

1 Transformations de Modélisation

En infographie, les transformations géométriques sont fondamentales pour manipuler la position, l'orientation, et la taille des objets dans une scène 2D ou 3D. Ces transformations sont généralement représentées par des matrices et appliquées aux coordonnées des points définissant les objets. Ce chapitre explore différentes transformations de modélisation, leur composition, l'organisation hiérarchique des scènes et les projections pour visualiser des objets 3D sur un écran 2D.

1.1 Glissement (Shear)

Définition 1.1. Le **glissement**, aussi appelé cisaillement (shear), est une transformation qui déforme un objet en déplaçant ses points le long de lignes parallèles. C'est un étirement non uniforme suivant un axe, dont l'amplitude dépend de la distance à une ligne ou un plan de référence.

1.1.1 Glissement parallèle à l'axe x

La matrice d'un glissement parallèle à l'axe des x, où la coordonnée x d'un point est augmentée proportionnellement à sa coordonnée y avec un rapport k, est donnée par :

$$M_{shear_x} = \begin{pmatrix} 1 & k & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Un point $P = (x, y, z, 1)^T$ est transformé en $P' = (x + ky, y, z, 1)^T$.

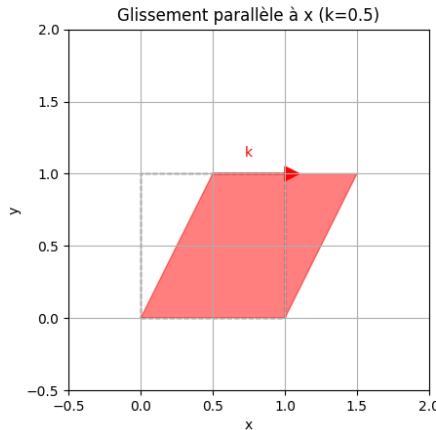


Figure 1: Illustration d'un glissement parallèle à l'axe x.

1.1.2 Glissement parallèle à l'axe x avec ligne de base

Si le glissement parallèle à l'axe x de rapport k s'effectue par rapport à une ligne de base horizontale $y = y_{ref}$, la matrice de transformation devient :

$$M_{shear_x_ref} = \begin{pmatrix} 1 & k & 0 & -k \cdot y_{ref} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Un point $P = (x, y, z, 1)^T$ est transformé en $P' = (x + k(y - y_{ref}), y, z, 1)^T$.

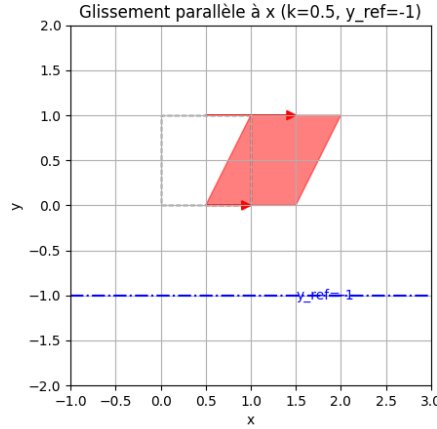


Figure 2: Illustration d'un glissement parallèle à l'axe x par rapport à une ligne de base y_{ref} .

2 Composition de Transformations

Il est souvent nécessaire d'appliquer plusieurs transformations successives à un objet. La composition de transformations consiste à combiner plusieurs matrices de transformation en une seule matrice équivalente.

2.1 Multiplication de matrices

Si un point p est d'abord transformé par une matrice M_1 puis par une matrice M_2 , le point résultant p'' est obtenu par $p'' = M_2 \cdot (M_1 \cdot p)$. Grâce à l'associativité de la multiplication matricielle, cela peut s'écrire $p'' = (M_2 \cdot M_1) \cdot p$. La matrice composée $M = M_2 \cdot M_1$ représente l'effet combiné des deux transformations.

Exemple 2.1 (Homothétie puis Translation). Considérons un point p auquel on applique d'abord une mise à l'échelle $S(s_x, s_y)$ puis une translation $T(t_x, t_y)$. Le point transformé p' est $p' = T \cdot (S \cdot p) = (T \cdot S) \cdot p$.

Soit $S = S(2, 2)$ et $T = T(3, 1)$. Les matrices correspondantes en coordonnées homogènes 2D (en étendant à 3x3 pour simplifier) sont :

$$S = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

La matrice composée $T \cdot S$ est :

$$T \cdot S = \begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} (1 \cdot 2 + 0 \cdot 0 + 3 \cdot 0) & (1 \cdot 0 + 0 \cdot 2 + 3 \cdot 0) & (1 \cdot 0 + 0 \cdot 0 + 3 \cdot 1) \\ (0 \cdot 2 + 1 \cdot 0 + 1 \cdot 0) & (0 \cdot 0 + 1 \cdot 2 + 1 \cdot 0) & (0 \cdot 0 + 1 \cdot 0 + 1 \cdot 1) \\ (0 \cdot 2 + 0 \cdot 0 + 1 \cdot 0) & (0 \cdot 0 + 0 \cdot 2 + 1 \cdot 0) & (0 \cdot 0 + 0 \cdot 0 + 1 \cdot 1) \end{pmatrix} = \begin{pmatrix} 2 & 0 & 3 \\ 0 & 2 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

Appliquons cela à un point $p = (1, 1)^T$, représenté en coordonnées homogènes par $p_{hom} = (1, 1, 1)^T$. $S \cdot p_{hom} = (2, 2, 1)^T$. Le point est mis à l'échelle à $(2, 2)$. $T \cdot (S \cdot p_{hom}) = T \cdot (2, 2, 1)^T = (2 + 3, 2 + 1, 1)^T = (5, 3, 1)^T$. Le point final est $(5, 3)$. Vérifions avec la matrice composée : $(T \cdot S) \cdot p_{hom} = \begin{pmatrix} 2 & 0 & 3 \\ 0 & 2 & 1 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \cdot 1 + 0 \cdot 1 + 3 \cdot 1 \\ 0 \cdot 1 + 2 \cdot 1 + 1 \cdot 1 \\ 0 \cdot 1 + 0 \cdot 1 + 1 \cdot 1 \end{pmatrix} = \begin{pmatrix} 5 \\ 3 \\ 1 \end{pmatrix}$. Le résultat est identique.

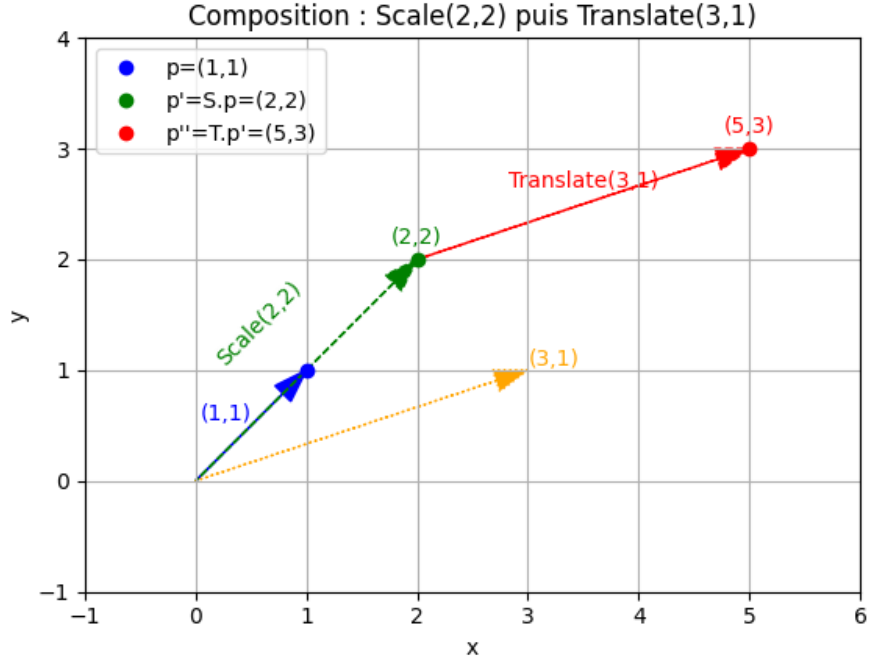


Figure 3: Composition d'une homothétie suivie d'une translation.

2.2 Non-commutativité

La multiplication de matrices n'est généralement pas commutative, c'est-à-dire $M_1 \cdot M_2 \neq M_2 \cdot M_1$. L'ordre dans lequel les transformations sont appliquées est donc crucial.

Exemple 2.2 (Translation puis Homothétie). Reprenons les transformations $S = S(2,2)$ et $T = T(3,1)$. Appliquons-les dans l'ordre inverse : d'abord la translation, puis l'homothétie. Le point transformé est $p' = S \cdot (T \cdot p) = (S \cdot T) \cdot p$.

La matrice composée $S \cdot T$ est :

$$S \cdot T = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} (2 \cdot 1 + 0 \cdot 0 + 0 \cdot 0) & (2 \cdot 0 + 0 \cdot 1 + 0 \cdot 0) & (2 \cdot 3 + 0 \cdot 1 + 0 \cdot 1) \\ (0 \cdot 1 + 2 \cdot 0 + 0 \cdot 0) & (0 \cdot 0 + 2 \cdot 1 + 0 \cdot 0) & (0 \cdot 3 + 2 \cdot 1 + 0 \cdot 1) \\ (0 \cdot 1 + 0 \cdot 0 + 1 \cdot 0) & (0 \cdot 0 + 0 \cdot 1 + 1 \cdot 0) & (0 \cdot 3 + 0 \cdot 1 + 1 \cdot 1) \end{pmatrix} = \begin{pmatrix} 2 & 0 & 6 \\ 0 & 2 & 2 \\ 0 & 0 & 1 \end{pmatrix}$$

On observe que $S \cdot T \neq T \cdot S$.

Appliquons cela au point $p = (1,1)^T$, soit $p_{hom} = (1,1,1)^T$. $T \cdot p_{hom} = (1+3, 1+1, 1)^T = (4,2,1)^T$. Le point est translaté à $(4,2)$. $S \cdot (T \cdot p_{hom}) = S \cdot (4,2,1)^T = (2 \cdot 4, 2 \cdot 2, 1)^T = (8,4,1)^T$. Le point final est

$$(8,4). \text{ Vérifions avec la matrice composée : } (S \cdot T) \cdot p_{hom} = \begin{pmatrix} 2 & 0 & 6 \\ 0 & 2 & 2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \cdot 1 + 0 \cdot 1 + 6 \cdot 1 \\ 0 \cdot 1 + 2 \cdot 1 + 2 \cdot 1 \\ 0 \cdot 1 + 0 \cdot 1 + 1 \cdot 1 \end{pmatrix} = \begin{pmatrix} 8 \\ 4 \\ 1 \end{pmatrix}. \text{ Le résultat est } (8,4), \text{ ce qui est différent de } (5,3) \text{ obtenu précédemment.}$$

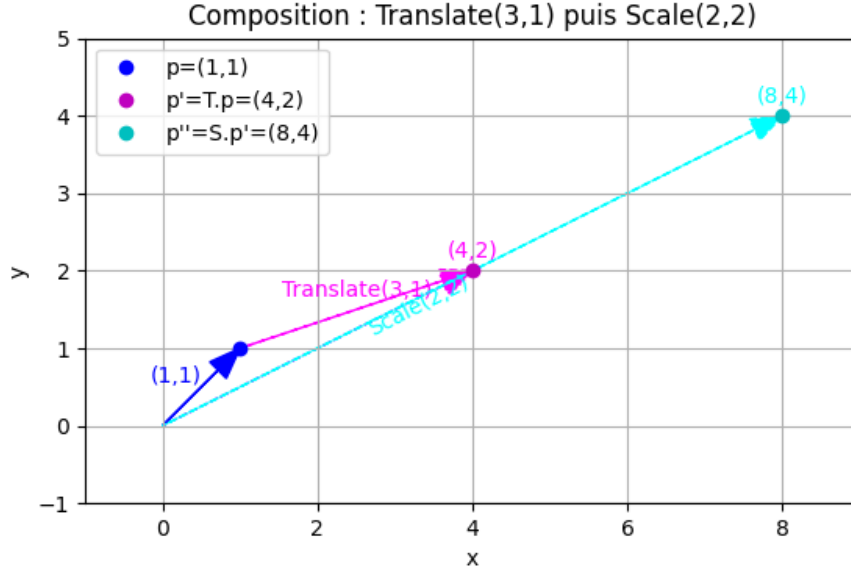


Figure 4: Composition d'une translation suivie d'une homothétie (comparez avec la Figure 3).

2.3 Ordre d'application

Comme $p'' = M_2 \cdot M_1 \cdot p$, la transformation M_1 est appliquée en premier au point p , suivie de la transformation M_2 appliquée au résultat. L'ordre d'application des transformations se lit donc ****de droite à gauche**** dans le produit matriciel.

2.4 Cas général

Pour une séquence de transformations représentées par les matrices M_1, M_2, \dots, M_n , appliquées dans cet ordre à un point p , le point transformé p' est donné par :

$$p' = M_n \cdot M_{n-1} \cdot \dots \cdot M_2 \cdot M_1 \cdot p$$

On peut précalculer la matrice de transformation globale $M = M_n \cdot M_{n-1} \cdot \dots \cdot M_2 \cdot M_1$. Alors, $p' = M \cdot p$.

2.5 Transformations par rapport à un point arbitraire

Les transformations élémentaires (rotation, mise à l'échelle) sont définies par rapport à l'origine du repère. Pour effectuer une transformation par rapport à un point arbitraire $P(t_x, t_y, t_z)$, on utilise une composition de trois transformations :

1. Ramener le point P à l'origine en appliquant une translation $T(-t_x, -t_y, -t_z)$.
2. Appliquer la transformation souhaitée (par exemple, une rotation $R(\theta)$ ou une mise à l'échelle $S(s_x, s_y, s_z)$).
3. Remplacer le point P à sa position initiale en appliquant la translation inverse $T(t_x, t_y, t_z)$.

La matrice de transformation résultante M est le produit de ces trois matrices :

$$M = T(t_x, t_y, t_z) \cdot R(\theta) \cdot T(-t_x, -t_y, -t_z)$$

ou

$$M = T(t_x, t_y, t_z) \cdot S(s_x, s_y, s_z) \cdot T(-t_x, -t_y, -t_z)$$

Exemple 2.3 (Rotation autour d'un axe passant par P). On désire effectuer une rotation d'angle θ_z autour d'un axe parallèle à l'axe z et passant par le point $P(T_x, T_y, T_z)$. La matrice de transformation M est :

$$M = T(T_x, T_y, T_z) \cdot R_z(\theta_z) \cdot T(-T_x, -T_y, -T_z)$$

où $R_z(\theta_z)$ est la matrice de rotation autour de l'axe z :

$$R_z(\theta_z) = \begin{pmatrix} \cos \theta_z & -\sin \theta_z & 0 & 0 \\ \sin \theta_z & \cos \theta_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Les matrices de translation sont :

$$T(T_x, T_y, T_z) = \begin{pmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad T(-T_x, -T_y, -T_z) = \begin{pmatrix} 1 & 0 & 0 & -T_x \\ 0 & 1 & 0 & -T_y \\ 0 & 0 & 1 & -T_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Effectuons le produit matriciel $M = T \cdot R_z \cdot T^{-1}$:

$$\begin{aligned} R_z \cdot T^{-1} &= \begin{pmatrix} \cos \theta_z & -\sin \theta_z & 0 & 0 \\ \sin \theta_z & \cos \theta_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -T_x \\ 0 & 1 & 0 & -T_y \\ 0 & 0 & 1 & -T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos \theta_z & -\sin \theta_z & 0 & -T_x \cos \theta_z + T_y \sin \theta_z \\ \sin \theta_z & \cos \theta_z & 0 & -T_x \sin \theta_z - T_y \cos \theta_z \\ 0 & 0 & 1 & -T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

$$\begin{aligned} M = T \cdot (R_z \cdot T^{-1}) &= \begin{pmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos \theta_z & -\sin \theta_z & 0 & -T_x \cos \theta_z + T_y \sin \theta_z \\ \sin \theta_z & \cos \theta_z & 0 & -T_x \sin \theta_z - T_y \cos \theta_z \\ 0 & 0 & 1 & -T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos \theta_z & -\sin \theta_z & 0 & -T_x \cos \theta_z + T_y \sin \theta_z + T_x \\ \sin \theta_z & \cos \theta_z & 0 & -T_x \sin \theta_z - T_y \cos \theta_z + T_y \\ 0 & 0 & 1 & -T_z + T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos \theta_z & -\sin \theta_z & 0 & T_x(1 - \cos \theta_z) + T_y \sin \theta_z \\ \sin \theta_z & \cos \theta_z & 0 & T_y(1 - \cos \theta_z) - T_x \sin \theta_z \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

Cette matrice M applique directement la rotation souhaitée autour de l'axe passant par $P(T_x, T_y, T_z)$. (Note : La diapositive 4 contenait une erreur dans le calcul final de la matrice M, elle a été corrigée ici.)

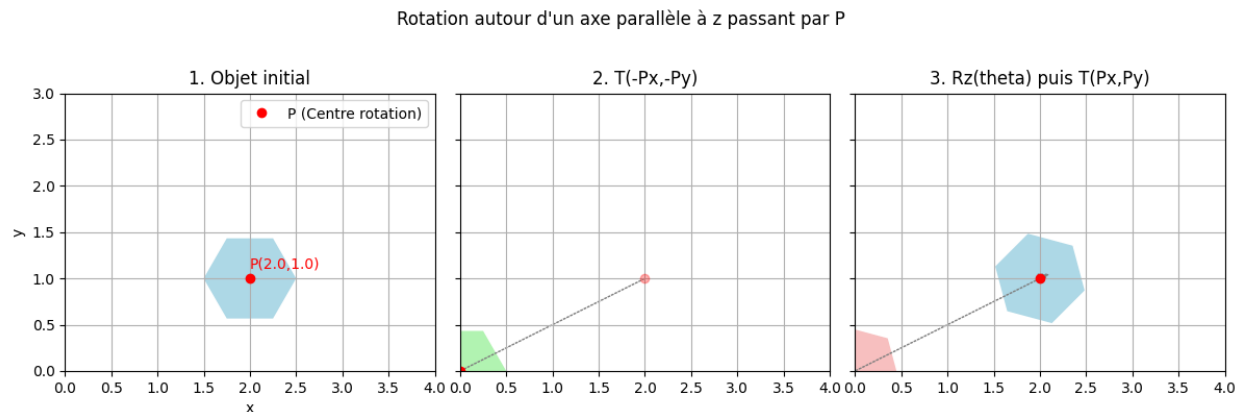


Figure 5: Étapes de la rotation d'un objet autour d'un point P.

3 Modélisation Hiérarchique

Pour décrire des objets ou des scènes complexes composés de plusieurs parties, on utilise souvent une approche de modélisation hiérarchique.

3.1 Concept et avantages

- Un modèle hiérarchique permet de décrire facilement des objets complexes composés d'objets plus simples (primitives).
- La scène est organisée sous forme d'une structure arborescente (ou plus généralement, un graphe).
- Les objets ne sont plus définis par leur transformation absolue par rapport au repère global (monde), mais par leur transformation relative par rapport à leur parent dans l'arbre.
- Le repère associé à la racine de l'arbre est le repère de la scène (repère monde).
- À chaque nœud de l'arbre est associé un repère local.
- À chaque arc (reliant un parent à un enfant) est associée une transformation géométrique qui positionne l'objet fils dans le repère de son père.
- Un même objet (ou sous-arbre) peut être inclus plusieurs fois dans la hiérarchie (instanciation), ce qui permet de réutiliser des modèles.
- La structure résultante est un graphe orienté acyclique (Directed Acyclic Graph - DAG), car un objet ne peut pas être son propre ancêtre.

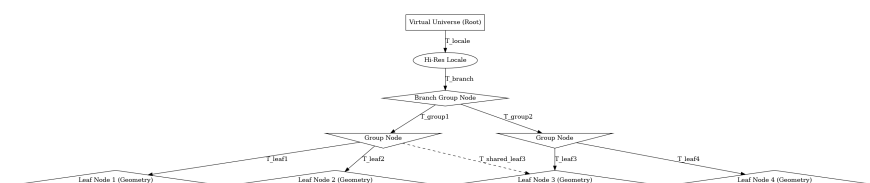


Figure 6: Exemple de structure hiérarchique sous forme de Graphe Orienté Acyclique (DAG).

3.2 Construction du graphe

La construction d'un modèle hiérarchique se fait souvent par un processus descendant ("top-down") :

1. On part de l'objet complet (racine).
2. On le décompose récursivement en parties plus simples (nœuds internes).
3. On continue la décomposition jusqu'à obtenir des objets élémentaires indécomposables, appelés primitives géométriques (feuilles de l'arbre).

La séquence de transformations le long d'un chemin depuis la racine jusqu'à un nœud feuille définit la position et l'orientation finale de la primitive correspondante dans le repère monde. Cette séquence est appelée la chaîne cinématique.

Exemple 3.1 (Modélisation d'une maison simple). On peut modéliser une maison comme un assemblage d'un corps principal et d'un toit. Le corps principal peut être décomposé en murs, fenêtres, porte. Les fenêtres peuvent être composées d'un cadre et d'une vitre, etc.

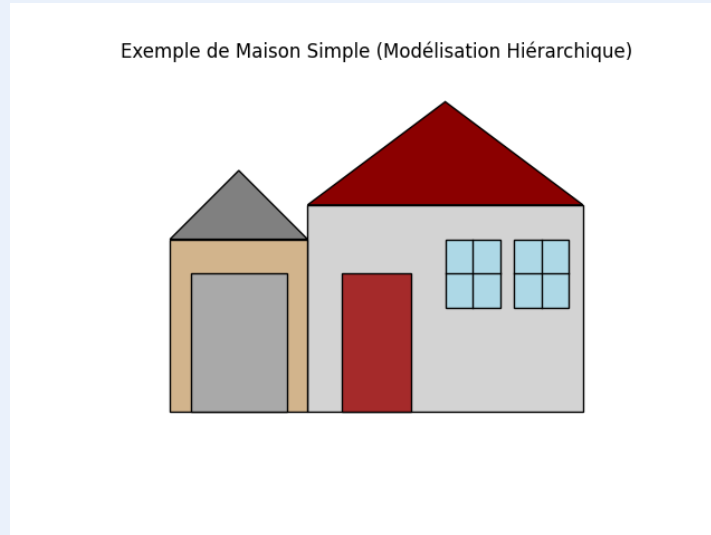


Figure 7: Illustration d'une maison simple pouvant être modélisée hiérarchiquement.

Le graphe de scène correspondant pourrait ressembler à ceci :

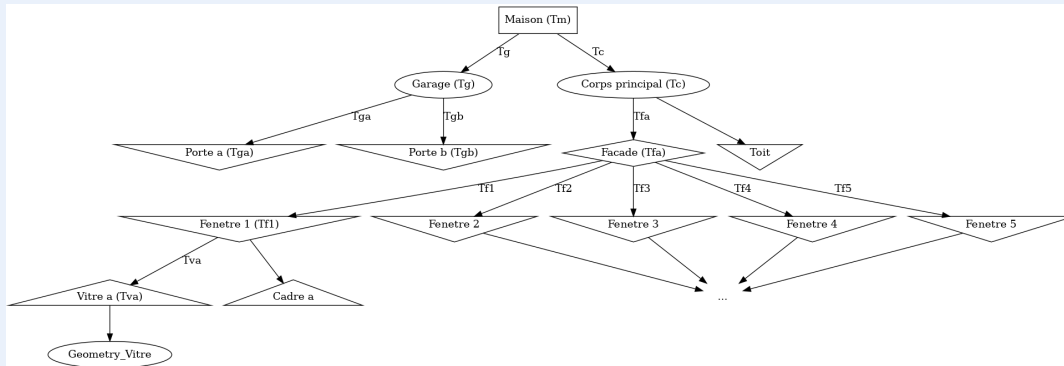


Figure 8: Exemple de graphe de scène pour la modélisation d'une maison.

3.3 Pile de transformations

Pour déterminer la transformation globale d'un objet feuille dans le repère monde, on multiplie toutes les matrices de transformation rencontrées le long du chemin depuis la racine jusqu'à la feuille. C'est la "Current Transformation Matrix" (CTM). Par exemple, pour trouver la transformation de la 'Vitre a' dans la Figure 8 par rapport à la 'Maison' (racine), on calcule :

$$M_{vitreA} = T_m \cdot T_c \cdot T_{fa} \cdot T_{f1} \cdot T_{va}$$

où chaque T représente la matrice de transformation associée à l'arc correspondant dans le graphe. L'implémentation se fait souvent à l'aide d'une pile (stack) pour gérer la CTM lors du parcours du graphe.

3.4 Instanciation (Instancing)

L'instanciation permet de réutiliser une définition d'objet (géométrie ou sous-graphe) à plusieurs endroits dans la scène sans dupliquer sa description.

- On définit l'objet une seule fois.
- Dans le graphe de scène, on utilise des nœuds "pointeurs" ou des références vers cette définition unique.
- Chaque instance (chaque pointeur) peut avoir sa propre transformation associée sur l'arc entrant, permettant de positionner, orienter et mettre à l'échelle chaque copie différemment.

Cela économise de la mémoire et facilite les modifications : changer la définition originale affecte toutes ses instances.

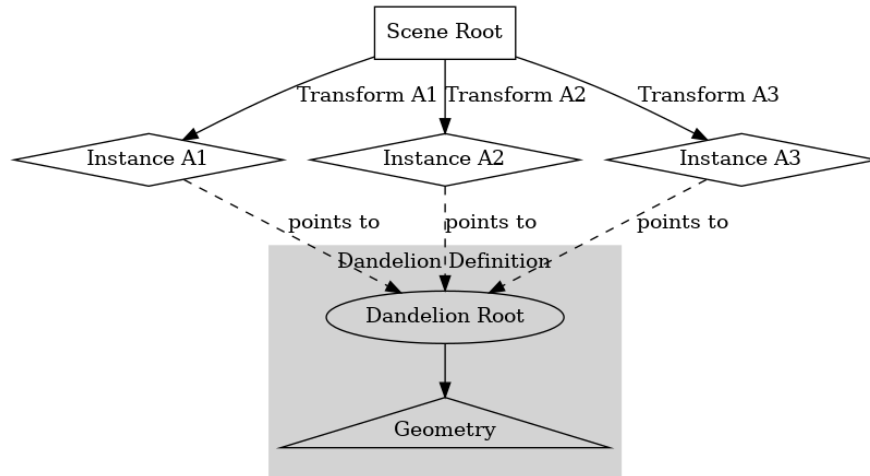


Figure 9: Graphe de scène illustrant l'instanciation : plusieurs nœuds (Instances A1, A2, A3) pointent vers la même définition d'objet (Dandelion) mais avec des transformations différentes.

4 Changement de Repère

Il est souvent nécessaire d'exprimer les coordonnées d'un point ou la description d'un objet dans différents systèmes de coordonnées (repères). Par exemple :

- Repère objet : Coordonnées locales à l'objet.
- Repère scène (ou monde) : Coordonnées globales de la scène.
- Repère caméra (ou vue) : Coordonnées par rapport à l'observateur.

Le passage d'un repère à un autre s'effectue à l'aide d'une matrice de changement de repère.

4.1 Matrice de passage

Soient deux repères R_1 (origine O_1 , axes X_1, Y_1, Z_1) et R_2 (origine O_2 , axes X_2, Y_2, Z_2). Soit un point P dont les coordonnées sont $P_1 = (x_1, y_1, z_1, 1)^T$ dans R_1 et $P_2 = (x_2, y_2, z_2, 1)^T$ dans R_2 .

La matrice de passage $M_{2 \rightarrow 1}$ permet de passer des coordonnées dans R_2 aux coordonnées dans R_1 :

$$P_1 = M_{2 \rightarrow 1} \cdot P_2$$

La matrice $M_{2 \rightarrow 1}$ exprime les vecteurs de base et l'origine de R_2 dans le repère R_1 . Si $O_2 = (T_x, T_y, T_z)$ dans R_1 et les vecteurs directeurs des axes X_2, Y_2, Z_2 ont pour coordonnées (R_{11}, R_{21}, R_{31}) , (R_{12}, R_{22}, R_{32}) , et (R_{13}, R_{23}, R_{33}) respectivement dans R_1 , alors :

$$M_{2 \rightarrow 1} = \begin{pmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} X_2|_{R1} & Y_2|_{R1} & Z_2|_{R1} & O_2|_{R1} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

La sous-matrice 3x3 contenant les R_{ij} est une matrice de rotation (si les repères sont orthonormés et de même orientation) et le vecteur (T_x, T_y, T_z) est le vecteur de translation de O_1 vers O_2 exprimé dans R_1 .

Pour passer des coordonnées de R_1 à R_2 , on utilise la matrice inverse $M_{1 \rightarrow 2} = (M_{2 \rightarrow 1})^{-1}$:

$$P_2 = M_{1 \rightarrow 2} \cdot P_1 = (M_{2 \rightarrow 1})^{-1} \cdot P_1$$

Si $M_{2 \rightarrow 1}$ représente une transformation rigide (rotation + translation), son inverse est relativement simple à calculer :

$$(M_{2 \rightarrow 1})^{-1} = \begin{pmatrix} R^T & -R^T \cdot T \\ 0 & 1 \end{pmatrix}$$

où R est la matrice de rotation 3x3 et T est le vecteur de translation $(T_x, T_y, T_z)^T$.

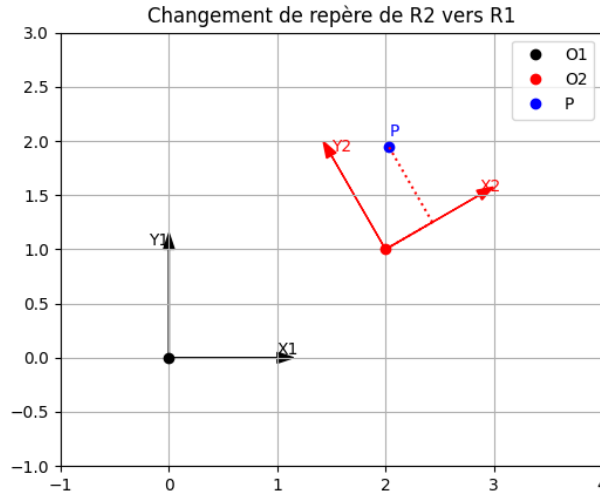


Figure 10: Illustration du changement de repère. Les coordonnées du point P peuvent être exprimées dans R_1 ou R_2 .

5 Projections

La projection est le processus qui transforme les objets d'un espace 3D en une représentation 2D, telle qu'elle apparaîtrait sur un écran ou une image. C'est une étape essentielle du pipeline graphique.

5.1 Pipeline Graphique simplifié

Les étapes principales pour afficher une scène 3D incluent :

1. **Transformations de modélisation** : Positionner/orienter/mettre à l'échelle les objets dans le repère monde.
2. **Transformation d'affichage (vue)** : Placer la caméra et transformer les coordonnées du repère monde vers le repère caméra.
3. **Illumination (Shading)** : Calculer les couleurs des objets en fonction des lumières et des matériaux (pas traité ici).
4. **Clipping** : Éliminer les parties de la scène qui sont en dehors du volume de vue de la caméra.
5. **Projection** : Transformer les coordonnées 3D (du repère caméra) en coordonnées 2D sur le plan de projection.
6. **Transformation écran (Viewport)** : Mapper les coordonnées 2D projetées vers les coordonnées pixels de l'écran.
7. **Pixelisation (Rasterization)** : Déterminer quels pixels sont couverts par chaque primitive projetée.
8. **Visibilité / Rendu** : Déterminer quelles surfaces sont visibles (élimination des parties cachées) et affecter les couleurs finales aux pixels.

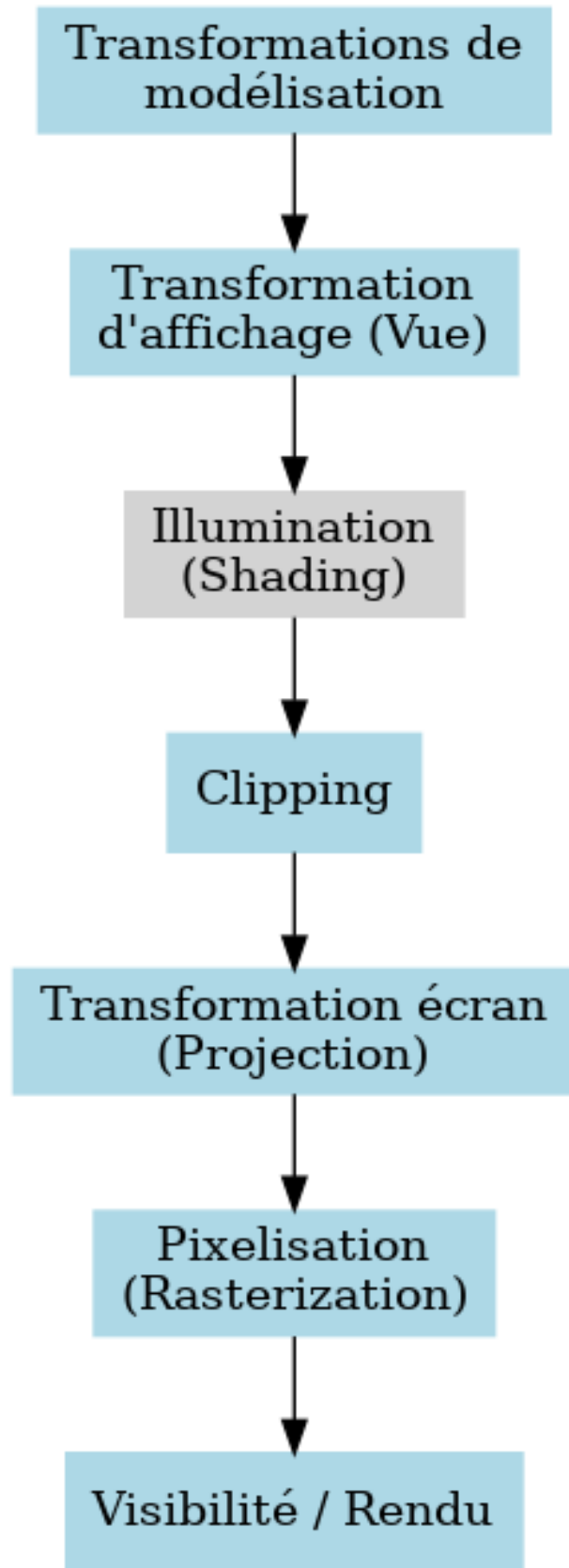


Figure 11: Étapes principales du pipeline graphique.

5.2 Définition et Types

La projection consiste à définir des **projecteurs** (lignes de projection) qui passent par chaque point de l'objet 3D et intersectent un **plan de projection** (ou plan de vue). L'ensemble de ces intersections forme l'image 2D projetée de l'objet.

Il existe deux grandes familles de projections géométriques :

1. **Projection Perspective** : Les projecteurs convergent en un point unique appelé **Centre de Projection (CP)** ou point de vue. Ce CP est situé à une distance finie du plan de projection. Elle simule la vision humaine et photographique (les objets lointains paraissent plus petits).
2. **Projection Parallèle** : Les projecteurs sont parallèles entre eux. Cela correspond à un CP situé à l'infini. Elle préserve les rapports de taille et les angles pour les faces parallèles au plan de projection, utile pour les dessins techniques.

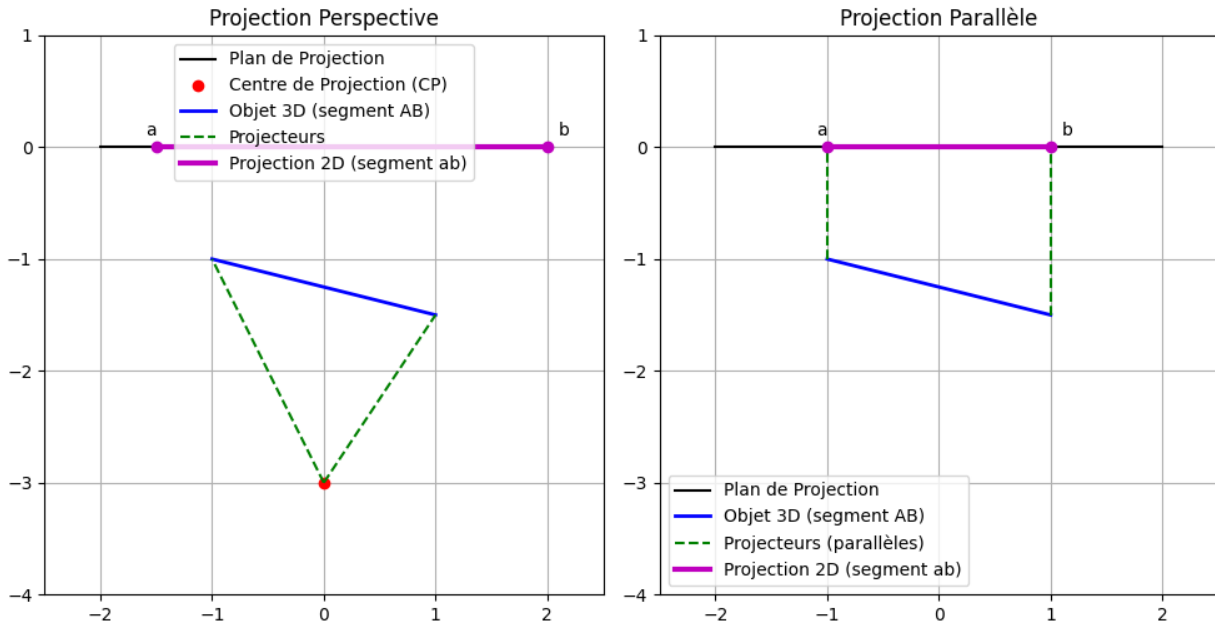


Figure 12: Comparaison entre projection perspective (gauche) et parallèle (droite).

5.3 Taxonomie des Projections

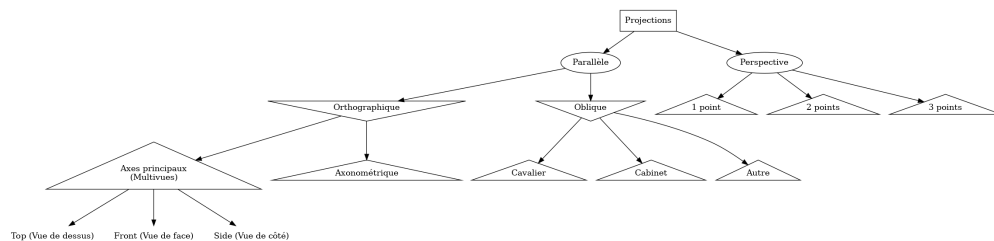


Figure 13: Classification des projections géométriques.

5.4 Projections Parallèles

Dans une projection parallèle, la direction de projection (DP) est constante pour tous les points. **Propriétés** :

- Les lignes parallèles dans l'espace 3D restent parallèles dans l'image 2D projetée.
- Les rapports des distances le long d'une direction donnée sont conservés.
- Peu réaliste (pas d'effet de perspective), mais utile pour les dessins techniques et les mesures exactes car les dimensions relatives sont préservées pour les faces parallèles au plan de projection.

5.4.1 Projection Orthographique

Les projecteurs sont parallèles entre eux ET perpendiculaires au plan de projection.

- **Vues Multiples (Axes principaux) :** Le plan de projection est parallèle à l'un des plans de coordonnées principaux (xy, yz, xz). On obtient les vues de face (Front), de dessus (Top), et de côté (Side). Mathématiquement, cela revient à annuler une des coordonnées. Par exemple, pour une projection sur le plan $z=0$ (vue de dessus), on met la coordonnée z à 0.
- **Axonométrique :** Le plan de projection n'est pas parallèle aux plans de coordonnées. Plusieurs faces de l'objet sont visibles simultanément. Le vecteur normal au plan de projection n'est parallèle à aucun des axes de coordonnées. (Isométrique, Dimétrique, Trimétrique selon les angles).

Matrice de Projection Orthographique (sur plan $z=0$) : Pour projeter un point $P = (x, y, z, 1)^T$ sur le plan $z = 0$ (vue de dessus), le point projeté est $P' = (x, y, 0, 1)^T$. La matrice correspondante est :

$$M_{ortho-z} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

De même pour les projections sur les plans $x=0$ et $y=0$:

$$M_{ortho-x} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad M_{ortho-y} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

5.4.2 Projection Oblique

Les projecteurs sont parallèles entre eux mais **non perpendiculaires** au plan de projection. L'angle ϕ entre les projecteurs et le plan est différent de 90° .

- Les faces parallèles au plan de projection sont projetées sans déformation.
- Les faces perpendiculaires au plan de projection sont déformées (lignes projetées avec un angle α et un facteur de raccourcissement f).
- **Projection Cavalière :** L'angle $\phi = 45^\circ$. Les lignes perpendiculaires au plan de projection ne sont pas raccourcies ($f = 1$). L'angle α est souvent choisi à 30° ou 45° . Apparence déformée.
- **Projection Cabinet :** L'angle $\phi \approx 63.4^\circ$ ($\cot \phi = 1/2$). Les lignes perpendiculaires au plan sont raccourcies de moitié ($f = 1/2$). Apparence plus réaliste que la cavalière. L'angle α est souvent choisi à 30° ou 45° .

La matrice d'une projection oblique est une combinaison d'une transformation de glissement (shear) suivie d'une projection orthographique.

5.5 Projection Perspective

Dans une projection perspective, les projecteurs émanent d'un point unique, le Centre de Projection (CP).

Propriétés :

- Non-linéaire (en coordonnées cartésiennes, mais linéaire en coordonnées homogènes).
- Ne conserve pas le parallélisme (sauf pour les lignes parallèles au plan de projection).
- Ne conserve pas les angles ni les rapports de distance (sauf cas particuliers).
- La taille projetée d'un objet diminue lorsque sa distance au CP augmente (effet de perspective).
- Les ensembles de lignes parallèles dans l'espace 3D (qui ne sont pas parallèles au plan de projection) convergent vers un point unique dans l'image projetée, appelé **point de fuite**.
- Le nombre de points de fuite (1, 2 ou 3) dépend de l'orientation des axes principaux de l'objet par rapport au plan de projection.

5.5.1 Calcul de la Projection Perspective (Vue de côté simplifiée)

Considérons une caméra simple (sténopé) avec le CP à l'origine (0,0,0) et le plan de projection à $z = -f$ (ou $z = f$ selon la convention, ici utilisons $z = f$ comme sur certaines slides, bien que $z = -f$ soit plus courant pour un système droitier regardant vers $-z$). Soit un point 3D $P(x, y, z)$ et son projeté $q(u, v)$ sur le plan $z = f$.

Par Thalès (triangles semblables) :

$$\frac{u}{x} = \frac{f}{z} \implies u = \frac{x \cdot f}{z} = \frac{x}{z/f}$$

$$\frac{v}{y} = \frac{f}{z} \implies v = \frac{y \cdot f}{z} = \frac{y}{z/f}$$

Si $f = 1$ (plan de projection unitaire) :

$$u = \frac{x}{z}, \quad v = \frac{y}{z}$$

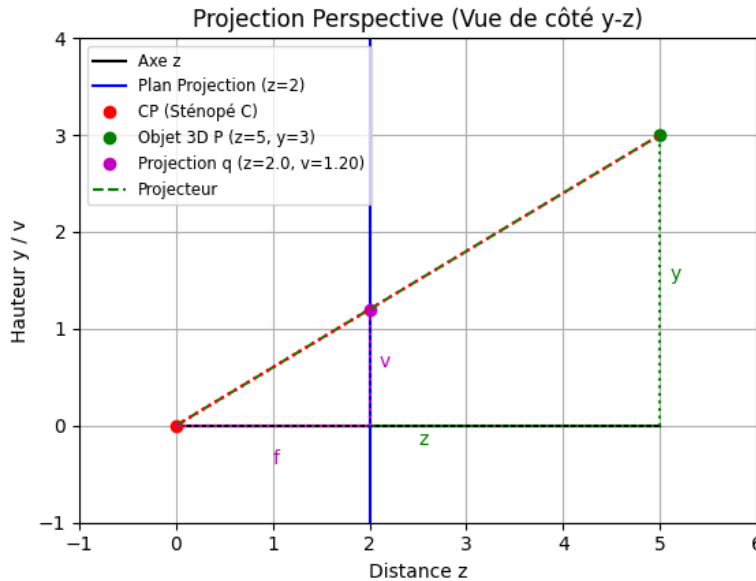


Figure 14: Calcul de la projection perspective en utilisant les triangles semblables (vue de côté y-z).

5.5.2 Matrice de Projection Perspective (simple)

En coordonnées homogènes, la projection perspective $u = xf/z$, $v = yf/z$ peut être représentée par une matrice. L'astuce est que la division par z est effectuée lors de la renormalisation des coordonnées homogènes (division par la 4ème composante w). On cherche une matrice M_{persp} telle que :

$$\begin{pmatrix} x' \\ y' \\ z' \\ w' \end{pmatrix} = M_{persp} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad \text{et} \quad u = \frac{x'}{w'}, v = \frac{y'}{w'}$$

Une matrice possible (qui projette sur $z = 0$ après division, et préserve z pour le Z-buffering, avec CP à l'origine) est :

$$M_{persp} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & f & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad \text{ou plus couramment} \quad M_{persp} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix}$$

Appliquons la seconde matrice :

$$\begin{pmatrix} x' \\ y' \\ z' \\ w' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ z/f \end{pmatrix}$$

Après division par $w' = z/f$, on obtient les coordonnées cartésiennes projetées :

$$u = \frac{x'}{w'} = \frac{x}{z/f} = \frac{xf}{z}$$

$$v = \frac{y'}{w'} = \frac{y}{z/f} = \frac{yf}{z}$$

La coordonnée $z_{proj} = z'/w' = z/(z/f) = f$ est constante (tous les points se projettent sur le plan $z = f$), ce qui n'est pas idéal pour la gestion de la profondeur (Z-buffer). Des matrices plus complexes sont utilisées en pratique pour mapper l'intervalle de profondeur $[z_{near}, z_{far}]$ vers un intervalle normalisé (e.g., $[-1, 1]$).

La matrice présentée sur la diapositive 16 (droite) semble correspondre à une projection sur $z = 0$ avec le CP à $(0, 0, -f)$ regardant vers $+z$. $x_p = \frac{x}{-z/f+1}$, $y_p = \frac{y}{-z/f+1}$. La matrice homogène correspondante est :

$$M_{persp} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/f & 1 \end{pmatrix}$$

Vérifions :

$$\begin{pmatrix} x' \\ y' \\ z' \\ w' \end{pmatrix} = M_{persp} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ -z/f + 1 \end{pmatrix}$$

Après division par $w' = -z/f + 1$:

$$x_p = \frac{x}{-z/f + 1}, \quad y_p = \frac{y}{-z/f + 1}$$

Ceci correspond aux formules données.

5.5.3 Effet de la distance focale (f)

- Plus f est **grand** (téléobjectif), plus la projection tend vers une projection parallèle (orthographique). La profondeur a peu d'effet sur la taille projetée, la perspective est moins marquée ("écrasée").
- Plus f est **petit** (grand angle), plus la projection diverge. La profondeur a une influence considérable, les effets de perspective sont exagérés (distorsion possible près des bords).

Effet de la distance focale (f) sur la projection perspective

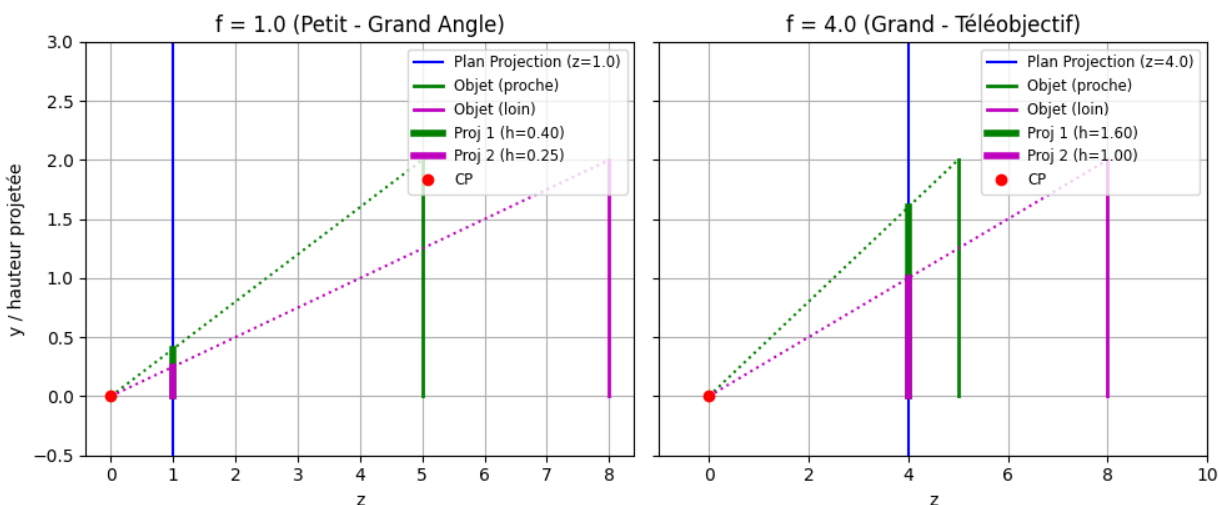


Figure 15: Comparaison de la taille projetée pour un objet à deux distances différentes, avec une petite distance focale (gauche) et une grande distance focale (droite). La différence de taille projetée est plus marquée avec un petit f .

5.5.4 Volume de vue (View Frustum) et Clipping

En pratique, on ne projette pas tout l'espace 3D. On définit un volume de vue (View Frustum), généralement une pyramide tronquée, qui délimite la portion de l'espace visible par la caméra. Ce volume est défini par 6 plans :

- Plan proche (Near plane) : $z = z_{near}$ (ou $z = -n$)
- Plan lointain (Far plane) : $z = z_{far}$ (ou $z = -f$)
- Plans gauche/droite/haut/bas (Left/Right/Top/Bottom) : Ces plans passent par le CP et les bords de la "fenêtre" définie sur le plan proche.

Le processus de **clipping** consiste à éliminer toutes les primitives géométriques (ou parties de primitives) qui se trouvent en dehors de ce volume de vue.

Pourquoi les plans Near et Far ?

- Éviter la division par zéro (ou un nombre très petit) dans les calculs de projection si z est proche de 0 (CP). Le plan Near impose $z \geq z_{near} > 0$.
- Gérer les primitives qui pourraient se trouver derrière la caméra.
- Limiter la plage de valeurs de profondeur à traiter, ce qui est crucial pour la précision du **Z-buffer** (tampon de profondeur). Le Z-buffer stocke la valeur de profondeur pour chaque pixel afin de déterminer

quelle surface est visible. Il utilise une précision limitée (e.g., float 16/24/32 bits). Une plage de profondeur $[z_{near}, z_{far}]$ trop grande, surtout avec un z_{near} très petit, peut entraîner des erreurs d'arrondi et des artefacts visuels ("Z-fighting") où des surfaces proches se "battent" pour savoir laquelle est devant.

5.5.5 Passage en Coordonnées Normalisées (NDC)

Avant la projection finale en 2D et le mapping vers l'écran, il est courant de normaliser le volume de vue (le frustum perspectif) en un volume canonique, généralement un cube $[-1, 1]^3$. Cet espace est appelé Normalized Device Coordinates (NDC). Cette transformation simplifie les étapes ultérieures comme le clipping (qui se fait alors contre les faces du cube unité) et la projection orthographique finale (qui devient triviale).

La matrice de transformation $M_{persp \rightarrow NDC}$ qui mappe le frustum perspectif défini par (*left, right, bottom, top, near, far*) vers le cube NDC $[-1, 1]^3$ est donnée par :

$$M_{persp \rightarrow NDC} = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & -\frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

(Note: Cette matrice suppose un système de coordonnées caméra regardant vers -z, avec $near=n$ et $far=f$. Des variations existent selon les conventions +/-z, n/f positifs/négatifs).

Après application de cette matrice et division par w , un point (x_c, y_c, z_c) dans le repère caméra qui était à l'intérieur du frustum se retrouvera avec des coordonnées $(x_{ndc}, y_{ndc}, z_{ndc})$ comprises entre -1 et 1.

5.5.6 Transformation complète

La transformation complète d'un point P_w en coordonnées monde vers ses coordonnées finales P_{screen} sur l'écran implique la chaîne suivante :

$$P_{screen} = M_{viewport} \cdot M_{ortho} \cdot M_{persp \rightarrow NDC} \cdot M_{world \rightarrow cam} \cdot P_w$$

où :

- $M_{world \rightarrow cam}$: Transformation du repère monde au repère caméra (changement de base).
- $M_{persp \rightarrow NDC}$: Transformation du frustum perspectif vers le cube NDC.
- M_{ortho} : Projection orthographique triviale (ignorer z, ou la matrice identité si z est déjà géré).
- $M_{viewport}$: Transformation des coordonnées NDC vers les coordonnées pixels de la fenêtre d'affichage (Viewport).

Souvent, $M_{ortho} \cdot M_{persp \rightarrow NDC}$ est combinée en une seule matrice de projection. La combinaison $M_{proj} \cdot M_{view} = M_{persp \rightarrow NDC} \cdot M_{world \rightarrow cam}$ est fréquemment précalculée.