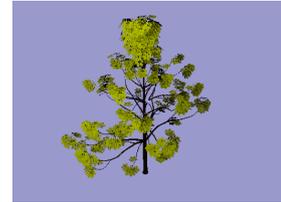


dynamic point sampling

Marc Stamminger

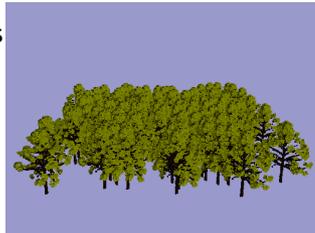
motivation

- tree created by AMAP
- 150,000 triangles
- 8 fps

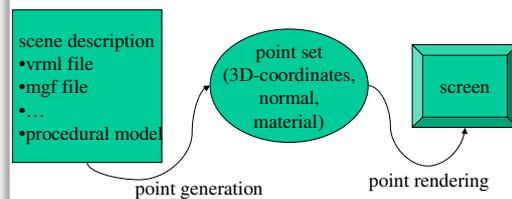


motivation

- level of detail
- 100 trees
- 270,000 points
- 20 fps



point rendering pipeline



point generation

- Surfels (Pfister et al., SIG2000)
 - (orthographic) views
- Q-Splat (Rusinkiewicz et al., SIG2000)
 - filtered triangle mesh hierarchy
- Randomized z-Buffer (Wand et al., SIG2001)
 - random points

point rendering

- in software
 - filtering
 - texturing
 - hole filling
- in hardware
 - as points
 - as polygonal disks
 - as splats

our approach



- dynamic point generation for
 - procedural objects
 - terrains
 - complex dynamic objects
 - point rendering with OpenGL's GL_POINT
 - very fast ($> 10^7$ points per second)
 - OpenGL does lighting

Point-Based Computer Graphics

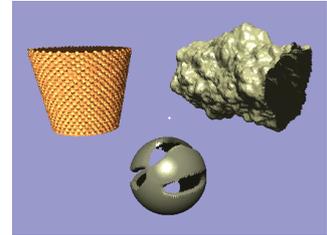
Marc Stamminger

7

results



- points are well suited for
- procedural geometry



Point-Based Computer Graphics

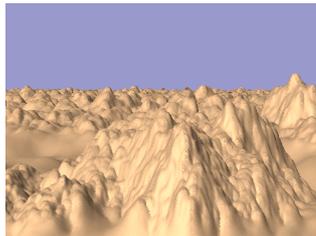
Marc Stamminger

8

results



- points are well suited for
- procedural geometry
- terrains



Point-Based Computer Graphics

Marc Stamminger

9

results



- points are well suited for
- procedural geometry
- terrains
- complex geometry



Point-Based Computer Graphics

Marc Stamminger

10

results



- points are well suited for
- procedural geometry
- terrains
- complex geometry
- combinations



Point-Based Computer Graphics

Marc Stamminger

11

results



- points are well suited for
- procedural geometry
- terrains
- complex geometry
- combinations
- eco systems



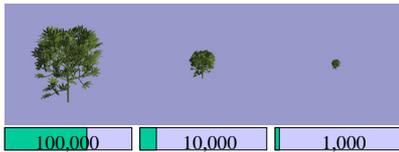
Point-Based Computer Graphics

Marc Stamminger

12

complex polygonal geometry

- generate list of randomly distributed samples
- for every frame: compute n , render the first n



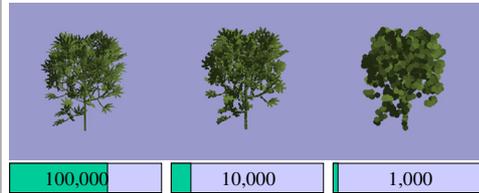
Point-Based Computer Graphics

Marc Stamminger

13

complex polygonal geometry

- easy speed / quality trade off
- frame rate control



Point-Based Computer Graphics

Marc Stamminger

14

sample densities

- adapt point densities to **image space (2D)**
- or: adapt to **post-perspective space (3D)**

Point-Based Computer Graphics

Marc Stamminger

15

densities complex geometry

- world space -> post-perspective:
 - area decreases by squared distance
 - goal: uniform post-perspective point density
 - point number \sim area/ d^2

Point-Based Computer Graphics

Marc Stamminger

16

modified complex geometry

- simple modifications on the fly



30 fps

Point-Based Computer Graphics

Marc Stamminger

17

complex geometry

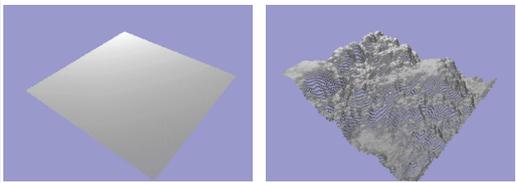
- video „complex geometry“
- download at <http://www-sop.inria.fr/reves/research>

Point-Based Computer Graphics

Marc Stamminger

18

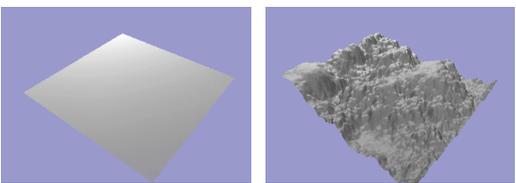
displaced geometry

25,000 points 25,000 points

Point-Based Computer Graphics Marc Stamminger 19

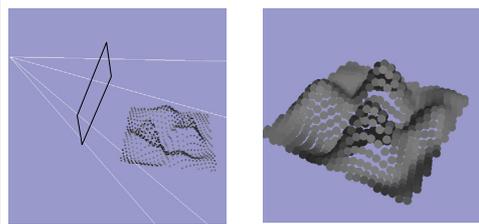
displaced geometry

25,000 points 100,000 points

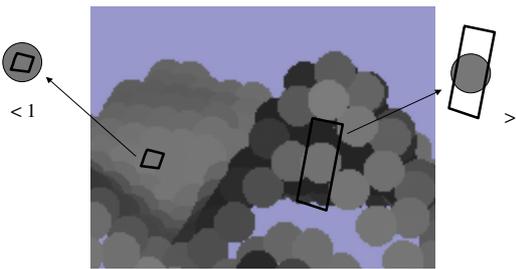
Point-Based Computer Graphics Marc Stamminger 20

adaptive sampling

Point-Based Computer Graphics Marc Stamminger 21

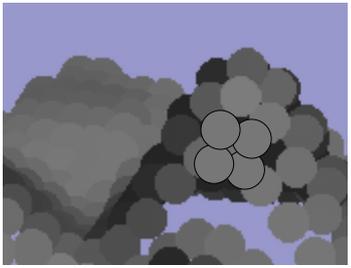
undersampling factor

undersampling in 2D image space, not post-perspective !

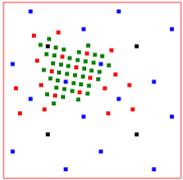
Point-Based Computer Graphics Marc Stamminger 22

undersampling factor

Point-Based Computer Graphics Marc Stamminger 23

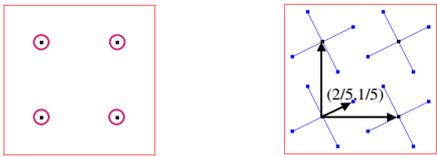
adaptive point generation

adaptive sample pattern

Point-Based Computer Graphics Marc Stamminger 24

$\sqrt{5}$ sampling 

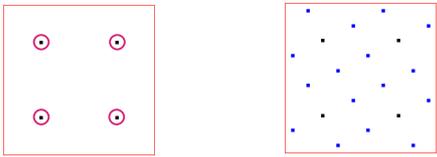


initial samples,
all undersampled

newly inserted
samples

Point-Based Computer Graphics Marc Stamminger 25

$\sqrt{5}$ sampling 

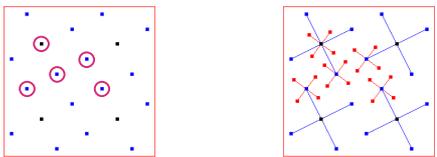


initial samples,
all undersampled

newly inserted
samples

Point-Based Computer Graphics Marc Stamminger 26

$\sqrt{5}$ sampling 

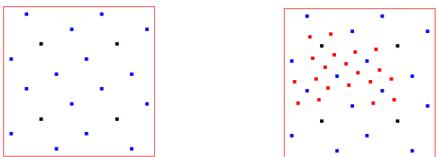


undersampled
samples

newly inserted
samples

Point-Based Computer Graphics Marc Stamminger 27

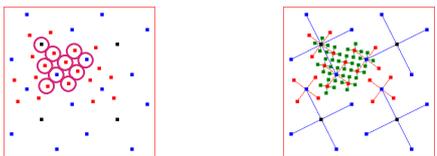
$\sqrt{5}$ sampling 



newly inserted
samples

Point-Based Computer Graphics Marc Stamminger 28

$\sqrt{5}$ sampling 

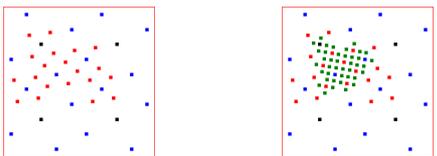


undersampled
samples

newly inserted
samples

Point-Based Computer Graphics Marc Stamminger 29

$\sqrt{5}$ sampling 



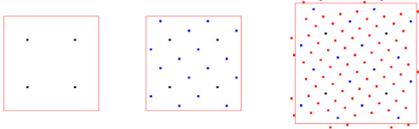
newly inserted
samples

Point-Based Computer Graphics Marc Stamminger 30

$\sqrt{5}$ sampling



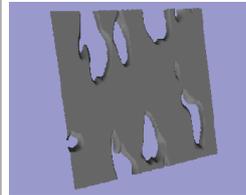
- rotated, nested grids
 - grid distance decreases by $1/\sqrt{5}$
 - rotation angle $\approx 27^\circ$
- special attention to boundaries



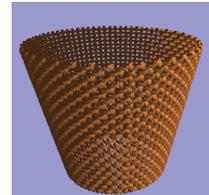
Point-Based Computer Graphics

Marc Stamminger 31

procedural modifiers



original geometry: square



original geometry: truncated cone

Point-Based Computer Graphics

Marc Stamminger 32

video

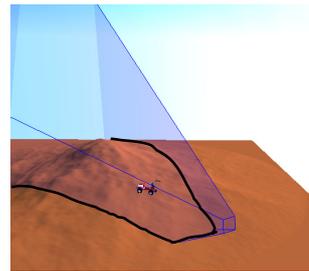


- video „ $\sqrt{5}$ sampling“
- download at <http://www-sop.inria.fr/reves/research>

Point-Based Computer Graphics

Marc Stamminger 33

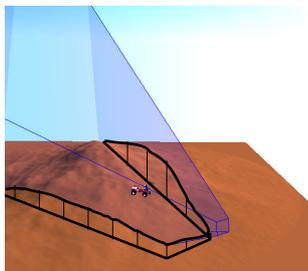
terrains



Point-Based Computer Graphics

Marc Stamminger 34

terrains



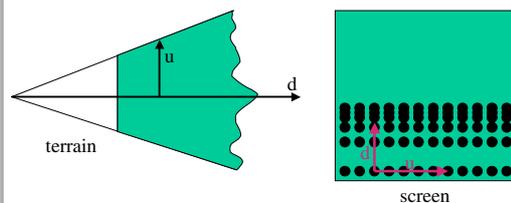
Point-Based Computer Graphics

Marc Stamminger 35

terrain parameterization



- parameterize sector by (d, u)



Point-Based Computer Graphics

Marc Stamminger 36

terrain parameterization

$$d(v) = \frac{1}{\frac{1}{d_{\min}} - v \left(\frac{1}{d_{\max}} - \frac{1}{d_{\min}} \right)}$$

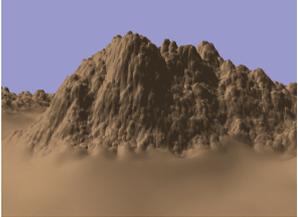
looking straight ahead looking up looking down



Point-Based Computer Graphics Marc Stamminger 37

terrain algorithm

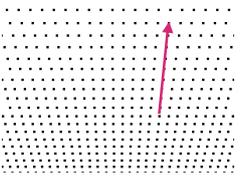
- $\sqrt{5}$ sampling scheme
- undersampling factor
 - parameterization distortions
 - perspective distortions
 - displacement



Point-Based Computer Graphics Marc Stamminger 38

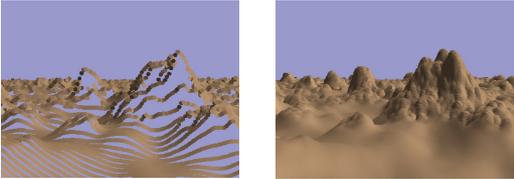
terrain occlusion culling

- elevation direction in image space along v
- simplifies occlusion culling



Point-Based Computer Graphics Marc Stamminger 39

terrain occlusion culling



occlusion culling, regular sampling occlusion culling, with adaptive sampling

Point-Based Computer Graphics Marc Stamminger 40

video

- video „terrain rendering“
- download at <http://www-sop.inria.fr/revs/research>

Point-Based Computer Graphics Marc Stamminger 41

eco systems

- level of detail:
 - polygonal model
 - ↓
 - replace polygons by points and lines
 - ↓
 - reduce number of points and lines

Point-Based Computer Graphics Marc Stamminger 42

eco systems



- example

points

lines

polygons



eco systems



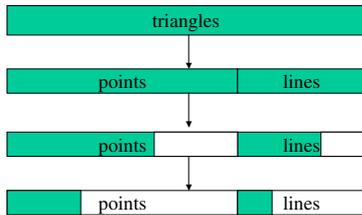
- modeller (xfrog) delivers:

- triangle set T_p
- random point set representing T_p
- triangle set T_l
- random line set L representing T_l ($|L| < T_l$)

eco systems



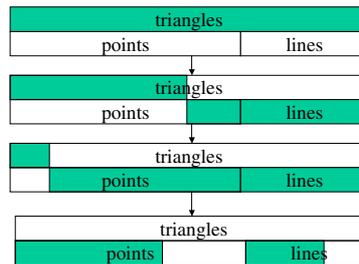
- level-of-detail 1



eco systems



- level-of-detail 2



eco systems



- criterion for point / line number (per object)

- user parameter:
point size d_p / line width d_l
- approximate screen space area of object:
 $A' = A * 0.5 / d^2$
- #points $\sim A' / d_p^2$
- #lines $\sim A' / d_p$

eco systems



- video „eco system rendering“

- download at

<http://www-sop.inria.fr/revs/research>