

Techniques de Rendu pour la
Synthèse d'Images

Master 2
Simulation des mondes virtuels



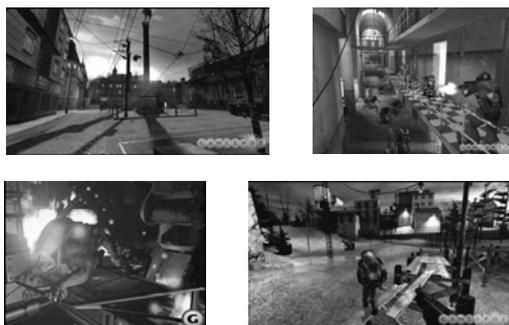
Sources

- Frédo Durand, Jovan Popović, *Computer Graphics: Introduction*, MIT 2007
- Joëlle Thollot, Marie-Paule Cani, ARTIS/EVASION
- SIGGRAPH,
- Etc.

<http://www.OpenGL.org>

2

Computer Graphics is Fun



3

Computer Graphics is Big Business



4

Computer Graphics Help Make Things



5

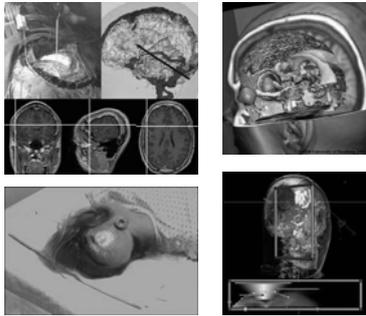
And houses, too



[Jovan Popović, *Computer Graphics: Introduction*, MIT 2007]

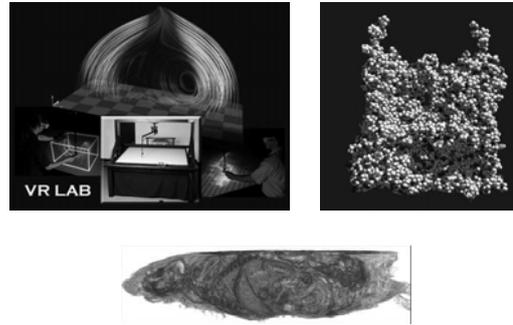
6

CG saves lives



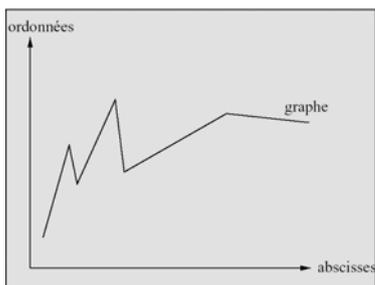
7

CG lead to new discoveries



8

Historiquement: Lignes 2D



9

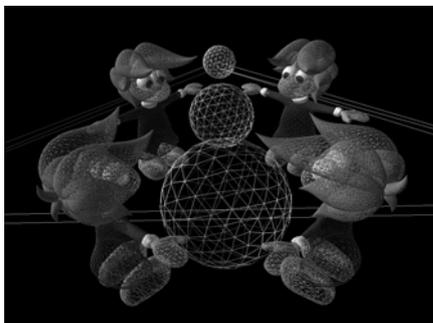
Image 2D



L'ombre, une image naturelle, Wikipedia

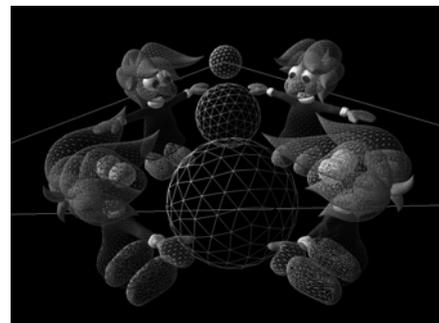
10

Image ~ lignes 3D



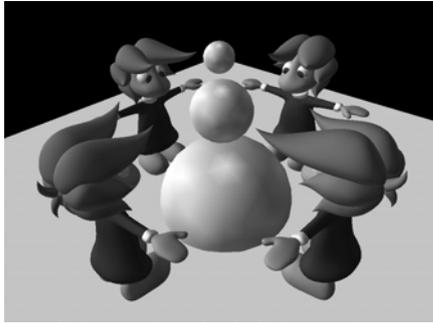
11

Image ~ lignes cachées *backface culling*



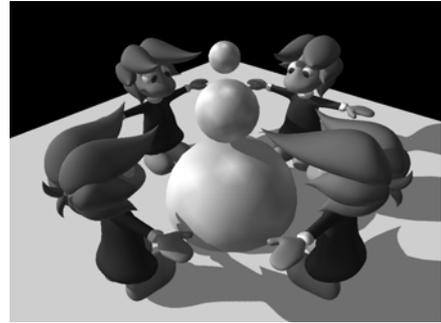
12

Image 3D



13

+ Ombres



14

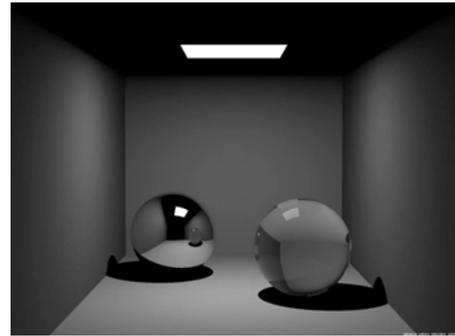
Photo-réalisme



15

Illumination

1 / 4

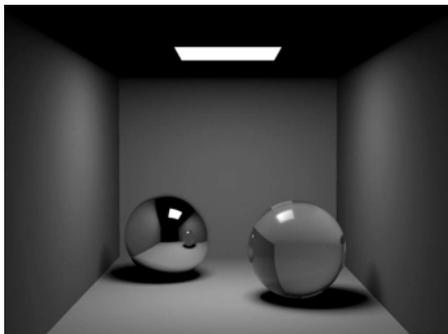


Ray tracing

16

Illumination

2 / 4

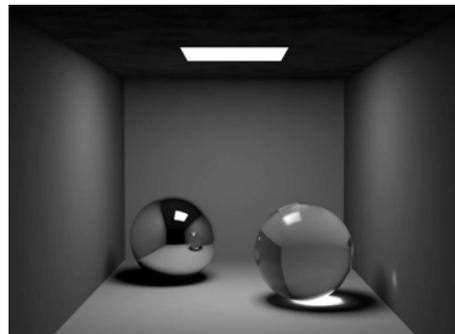


+ ombres douces

17

Illumination

3 / 4

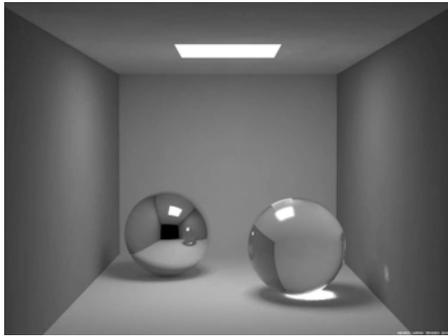


+ caustiques

18

Illumination

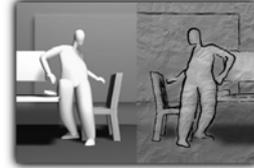
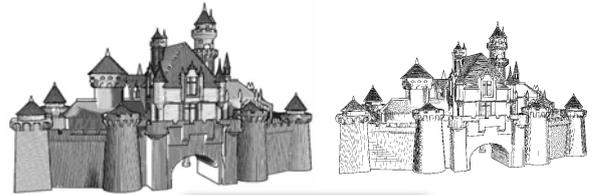
4/4



+ globale

19

NPR ~ Rendu expressif

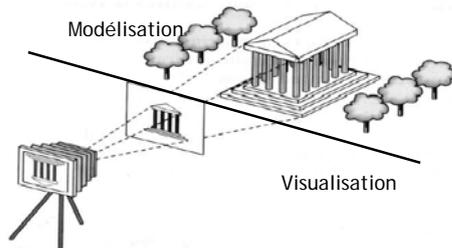


© 1998 Jörg Hamul

© 1998 Cassidy Curtis

20

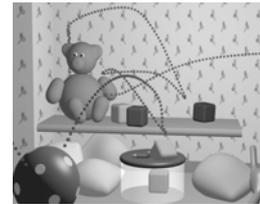
Synthèse d'Images



21

Production d'un film de synthèse

1. Modélisation
des objets, d'une scène
2. Rendu d'une image
à partir des objets,
matières, éclairages,
caméras...
3. Animation
spécifier ou calculer
mouvements et
déformations



22

Ce que vous apprendrez

- Fondements du rendu
- Présentation rapide des méthodes récentes
- Vocabulaire complet
- Pratique d'OpenGL
- Programmation C++
- Études de cas
 - Connaissance globale de ce qui existe
 - Capacité à faire des choix de méthode

23

Ce que vous n'apprendrez pas

- La modélisation et l'animation
- Les détails
- L'utilisation de logiciel de synthèse d'images (Blender, 3DS, Maya, XSI)
- La conception artistique

24

Fonctionnement du cours

<http://artis.inrialpes.fr/Enseignement/>

- 10 cours , ~25 papiers, 1 mini débat
- 4 TPs *OpenGL*, 1 TP *CUDA*
- 1 TP *ray tracing* + 1 mini projet
Binômes. Soutenance le vendredi 11 janvier
- examen écrit : cours + papiers
vendredi 25 janvier

25

Programme

- Bases: OpenGL, pipeline, *vertex shader*
- Illumination locale, *Fragment shader*
- Textures
- Ombres
- Lancé de rayon
- Représentations alternatives
- Rendu expressif
- Illumination globale
- *GPGPU*

26

Techniques de Rendu pour la Synthèse d'Images

Cours 1 : Introduction, pipeline graphique



28

Cours 1 et 2 : Les bases

- Historique
- Pipeline graphique
- Transformations
- Image, couleur
- Rasterisation
- Visibilité
- Illumination

Plan

- Historique
- Pipeline graphique
- Transformations
- Image, couleur
- Rasterisation
- Visibilité
- Illumination

29

Historique

1 / 2

- **1950** : MIT, premier écran (vectoriel) contrôlé par ordinateur.
- **1950-60** : SAGE, contrôle d'un écran par un crayon lumineux, premier système graphique interactif.
- **1963** : Sketchpad (thèse de Ivan Sutherland), propose un premier modèle complet de système graphique interactif (sélectionner, pointer, dessiner, éditer); identifie les structures de données et algorithmes nécessaires. Avancée majeure dans le domaine du graphisme.
- **1960-70** : premiers systèmes de DAO (General Motors, Bell, NASA). Algorithmes de base du graphisme.
- **1969** : premiers capteurs CCD (cameras numériques), Bell.

30

Historique

2/2

- **1970** : premier terminal graphique (Tektronix). ADIS graphics package : possibilité de lire/écrire un pixel, dessiner une ligne, déplacer des portions de l'image, la sortie étant dirigée sur des zones rectangulaires appelées *fenêtres*.
- **1970-80** : barrière économique dépassée, mémoires et écrans peu chers sont disponibles, écrans matriciels, apparition de la couleur, cartes d'extensions graphiques (Apple II). Algorithmes de traitements d'images, premiers algorithmes de vision artificielle.
- **1980-90** : premières interfaces graphiques (xerox). Photo-réalisme (lancer de rayons, radiosité). Animations.
- **1984** : Le langage PostScript. Le macintosh.
- **1990-...** : systèmes 3D interactifs, algorithmes câblés, librairie graphique OpenGL, cameras CCD peu chères, développement de la vision artificielle. Réalité virtuelle.

31

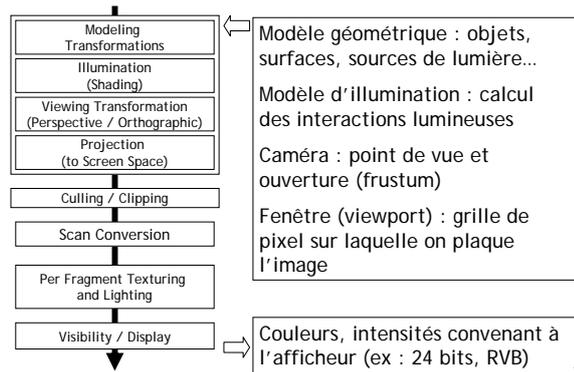
Plan

- Historique
- Pipeline graphique
- Transformations
- Image, couleur
- Rasterisation
- Visibilité
- Illumination

32

Le pipeline graphique

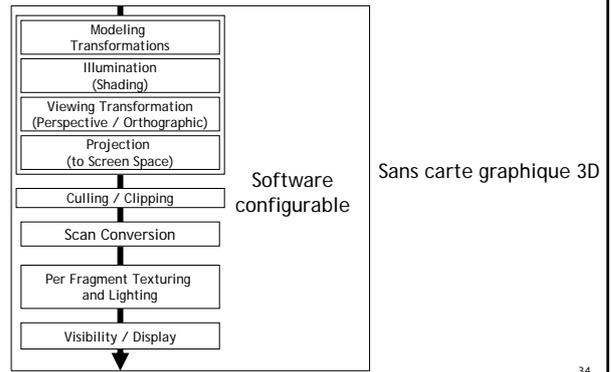
1 / 6



33

Le pipeline graphique

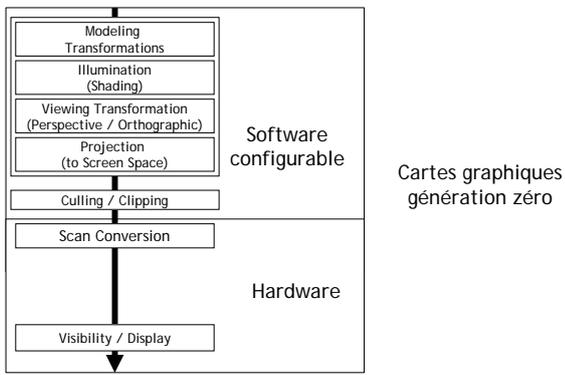
2 / 6



34

Le pipeline graphique

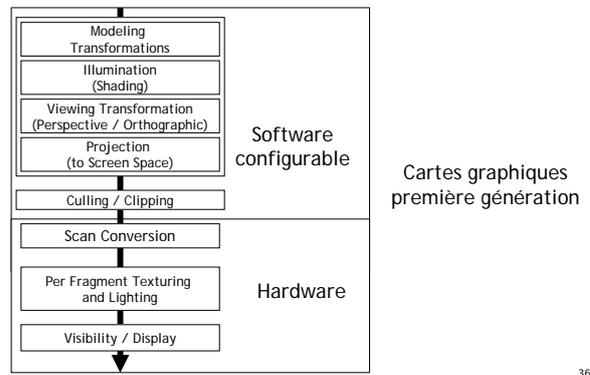
3 / 6



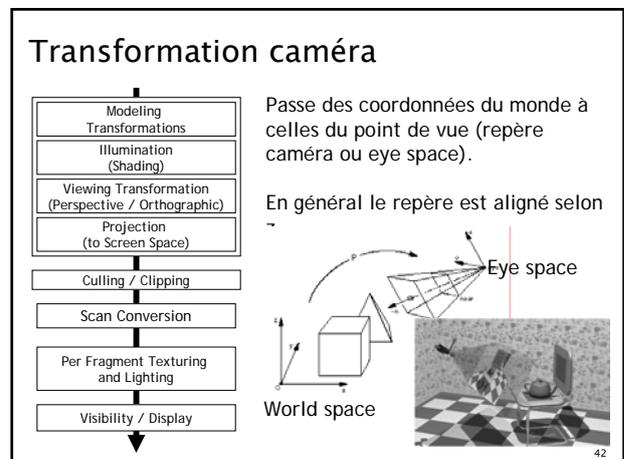
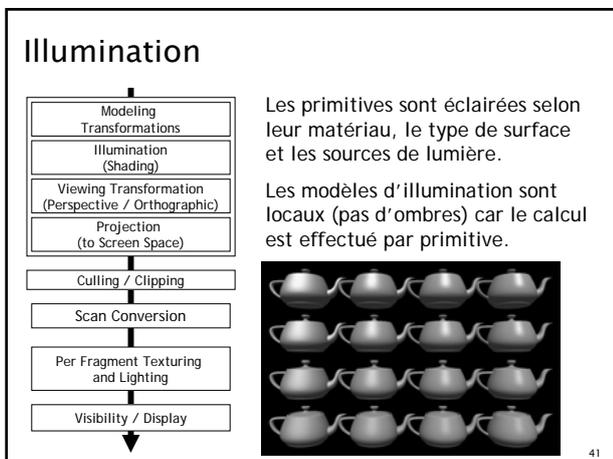
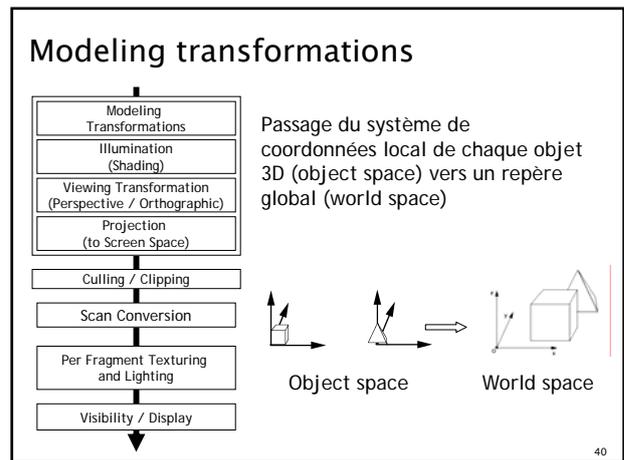
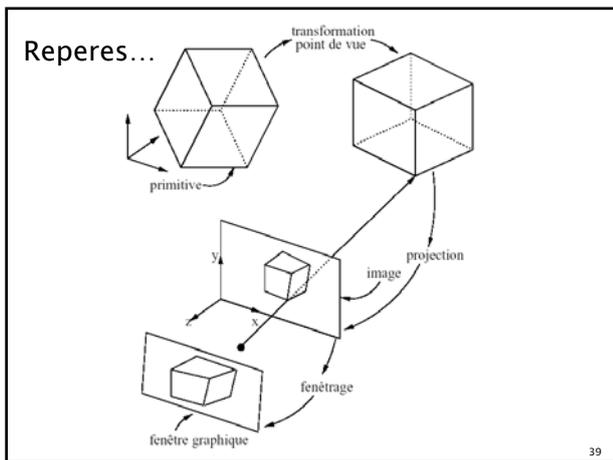
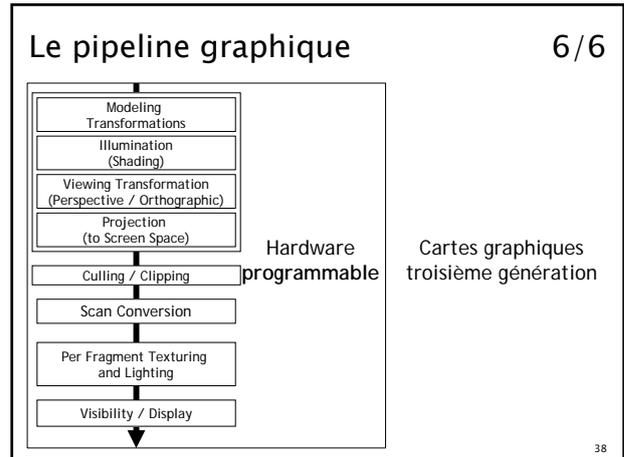
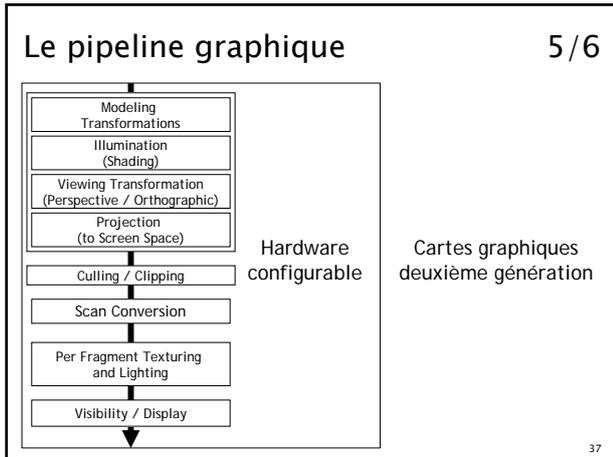
35

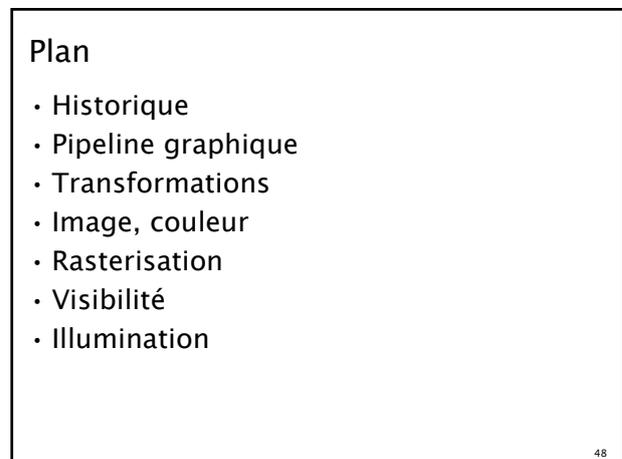
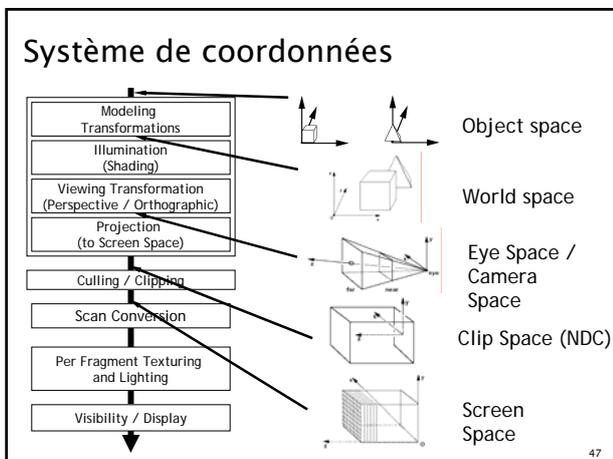
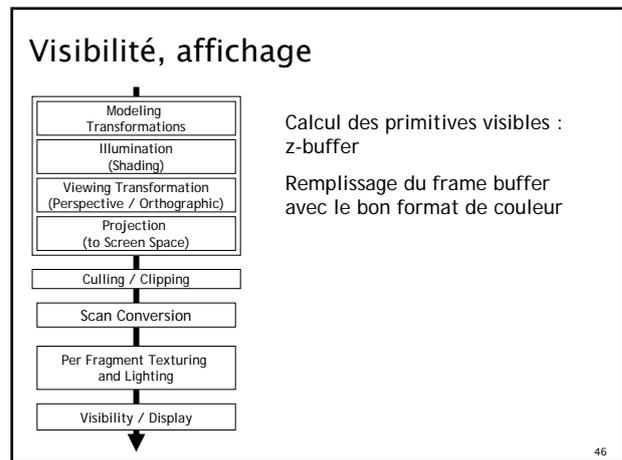
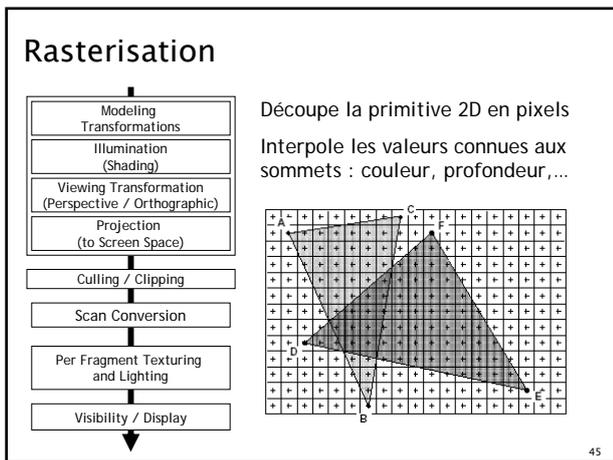
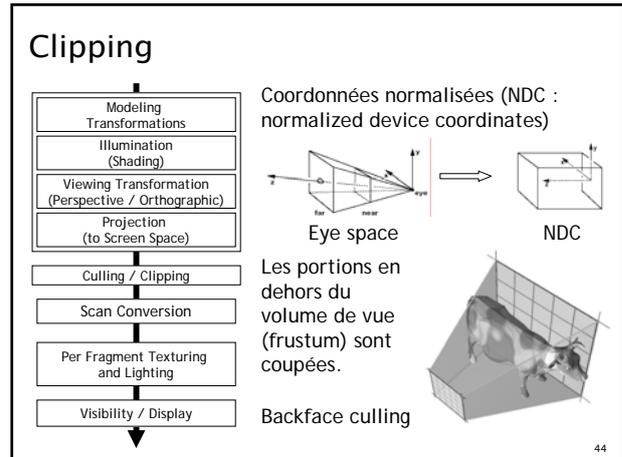
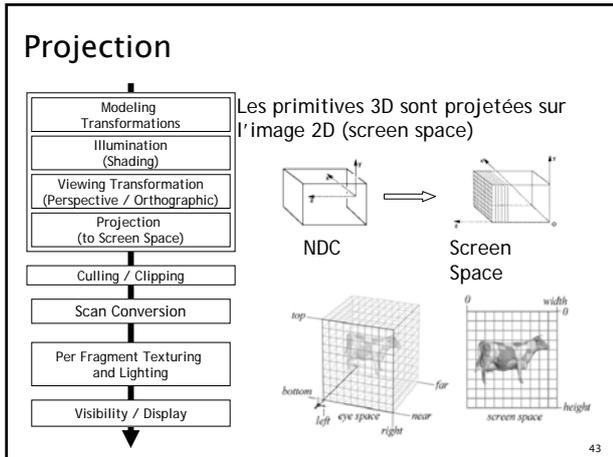
Le pipeline graphique

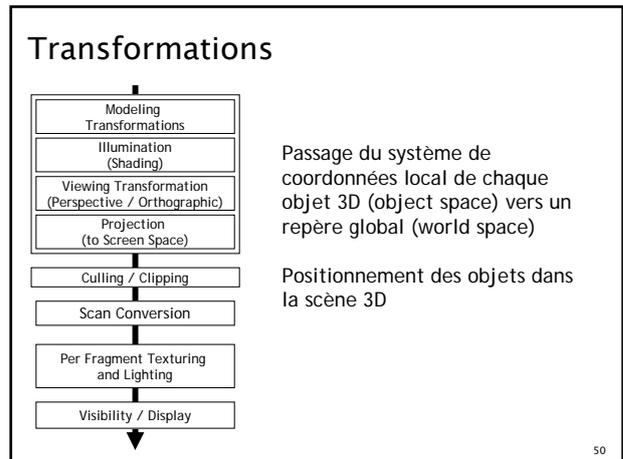
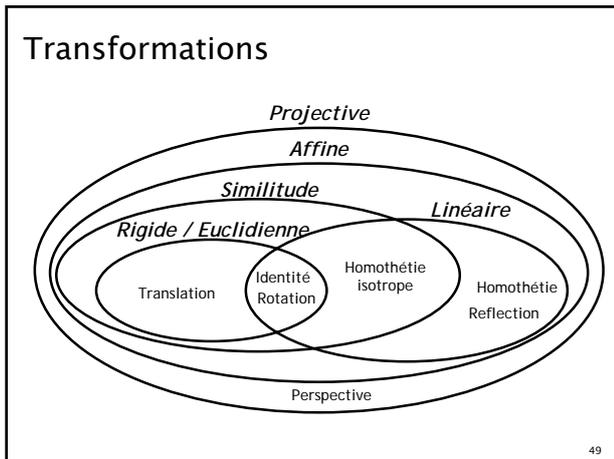
4 / 6



36







Coordonnées homogènes

- Représentation matricielle uniforme de tous les types de transformations

$$\begin{pmatrix} x' \\ y' \\ z' \\ w' \end{pmatrix} = \begin{pmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix}$$

$$p' = M p$$

51

Coordonnées homogènes

- La plupart du temps $w = 1$
- Si on multiplie un vecteur par une transformation affine w n'est pas modifié
- On divise par w pour revenir en cartésien

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

52

Transformations affines

Translations :

$$T = \begin{pmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Rotations : Représentation par les angles d'Euler : $R = R_x \cdot R_y \cdot R_z$

$$R_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi & 0 \\ 0 & \sin \phi & \cos \phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$R_y = \begin{pmatrix} \cos \phi & 0 & \sin \phi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \phi & 0 & \cos \phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$R_z = \begin{pmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Changements d'échelles :

$$T = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

53

Associatif ~ compositions

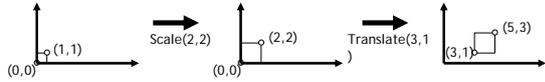
Multiplication de matrices : $p' = T(S p) = TS p$

$$TS = \begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 0 & 3 \\ 0 & 2 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

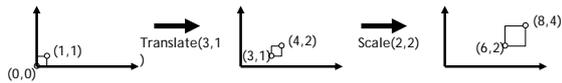
54

Non-commutatif !!!

homothétie puis translation : $p' = T (S p) = TS p$



translation puis homothétie : $p' = S (T p) = ST p$



Hiérarchie de transformations

Objet complexe

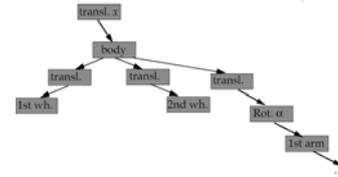
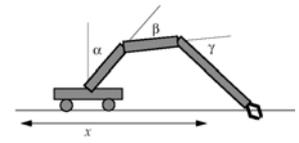
Coordonnées relatives (ex : la roue par rapport au socle)

Concaténation de transformations

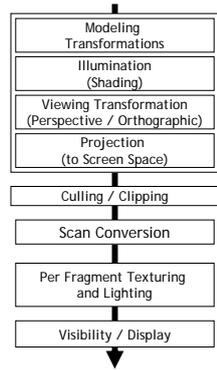
Hiérarchie

⇒ dessiner = parcourir un arbre

⇒ conserve la cohérence



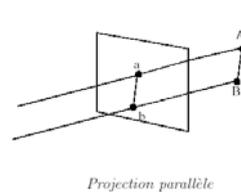
Projections



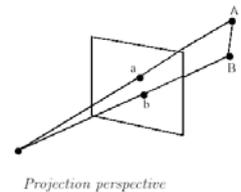
Passes des coordonnées du monde 3D aux coordonnées écran 2D.

Projections

- Utilisées en synthèse et en vision (modèles de caméras)
- Deux grandes familles :



Projection parallèle



Projection perspective

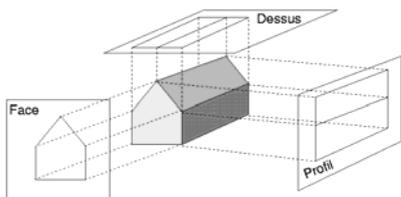
Les projections parallèles :

projection orthographique lorsque la direction de projection est perpendiculaire au plan de projection,

- projection oblique sinon.

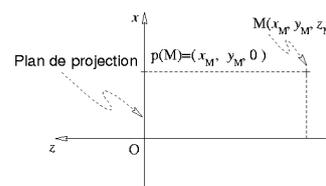
Propriétés géométriques des projections parallèles :

- conservent le parallélisme des droites.
- conservent les rapports des distances parallèles.



Matrice en coordonnées homogènes de la projection orthographique canonique :

$$P(x_M, y_M, z_M, w_M) = (x_M, y_M, 0, 1)$$

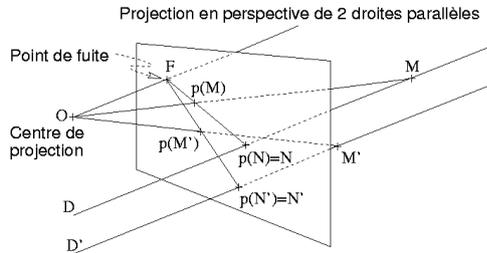


Matrice de la projection orthographique sur xOy

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

Les projections perspectives :

L'image d'un point M par une projection en perspective sur le plan P de centre O est l'intersection de la droite OM avec le plan P.
 Une projection en perspective dont le centre de projection est à l'infini est une projection parallèle.



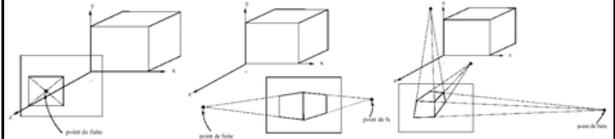
61

Point de fuite :

Si une droite D coupe le plan de projection, il existe un point F, appelé **point de fuite** appartenant à la projection de toute droite parallèle à D.



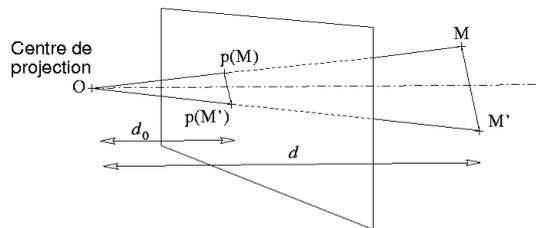
On différencie les projections en perspective par le nombre de points de fuite pour les directions des axes du repère.



Propriétés géométriques des projections en perspective

- Les projections perspectives ne conservent pas le parallélisme des droites non parallèles au plan de projection.
- La taille d'un objet est inversement proportionnelle à sa distance au point de projection :

$$|p(M) p(M')| = d_0 \cdot 1/d \cdot |MM'|$$



63

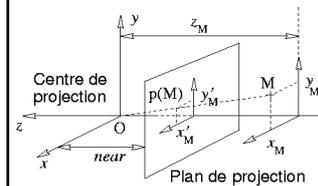
Calcul des coordonnées projetées en perspective

On se place dans le cas d'une projection canonique :

Centre de projection à l'origine et plan de projection parallèle à xOy.

Coordonnées dans le plan de projection

$$P(x_M, y_M) = ((near \cdot x_M) / (-z_M), (near \cdot y_M) / (-z_M))$$



Matrice de la projection de centre O sur le plan $z = near$

$$\begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/near & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

64

Plan

- Historique
- Pipeline graphique
- Transformations
- Image, couleur
- Rasterisation
- Visibilité
- Illumination

65