Analyse et Reconnaissance d'Images

James L. Crowley

M2R IVR — Premier Semestre 2007/2008

Séance 1 — 8 Octobre 2007

Formation d'Image :
Coordonnées Homogènes et Modèles Projectifs des Caméras

Plan de la Séance :

Coordonnées Homogènes en Notation Tensorielle

Les coordonnées homogènes repose sur une notation dans laquelle les vecteurs en N dimensions sont représentées par un vecteur en N+1 dimensions.

Les coordonnées homogènes sont un outil de base en vision, en robotique et en synthèse d'images.

Exemple : un point

Soit un plan Euclidienne en R^2 composé de points,

En notation classique, un point est un vecteur : $\vec{P} = \begin{pmatrix} x \\ y \end{pmatrix}$

En notation homogène, un point est un vecteur $\vec{P} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

En coordonnées homogènes, les points sont invariant au multiplication par un constant.

$(i, j, 1)^T = a \cdot (i, j, 1)^T = (ai, aj, a)^T$

On note que a, b : $a \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = b \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

Vecteur et Matrice en Notation Tensorielle

En notation tensorielle, la signe " $\rightarrow$ " est remplacé par un indice en super-scripte ou sous-scripte. Une super-scripte signifie un vecteur colonne. Par exemple, le point est indiqué par un vecteur p^i

$$P^i = \begin{pmatrix} p^1 \\ p^2 \\ p^3 \end{pmatrix}$$

une sous-scripte indique un vecteur ligne
Par exemple, la droite est indiquée par le vecteur l_i :

$$L_i = (l_1, l_2, l_3)$$

Une matrice est une ligne de vecteurs (ou une vecteurs de lignes).

$$M_i^j = \begin{pmatrix} m_1^1 & m_2^1 & m_3^1 \\ m_1^2 & m_2^2 & m_3^2 \\ m_1^3 & m_2^3 & m_3^3 \end{pmatrix}$$

Le produit d'un vecteur avec une indice en super scripte et une vecteur avec le même indice en sous-scripte signifie une produit scalaire. Ceci est fait par une sommation implicite des indices. (La Convention de Sommation d'Einstein).

Donc, le produit indique une annulation des souscriptes et superscriptes.

$$L_i P^i = l_1 p^1 + l_2 p^2 + l_3 p^3$$

Cette sommation et commutative

$$L_i P^i = P^i L_i$$

Pour la produit d'une matrice et une vecteur, ceci donne une nouvelle vecteur.

$$P^j = M^j_i P^i$$

Ceci représent une tranformation du repère "i" vers le repère "j".

L'équation d'une droite

Dans un plan Euclidienne en R^2, en notation "classique", une droite est définie par une équation

$$a x + b y + x = 0.$$

On peut exprimer cette équation comme la produit de deux vecteurs :

$$\vec{L} \cdot \vec{P} = 0$$

ou $\vec{L} = (a \ \ b \ \ c)$ et $\vec{P} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

En notation tensorielle, cette equation est exprimé :

$$L_i P^i = 0 \qquad \text{avec } i=1, 2, 3.$$

La sommation des indices est implicites.

Le produit Croisé

Une droite est définie par deux points. Un point est défini par le croisement de deux droites. Il y a une dualité parfaite entre les points et les droites.

En notation classique, $ax + by + c = 0$
le manière de déterminer la droite pour deux points est :

où $a= (y_1 - y_2)$ $b= (x_2 - x_1)$

$$
\begin{aligned}
c &= -(a\, x_1 + b\, y_1) = - x_1(y_1 - y_2) - y_1(x_2 - x_1) \\
&= - x_1 y_1 + x_1 y_2 - y_1 x_2 + y_1 x_1 \\
&= x_1 y_2 - y_1 x_2
\end{aligned}
$$

La droite est $x (y_2 - y_1) + y (x_1 - x_2) - x_1 y_2 + y_1 x_2 = 0$

Ceci peut être calculé par la déterminante, avec les variables libres dans le premier colonne :

$$
\begin{vmatrix} x & y & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{vmatrix} = x(y_1-y_2) + y(x_2-x_1) + x_1y_2 - x_2y_1 = 0
$$

donc $a= (y_1 - y_2)$ $b= (x_2 - x_1)$ $c = x_1 y_2 - y_1 x_2$

Il s'agit d'une méthode générale de déterminer les paramètres d’une équation linéaire à partir des contraintes. Ca marche aussi pour trouver la point d'intersection de deux lignes.

On peut, également, écrire la déterminante comme un produit croisé.

$$
L^T = \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \vec{P}_1 \times \vec{P}_2 = \begin{pmatrix} 0 & -1 & y_1 \\ 1 & 0 & -x_1 \\ -y_1 & x_1 & 0 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = \begin{pmatrix} y_1-y_2 \\ x_2-x_1 \\ x_1y_2-x_2y_1 \end{pmatrix}
$$

Intersection de deux droites

Pour le calcul d'un point d'intersection de deux droites.

soit deux droites : L: $a_1x + b_1y + c_1 = 0$ et M : $a_2x + b_2y + c_2 = 0$.

En notation classique : Soit $\vec{L} = (a_1\ b_1\ c_1)$ et $\vec{M} = (a_2\ b_2\ c_2)$

Le point d'intersection est $x = \dfrac{b_1f - c_1b_2}{a_1b_2 - b_1a_2}$ $y = \dfrac{c_1a_2 - a_1c_2}{a_1b_2 - b_1a_2}$

Pour le demontrer :

$$\vec{P} = \vec{L} \times \vec{M} = \begin{pmatrix} 0 & -c_1 & b_1 \\ c_1 & 0 & -a_1 \\ -b_1 & a_1 & 0 \end{pmatrix} \begin{pmatrix} a_2 \\ b_2 \\ c_2 \end{pmatrix} = \begin{pmatrix} b_1c_2 - c_1b_2 \\ c_1a_2 - a_1c_2 \\ a_1b_2 - a_2b_1 \end{pmatrix} = \begin{pmatrix} \dfrac{b_1c_2 - c_1b_2}{a_1b_2 - a_2b_1} \\ \dfrac{c_1a_2 - a_1c_2}{a_1b_2 - a_2b_1} \\ 1 \end{pmatrix}$$

Le operateur "x" est équivalent à une déterminant.
Un point est l'intérsection d'une infini de droites.
Soit deux droites (a_1, b_1, c_1) et (a_2, b_2, c_2).
et soit une droit "variable" avec coéfficients a, b, c :

$$a \cdot x + b \cdot y + c = 0.$$

On peut trouver les coordonées du point grace au déterminant.

$$\begin{vmatrix} a & b & c \\ a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \end{vmatrix} = 0 = a(b_1f - c_1b_2) + b(c_1d - a_1c_2) + c(a_1e - b_1d) = 0$$

ou bien $$a \cdot \frac{b_1c_2 - c_1b_2}{a_1b_2 - b_1a_2} + b \cdot \frac{c_1a_2 - a_1c_2}{a_1b_2 - b_1a_2} + c = 0 = a \cdot x + b \cdot y + c$$

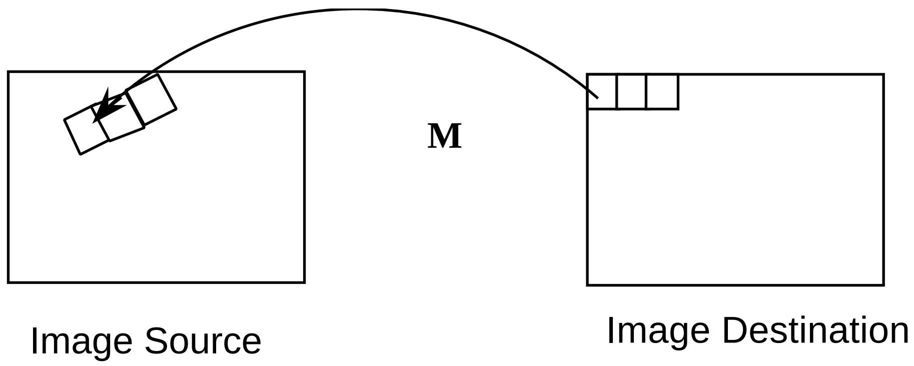
donc $$x = \frac{b_1c_2 - c_1b_2}{a_1b_2 - b_1a_2} \quad \text{et } y = \frac{c_1a_2 - a_1c_2}{a_1b_2 - b_1a_2} \quad \text{et } 1 = 1.$$

Transformations en Coordonnées Homogènes.

Les coordonnés homogènes fournissent une notation uniforme pour les transformations.

Par exemple, les transformations dans un plan sont décrites par une matrice homogène 3 x 3.

Transformations d'images



Pour chaque pixel de l'image de destination (x_d, y_d), on calcule une position (x_s, y_s) dans l'image de la source.

$$P^s = \mathbf{M}^s_d \, P^d$$

ou

$$\begin{pmatrix} p^1 \\ p^2 \\ p^3 \end{pmatrix} = \begin{pmatrix} m^1_1 & m^1_2 & m^1_3 \\ m^2_1 & m^2_2 & m^2_3 \\ m^3_1 & m^3_2 & m^3_3 \end{pmatrix} \begin{pmatrix} x_d \\ y_d \\ 1 \end{pmatrix}$$

$$x_s = \frac{p^1}{p^3} \qquad y_s = \frac{p^2}{p^3}$$

ou

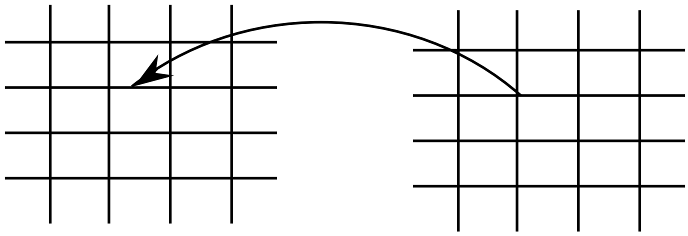
$$\begin{pmatrix} x_s \\ y_s \\ 1 \end{pmatrix} = \begin{pmatrix} p^1/p^3 \\ p^2/p^3 \\ 1 \end{pmatrix}$$

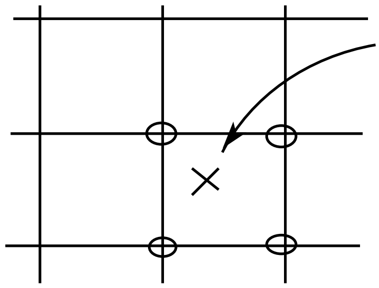
Ensuite la nouvelle valeur de pixel destination, P^2, est calculé en fonction du voisinage de la position source.

MAIS, $P^s = \begin{pmatrix} x_s \\ y_s \\ 1 \end{pmatrix}$ n'est pas des entier!

Quelle valeur faut-il prendre pour les pixels?

Pour chaque pixel du destination, (x_d, y_d) on calcul le position du source, (x_s, y_s). Ensuite, on détermine une valeur par interpolation avec les pixels voisins.





Il existe plusieurs fonctions d'interpolation.

Ordre zéro : Plus proche voisin.
Ordre unité : intérpolation linéaire et "bi-linéaire"
Ordre trois : Spline cubic.

Translation en Coordonnées Homogènes.

$x_2 = x_1 + t_x$,
$y_2 = y_1 + t_y$

$$\begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & t_x \\ 0 & 0 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix}$$

En notation tensorielle :

$$P^B = \mathbf{T}_A^B P^A \qquad \text{pour A, B = 1, 2, 3.}$$

Donc $\mathbf{T}_A^B$ est une transformation du repère A vers le repère B.

Les indices permet de noter les repères.

Rotation

(Repère main droite, rotation sens trigonométrique)
$x_2 = \mathrm{Cos}(\theta)\, x_1 + \mathrm{Sin}(\theta)\, y_1$,
$y_2 = \mathrm{Sin}(\theta)\, x_1 - \mathrm{Cos}(\theta) y_1$

$$\begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = \begin{pmatrix} \mathrm{Cos}(\theta) & \mathrm{Sin}(\theta) & 0 \\ -\mathrm{Sin}(\theta) & \mathrm{Cos}(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix}$$

Quand le repère tourne dans le sanse "θ", le vecteur est tourner dans le sense –θ

Translation et Rotation en Coordonnées Homogènes.

$$x_2 = \mathrm{Cos}(\theta)\, x_1 + \mathrm{Sin}(\theta)\, y_1 + t_x$$
$$x_2 = \mathrm{Sin}(\theta)\, x_1 - \mathrm{Cos}(\theta)\, y_1 + t_y$$

$$\begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = \begin{pmatrix} \mathrm{Cos}(\theta) & \mathrm{Sin}(\theta) & t_x \\ -\mathrm{Sin}(\theta) & \mathrm{Cos}(\theta) & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix}$$

Ceci modèlise la rotation, suivi de la translation.

En tensorielle, $P^B = \mathbf{R}_A^B\, P^A$

Transformations d'échelle, rotation et translation

Une transformation de similitude d'une image est définie par
une rotation, une translation, et un changement de taille des axes.

Soit un changement d'échelle s_x et s_y des axes x_1 et y_1 (repère source)
suivi d'une rotation de le repère d'angle θ dans le plan de l'image source,
suivi d'une translation t_x, t_y s'exprimes dans le repère de la destination.

Ces paramètres donne une transformation (s_x, s_y, θ, t_x, t_y) de

1) Un changement d'échelle des axes, puis
2) Une rotation des axes, puis
3) Une translation.

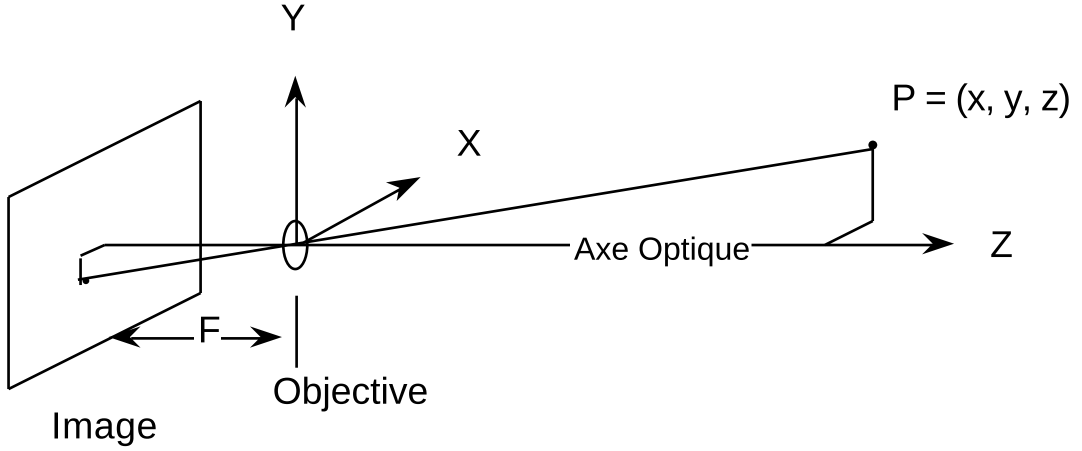
$$\begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = \begin{pmatrix} s_x \, Cos(\theta) & s_y \, Sin(\theta) & t_x \\ -s_x \, Sin(\theta) & s_y \, Cos(\theta) & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix}$$

ou bien

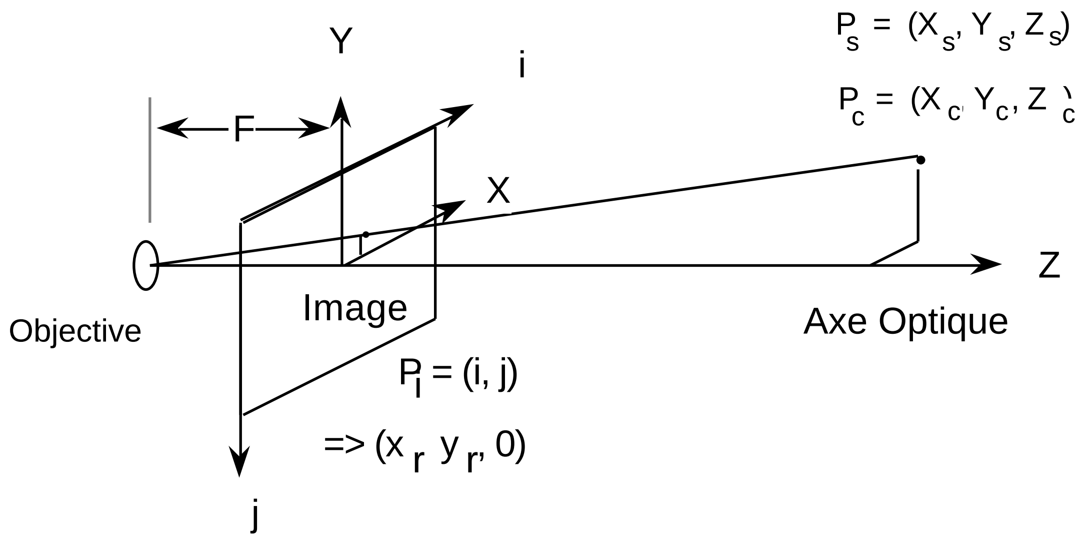
$$x_2 = s_x Cos(\theta) \, x_1 + s_y \, Sin(\theta) y_1 + t_x,$$
$$y_2 = s_y Sin(\theta) \, x_1 - s_x \, Cos(\theta) y_1 + t_y$$

Modèle de la Caméra

Modèle de la Caméra   Géométrie Physique



Modèle Mathématique: Projection Centrale



Les Repères

Coordonnées de la Scène :

Point Scène : $P^s = (x_s, y_s, z_s, 1)^T$

Coordonnées de la Caméra :

Point Caméra : $P^c = (x_c, y_c, z_c, 1)^T$

Point Image : $P^r = (x_r, y_r, 1)^T$

Coordonnées de l'Image :

Point Image : $P^i = (i, j, 1)^T$

Nota : En coordonnées homogènes, les points sont invariant au multiplication par un constant.

$$(i, j, 1)^T = w\cdot(i, j, 1)^T = (wi, wj, w)^T$$

Transformations entres reperes

La modèle de la caméra est composé d'une composition de transformations.

Transformation entre repère <u>Caméra</u> et repère <u>Scène</u> :

$$P^c = \mathbf{T}_s^c P^s$$

Matrice de Projection ${}_c^r\mathbf{P}$ du repère <u>Caméra</u> vers le repère <u>Rétine</u>

$$Q^r = \mathbf{P}_c^r P^c$$

Transformation entre repère image et repère caméra

$$Q^i = \mathbf{C}_r^i P^r$$

Composition : $Q^i = \mathbf{C}_r^i \mathbf{P}_c^r \mathbf{T}_s^c P^s = \mathbf{M}_s^i P^s$

La Transformation Scène - Caméra

Dans les coordonnées homogènes, les translations, rotations, transformées affines et perspectives sont représentées par les produits des matrices.

La transformation $\mathbf{T}_s^c$ a la forme :

$$\mathbf{T}_s^c = \begin{pmatrix} & \mathbf{R}_s^c & & \begin{matrix} x_s \\ y_s \\ z_s \end{matrix} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

ou (x_s, y_s, z_s) est la position du repère scène dans le repere caméra.
et $\mathbf{R}_s^c$ est l'orientation du repère scène dans le repère caméra.

Translation	Addition en espace cartésien Multiplications en Coordonnées Homogènes

Elle est composé de $\mathbf{T}_s^c = \mathbf{T}_\gamma^c \mathbf{R}_\phi^\gamma \mathbf{R}_\theta^\phi \mathbf{R}_q^\theta \mathbf{S}_s^q$

Changement d'echelle :

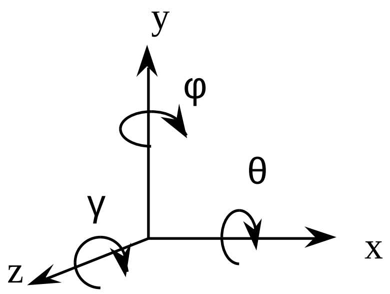
$\mathbf{S}_s^q$ est une changement d'echelle entre S est C. Elle exprime s en "c".

$$\mathbf{S}_s^q = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Rotation:

$$\mathbf{R}_\phi^\gamma \; \mathbf{R}_\theta^\phi \; \mathbf{R}_q^\theta$$

En 3D



Au tour de l'axe X :

$$\mathbf{R}_q^\theta = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \mathrm{Cos}(\theta) & \mathrm{Sin}(\theta) & 0 \\ 0 & -\mathrm{Sin}(\theta) & \mathrm{Cos}(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

l'axe Y :

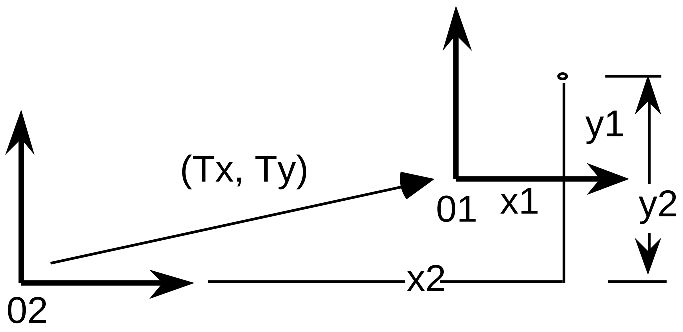
$$\mathbf{R}_\theta^\phi = \begin{pmatrix} \mathrm{Cos}(\varphi) & 0 & -\mathrm{Sin}(\varphi) & 0 \\ 0 & 1 & 0 & 0 \\ \mathrm{Sin}(\varphi) & 0 & \mathrm{Cos}(\varphi) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

l'axe Z :

$$\mathbf{R}_\phi^\gamma = \begin{pmatrix} \mathrm{Cos}(\gamma) & \mathrm{Sin}(\gamma) & 0 & 0 \\ -\mathrm{Sin}(\gamma) & \mathrm{Cos}(\gamma) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Translation:

$\mathbf{T}_{\gamma}^{c}$ es composé de la position de l'ancien repère dane le nouveau



(t_x, t_y) est la position de O_1 dans O_2. $\mathbf{T}_s^c = \begin{pmatrix} & \mathbf{R}_s^c & & x_s \\ & & & y_s \\ & & & z_s \\ 0 & 0 & 0 & 1 \end{pmatrix}$

En Générale :

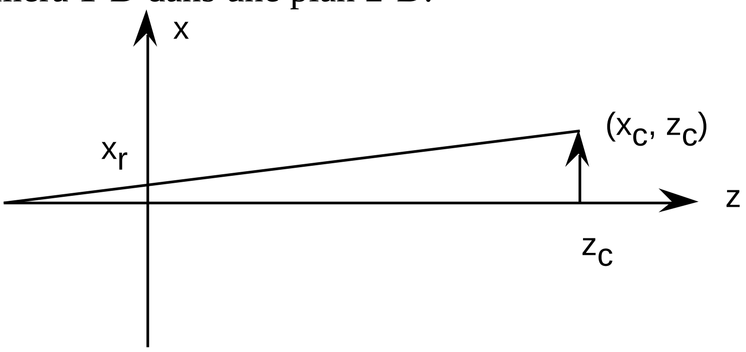
$$\mathbf{T}_s^c = \begin{pmatrix} & \mathbf{R}_s^c & & x_s \\ & & & y_s \\ & & & z_s \\ 0 & 0 & 0 & 1 \end{pmatrix} \text{ ou } \mathbf{R} = \mathbf{R}_Z(\gamma)\,\mathbf{R}_Y(\varphi)\,\mathbf{R}_X(\theta)$$

Rotation puis translation.

La Projection Caméra - Rétine (Le modèle Stenope)

La projection de la caméra vers la rétine est une projection perspective.

Considère une caméra 1-D dans une plan 2-D.



Dans le repère de la caméra :

$(x_c, y_c, z_c, 1)$: Point dans la scène en repère caméra
$(x_r, y_r, 1)$: Point dans la rétine en repère rétine.

Par triangles semblables :

$$\frac{x_r}{F} = \frac{x_c}{z_c} \Leftrightarrow x_r = x_c \frac{F}{z_c} \Leftrightarrow x_r \frac{z_c}{F} = x_c$$

$$\frac{y_r}{F} = \frac{y_c}{z_c} \Leftrightarrow y_r = y_c \frac{F}{z_c} \Leftrightarrow y_r \frac{z_c}{F} = y_c$$

soit : $w = \frac{z_c}{F}$

alors : $w\, x_r = x_c$
$w\, y_r = y_c$
$w = \frac{z_c}{F}$

En matrice :

$$\begin{pmatrix} w x_r \\ w y_r \\ w \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{F} & 0 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix}$$

La transformation du repère Caméra vers le repère Rétine est :

$$P^r = \begin{pmatrix} w x_r \\ w y_r \\ w \end{pmatrix} = \mathbf{P}_c^r \; P^c \quad \text{tel que} \quad \begin{pmatrix} x_r \\ y_r \\ 1 \end{pmatrix} = \begin{pmatrix} p^1/p^3 \\ p^2/p^3 \\ 1 \end{pmatrix}$$

et donc : $\mathbf{P}_c^r = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{F} & 0 \end{pmatrix}$

Noter que $\mathbf{P}_c^r$ n'est pas inversible .

Si l’origine est dans la rétine, par triangles semblables :

$$\frac{x_r}{F} = \frac{x_c}{(F + z_c)} \qquad \Rightarrow \qquad x_r = \frac{x_c F}{(F + z_c)}$$

Equations de perspective:

$$x_r = \frac{x_c F}{(F+z_c)} \qquad\qquad y_r = \frac{y_c F}{(F+z_c)}$$

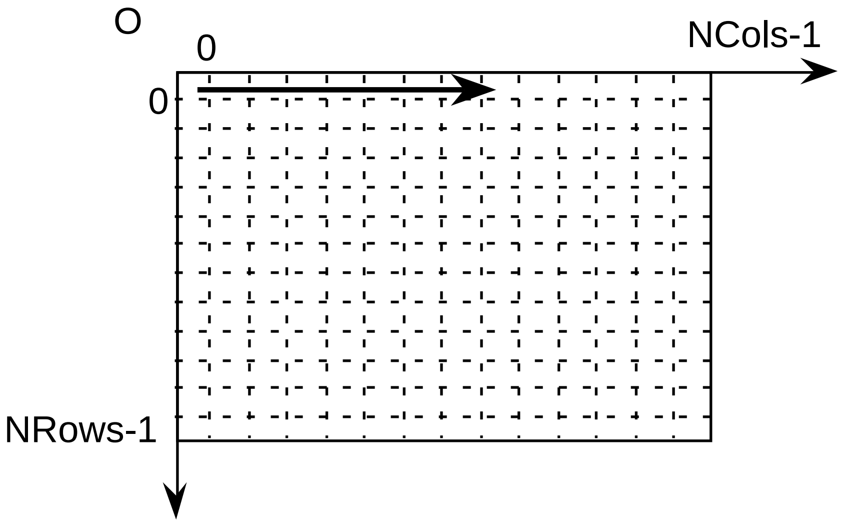
$$z_r = 0$$

et puis : $\mathbf{P}_c^r = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{F} & 1 \end{pmatrix}$

La transformation Rétine - Image

Les paramètres intrinsèques de la caméra

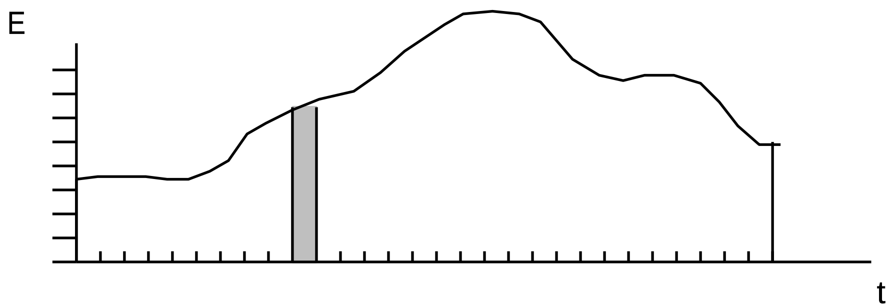
La Trame : L'image est composée des "pixels" (<u>pic</u>ture <u>el</u>ements)



Question : Est-ce-que les pixels sont "carrés"?
Réponse : non.
Le nombre de pixels de l'image dépend de la matérielle.

Par exemple : VGA : 640 x 480

<u>Echantillonage et Numérisation</u>



Les Paramètre Intrinsèque de la Caméra

F :	Distance Focale
C_i, C_j:	Centre Optique de l'Image (en pixels)
D_i, D_j :	Taille Physique d'un Pixel dans la Rétine (pixel/mm)

$i = x_r D_i$ (mm · pixel/mm) + C_i (pixel)
$j = y_r D_j$ (mm · pixel/mm) + C_j (pixel)

Transformation entre repère de l'image et repère de la rétine :

$$P^i = \mathbf{C}_r^i \; P^r$$

$$\begin{pmatrix} i \\ j \\ 1 \end{pmatrix} = \begin{pmatrix} D_i & 0 & C_i \\ 0 & D_j & C_j \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_r \\ y_r \\ 1 \end{pmatrix}$$

ou bien:

$$\begin{pmatrix} wi \\ wj \\ w \end{pmatrix} = \begin{pmatrix} D_i & 0 & C_i \\ 0 & D_j & C_j \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} wx_r \\ wy_r \\ w \end{pmatrix}$$

La Composition de la Projection Scène - Image

$$\boxed{P^i = \mathbf{C}_r^i \; \mathbf{P}_c^r \; \mathbf{T}_s^c \, P^s = \mathbf{M}_s^i \; P^s}$$

$$\begin{bmatrix} w\ i \\ w\ j \\ w \end{bmatrix} = \mathbf{M}_s^i \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix}$$

et donc

$$i = \frac{w\ i}{w} = \frac{M_s^1 \cdot P^s}{M_s^3 \cdot P^s} \qquad j = \frac{w\ j}{w} = \frac{M_s^2 \cdot P^s}{M_s^3 \cdot P^s}$$

ou bien

$$i = \frac{w\ i}{w} = \frac{M_{11} X_s + M_{12} Y_s + M_{13} Z_s + M_{14}}{M_{31} X_s + M_{32} Y_s + M_{33} Z_s + M_{34}}$$

$$j = \frac{w\ j}{w} = \frac{M_{21} X_s + M_{22} Y_s + M_{23} Z_s + M_{24}}{M_{31} X_s + M_{32} Y_s + M_{33} Z_s + M_{34}}$$

Projection inverse, du repère <u>image</u> vers le repère <u>scène</u> :

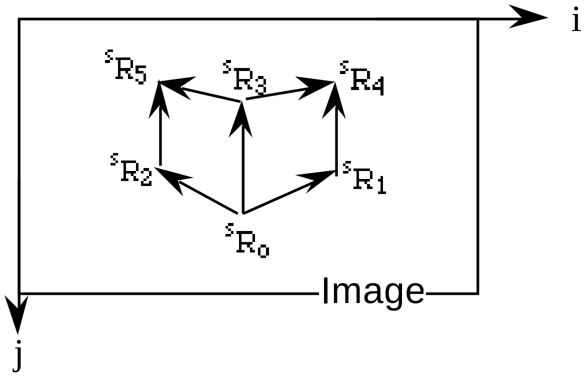
$$P^s = \mathbf{T}_c^s \; \mathbf{P}_r^c \; \mathbf{C}_i^r \; P^i$$

<u>Cependant</u>, l'inversion de la transformation perspective $\mathbf{P}_r^c$ implique la connaissance de la profondeur, z_C, pour chaque pixel.

La Calibrage

Comment obtenir $\mathbf{M}_s^i$? Par calibrage.

On définit un mire composé de points pour lequels les positions sont connues R_k^s.



La matrice $\mathbf{M}_s^i$ est composé de 3x4=12 coéfficients. Cépendant, $\mathbf{M}_s^i$ est homogène, avec rang 12-1 = 11.

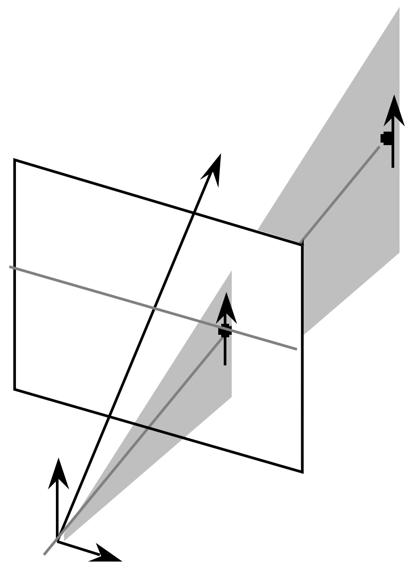
On détermine une correspondance entre les points dans le scènes (R_k^s) et leurs images (P_k^i). Chaque point donne deux équations pour les 11 inconnus.
Il faut au moins $5\frac{1}{2}$ correspondances pour les 11 inconnus.

Pour chaque point de calibrage R_k^s et sa projection P_k^s, on peut écrire :

$$i_k = \frac{w_k\ i_k}{w_k} = \frac{M_s^1 \cdot R_k^s}{M_s^3 \cdot R_k^s} \qquad j_k = \frac{w_k\ j_k}{w_k} = \frac{M_s^2 \cdot R_k^s}{M_s^3 \cdot R_k^s}$$

Avec lequel on peut déterminer 2 équations pour les 11 inconnus

$$(M_s^1 \cdot R_k^s) - i_k\,(M_s^3 \cdot R_k^s) = 0 \qquad (M_s^2 \cdot R_k^s) - j_k\,(M_s^3 \cdot R_k^s) = 0$$



L'équation $(M^1_s \cdot R^s_k) - i_k\,(M^3_s \cdot R^s_k) = 0$ est une plan passant par l'origine de la caméra est la colone $i=i_k$.

L'équation $(M^2_s \cdot R^s_k) - j_k\,(M^3_s \cdot R^s_k) = 0$ est une plan passant par l'origine de la caméra est la ligne $j=j_k$.

Avec notation tensorielle, les équations de la calibrage sont explicites.

soit $P^i = \begin{pmatrix} wi \\ wj \\ w \end{pmatrix}$ On écrit : $P^i = \mathbf{M}^i_s\ R^s$

Avec K points de la scène R^S_k et leur correspondance de l'image P^i_k on peut écrire

$$P^i_k = \mathbf{M}^i_s\ R^s_k$$

On ne connait pas P^i_k mais $i\,w = P^1_k/P^3_k$ et $j\,w = P^2_k/P^3_k$. Donc
Pour chaque point k, il y deux equations independants, i=1, 2.

$$\begin{pmatrix} p^1 \\ p^2 \\ p^3 \end{pmatrix} = \begin{pmatrix} wi \\ wj \\ w \end{pmatrix} \quad \text{donc} \quad \begin{pmatrix} i \\ j \\ 1 \end{pmatrix} = \begin{pmatrix} p^1/p^3 \\ p^2/p^3 \\ 1 \end{pmatrix}$$

et $P^3_k = \mathbf{M}^3_s R^3_k$

$$i = p^1/p^3 = \mathbf{M}^1_s\ R^s_k \,/\, \mathbf{M}^3_s\ R^s_k \qquad i\,\mathbf{M}^3_s\ R^s_k - \mathbf{M}^1_s\ R^s_k = 0$$
$$j = p^2/p^3 = \mathbf{M}^2_s\ R^s_k \,/\, \mathbf{M}^3_s\ R^s_k \qquad j\,\mathbf{M}^3_s\ R^s_k - \mathbf{M}^2_s\ R^s_k = 0$$

Pour chaque k, ceci donne deux équations :

$$\begin{pmatrix} R^1 & R^2 & R^3 & 1 & 0 & 0 & 0 & 0 & -iR^1 & -iR^2 & -iR^3 & -i \\ 0 & 0 & 0 & 0 & R^1 & R^2 & R^3 & 1 & -jR^1 & -jR^2 & -jR^3 & -j \end{pmatrix} \begin{pmatrix} M_1^1 \\ M_2^1 \\ M_3^1 \\ M_4^1 \\ M_1^2 \\ M_2^2 \\ M_3^2 \\ M_4^2 \\ M_1^3 \\ M_2^3 \\ M_3^3 \\ M_4^3 \end{pmatrix} = 0$$

Pour N points (non-coplanaires) on peut écrire un système de 2N équations :

$$\mathbf{A} \ \mathbf{M}_s^i = 0.$$

Le rank de A est 11. Le problème est de minimiser un critère :

$$\mathbf{C} = \| \mathbf{A} \ \mathbf{M}_s^i \|$$

On utilise les multiplieurs de Lagrange afin d'obtenir le $\mathbf{M}_s^i$ qui minimise **C**

Exemple : Soit un cube dont les coins sont détectés à :

$P^L{}_o = (101, 221)$ $P^L{}_1 = (144, 181)$ $P^L{}_2 = (22, 196)$

$P^L{}_3 = (105, 88)$ $P^L{}_4 = (145, 59)$ $P^L{}_5 = (23, 67)$

Par moindre de carré on obtient :

$$\mathbf{M}_s^i = \begin{pmatrix} 55.886873 & -79.292084 & 1.276703 & 101.917630 \\ -22.289319 & -17.878203 & -134.345576 & 221.300658 \\ 0.100734 & 0.038274 & -0.008458 & 1.000000 \end{pmatrix}$$

Dérivation alternative : Le produit croisé.

Dans une notation matricielle, on écrit :

$$\vec{P} \times \mathbf{M}_s^i \vec{R} = 0$$

Le terme $\vec{R}$ est facturé pour obtenir $\vec{P}\,\vec{R} \times \mathbf{M}_s^i = 0$

Ce qui donne : $$\begin{pmatrix} 0 & -wR^s & jwR^s \\ -wR^s & 0 & -iwR^s \\ wR^s & -wR^s & 0 \end{pmatrix} \begin{pmatrix} \mathbf{M}_s^1 \\ \mathbf{M}_s^2 \\ \mathbf{M}_s^3 \end{pmatrix} = 0$$

Parse que $\vec{R}$ et $\mathbf{M}_s^i$, sont les vecteurs, on obtient :

$$\begin{pmatrix} 0 & 0 & 0 & 0 & wX & wY & wZ & w1 & -jwX & -jwY & -jwZ & -jw1 \\ -wX & -wY & -wZ & -w & 0 & 0 & 0 & 0 & -iwX & -iwY & -iwZ & -iw1 \\ wX & wY & wZ & w & -wX & -wY & -wZ & -w & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{M}_1^1 \\ \mathbf{M}_2^1 \\ \mathbf{M}_3^1 \\ \mathbf{M}_4^1 \\ \mathbf{M}_1^2 \\ \mathbf{M}_2^2 \\ \mathbf{M}_3^2 \\ \mathbf{M}_4^2 \\ \mathbf{M}_1^3 \\ \mathbf{M}_2^3 \\ \mathbf{M}_3^3 \\ \mathbf{M}_4^3 \end{pmatrix} = 0$$

Dont deux equations sont indépendants.

Homographie entre deux plans

La projection d'un plan vers un autre plan est une transformation projective entre deux plans. Cette transformation s'appele une "homographie.
L'homographie est bijective.

Elle est bijective est facile a estimate et de "recetifié.

$$\boxed{Q^B = \mathbf{H}_A^B \; P^A}$$

En notation "classique".

$$\begin{pmatrix} w\ x_B \\ w\ y_B \\ w \end{pmatrix} = \mathbf{H}_A^B \begin{pmatrix} x_A \\ y_A \\ 1 \end{pmatrix} = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{pmatrix} \begin{pmatrix} x_A \\ y_A \\ 1 \end{pmatrix}$$

$$x_B = \frac{w\ x_B}{w} = \frac{m_{11}\, x_A + m_{12}\, y_A + m_{13}}{m_{31}\, x_A + m_{32}\, y_A + m_{33}}$$

$$y_B = \frac{w\ y_B}{w} = \frac{m_{21}\, x_A + m_{22}\, y_A + m_{23}}{m_{31}\, x_A + m_{32}\, y_A + m_{33}}$$

En notation "tensorielle"

$$\boxed{Q^B = \mathbf{H}_A^B \; P^A}$$

$$\begin{pmatrix} q^1 \\ q^2 \\ q^3 \end{pmatrix} = \mathbf{H}_A^B \begin{pmatrix} p^1 \\ p^2 \\ p^3 \end{pmatrix} = \begin{pmatrix} h_1^1 & h_2^1 & h_3^1 \\ h_1^2 & h_2^2 & h_3^2 \\ h_1^3 & h_2^3 & h_3^3 \end{pmatrix} \begin{pmatrix} p^1 \\ p^2 \\ p^3 \end{pmatrix}$$

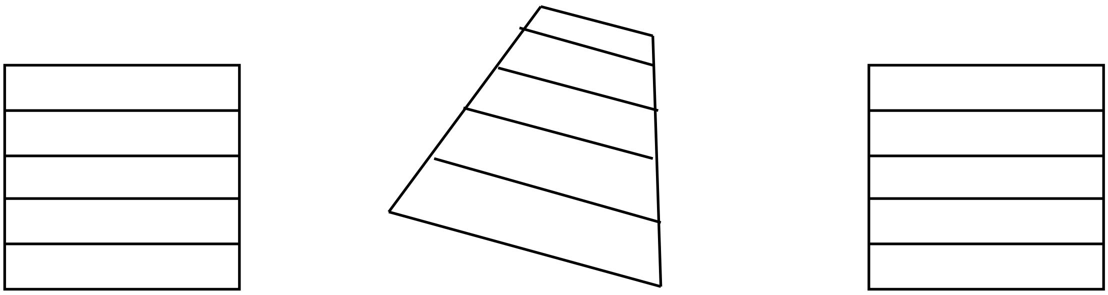
$$x_B = \frac{q^1}{q^3} \qquad y_B = \frac{q^2}{q^3}$$

Rectification de l'Image d'un plan.

La projection d'un plan 2D dans un scène 3D vers le retine 2D d'un caméra est une homographie. Elle est bijective est facile a "corriger".

Ceci a des nombreuse applications. Par exemple, l'image d'un tableau blanc peut être "rectifié" par l'homographie inverse.

$$\mathbf{H}_D^S = (\mathbf{H}_S^D)^{-1}$$



Dans le scène Dans l'image (source) L'image Rectifié (destination)

Pour chaque pixel de l'image destination, on calcul son position dans l'image source.

$$Q^S = \mathbf{H}_D^S \; P^D \quad \text{pour S, D = 1, 2, 3.}$$

Ensuite Pour chaque pixel P^D on détermine la valeur de la pixel au position

$$\begin{pmatrix} x_S \\ y_S \\ 1 \end{pmatrix} = \begin{pmatrix} q^1/q^3 \\ q^2/q^3 \\ 1 \end{pmatrix}$$

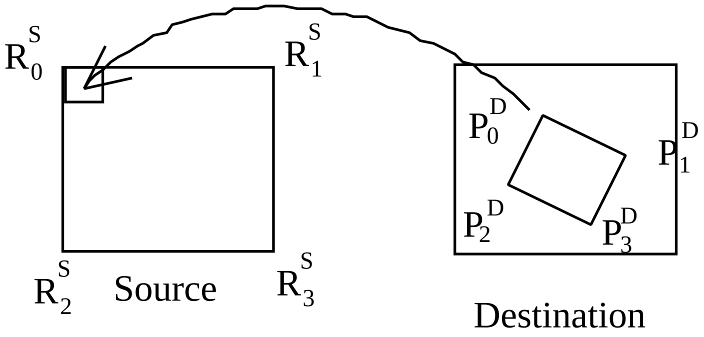
Ensuite

Pour chaque pixel, P^D :

SetPixel (Destination, P^D)= BiLinear_Interpolate(Source, ($\mathbf{H}_D^S$ P^D));

Estimation et Rectification par Homographie

La Homographie peut être déterminer par observation des 4 coins d'un carré.



Soit les qautre coins, k=1, 2, 3, 4 de l'image dans repere destination :

$$P_k^D = \begin{pmatrix} p_0^1 & p_1^1 & p_2^1 & p_3^1 \\ p_0^2 & p_1^2 & p_2^2 & p_3^2 \\ p_0^3 & p_1^3 & p_2^3 & p_3^3 \end{pmatrix}$$

On dit qu'ils corresponde au sommets de l'image source

$$R_k^S = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

On note $R_k^S = \mathbf{H}_D^S \; P_k^D$

Le 3 x 3 matrice inconnu, $\mathbf{H}_D^S$ a 9 coefficients.
Parce que $\mathbf{H}_D^S$ est en coordonées homogénes, on peut fixer $\mathbf{H}_3^3 = 1$.
Il nous reste 8 coéfficients à estimer.

Donc :

$$R_k^1 / R_k^3 = \mathbf{H}_D^1 \; P_k^D \; / \; \mathbf{H}_D^3 \; P_k^D$$

et

$$R_k^2 / R_k^3 = \mathbf{H}_D^2 \; P_k^D \; / \; \mathbf{H}_D^3 \; P_k^D$$

$$R_k^1 \; \mathbf{H}_D^3 \; P_k^D = R_k^3 \; \mathbf{H}_D^1 \; P_k^D$$
$$R_k^2 \; \mathbf{H}_D^3 \; P_k^D = R_k^3 \; \mathbf{H}_D^2 \; P_k^D$$

$$R_k^1 \; \mathbf{H}_D^3 \; P_k^D - R_k^3 \; \mathbf{H}_D^1 \; P_k^D = 0$$
$$R_k^2 \; \mathbf{H}_D^3 \; P_k^D - R_k^3 \; \mathbf{H}_D^2 \; P_k^D = 0$$

Ceci donne deux équations (S=1, 2) pour chaque coin (k=1,2,3,4) , donc 8 equations pour les 8 inconnues du $\mathbf{H}_D^S$. L'equation pour S =3 n'est pas indépendant de S =1 et S=2.