

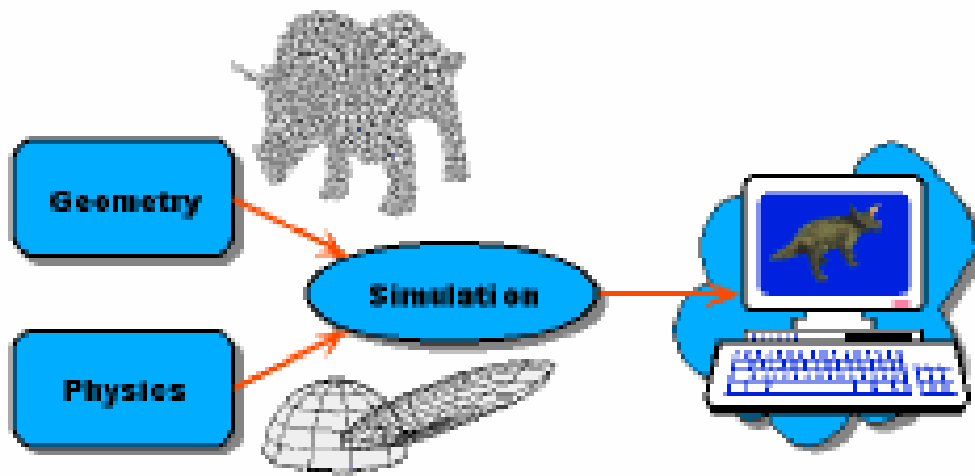
RENDU BASE-IMAGES

1. Introduction

le rendu vu jusqu'à présent était basé sur la géométrie (triangles, ...) et la physique

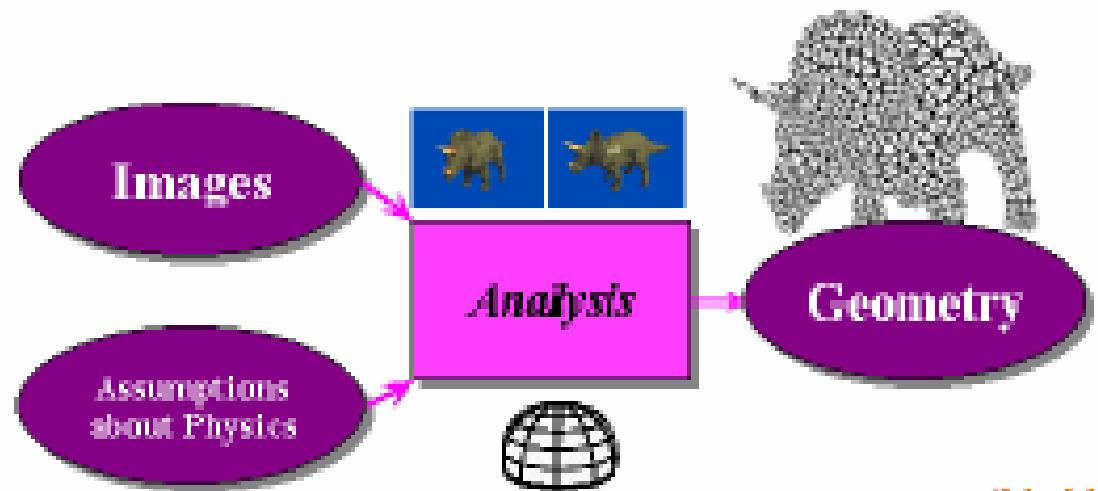
pour avoir plus de réalisme, l'idée est de partir d'un ensemble d'images acquises à partir de la réalité

⇒ mélange d'informatique graphique et de vision par ordinateur



Traditional
Computer
Graphics

Traditional
Computer
Vision



(McMillan)

ce domaine a beaucoup évolué et peut être schématisé par :

← moins de géométrie

plus de géométrie →

rendu sans
géométrie

rendu avec
géométrie implicite

rendu avec
géométrie explicite

lightfield
mosaïques concentriques
mosaïques

lumigraphe
morphing de vue
interpolation de vue

placage de textures
déformations 3D
textures dépendant
du point de vue

2. Rendu sans géométrie

Les méthodes de cette catégorie reposent sur la **fonction plénoptique**

2.1. Fonction plénoptique

définit en chaque point (x,y,z) de l'espace l'énergie radiante dans toutes les directions (θ, φ) , à tout instant t et pour toute longueur d'onde λ

$$p = P(x, y, z, \theta, \varphi, t, \lambda)$$



suivant les contraintes que l'on impose à cette fonction, on obtient des cas particuliers :

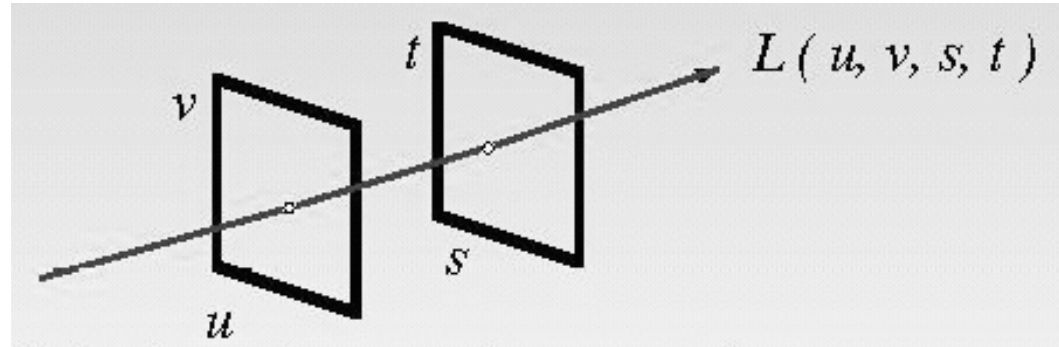
- ✓ si on ne tient pas compte du temps (scène statique), ni de la longueur d'onde, la fonction se réduit à 5 dimensions
- ✓ l'espace de vue est une boîte englobante \Rightarrow 4D : **lightfield, lumigraph**
- ✓ l'espace de vue est un plan englobant \Rightarrow 3D : **mosaïques concentriques**
- ✓ le point de vue est fixé \Rightarrow 2D : **panorama sphérique ou cylindrique**

2.2. Champ de lumière (light field) ou Lumigraph

➤ **principe** : sans occlusion, couleur identique sur une droite
⇒ permet de tourner autour d'un objet

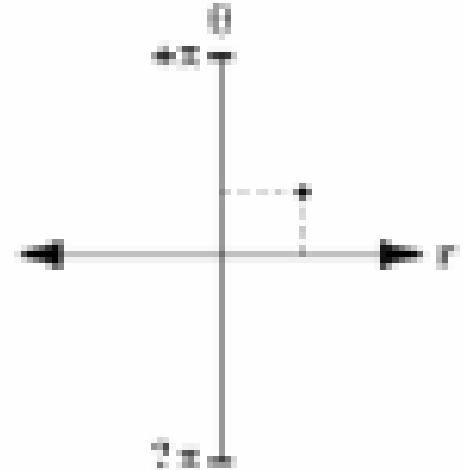
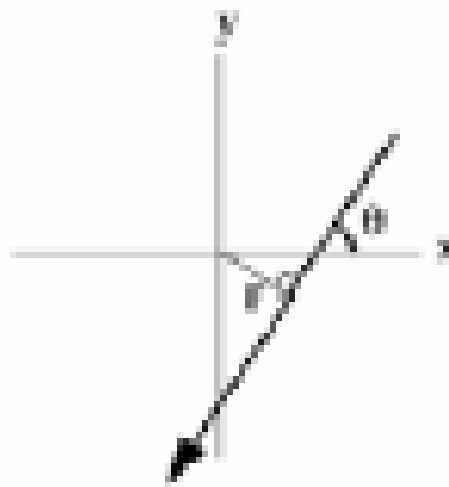
image est une tranche 2D d'une fonction 4D

➤ **paramétrisation** grâce à deux plan parallèles : intersection en (u, v) et (s, t) → $L(u, v, s, t)$



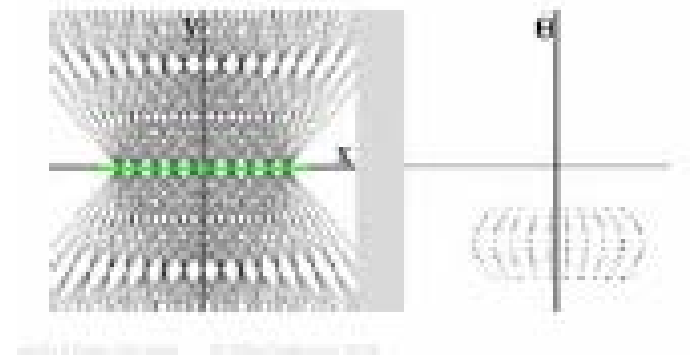
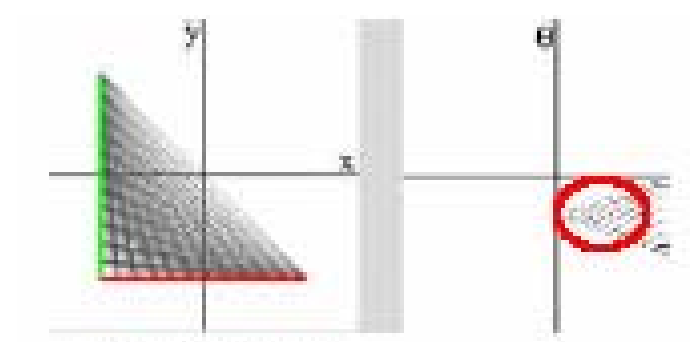
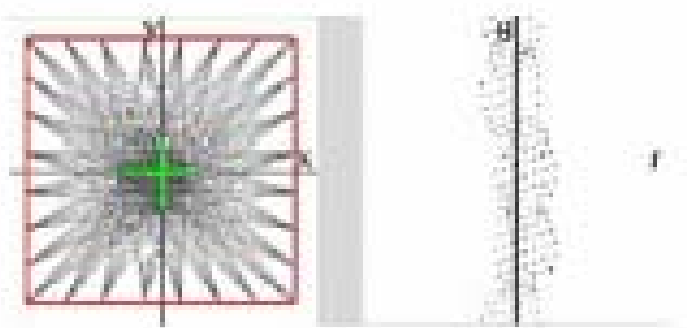
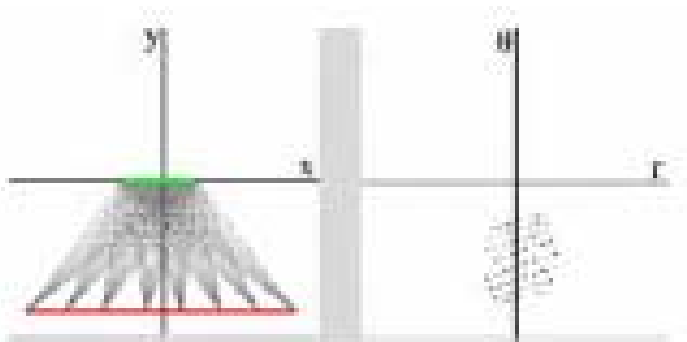
plusieurs jeux de plans pour avoir toutes les directions

espace des droites : droite définie par (r, θ)

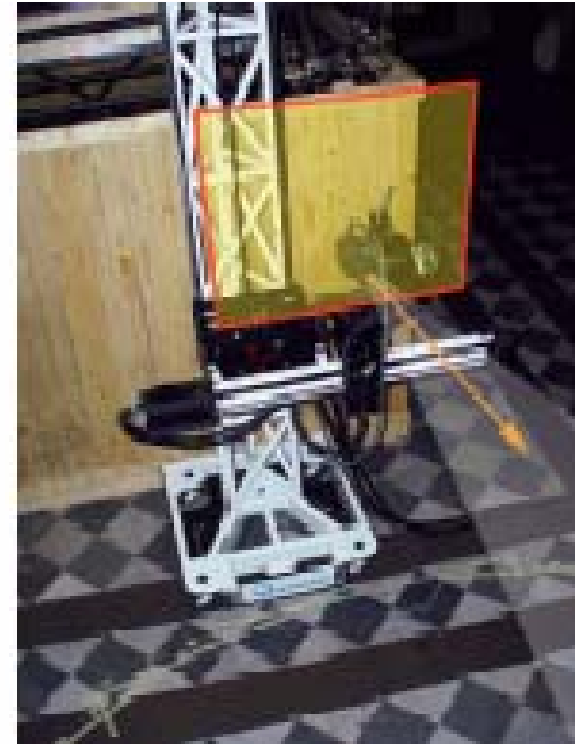


idéalement, il faudrait être
complet en θ
le plus large possible en r

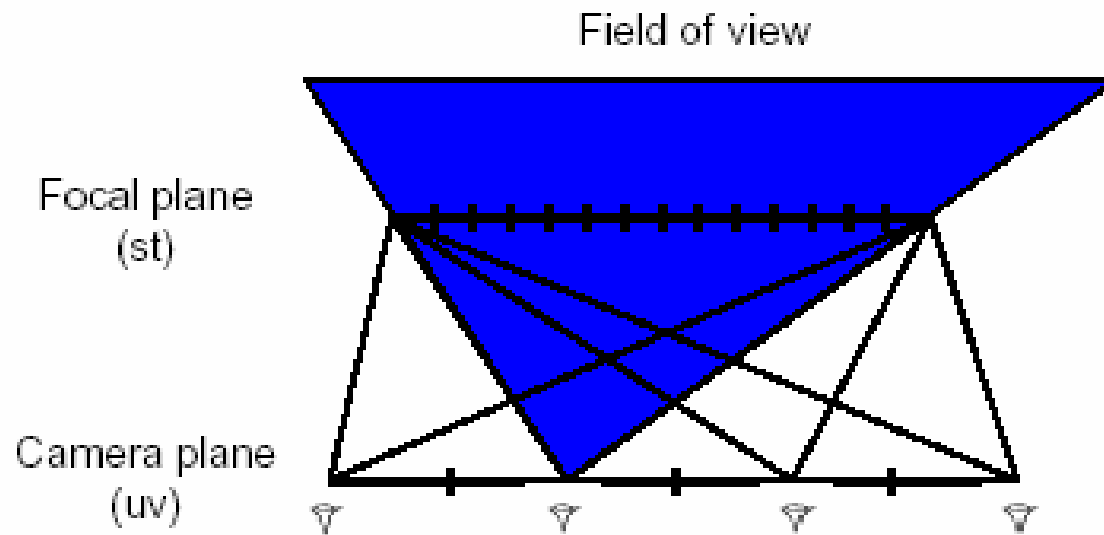
exemples d'échantillonnage



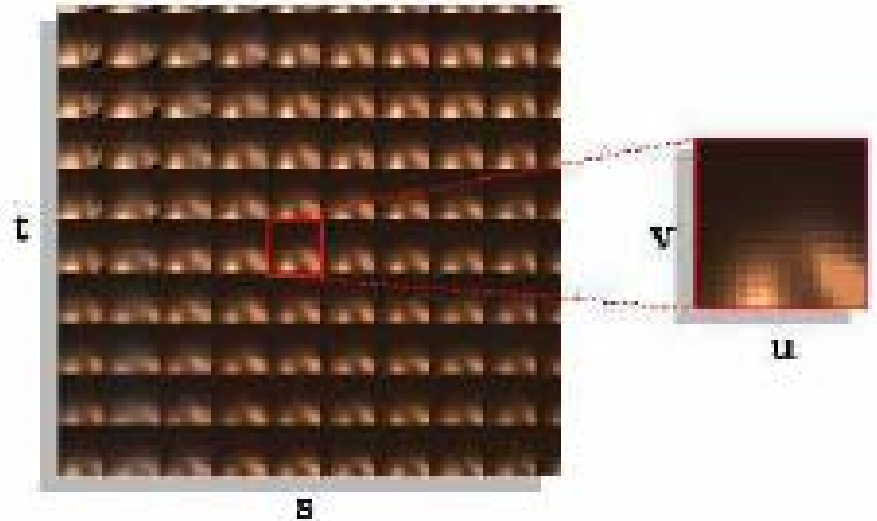
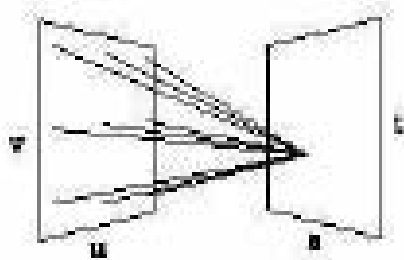
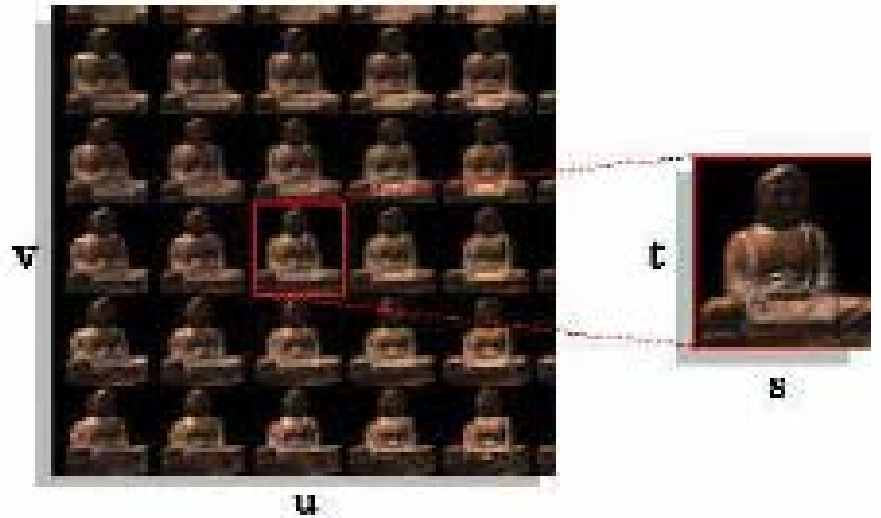
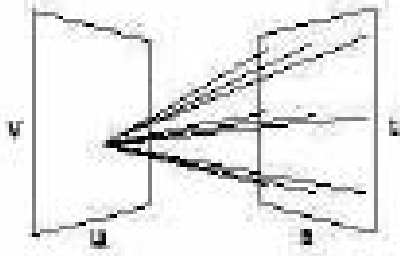
- **création** du champ de lumière :
caméra contrôlée par ordinateur
mouvement plan
pour avoir 4 vues, il faut faire tourner l'objet et la lumière en même temps



(Stanford university)

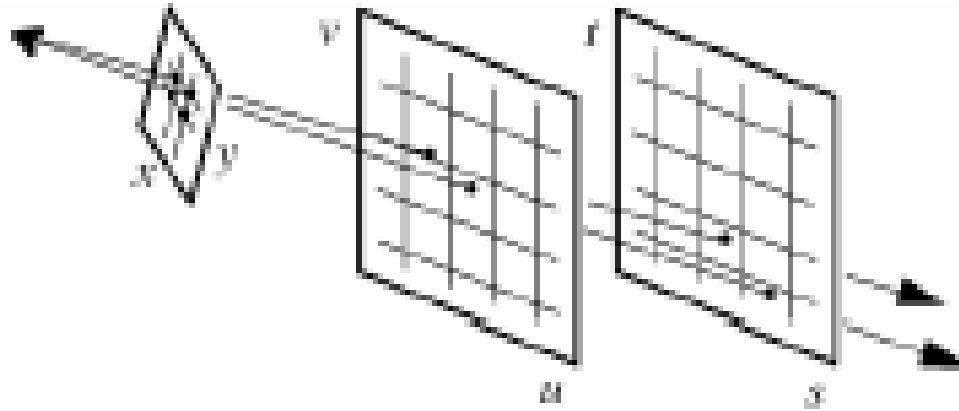


Exemple :

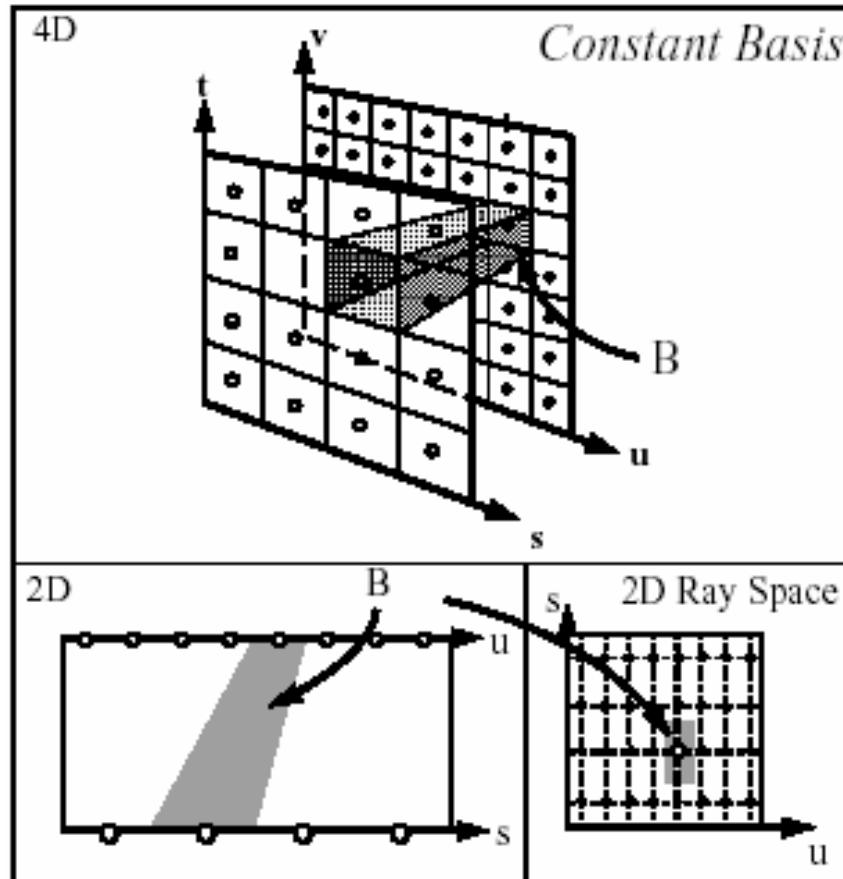


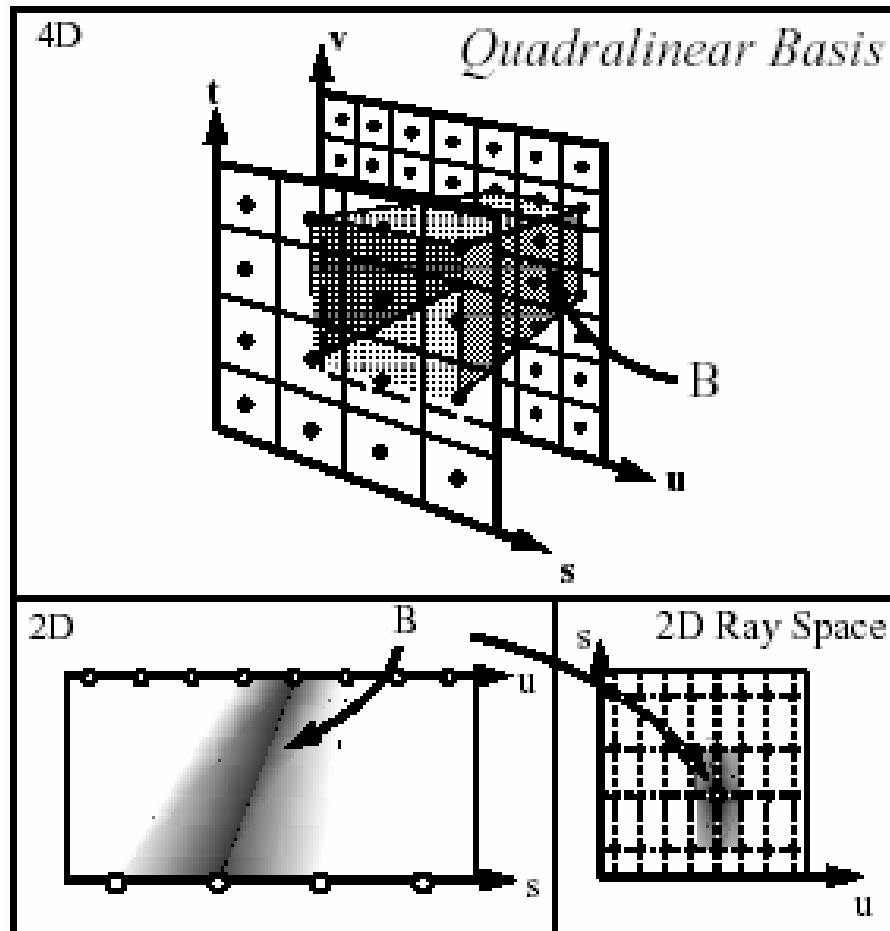
➤ **affichage** :

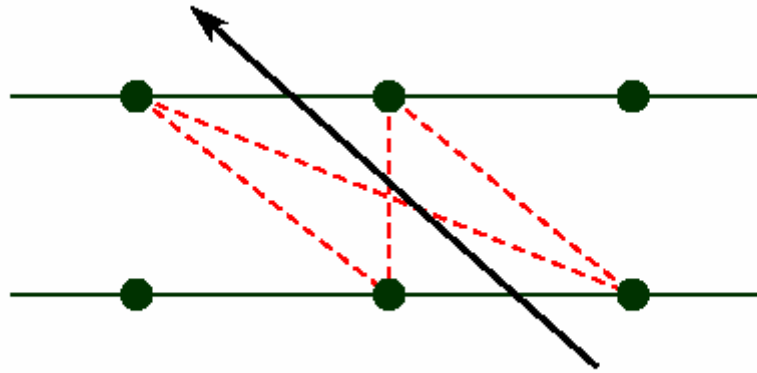
un rayon passant par le pixel (x,y) coupe les deux quadrilatères en (u,v) et (s,t)



en fait, il faut **interpoler** par rapport aux voisins :







(Gortler96)



plus proche



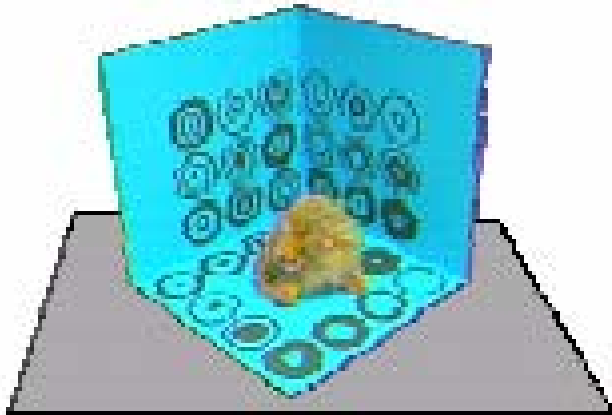
bilinéaire



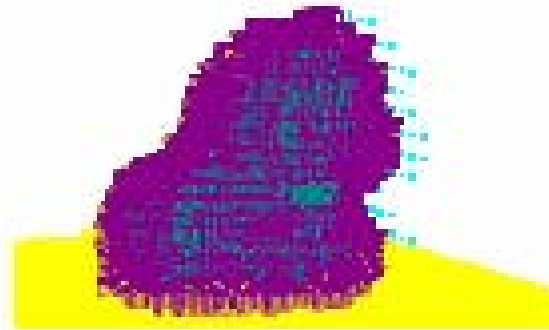
quadrilinéaire



le lumigraph



acquisition



modèle volumique



nouvelle vue

2.3. Mosaïques concentriques

la caméra est contrainte à se déplacer suivant des cercles concentriques dans un plan

⇒ 1 image (1 rayon plénoptique) est indexée par 3 paramètres : le rayon du cercle, l'angle de rotation et l'élévation verticale

une nouvelle vue est créée en combinant les images appropriées lors du rendu

avantage : moins gourmand en place mémoire que les light fields

inconvénient : ne prennent pas en compte les effets de parallaxe



2.4. Panoramas

le point de vue de capture est réduit à un point

Chen a construit des images panoramiques cylindriques à 360° , en projetant explicitement toutes les images sur un cylindre

on peut aussi associer une matrice de transformation (matrice de rotation et éventuellement changement de focale) à chaque vue

il faut utiliser des algorithmes d'alignement (globaux et/ou locaux) pour aligner les images



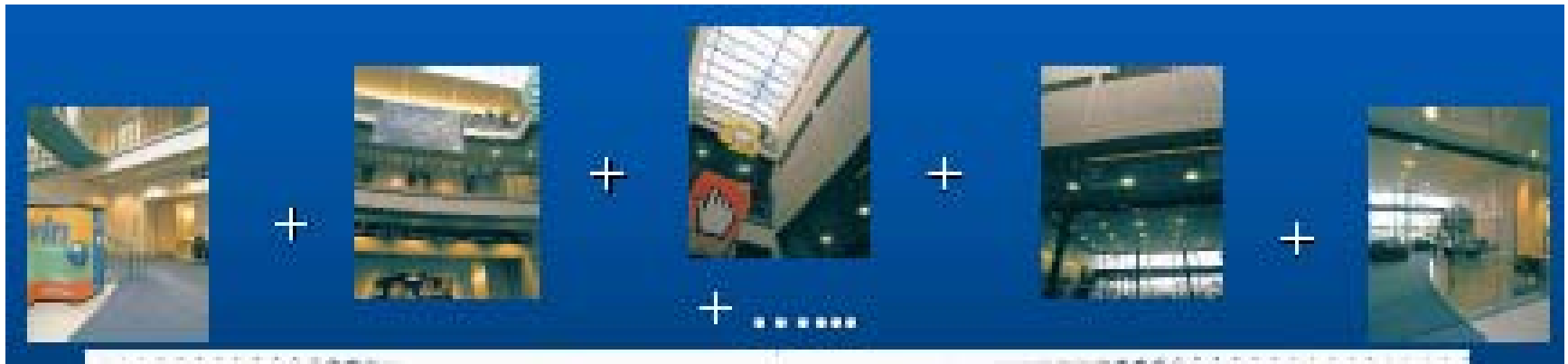
distorsion d'image



si la focale de la caméra est fixe et connue, on peut projeter chaque image sur un cylindre et la relation entre les images devient une simple translation



Exemple : carte sphérique obtenue à l'aide de 3 séquences d'images créées à l'aide d'une seule caméra avec 3 angles d'inclinaison (22 images, 22 images et 10 images)

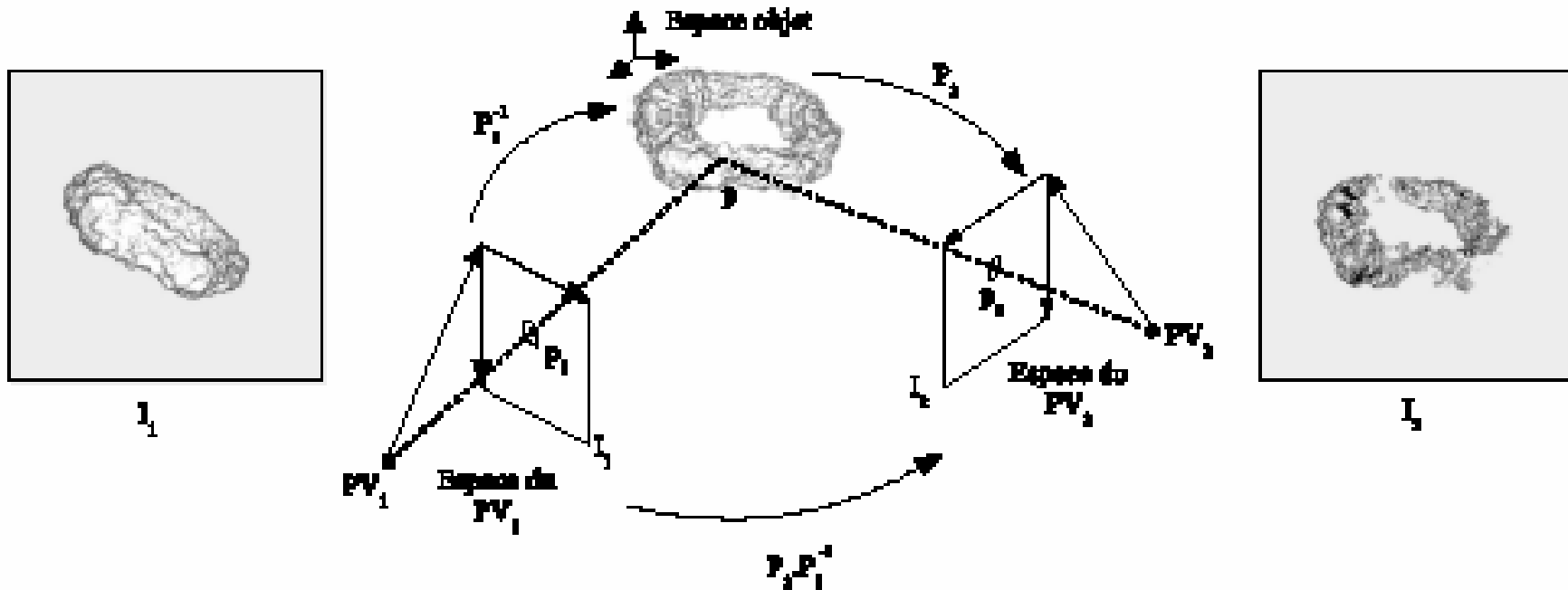


3. Rendu avec géométrie implicite

ici, l'information 3D est implicite : elle est calculée en utilisant seulement les calculs de projection usuels ou des calculs de profondeur

3.1. Reprojection de pixels

utilise deux changements de repère successifs pour chaque point de l'image initiale



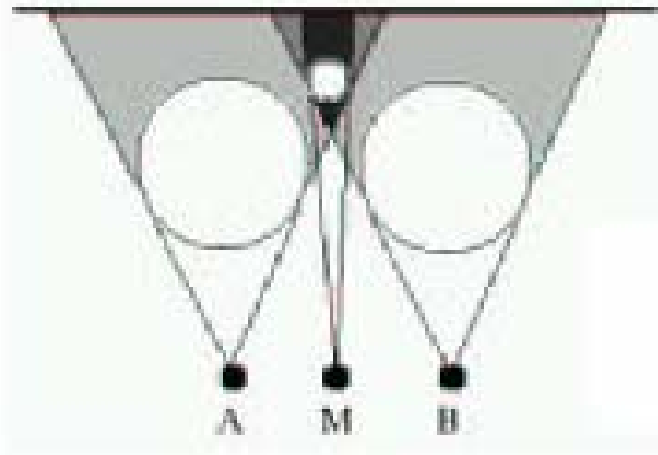
si P_1 est la matrice de projection associée à l'image 1,

$$p = P_1^{-1} \cdot p_1$$

l'image p_2 du point p dans l'image 2 est $p_2 = P_2 \cdot p$, soit

$$p_2 = P_2 \cdot P_1^{-1} \cdot p$$

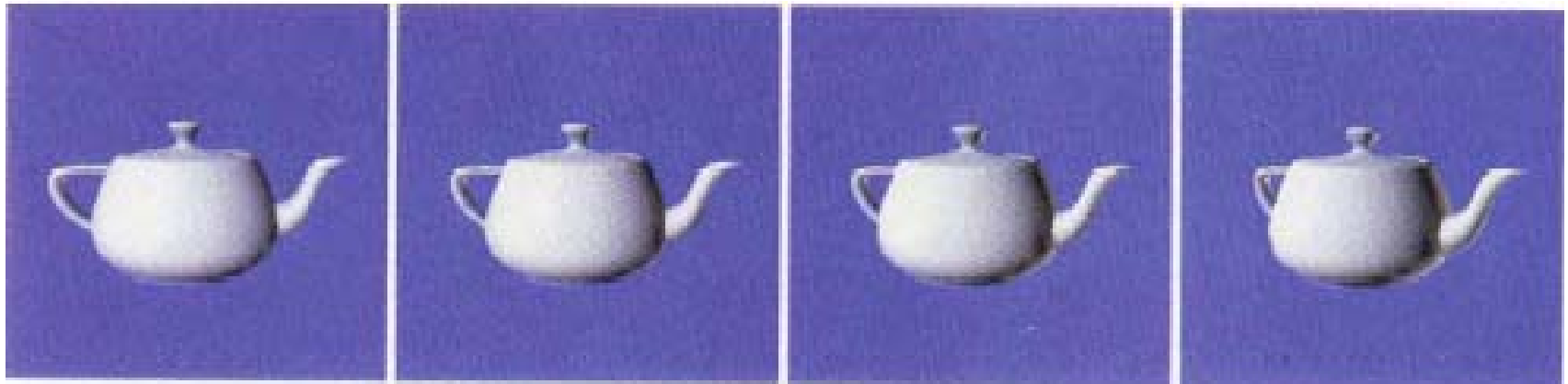
- ✓ nécessite un tampon de profondeur pour l'image 2 (plusieurs points de l'image initiale peuvent se projeter sur le même pixel)
- ✓ problèmes de visibilité et d'échantillonnage pas traités (trous)



(McMillan)

3.2. Interpolation de vue (morphing)

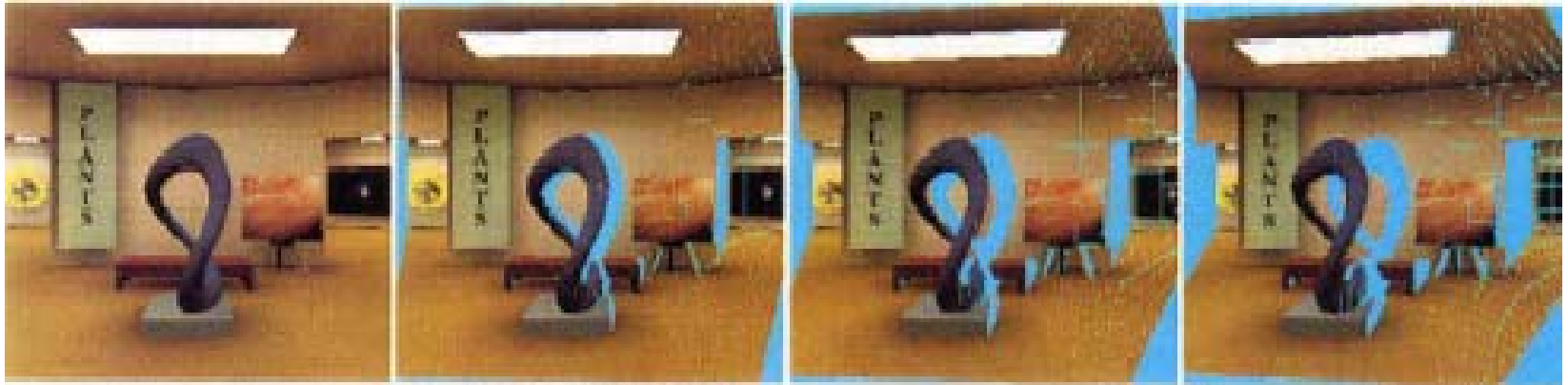
technique permettant de créer des images intermédiaires entre 2 images par interpolation des couleurs et des formes => des correspondances sont établies entre les 2 images, puis ces éléments sont interpolés



ne fonctionne que si les caméras de référence sont parallèles

3.3. Interpolation de point de vue

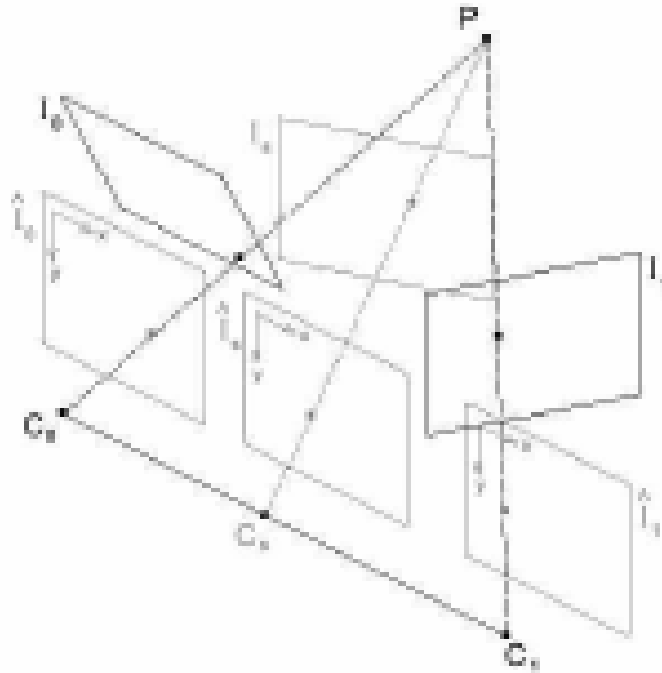
- ✓ établissement de correspondances entre images de référence
- ✓ chaque pixel se déplace dans l'image de son emplacement d'origine jusqu'à son emplacement d'arrivée en suivant une ligne droite

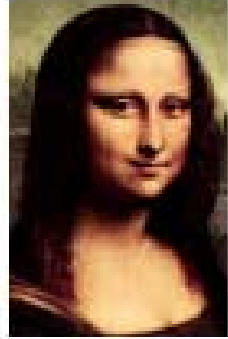
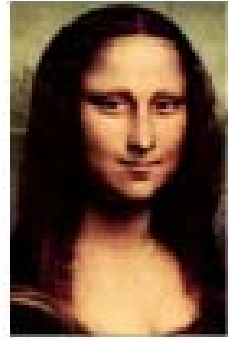
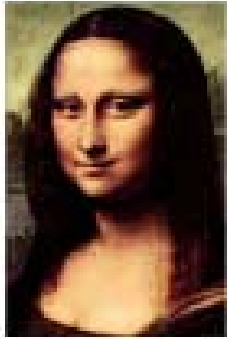


(Chen and Williams, 1993)

3.4. Déformation de point de vue

extension de la méthode précédente où les images de référence sont pré déformées (pour obtenir des images parallèles) ; une fois l'interpolation linéaire effectuée, l'image est re-déformée

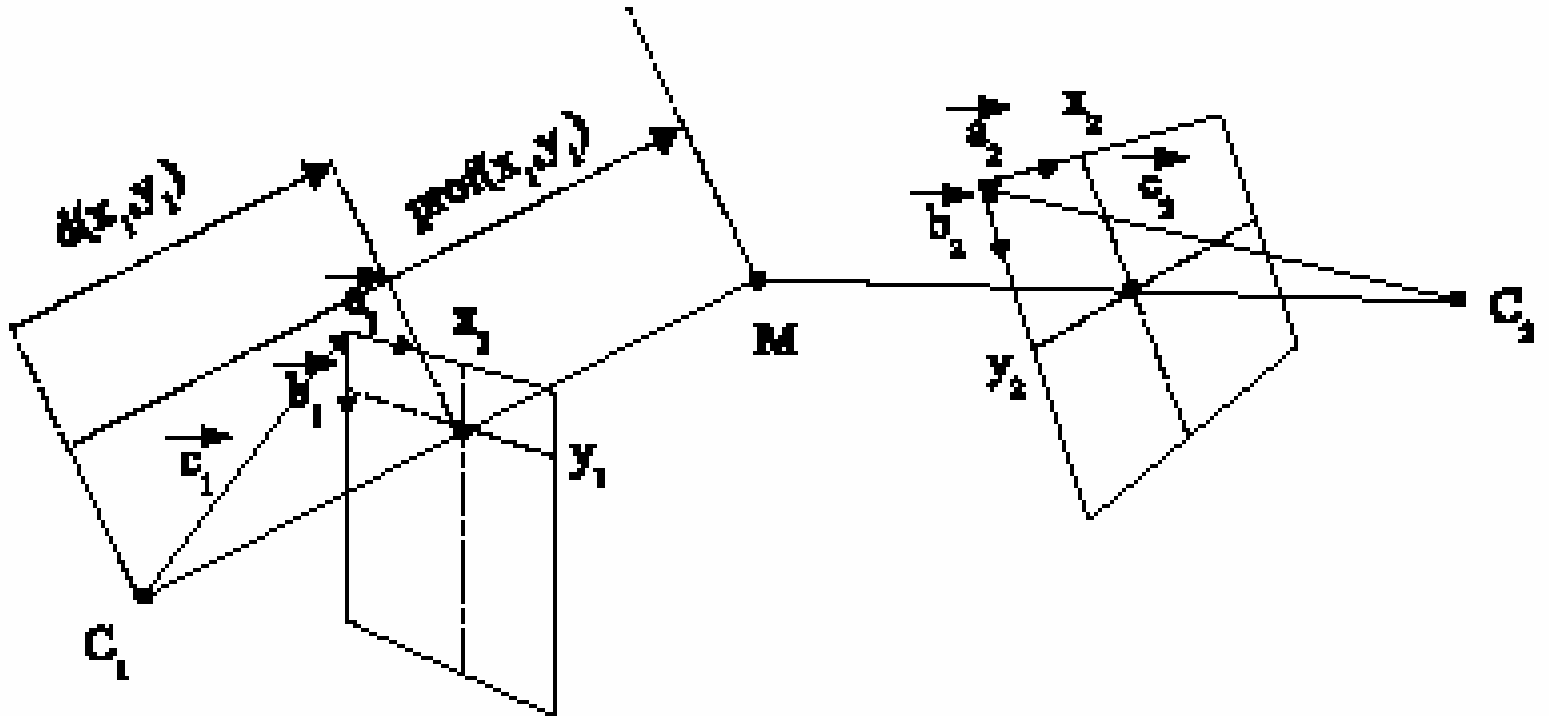




4. Rendu avec géométrie explicite

4.1. Déformations 3D

formalisation des équations de reprojection en utilisant la notion de disparité généralisée



$$d(x,y) = \text{profondeur}(x_1,y_1)/d(x_1,y_1)$$

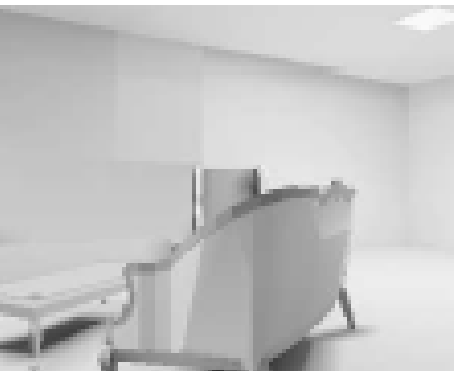


Image de référence 1

**Déplacement 3D
selon le nouveau
point de vue**



**Correction
de l'angle**

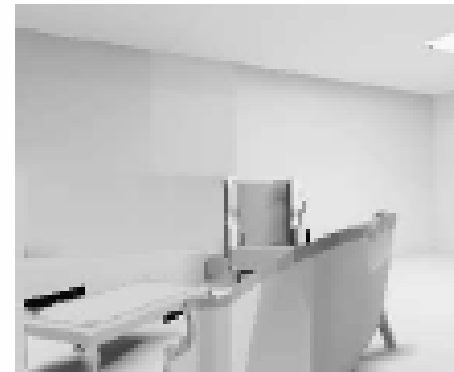
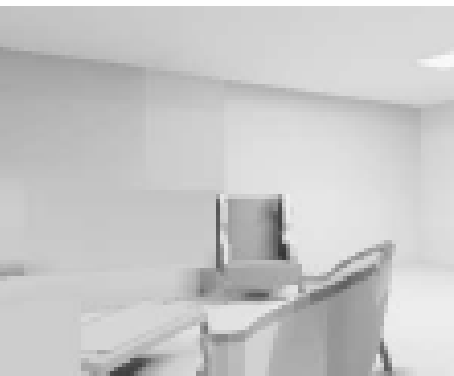


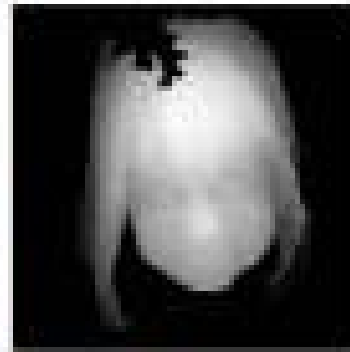
Image obtenue



**Déplacement 3D
selon le nouveau
point de vue**

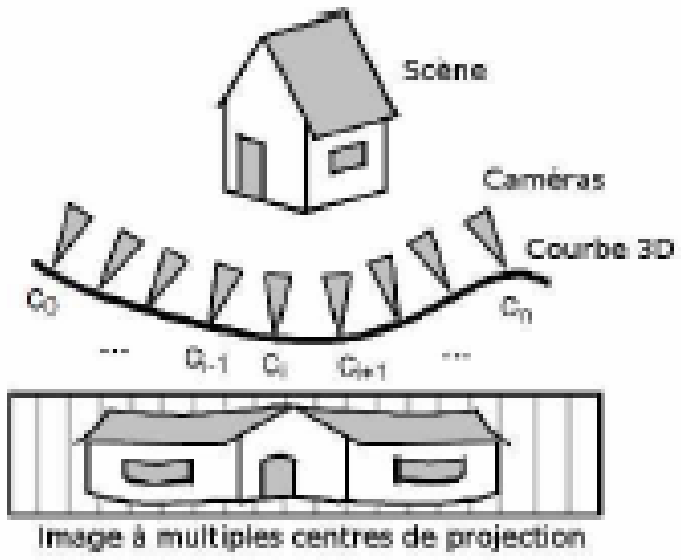


(Mark et al. 97)



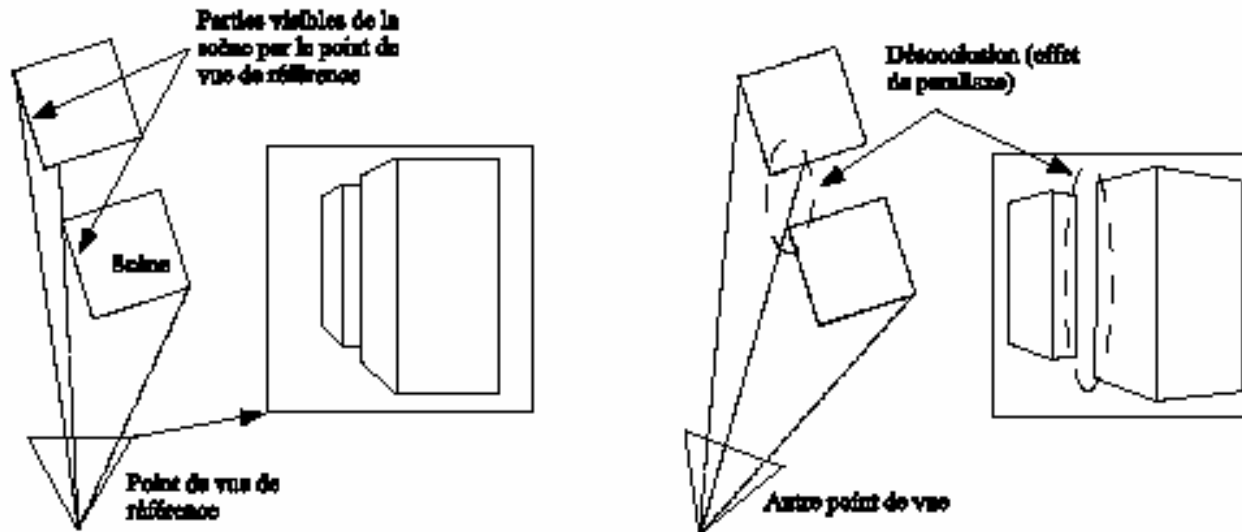
4.2. Images à multiples centres de projection

- ✓ on connaît un chemin 3D positionné par l'utilisateur
- ✓ une structure est créée avec l'image et la position de la caméra
- ✓ cette structure est utilisée pour effectuer le rendu de la scène pour tous les points de vue intermédiaires entre les positions d'échantillonnage



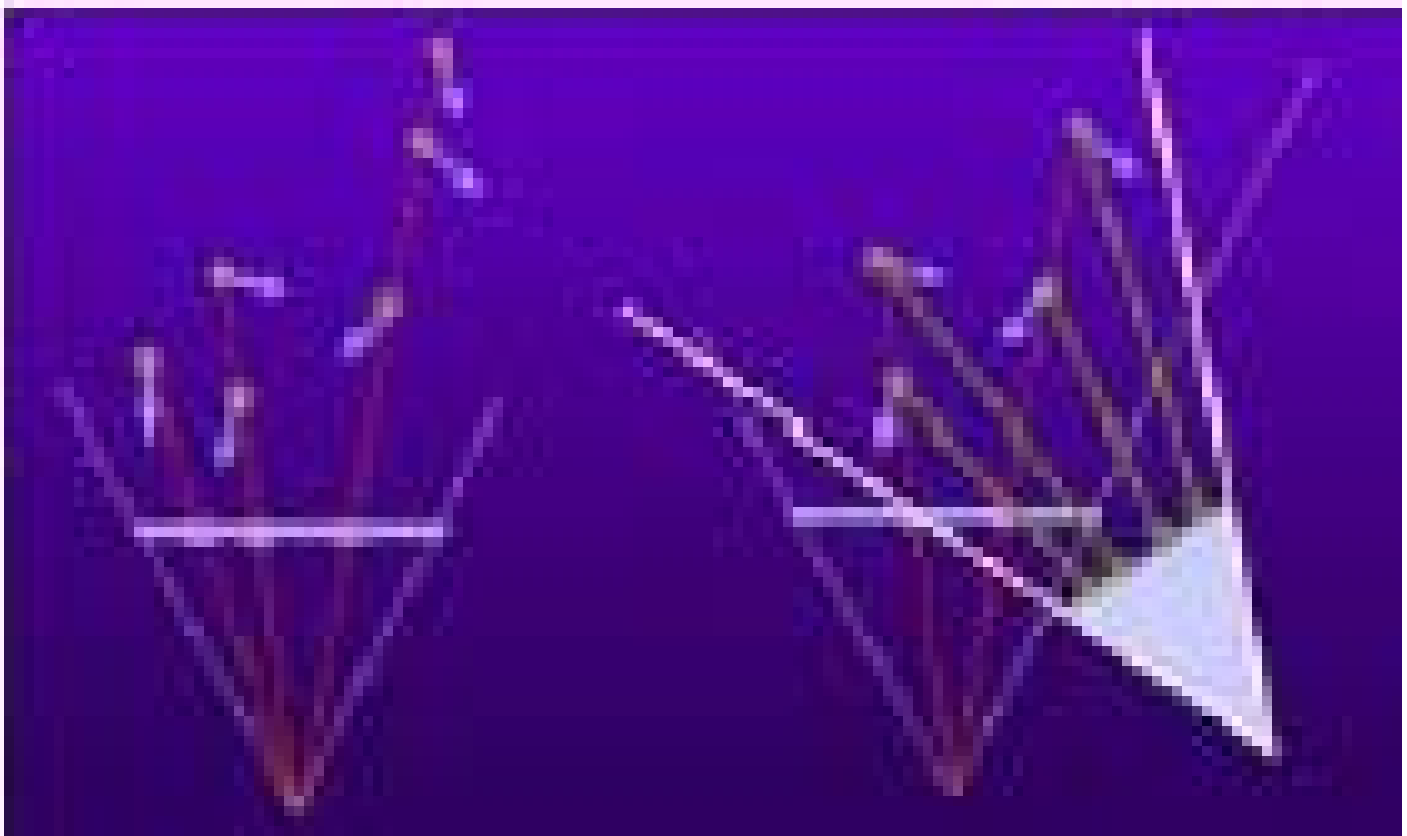
4.2. Images à plans de profondeur (layered depth images = LDI)

problèmes d'occlusion => une seule image ne suffit pas pour générer un nouveau point de vue ;



les LDI ont été introduites pour résoudre ce problème : une LDI stocke tout ce qui est derrière la surface visible

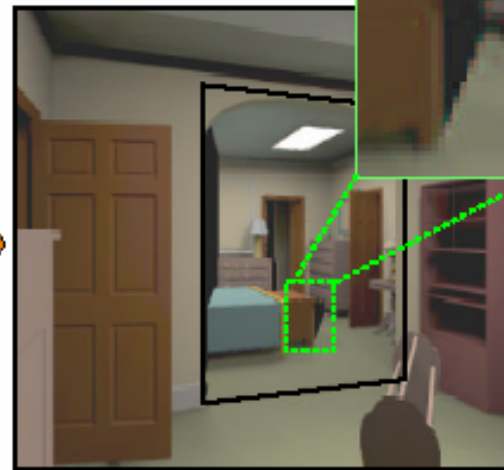
- ✓ trous comblés par ce qui est derrière
- ✓ projection pixel par pixel, d'arrière en avant
- ✓ utilisation de l'écrasement de point (splatting)



- Multiple samples per pixel at different depths



Reference Image



Warped Depth Image



Reference Image



Warped Layered Depth Image
(Popescu)



(Gortler et al)

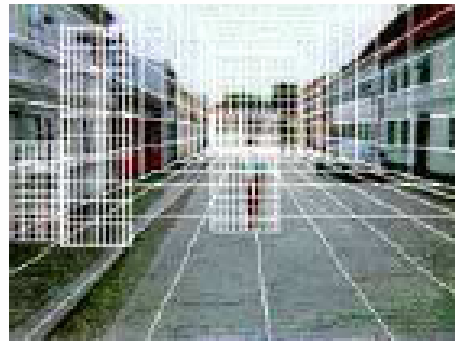
5. Imposteurs

- ✓ calcul hors écran d'une partie de la scène
- ✓ l'image créée est utilisée ensuite comme texture plaquée sur un rectangle positionné dans la scène

5.1. Billboards et sprites

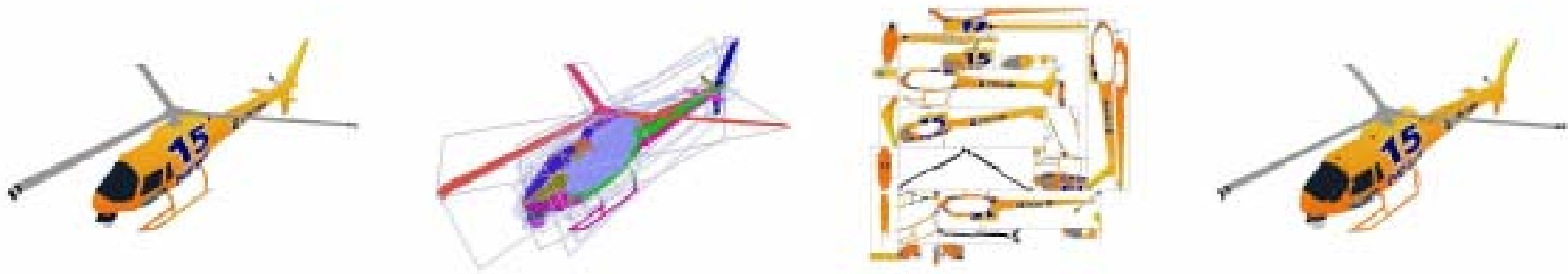
- ✓ billboard = plan texturé représentant un objet complexe placé dans la scène (nuages, arbres, ...)
 - => billboards toujours parallèles à l'écran
 - => billboards orientés vers l'observateur
 - => billboards fixes dans l'espace de la scène
- ✓ sprite = plus petit, rectangle 2D parallèle à l'écran

Billboards



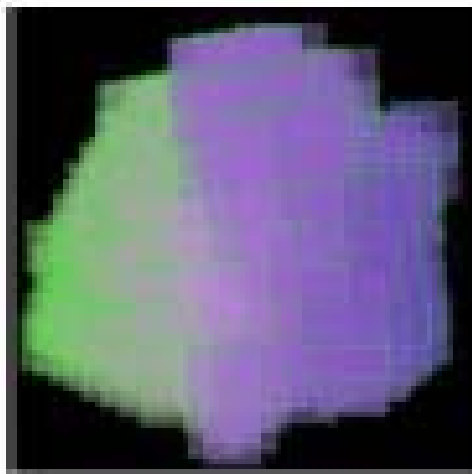
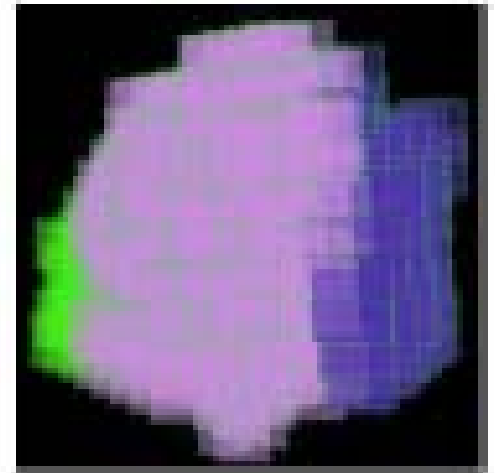
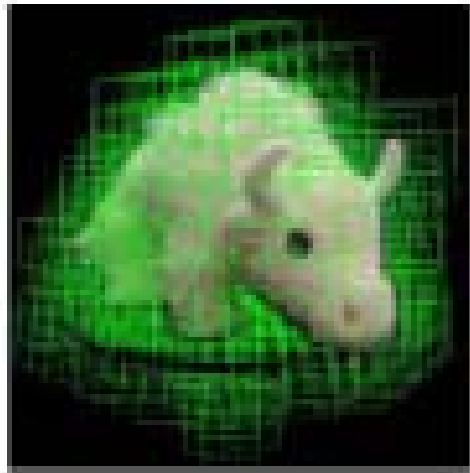
Horry et al 97)

- ✓ billboards extraits de la photographie
- ✓ ils sont déformés (warping) en fonction de leur point de fuite



(Décoret et al. 03)

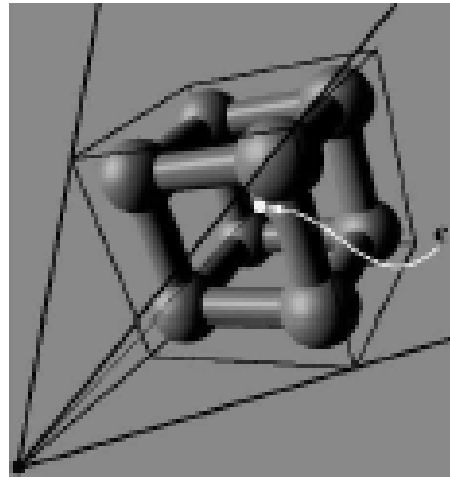
on remplace l'objet par la combinaison de plusieurs billboards représentant l'objet sous différents angles



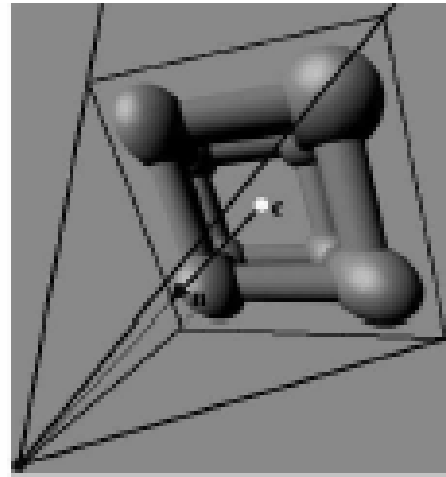
(Yamazaki et al. 02)

5.2. Imposteurs dynamiques

génération d'un imposteur par objet lors du rendu



objet original

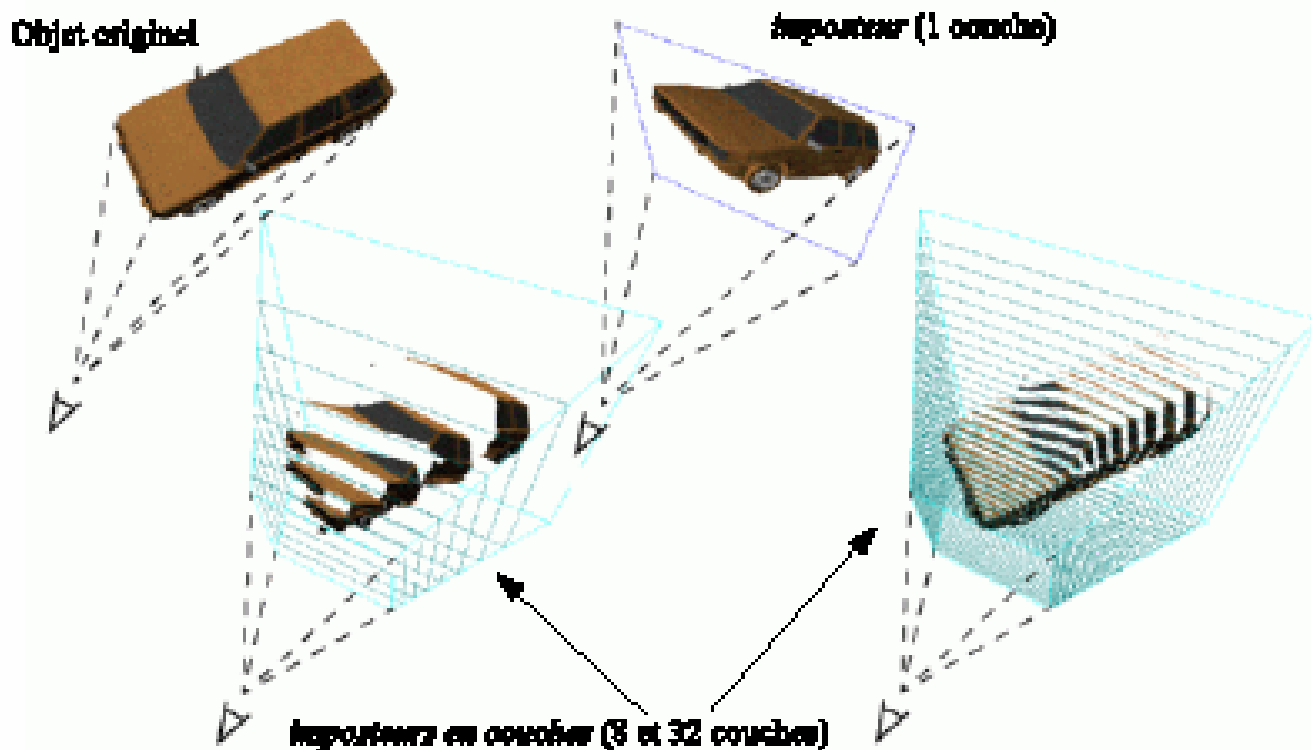


imposteur

(Schaufler 95)

5.3. Imposteurs en couches

un imposteur en couches est composé de plusieurs polygones transparents



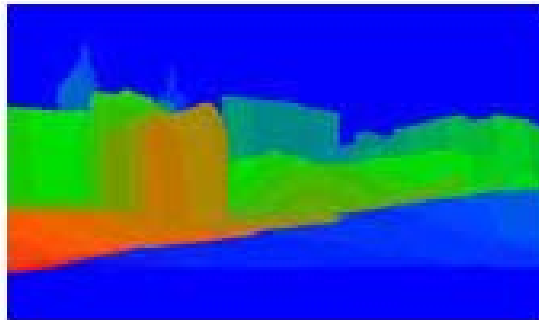
(Schaufler 98)

5.4. Imposteurs maillés

on utilise non plus une image mais un maillage de triangles (pour des zones à peu près plates) sur lequel on plaque l'image pour remplacer la géométrie distante



(a) Texture de l'imposteur



(b) Tampon de profondeur



(c) Imposteur maillé

(Sillion et al. 97)

6. Conclusion

l'utilisation d'images

- ✓ offre une qualité visuelle parfaite
- ✓ offre un affichage indépendant de la complexité géométrique
- ✓ nécessite des informations partielles sur le monde
- ✓ combien d'images, de quels points de vue ?
- ✓ comment interpoler entre images ?
- ✓ taille mémoire