Efficient models for representing sub-pixel appearances



Daniel MENEVEAUX Steve MARSCHNER Emilie GUY Wenzel JAKOB Fabrice NEYRET Professeur, Université de Poitiers Professeur, Cornell University Chercheuse, Dynamixyz Professeur Assistant, EPFL DR, CNRS Rapporteur Rapporteur Examinateur Examinateur Directeur de thèse

Rendering



Rendering



Introduction

Rendering



Introduction

Applications: the film industry



Dawn of the Planet of the Apes, Weta, 2014

Applications: the film industry



Moana, Disney, 2016



























600 000 light paths (1 path/px)

Rendering time: A few seconds



2.5 million light paths (4 paths/px)

Rendering time: A few seconds



10 million light paths (16 paths/px) Rendering time: A few seconds



40 million light paths (64 paths/px) Rendering time: 1 minute



2.5 billion light paths (4096 paths/px) Rendering time: 1 hour



Rendering: hours, tens of hours with render farms! 1 film: about 150000 final frames Huge problem for artists, directors, producers...

Ralph Breaks the Internet, Disney, 2018 32 GB of geometry







Introduction



Introduction



Massive amounts of detail



War for the Planet of the Apes, Weta, 2017

Massive amounts of detail



Coco, Pixar, 2017



Introduction



Olaf's Frozen Adventures, Disney, 2017



Olaf's Frozen Adventures, Disney, 2017



Olaf's Frozen Adventures, Disney, 2017



Moana, Disney, 2016


Useless details



Useless details



Introduction

Useless details









Introduction

Benefits of LODs	Problems
 Reduce: Memory usage Loading time Light path computation Noise Same image ! 	 Preserving the appearance: Difficult In the industry: naïve methods Academic research: special cases And some requirements: General, robust Seamless prefiltering

Benefits of LODs	Problems
 Reduce: Memory usage Loading time Light path computation Noise Same image ! 	 Preserving the appearance: Difficult In the industry: naïve methods Academic research: special cases And some requirements: General, robust Seamless prefiltering

Our goal: addressing these problems!

Overview of the thesis



Part 1 Prefiltering production assets



Previous work

Many academic publications:

- Geometry simplification
- Approximations of aggregate details (snow, sand, hair)
- Texture prefiltering
- Alternative representations (real-time)

Previous work

Many academic publications:

- Geometry simplification
- Approximation of aggregate details (snow, sand, hair)
- Texture prefiltering
- Alternative representations (real-time)

Previous work on geometry simplification

Hundreds of publications

• Sometimes useful:

- Not accurate for intricate geometry:
- Ecometry:
- Not always adapted to reflectance prefiltering



[Moon 2008] Prefiltering hair (MS)



[Schröder 2011] [Zhao 2011] [Jakob 2010] [Khungurn 2015] **Prefiltering fabrics**



[Moon 2007] [Meng 2015] [Muller 2016] Prefiltering granular materials



[Moon 2008] Prefiltering hair (MS)





[Schröder 2011] [Zhao 2011] [Jakob 2010] [Khungurn 2015] **Prefiltering fabrics**

[Moon 2007] [Meng 2015] [Muller 2016] Prefiltering granular materials

Explicit geometry











Previous work

	Mesh simplification	Volume approximations
Large surfaces	Useful	Inaccurate
Tiny geometry	Inaccurate	Useful

Automatic prefiltering with both meshes and volumes





• Cascading precomputation for seamless transitions:



• At rendering: mesh inside the volume



















Contributions:

More robust mesh simplification





Triangle meshes

Quad meshes

• Add support for reflectance prefiltering










Part 1: Prefiltering production assets



Part 1: Prefiltering production assets

Edge collapse & vertex placement Minimize a *two-sided* distance Same metric used for quad meshes Input mesh LOD, LOD, LOD. (foliage) Quadric error metric Volume preserving **Our:** minimization

Part 1: Prefiltering production assets



Mappings for reflectance prefiltering



Triangle meshes



Mappings for reflectance prefiltering





Triangle meshes

Quad meshes

Normal prefiltering in world space [Heitz et al. 2015] [Walter et al. 2014]



Our contributions



Our contributions



Appearance-preserving voxelization

Difficult problem:

- Appearance has many degrees of freedom
- We want few parameters

Need accurate occlusion estimation



Surface reflectance to volume scattering functions: **Microflake volume model** [Jakob et al. 2010] [Heitz et al. 2015]

Result of our hybrid pipeline

Video

Problems and future work

• Prefiltering surface-like appearance of disconnected elements





- Animation
- New volume models for prefiltering

Part 2 Downsampling heterogeneous volumes









Part 2: Downsampling heterogeneous volumes





Microflake model [Jakob et al. 2010]

Parameters per voxel:

- **Density** of microflakes
- Microflake normal distribution
- Microflake albedo

Part 2: Downsampling heterogeneous volumes



[Zhao, Wu et al. 2016]



Voxels at the scale of fibers

[Zhao, Wu et al. 2016]





Voxels at the scale of fibers

• Very realistic

[Zhao, Wu et al. 2016]

• Rendering is long (hours)

Part 2: Downsampling heterogeneous volumes











Goal: appearance-preserving, for any light and view



Low-resolution voxel

Part 2: Downsampling heterogeneous volumes



Low-resolution voxel

Density: $\frac{1}{N}\sum \rho_i$

Part 2: Downsampling heterogeneous volumes



Low-resolution voxel

Density: $\frac{1}{N} \sum \rho_i$ Albedos: $\frac{\sum \rho_i \alpha_i}{\sum \rho_i}$



Density:
$$\frac{1}{N} \sum \rho_i$$
 Albedos: $\frac{\sum \rho_i \alpha_i}{\sum \rho_i}$

Normal distributions:

[Heitz et al. 2015] [Zhao, Wu et al. 2016]



500[^]3 voxels

Linear downsampling



16^3 voxels (< 1%)



Linear downsampling

500^3 voxels



16^3 voxels (< 1%)

Part 2: Downsampling heterogeneous volumes

Input



Input (density x20)







Comparison:



Part 2: Downsampling heterogeneous volumes



What's happening? Problem #1: Transparency problem Problem #2: Local shadowing problem

What's happening? **Problem #1: Transparency problem** Problem #2: Local shadowing problem

Problem #1: transparency






Transparency: 50%



Transparency: 50%





Transparency: 50%

Transparency: 50%

Problem with linear downsampling

What's happening? **Problem #1: Transparency** Problem #2: Local shadowing

Problem with linear downsampling

What's happening? Problem #1: Transparency Problem #2: Local shadowing











Local shadowing Multiple scattering = saturated perceived color





Input

Naïve (correct transp.)



Input

Naïve (correct transp.)

Previous work: [Zhao, Wu et al. 2016]



Reference, 25.4 GB

LODs, 45.6 MB

Much more accurate than naïve linear downsampling

Previous work: [Zhao, Wu et al. 2016]

Algorithm:

- 1) Downsample linearly
- 2) Render input volume (several views & lights)
- 3) Iterative optimization:
 - Render current LOD (several views & lights)
 - Update scattering parameters

Problems:

- Long precomputation time
- Does not support well semi-transparent media
- Does not support well **spatially-varying appearances**
- Linear downsampling of density



Our approach

• Correct transparency (non-linear)

Our approach

• Correct transparency (non-linear)

• Local estimation of parameters About x10 faster than [Zhao, Wu et al. 2016] Overcome some limitations



o Now mic

New microflake model for correct local shadowing

Part 2: Downsampling heterogeneous volumes

Our approach

• Correct transparency (non-linear)

 Local estimation of parameters About x10 faster than [Zhao, Wu et al. 2016] Overcome some limitations







Standard model



Our model Isotropic shadowing (+1 parameter)





Difficult: get closed-form expressions for scattering functions, sampling



Difficult: get **closed-form expressions** for scattering functions, sampling Model of **transparent media** with **multiple scattering** at micro-scales

Input









Naïve (correct transp.)











Local estimation of parameters

We estimate:



- Transparency (easy)
- Single scattering albedo & normal distributions [Heitz et al. 2015] (linear)

Local estimation of parameters

We estimate:

 \rightarrow

- Transparency (easy)
- Single scattering albedo & normal distributions [Heitz et al. 2015] (linear)
- Density and shadowing parameters
- Multiple scattering albedo

Good property: no optimization needed









Errors:














































Results



LOD 2 [Zhao, Wu et al. 2016] 12 CPU core hours Input

LOD 2 Our, 1 CPU hours (2 minutes)

Part 2: Downsampling heterogeneous volumes

Results



Input

Results



LOD 2

Limitations



Local shadowing Multiple scattering Saturated colors



Conclusion

Future work

Sub-resolution polygons in hybrid LODs

Glints, surfaces & volumes

Non-classical volume models [d'Eon 2018] [Jarabo et al. 2018]







