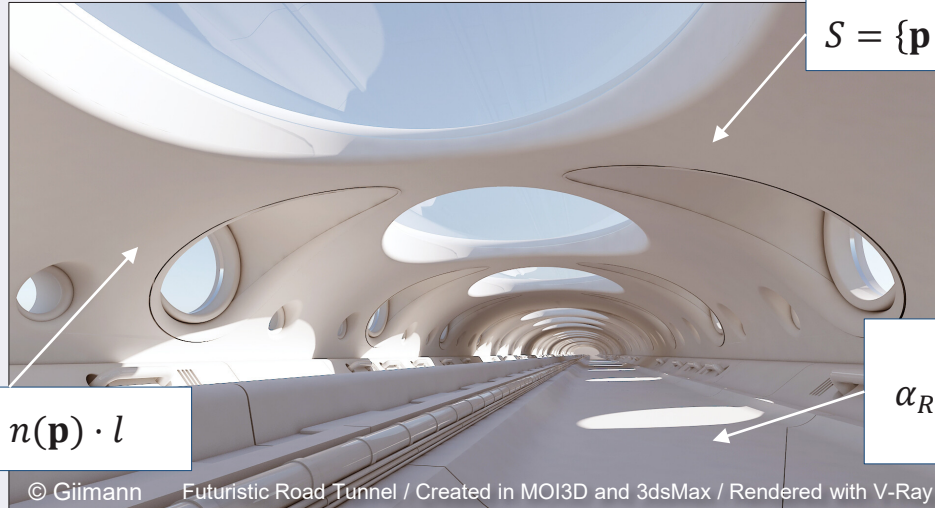


# Computer Graphics



# Computer Graphics

From mathematics ...



$$S = \{\mathbf{p} \in \mathbf{R}^3, f(\mathbf{p}) = 0\}$$

$$d(\mathbf{p}) = n(\mathbf{p}) \cdot l$$

$$\alpha_R(\mathbf{p}) \approx \frac{1}{n} \sum_{i=0}^n \delta_i$$

© Giimann Futuristic Road Tunnel / Created in MOI3D and 3dsMax / Rendered with V-Ray

... to the screen

E. Galin  
Université Lyon 1

# Computer Graphics

**Introduction**

Mathematics

Modeling

Color and Texturing

Shading

Realistic Rendering

Acceleration

# Computer Graphics

## Pipeline de création

# Pipeline de génération

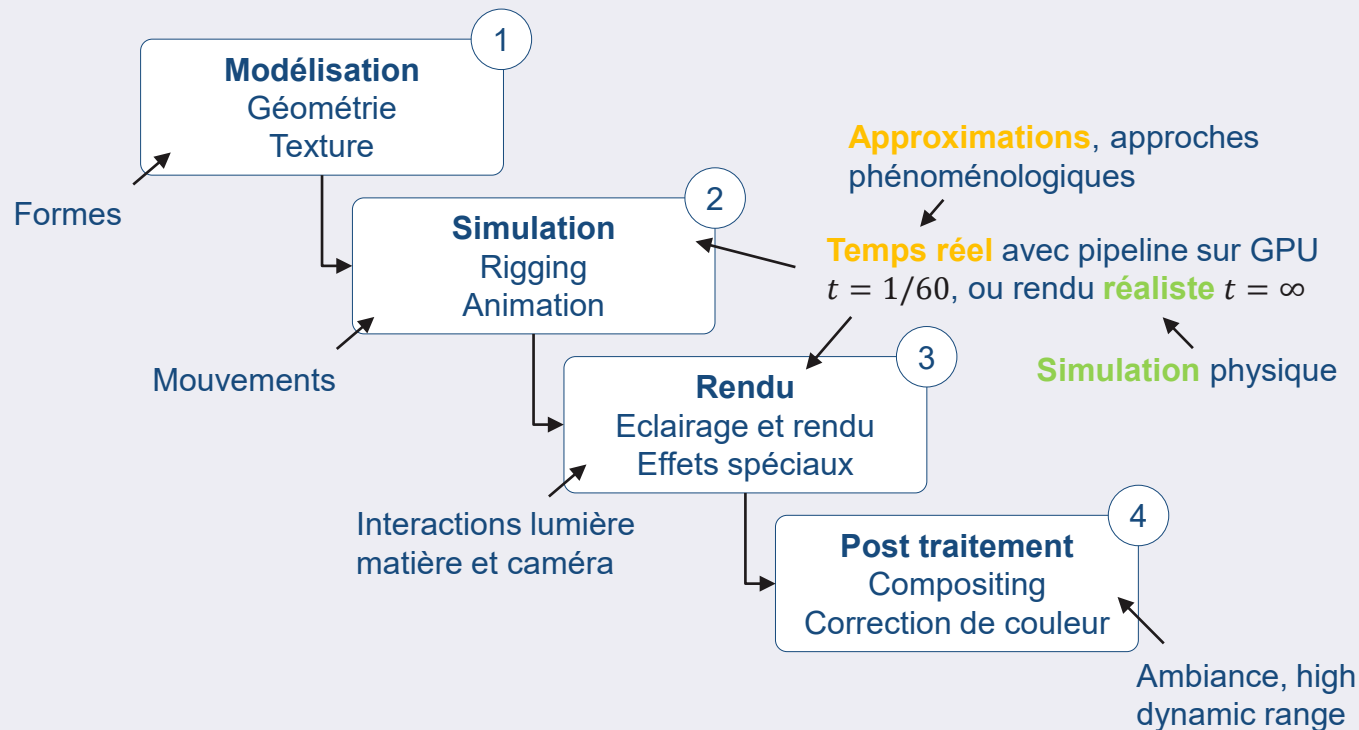
Introduction

Temps réel

Rendu réaliste

## Etapes

Modélisation, animation, rendu, post traitement  
Processus parfois partiellement entremêlés



eric.galin@liris.cnrs.fr

<http://liris.cnrs.fr/~egalain>

Fundamentals of Computer Graphics, Steve Marschner, Peter Shirley, ISBN 9780367505035, 2021 by A K Peters/CRC Press

# Computer Graphics

## Rasterization





Billboards



Triangle mesh



Low resolution mesh



Billboards



**Modeling**  
Geometry  
Materials





Animated texture

Sky sphere

Vidéo

Transparent billboards

Articulated body

Ambient

Faked reflections

Animated sprites

Snow particles

**Rendering**  
Lighting  
Camera



# Pipeline de rendu par rasterisation

Introduction

Temps réel

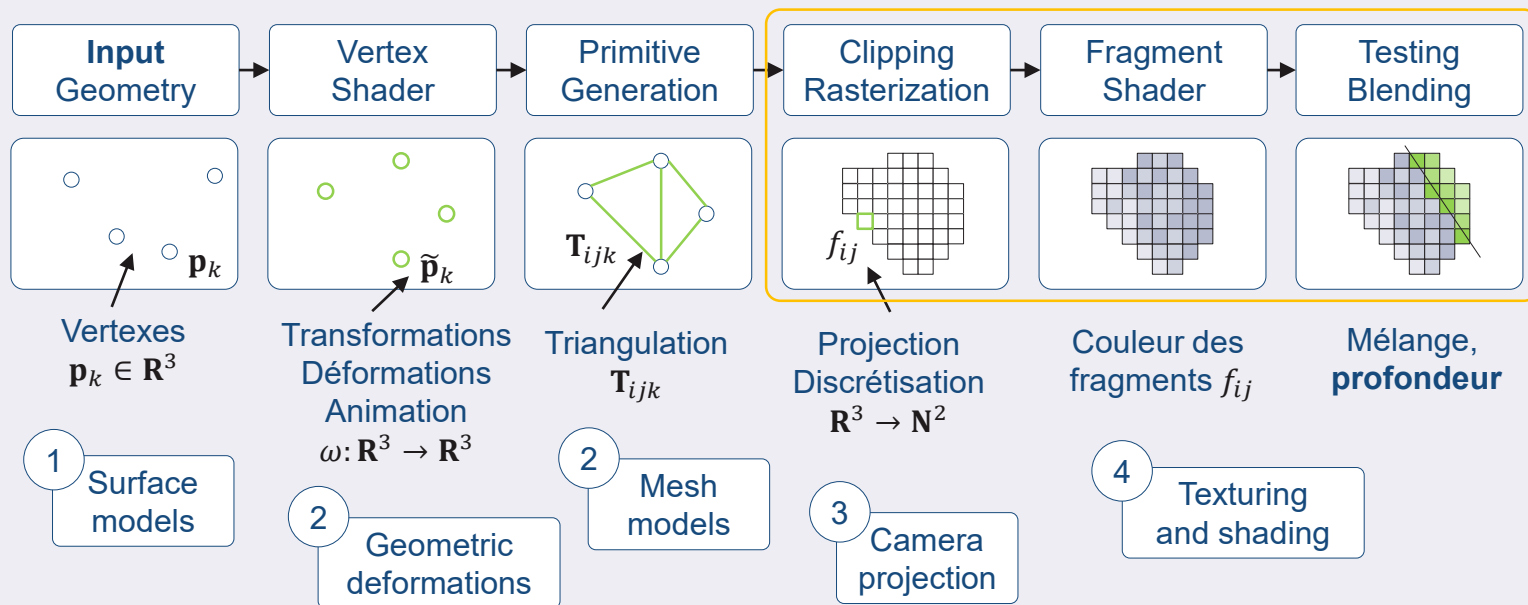
Rendu réaliste

## Principe

Transformation, projection et discrétisation d'un modèle **continu** dans l'espace image **discret**

Modèles, animations et déformations doivent être adaptés au temps réel

Les ombres, la visibilité, l'éclairage sont **approximés**



eric.galin@liris.cnrs.fr  
<http://liris.cnrs.fr/~egalain>

# Depth Buffer

Introduction

Temps réel

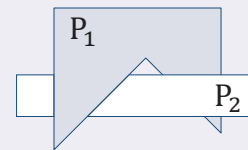
Rendu réaliste

## Algorithmes de classement de profondeur

Algorithme du peintre : affichage dans un **ordre** de distance décroissante au point d'observation

Méthode de subdivision [Warnock 1969]

On ne peut **pas** trier des polygones ou des triangles



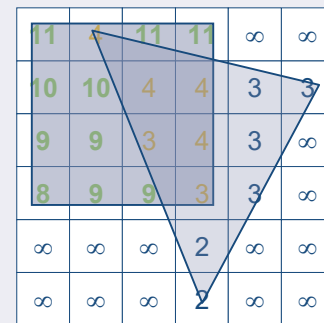
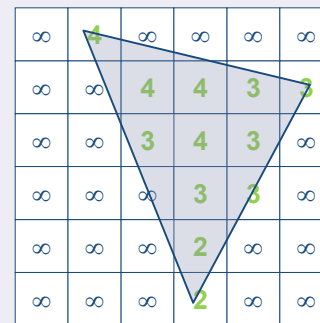
## Depth Buffer

Discrétisation de tous les polygones  $\tilde{T}$  selon une grille de pixels  $E$

Algorithme parallèle pour les pixels  $p_{ij}$  de chaque triangle

```
Initialize depth of pixels  $z_{ij}$  to  $\infty$ 
For all triangles  $T$ 
  Project  $T$  onto screen
  For all pixels  $p_{ij}$  of  $\tilde{T}$ 
    Compute depth of  $p_{ij}$ 
    If  $z_T < z_{ij}$  then
      Set  $z_{ij} = z$ 
      Compute color  $c_{ij}$ 
    Otherwise
      Nothing to update
```

Annotations:  $z$  buffer, Triangle in  $E$ , Distance to  $E$ , Hidden



J. Warnock. A hidden surface algorithm for computer generated halftone pictures. *Ph.D. Thesis*. University of Utah. 1969



eric.galin@liris.cnrs.fr  
http://liris.cnrs.fr/~egalain

# Computer Graphics

## Ray Tracing





Volumes

Materials

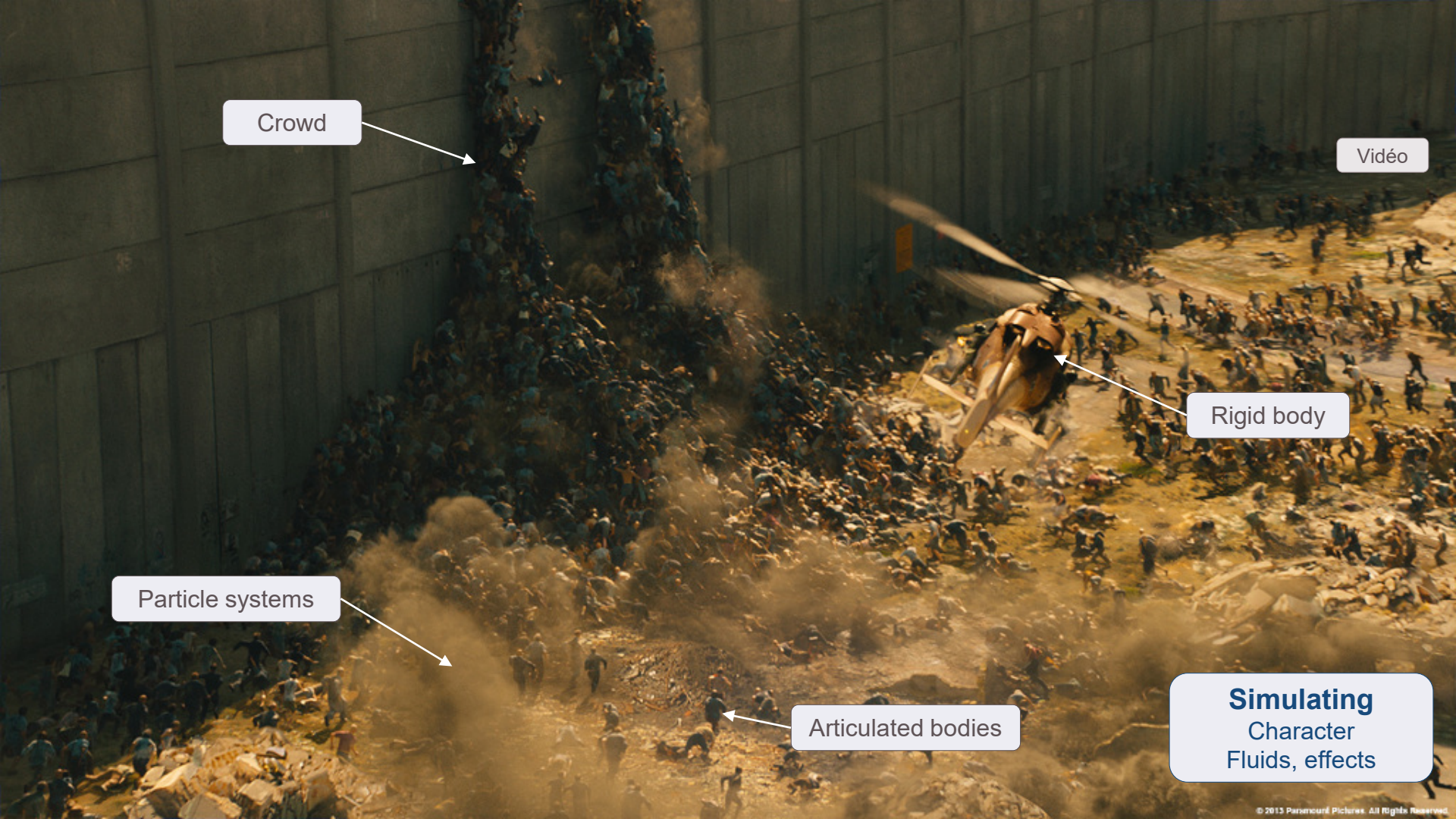
Surfaces

Fog density

Vidéo

**Modeling**  
Geometry  
Materials





Crowd

Vidéo

Rigid body

Particle systems

Articulated bodies

**Simulating**  
Character  
Fluids, effects



Particle systems

Surface reflection

Radiosity

**Rendering**

Lighting  
Camera





# Rendu réaliste

Introduction

Temps réel

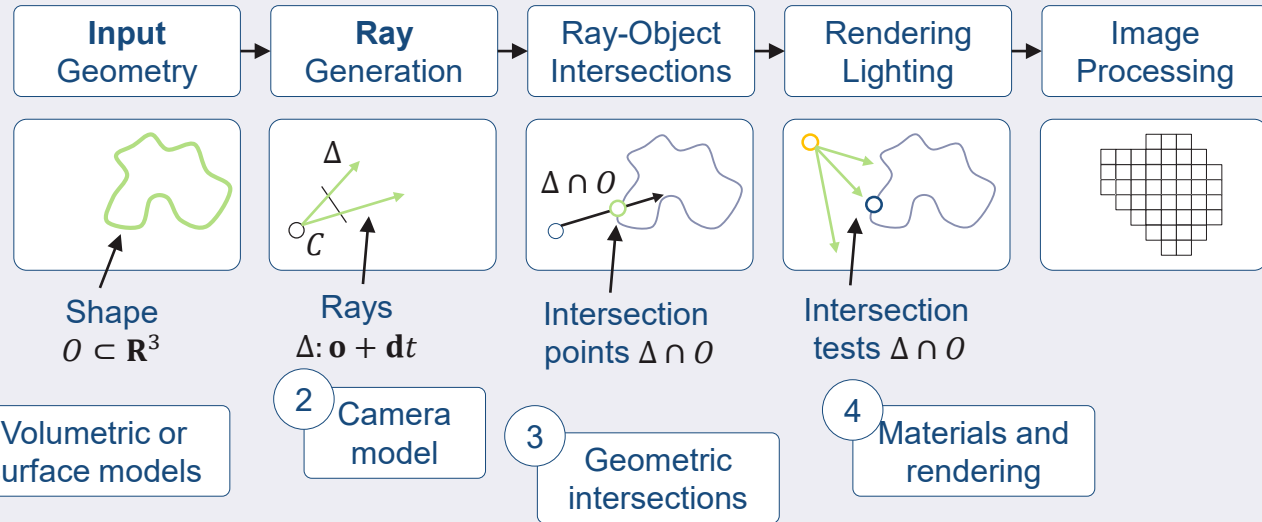
Rendu réaliste

## Principe

Lancer de rayons depuis l'espace image **discret** vers le modèle **continu**

Modèles, déformations, textures plus complexes

Les ombres, la visibilité, l'éclairage plus réalistes



eric.galin@liris.cnrs.fr  
<http://liris.cnrs.fr/~egalain>

Master

# Lancer de rayon

Introduction

Temps réel

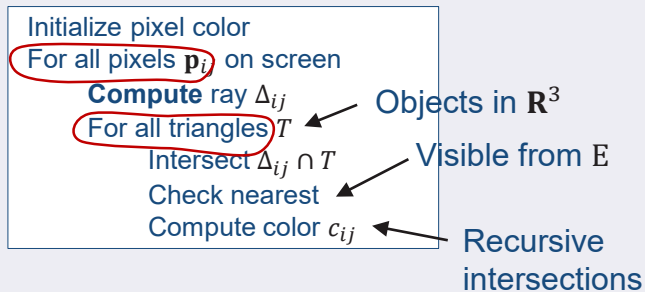
Rendu réaliste

## Algorithme

Lancer un rayon primaire  $\Delta_{ij}$  pour chaque pixel  $\mathbf{p}_{ij}$  depuis l'observateur E

Calculer la plus proche intersection en calculant  $\Delta_{ij} \cap T_k$  pour tous les triangles  $T_k$

Déterminer la couleur  $c_{ij}$  selon le modèle d'éclaircement



eric.galin@liris.cnrs.fr  
<http://liris.cnrs.fr/~egalin>