

**IFT 3740 & IFT 6845**  
**Animation par Ordinateur**

**9. Intéraction Réel - Synthétique**

Sébastien Roy

Département d'informatique et de recherche opérationnelle

Université de Montréal

été 2000

# Intéraction Réel - Synthétique

## Infographie

Génération d'images synthétiques ( Modèles → Images )

## Vision

Analyse **automatisée** d'image réelles ( Images → Modèles )

Plusieurs problèmes touchent à ces deux domaines:

### **Modélisation à partir d'images (image based modelling)**

Construire un modèle 3D d'un objet réel à partir de plusieurs images de cet objet, sous des points de vue différents.

### **Composition d'objets synthétiques dans une scène réelle**

Ajouter un dinosaure synthétique dans un décor réel (en général un vidéo).

### **Composition d'objets réels dans un univers synthétique**

Acteur réel filmé devant un "écran bleu", puis ajouté à la scène 3D. (Exemple simple: la météo)

### **Capture de mouvement (Mouvement réel → mouvement synthétique)**

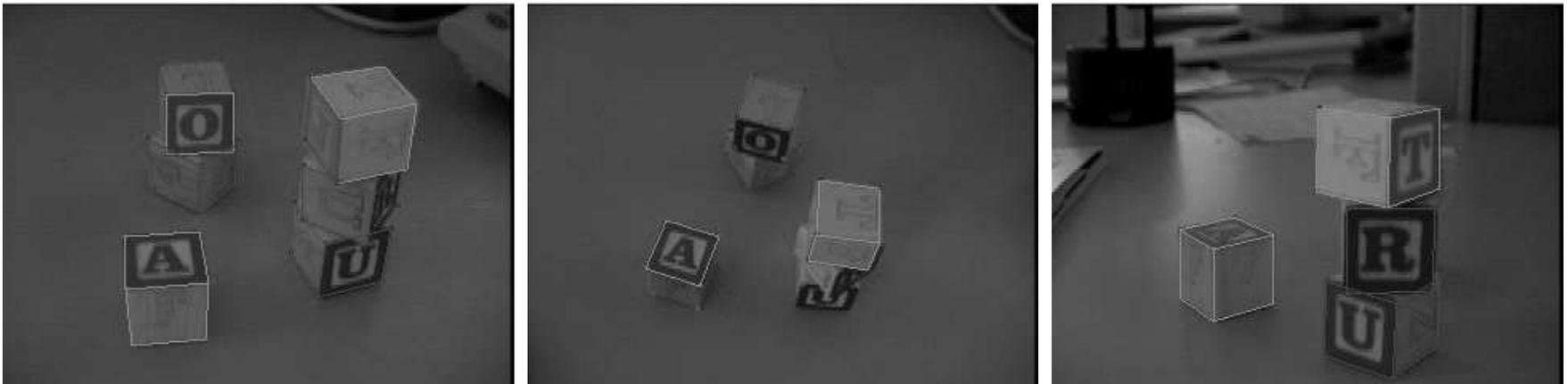
Enregistrer (par des méthodes optiques, magnétiques, ou ultrasons) le mouvement d'un acteur réel pour l'associer ensuite à un acteur synthétique.

## Modélisation à partir d'images (image based modelling)

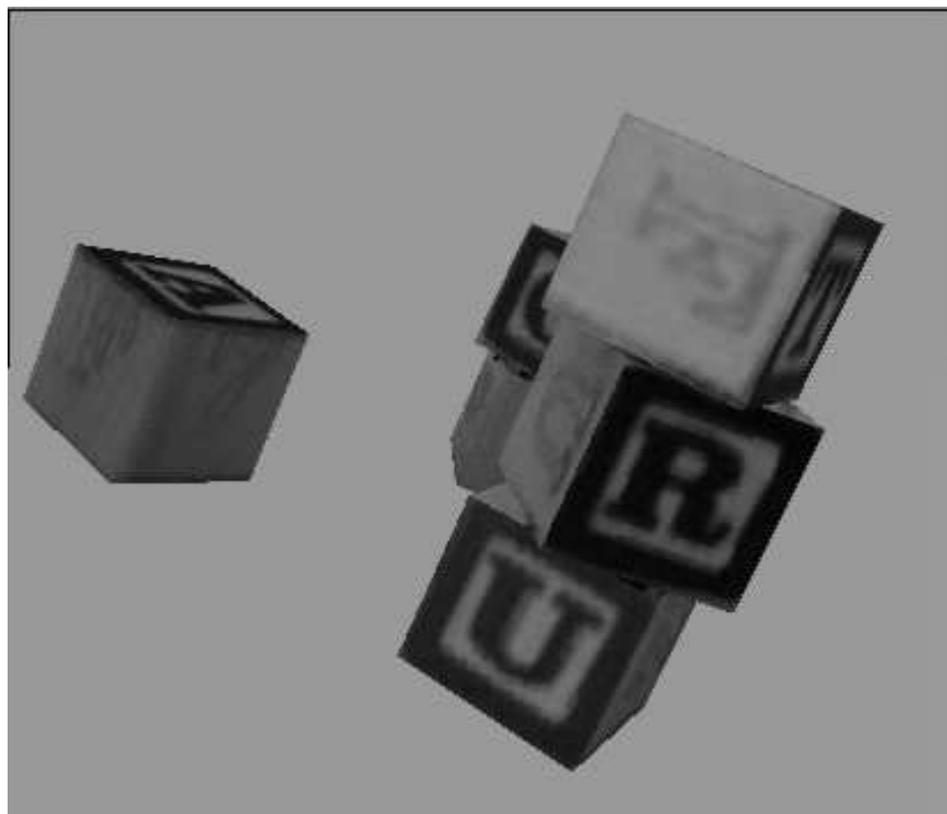
**But:** Construire un modèle 3D d'un objet réel à partir de plusieurs images de cet objet, sous des points de vue différents.

### Système Rekon (lab d'infographie, UdeM)

- Plusieurs images d'un même objet sous des points de vue différents
- Correspondance établie manuellement
- Caméra affine calibrée automatiquement
- Polygones et textures extraits automatiquement



**Résultat:** Un modèle 3D de l'objet réel.



# Modélisation à partir d'images (image based modelling)

## Problèmes à résoudre

- Mise en correspondance (ou *tracking*) entre deux ou plusieurs images (tel point de l'image 1 est à telle position dans l'image 2...)
- Déterminer les modèles des caméras à partir des points de correspondance.
- Chaque point de correspondance existe maintenant en 3D.
- Reconstruire des faces entre les points de correspondance.

## Questions importantes

- Est-ce que l'on connaît quelque chose de la scène?  
Chaque information *a priori* aide...
  - Information sur les objets
    - Dimensions absolues
    - Géométrie connue (cube, mur plat, lignes perpendiculaires, fenêtres alignées, etc.)
    - ...
  - Information sur la caméra
    - Paramètres internes (distance focale fixe? angle de vue connu?)
    - Paramètres externes (caméra sur un trépied ou libre?)
    - ...

# Le problème de mise en correspondance

## Mise en correspondance

Pour un point d'une image, trouver la nouvelle position de ce point dans les autres images.

Photo: On observe de grands déplacements (nécessite souvent une intervention manuelle)

Vidéo: Les déplacements sont petits (analyse automatique plus facile)

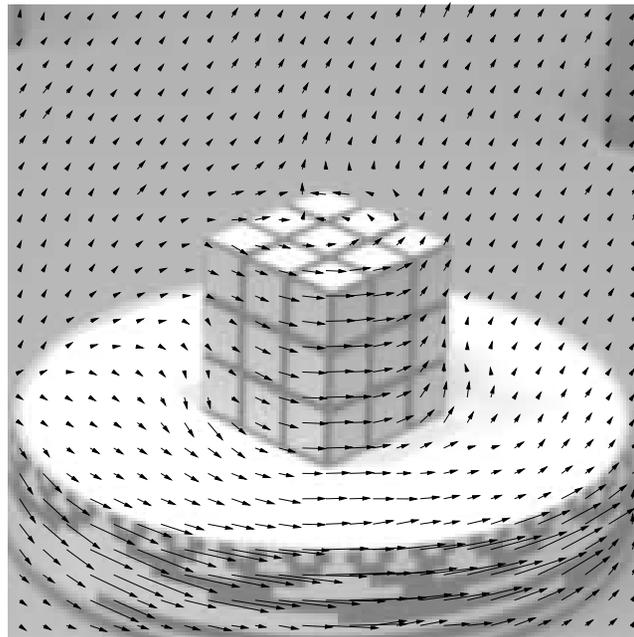
La composante "temps" transforme les déplacements en vitesse.

Le problème est celui du **tracking** ou du **flux optique**.

# Flux optique

## Flux optique

Champs de vitesse décrivant le mouvement apparent des patterns d'intensités de l'image.



# Le problème de la calibration de caméra

## Calibration de caméra

La caméra projète un point 3D du monde sur une image 2D, selon un modèle donné (va de très simple à très compliqué).

La calibration consiste à retrouver les paramètres du modèle de caméra à partir de paires (point 3D, point 2D projeté).

## Paramètre externes

Position et orientation de la caméra dans le monde.

S'exprime par une matrice de translation 3D et de rotation 3D.

## Paramètre internes

Tous les paramètres qui ne changent pas lorsqu'on déplace la caméra.

- Distance focale
- Centre de l'image
- Ratio des pixels
- Distorsion radiale (en général, considéré négligeable)
- ...

S'exprime par une matrice  $2 \times 3$  (transformation affine 2D).

# Calibration de caméra

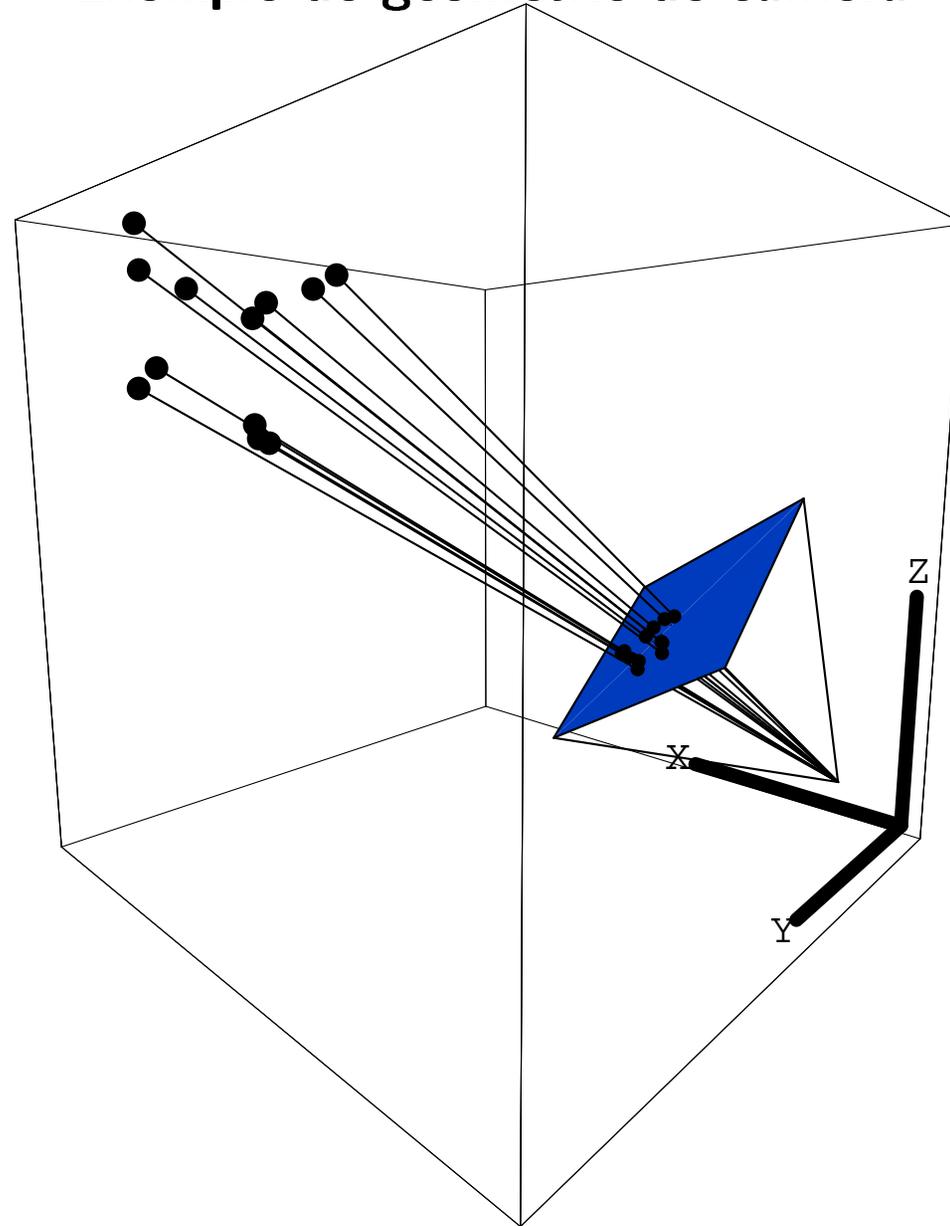
## Problème:

Calculer les paramètres d'une caméra en utilisant uniquement une ou plusieurs images d'un objet de calibration.

$$\begin{bmatrix} x \\ y \\ h \end{bmatrix}' = \mathbf{M} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}'$$

- Quels paramètres?
- Quel objet de calibration?

# Exemple de géométrie de caméra



# Types de calibration

## Choix du modèle de caméra

- orthographique
- perspective faible (ou orthographique + échelle)
- perspective
- affine

## Choix des Paramètres à déterminer

- paramètres internes et externes
- seulement les paramètres internes
- seulement les paramètres externes (paramètres internes connus)

## Choix du Type d'objet de calibration

- Objet de calibration complètement connu (cube)
- Objet connu en partie (lignes parallèles, surface plane, ...)
- Objet inconnu (scène arbitraire)

# Modèles de caméra

Perspective, Perspective faible, Orthographique

$$\begin{bmatrix} x \\ y \\ h \end{bmatrix}' = \mathbf{JART} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}'$$

**Affine**

$$\begin{bmatrix} x \\ y \\ h \end{bmatrix}' = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}'$$

## Projection J

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

perspective

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \bar{Z} \end{pmatrix}$$

perspective faible

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

orthographique

## Paramètres internes $\mathbf{A}$

Paramètres: Centre de l'image  $(o_x, o_y)$ , taille d'un pixel  $(s_x, s_y)$  et distance focale  $f$ .

$$\mathbf{A} = \begin{pmatrix} -\frac{f}{s_x} & 0 & o_x & 0 \\ 0 & -\frac{f}{s_y} & o_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} -f_x & 0 & o_x & 0 \\ 0 & -f_y & o_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Plus généralement, simple transformation affine  $2 \times 3$  de l'image:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

## Paramètres externes $\mathbf{R}$ et $\mathbf{T}$

### Matrice de rotation

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

avec la propriété d'orthonormalité  $\mathbf{R}^\top \mathbf{R} = \mathbf{I}$ .

### Matrice de translation

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

## Composition d'objets synthétiques dans une scène réelle

- On dispose d'une séquence vidéo d'un autobus.
- On désire ajouter un objet (dinosaur) sur le trottoir



# Composition d'objets réels dans un univers synthétique

Les trois composantes de base:

## 1. Caméra

Extraction des paramètres de la caméra.

## 2. Illumination

Estimation de la couleur et de la direction de l'illumination.  
(important pour le réalisme et les ombres)

## 3. Interaction Réel - Synthétique

Comment l'objet interagit avec la scène?

En général, on utilise une modélisation sommaire de la scène (ex: modèle de trottoir)

# Caméra

On peut utiliser des points de correspondance et résoudre le modèle de caméra.

De l'information *a priori* est très utile.

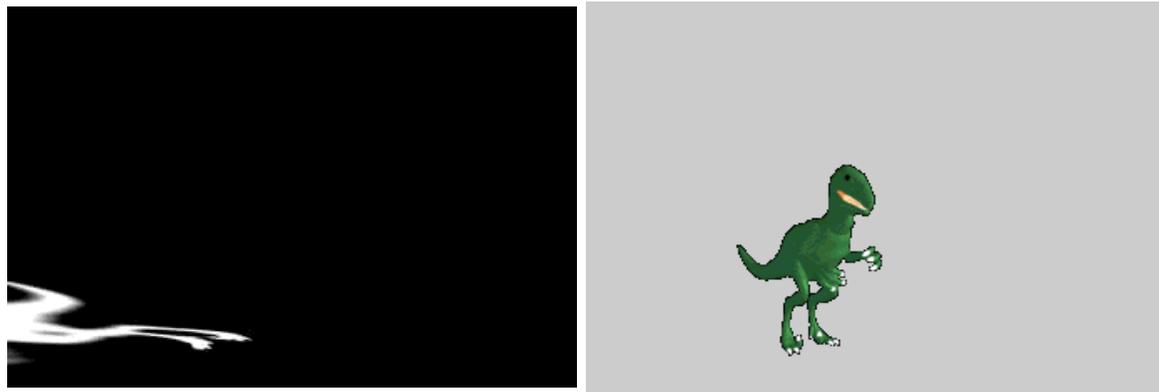
## **Cas spécial : caméra sans translation (panning)**

Avec une caméra sans translation, on peut estimer la rotation (inconnue) en suivant n'importe quel point fixe de la scène.

## Estimation de l'Illumination

- Problème difficile qui est souvent résolu à la main en estimant grossièrement la position et la couleur de l'éclairage.
- Pour être plus précis, on doit utiliser des objets de calibration aux propriétés surface connues (ex: boule en plâtre, boule miroir, etc..)

### Ombre et Objet rendu avec illumination



## Intéraction Réel - Synthétique

Pour modéliser sommairement, on utilise en général des points de correspondance.

### Cas spécial : lignes parallèles et points de fuite

Si un objet contient des ligne parallèle, alors on observe un point de fuite dans l'image. Ce point aide à déterminer la position en 3D de l'objet.

Avec 2 points de fuite pour un même objet, on peut retrouver la position 3D exacte.

