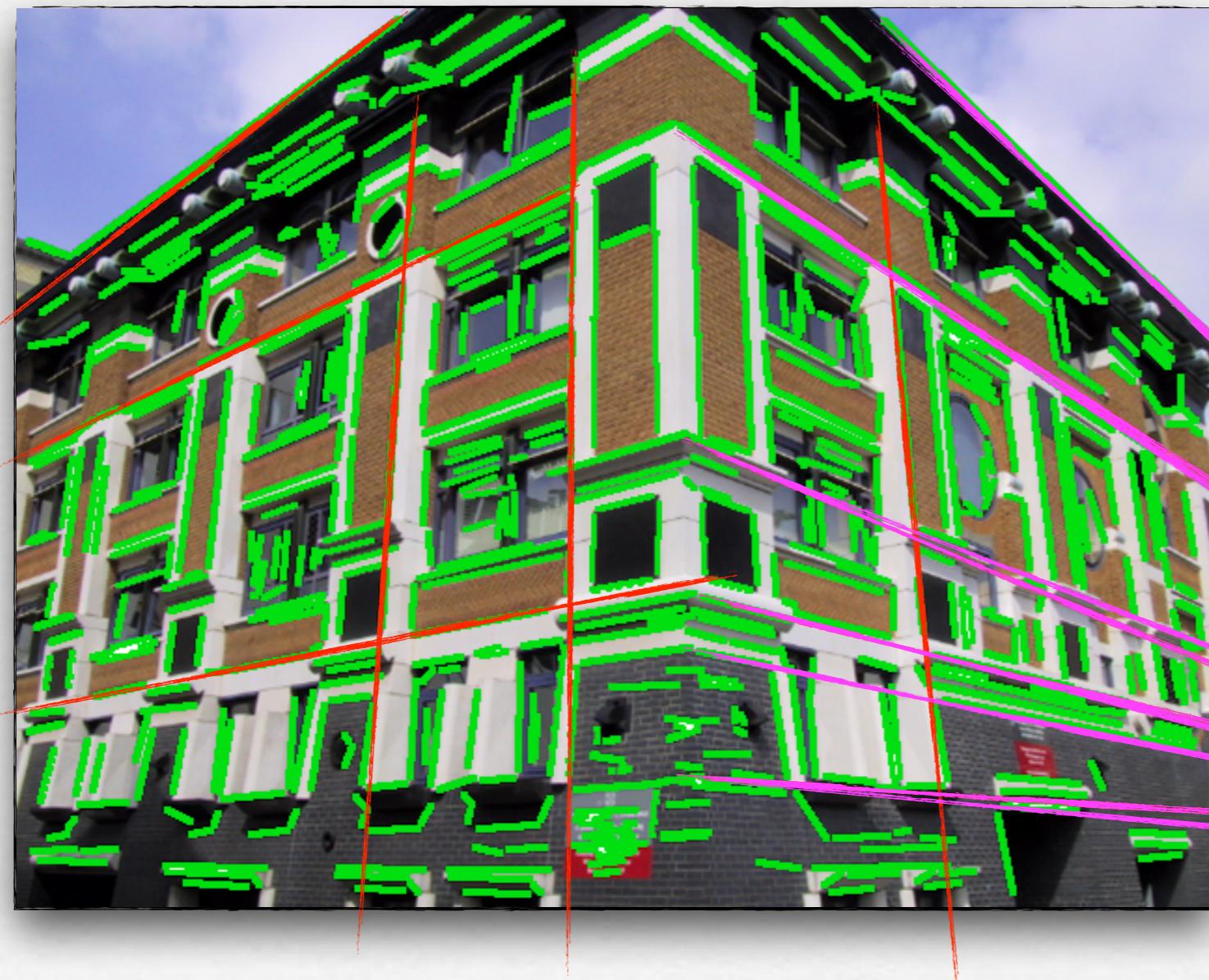


Détection des droites

4. Autocalibrage

2. Extraction des lignes et points de fuite

- Par RaNSaC



Courtoisie de Kenton Anderson

4. Autocalibrage

2. Extraction des lignes et points de fuite

- Par RaNSaC

Les 3 couleurs donnent les 3 orientations



4. Autocalibrage

3. Trouver la matrice de projection de la caméra

Considérons un point de la scène en coordonnées homogènes $P_s = (X_s, Y_s, Z_s, 1)$. Soit M la matrice de projection telle que $p = \lambda M P_s \equiv M P_s$ avec $M_{3 \times 4} = [M_1 \ M_2 \ M_3 \ M_4]$, où M_i est la i -ième colonne de M .

Prenons les trois premiers vecteurs de la base canonique de \mathbb{P}^3 qui représentent les directions des axes X , Y , et Z . Leur projection dans l'image se trouve à être les points de fuite des trois axes principaux :

$$v_x \equiv M \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad v_y \equiv M \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad v_z \equiv M \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

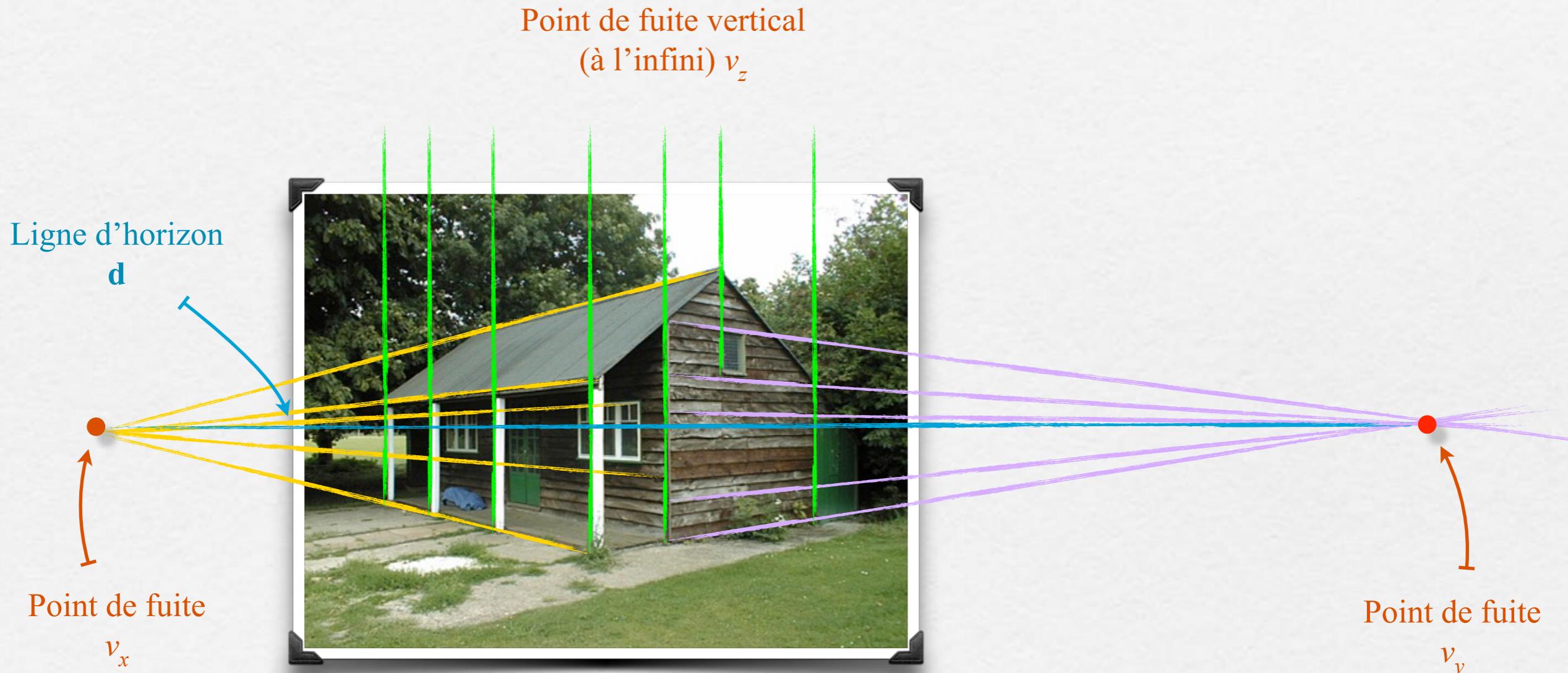
De plus, on a la ligne d'horizon qui correspond à la projection du plan à l'infini :

$$d = v_x \times v_y \equiv M \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

On peut alors valider que $M_1 = v_x$ $M_2 = v_y$ $M_3 = v_z$ $M_4 = \frac{d}{|d|}$ (facteur d'échelle arbitraire)

4. Autocalibrage

3. Trouver la matrice de projection de la caméra



4. Autocalibrage

3. Trouver la matrice de projection de la caméra

■ *Exemple*

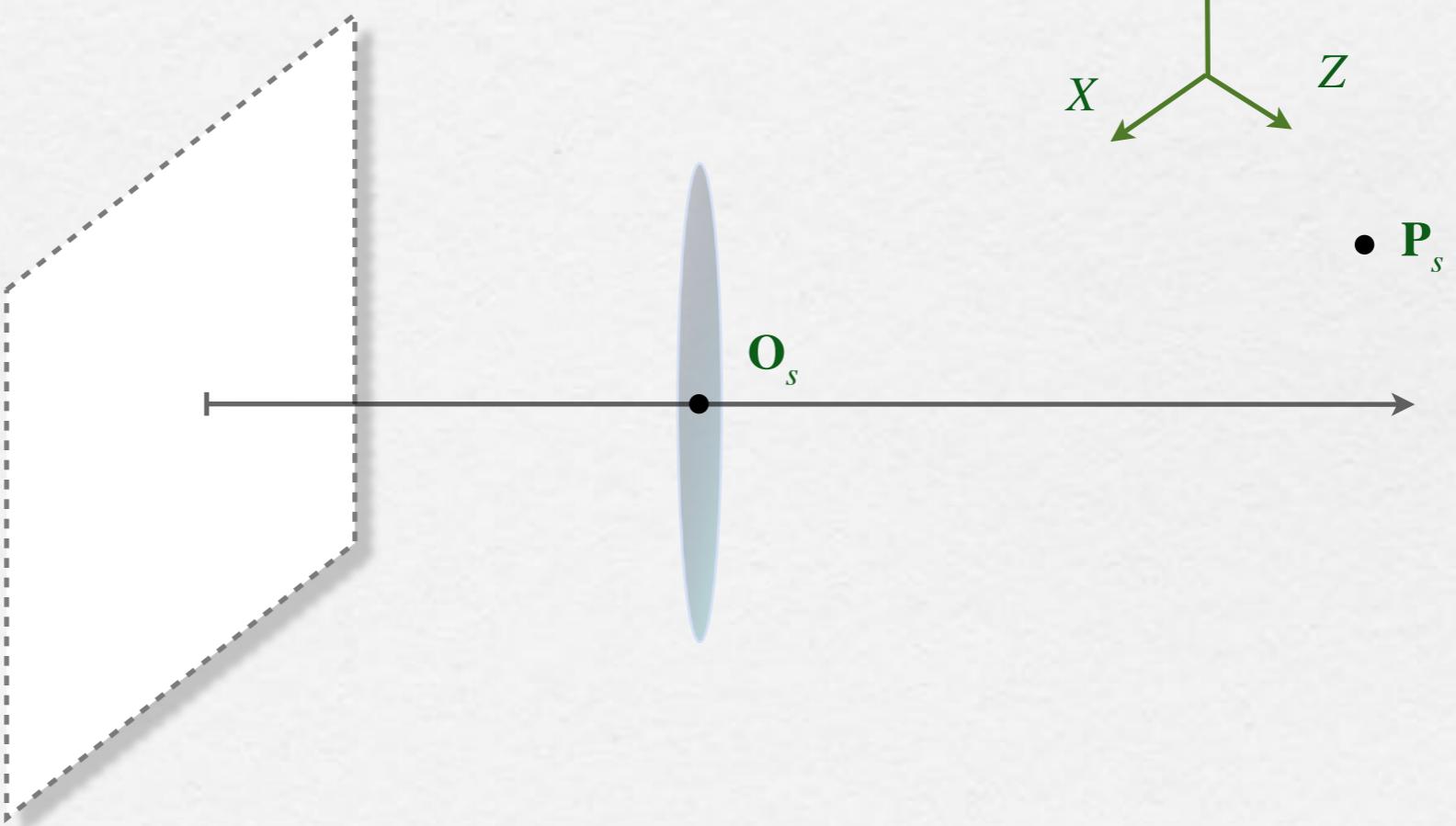


4. Autocalibrage

4. Position de la caméra

Objectif

- ✓ On veut déterminer \mathbf{O}_s , la position du centre optique dans la scène



Repère de la scène

4. Autocalibrage

4. Position de la caméra

Considérons le centre optique de la caméra dans $\mathbb{P}^3 : O_s = (X_o, Y_o, Z_o, T_o)$. On sait que si on projette ce dernier sur lui-même, on trouve

$$MO_s = [M_1 \ M_2 \ M_3 \ M_4] \begin{bmatrix} X_o \\ Y_o \\ Z_o \\ T_o \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

On peut réécrire la dernière équation sous la forme

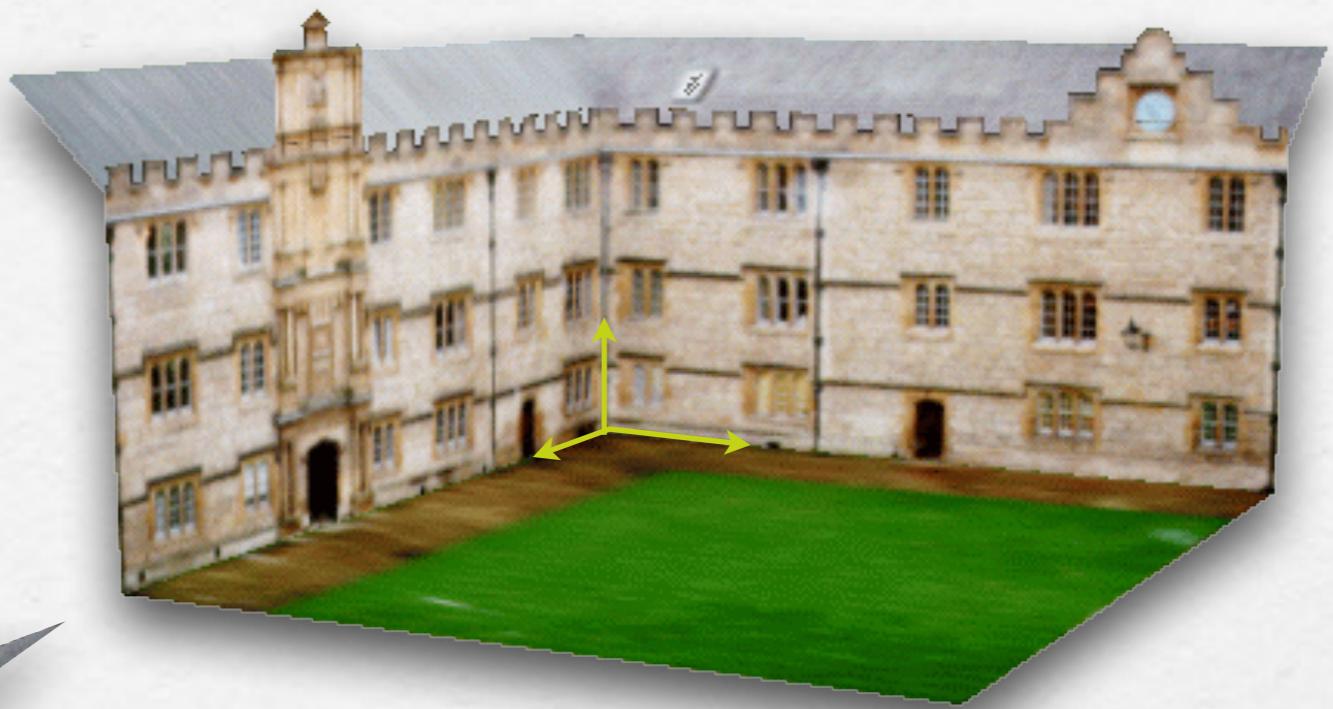
$$[M_1 \ M_2 \ M_3] \begin{bmatrix} \frac{X_o}{T_o} \\ \frac{Y_o}{T_o} \\ \frac{Z_o}{T_o} \end{bmatrix} = -M_4$$

Par la règle de Cramer, on peut ensuite montrer que la position de la caméra est donnée par

$$X_o = -\det[M_2 \ M_3 \ M_4] \quad Y_o = \det[M_1 \ M_3 \ M_4] \quad Z_o = -\det[M_1 \ M_2 \ M_4] \quad T_o = \det[M_1 \ M_2 \ M_3]$$

4. Autocalibrage

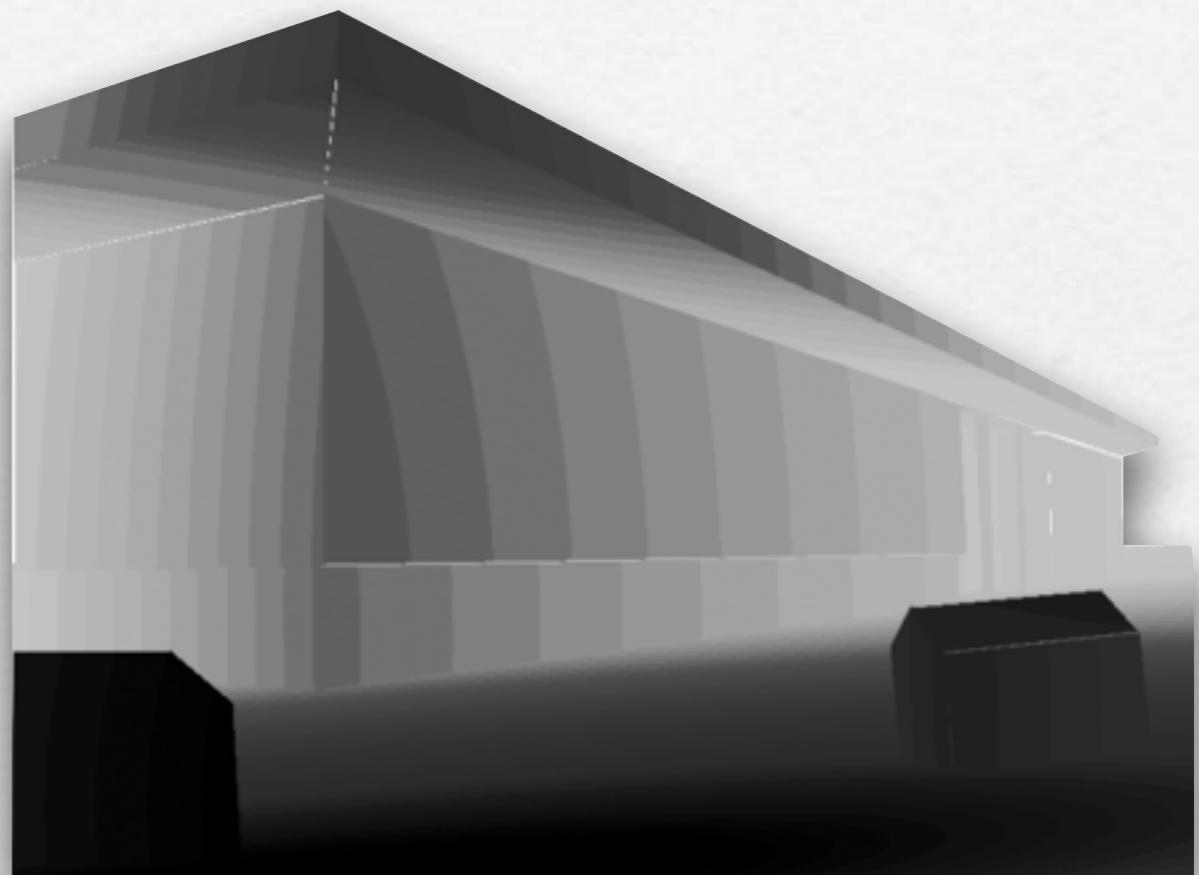
4. Position de la caméra



4. Autocalibrage

4. Position de la caméra

- Reconstruction à partir d'une seule vue



4. Autocalibrage

4. Position de la caméra

- Nouvelles vues



Références

Pour en savoir plus ...

- **R. Tsai**
A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 4, 1987
- **B.K.P. Horn**
Tsai's Camera Calibration Method Revisited. 2000
- **L.G. Shapiro and G.C. Stockman**
Computer Vision. Prentice Hall, 2001
- **E. Trucco et A. Verri**
Introductory Techniques for 3-D Computer Vision. Prentice Hall, 1998
- **R. Horaud et O. Monga**
Vision par ordinateur. 2nd édition, Hermès, 1995
- **R. Hartley and A. Zisserman**
Multiple View Geometry in Computer Vision. Cambridge University Press, 2003
- **K. Madsen, H.B. Nielsen, O. Tingleff**
Methods for Non-linear Least Squares Problems 2nd ed., Technical University of Denmark

Références

Pour en savoir plus ...

- ▶ **S. Tabbone**
Detecting Junction Using Laplacian of Gaussian Operator. Procs of the 12th International Conference on Pattern Recognition, Israel, 1994
- ▶ **D. Lowe**
Object recognition from local scale-invariant features. Procs of the International Conference on Computer Vision, Vol. 2, 1999
- ▶ **K. Derpanis**
The Harris Corner Detector. 2004
- ▶ **K. Madsen, H.B. Nielsen, O. Tingleff**
Methods for Non-linear Least Squares Problems. 2nd Edition, April 2004
- ▶ **V. Cantoni, L. Lombardi, M. Porta and N. Sicard**
Vanishing Point Detection: Representation Analysis and New Approaches, in Proc. 11th International Conference on Image Analysis and Processing, Italy, 2001
- ▶ **A. Criminisi, I. Reid and A. Zisserman**
Single View Metrology, International Journal on Computer Vision, 40(2), 2000
- ▶ **E. Lutton, H. Maitre and J. Lopez-Krahe**
Contribution to the Determination of Vanishing Points Using Hough Transform, IEEE Transaction on Pattern Analysis and Machine Intelligence, 21(3):282-288, 1999