APPLICATIONS OF SDF IN CG AND GAMES

MILO YIP

2018/5/29

MOREFUN
STUDIO GROUP
魔方工作室群

MILO YIP

- Technical Director, Morefun Studio Group, Tencent
- Bachelor of Cognitive Science, HKU
- MPhil in SEEM, CUHK
- Game Credits
  - Cloudy with a Chance of Meatballs (XB360/PS3/Wii/PC)
  - Alice: Madness Returns (XB360/PS3/PC)
  - R & D Techs used in 斗战神(PC), 天涯明月刀(PC), (iOS/Android)
- Translator of Game Engine Architecture
- Author of RapidJSON (credited by Overwatch, God of War)
- https://github.com/miloyip
- https://www.zhihu.com/people/miloyip

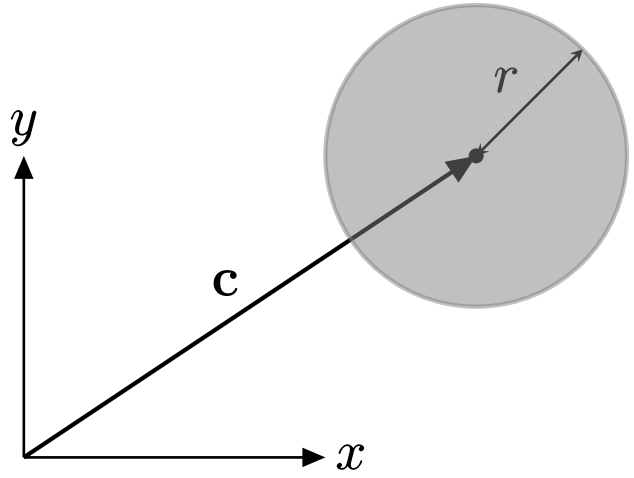
TABLE OF CONTENTS

1. INTRDOUCTION

REPRESENTATION OF SHAPES

In Maths, a shape can be represented as point set,

How to represent a disk with center $\mathbf{c}$ and radius r?



EXPLICIT REPRESENTATION

- Explicit representation also called parametric representation
- E.g. let 2 parameters of a disk be $\theta \in [0, 2\pi), t \in [0, 1]$

$$x(\theta, t) = c_x + tr\cos\theta$$
$$y(\theta, t) = c_y + tr\sin\theta$$

- Generate a point in the shape by parameter (θ, t)
- To determine if a point is in the shape, solve the equation
  - YES if there is a solution

IMPLICIT REPRESENTATION

- Implicit representation does not have parameters
- A disk can be represented as

$$\|\mathbf{x} - \mathbf{c}\|^2 \leq r^2$$

- All shapes can be represented by implicit equation

$$f(\mathbf{x}) \leq 0$$

- Easier to determine if a point is in the shape.
- But harder to generate a point in the shape.

DRAWBACKS OF THESE REPRESENTATIONS

- Can only represent if a point is in the shape (boolean result)
- No other information
- How to provide more useful information?

FIELDS

- A field is an assignment of values to each point in space
- Scalar field
  - E.g. tempreture field, electric potential field
- Vector field
  - E.g. velocity field, gravitational field

REPRESENT SHAPE BY DISTANCE FIELD

- Distance field represents the shortest distance between point in space and the shape (scalar field)
  - Often use Euclidean distance
- Define $d : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ for a shape (point set) S

$$d(\mathbf{x}) = \min_{\mathbf{y} \in S} \|\mathbf{x} - \mathbf{y}\|$$

- E.g. a disk is represented as

$$d(\mathbf{x}) = \max(\|\mathbf{x} - \mathbf{c}\| - r, 0)$$

SIGNED DISTANCE FIELD

- SDF also considered the interior of the shape
  - i.e. the signed distance to the boundary of shape
  - May define the exterior as +ve, the interior as -ve
- Define $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ for a shape S

$$\phi(\mathbf{x}) = \begin{cases} \min\limits_{\mathbf{y} \in S} \|\mathbf{x} - \mathbf{y}\| & \text{if } \mathbf{x} \notin S \\ -\min\limits_{\mathbf{y} \notin S} \|\mathbf{x} - \mathbf{y}\| & \text{if } \mathbf{x} \in S \end{cases}$$

- E.g. a disk is represented as

$$\phi(\mathbf{x}) = \|\mathbf{x} - \mathbf{c}\| - r$$

- Similar to implicit representation, it can determine if a point is in a shape by $\phi(\mathbf{x}) \leq 0$

2. APPLICATIONS IN GAME GRAPHICS

ALIASING ISSUE IN FONT

- When magnifying font texture, serious aliasing exists even after filtering

- [Green2007] solves this by SDF

PREGENERATED SDF TEXTURE

- [Green2007] use brute force $O(n^4)$ for $n \times n$ texels
- We can modify [Meijster2002]'s $O(n^2)$ algorithm for SDF generation

1024x1024 Monochrome → 1024x1024 SDF → 64x64 SDF

SAMPLING SDF WITH FILTERING

64x64 8-bit 灰度　　64x64 8-bit SDF

USING SHADER FOR FONT EFFECTS

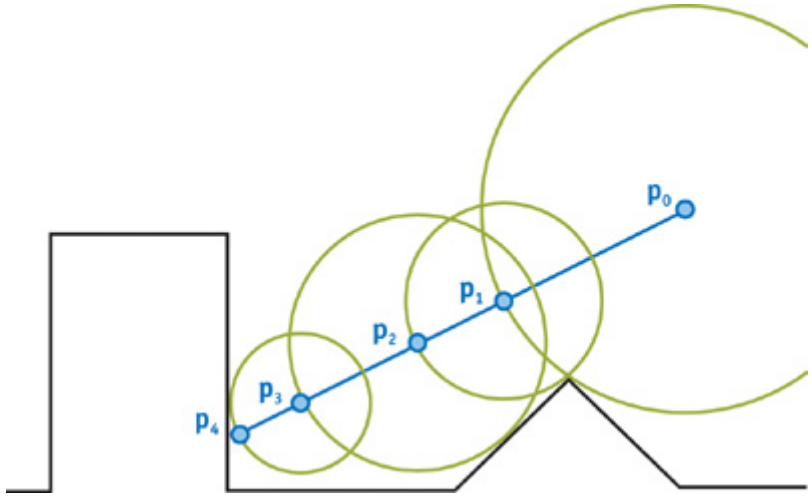
PROCEDURAL SDF

- In addition to pregenerated SDF texture
- We can also represent SDF procedurally

http://iquilezles.org/www/articles/distfunctions/distfunctions.h

RAY MARCHING

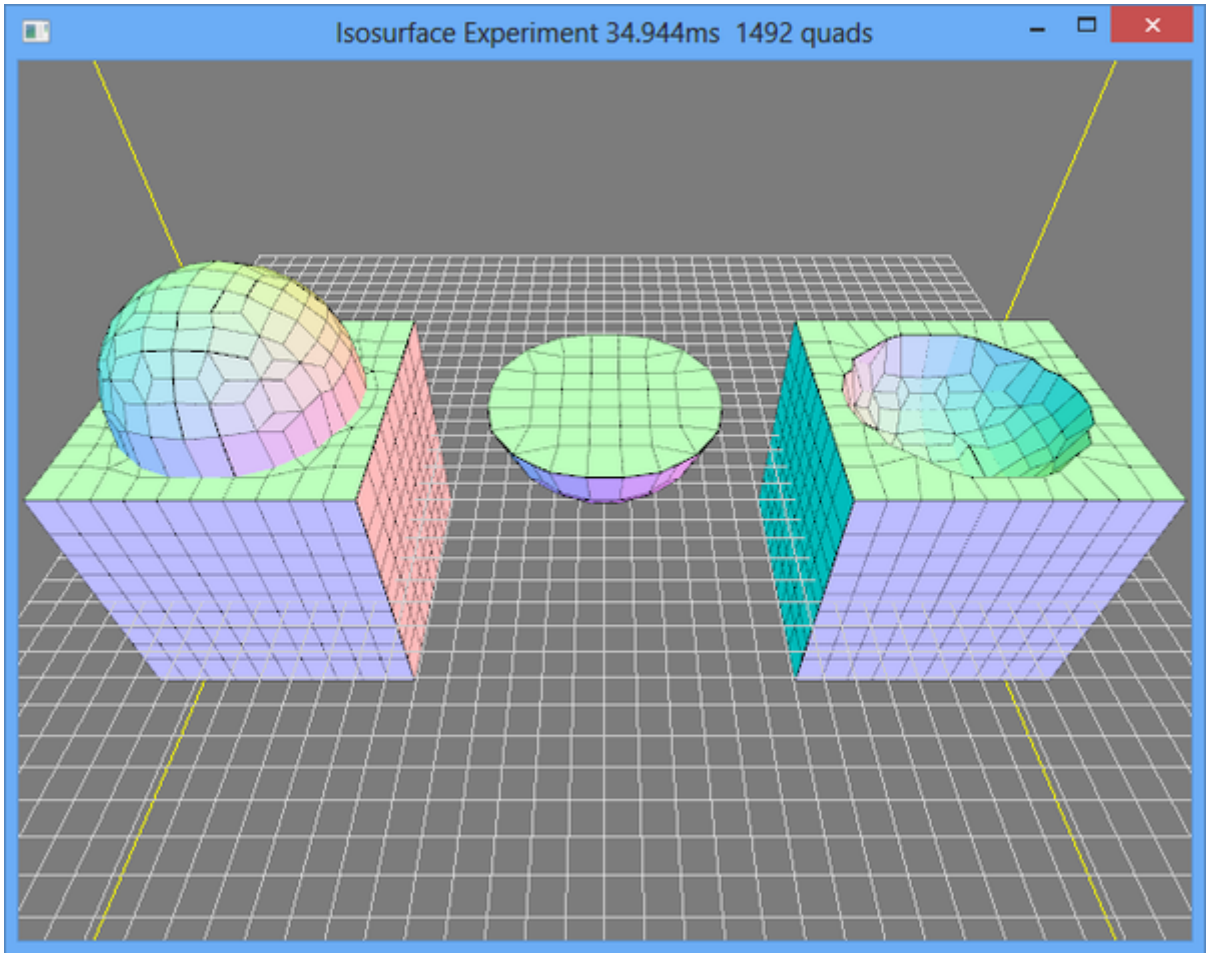
- Implement ray tracing via ray marching a SDF



Rendering SDF of Various Shapes

CONSTRUCTIVE SOLID GEOMETRY

- SDF can implement CSG easily



CSG OPERATIONS ON SDF

- Union

$$\phi_{A \cup B} = \min(\phi_A, \phi_B)$$

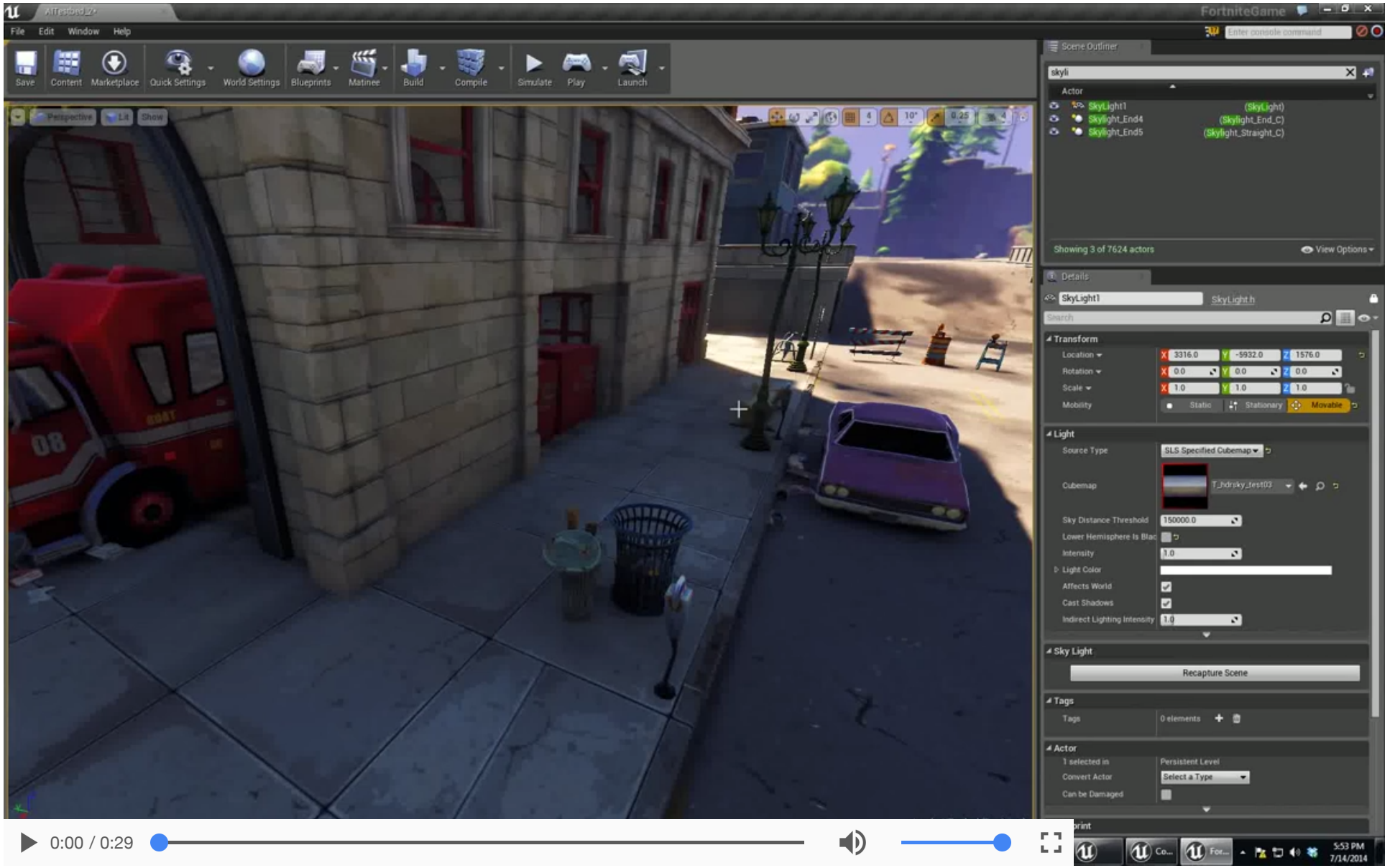
- Intersection

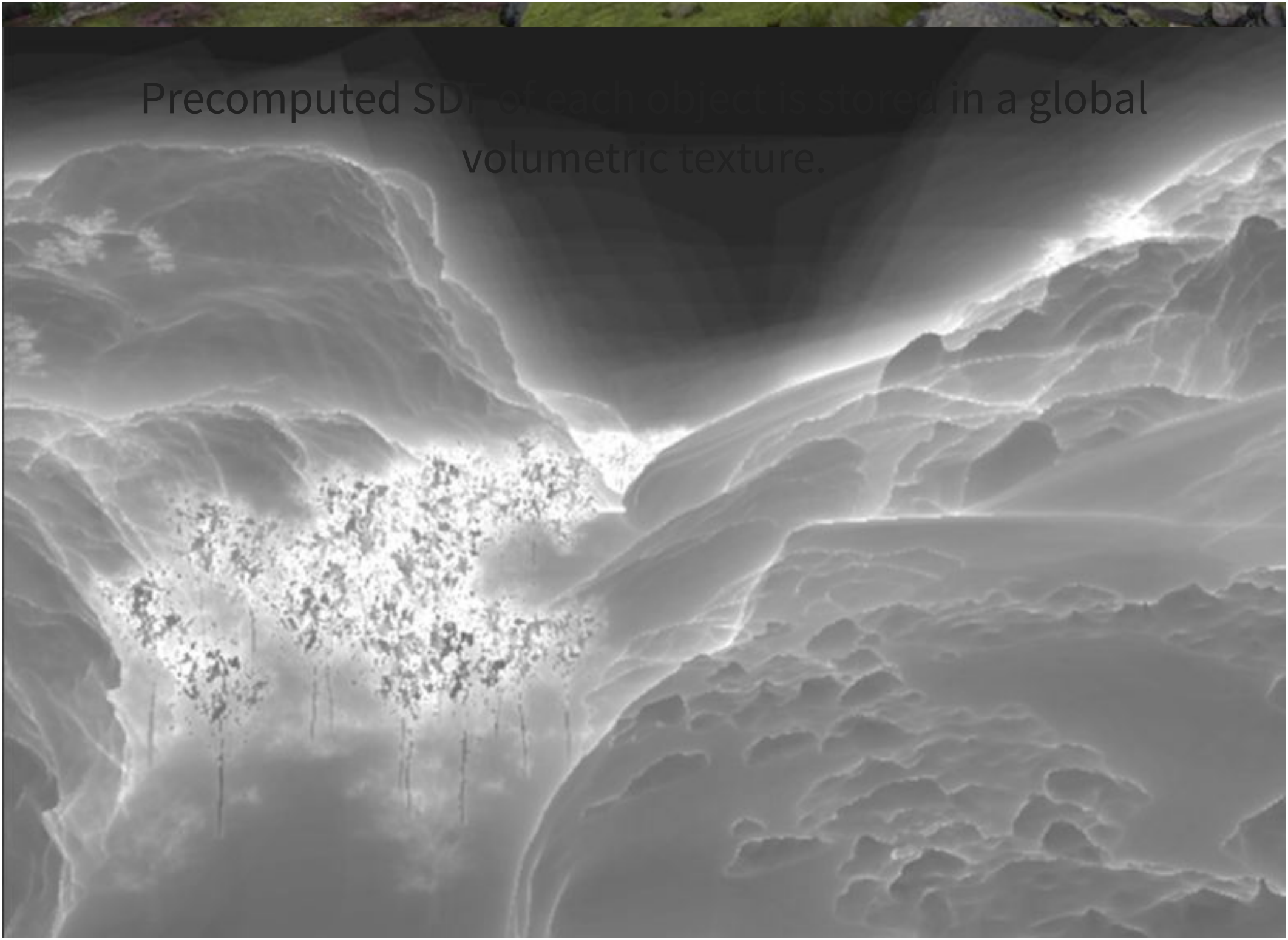
$$\phi_{A \cap B} = \max(\phi_A, \phi_B)$$

- Relative complement (subtraction)

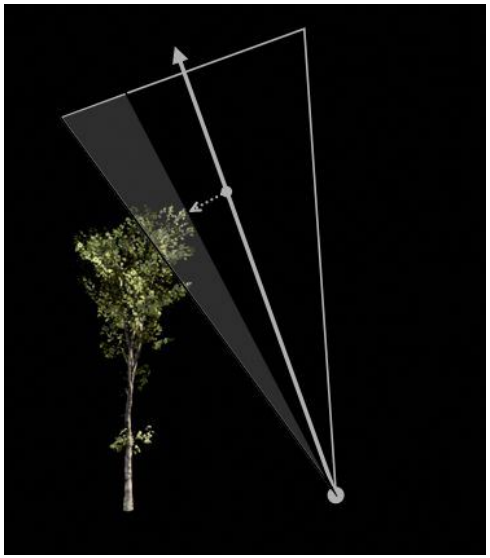
$$\phi_{A \setminus B} = \phi_{A \cap B^C} = \max(\phi_A, -\phi_B)$$

UE4 SDF AMBIENT OCCLUSION




Precomputed SDF of each object is stored in a global volumetric texture.

AMBIENT OCCLUSION: CONE TRACING

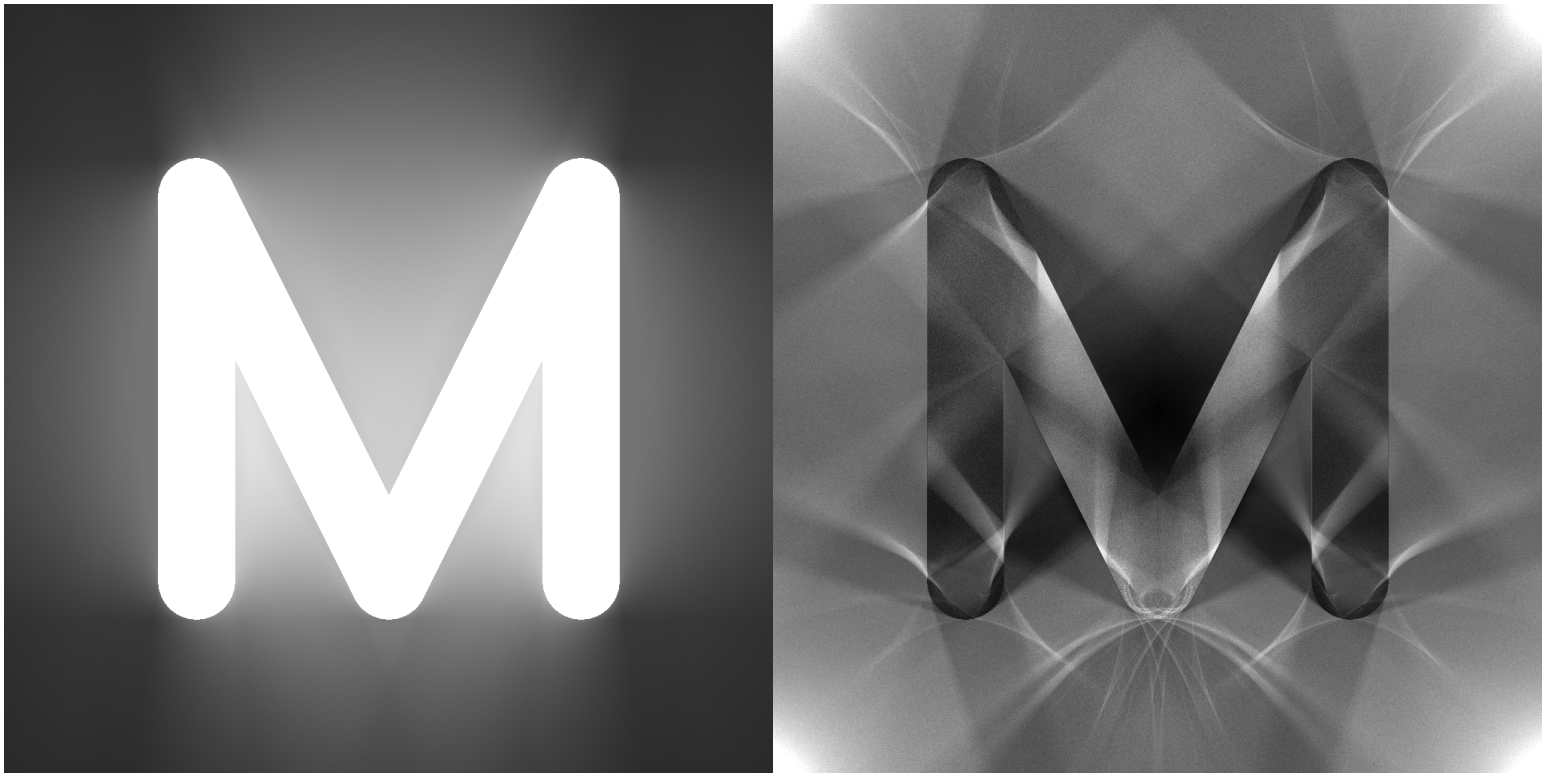


- Cone tracing to SDF on several directions
- Generate AO and bent normal[Wright2015]
- https://docs.unrealengine.com/latest/INT/Engine/Rendering/LightingAndShadows/DistanceFieldAmbientOcclusion/

RAYTRACED SOFT SHADOW

2D SDF/RAY MARCHING TUTORIAL

《用 C 语言画光》



- https://zhuanlan.zhihu.com/p/30816284
- https://github.com/miloyip/light2d

3. APPLICATIONS IN GAME PHYSICS

CHARACTER MOVEMENT AND COLLISION WITH ENVIRONMENT

- 2D Gameplay often uses grid to represent movable areas

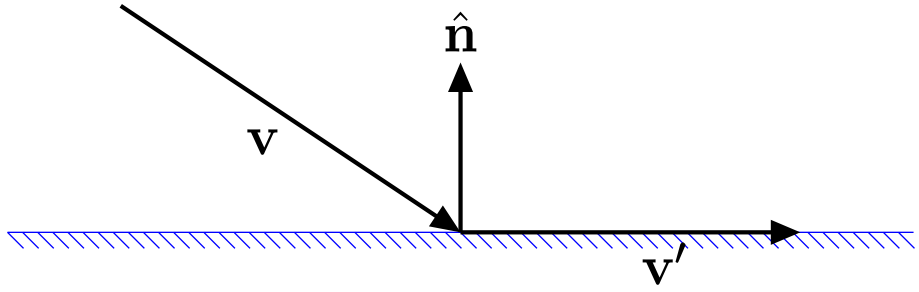
- How to detect collision between disk-shaped character & environment?

PREGENERATED SDF

- Convert obstacle areas into SDF
- Sample SDF at character's position with bilinear filtering
- If $\phi(\mathbf{x}) \leq r_{\text{avatar}}$ then collision occurs
- $O(1)$ time complexity!
- Demo

REQUIREMENT: COLLISION RESPONSE

- When colliding with environment, we need avater sliding at the boundary



$$\mathbf{v}' = \mathbf{v} - (\mathbf{v} \cdot \hat{\mathbf{n}})\hat{\mathbf{n}}$$

  - How to obtain the normal vector of collision $\hat{\mathbf{n}}$?
  - Do we need to pre-compute the normal vector field?

GRADIENT OPERATOR

- The gradient of a scalar field represents the direction of greatest change
- Leverage gradient of SDF as normal vector of boundary

$$\nabla \phi(\mathbf{x}) = \left[\frac{\partial \phi}{\partial x} \quad \frac{\partial \phi}{\partial y} \right]$$

- SDF has a special property

$$\|\nabla \phi\| = 1$$

- So we can use gradient of SDF as normal vector directly

COMPUTE GRADIENT NUMERICALLY

- $\phi(\mathbf{x})$ is almost everywhere differentiable
- May simply use finite difference

$$\nabla\phi(x,y) \approx \begin{bmatrix} \dfrac{\phi(x+\Delta x, y) - \phi(x-\Delta x, y)}{2\Delta x} \\ \dfrac{\phi(x, y+\Delta y) - \phi(x, y-\Delta y)}{2\Delta y} \end{bmatrix}^{\mathrm{T}}$$

- Demo

REQUIREMENT: AVATER STUCKING ISSUE

- When a character is stucked in obstacle after a hit, how to move it to a legal position?

MOVE AVATER TO NEAREST POSITION

- Use $\nabla\phi(\mathbf{x})$ again

$$\mathbf{x}' = \mathbf{x} + \nabla\phi\,(\mathbf{x})(r_{\text{avatar}} - \phi(\mathbf{x}))$$

- Only if the obstacle is convex, the solution is the nearest point
- Otherwise single iteration may not find a legal position
- Demo

REQUIREMENT: DASH

- When an avatar can dash to a certain direction, how to determine its position just before hitting an obstacle?

DISK CASTING

- In addition to ray tracing, ray marching can also do disk casting

```
Vector2 DiskCast(Vector2 origin, FVector2 direction, float radius,
                 float maxDistance, out float t) {
    t = 0.0f;
    while (true) {
        Vector2 p = origin + direction * t;
        float sd = SDF(p);
        if (sd <= radius)
            return p;
        t += sd - radius;
        if (t >= maxDistance) {
            t = maxDistance;
            return origin + direction * t;
        }
    }
}
```

- Demo

COMPUTATIONAL FLUID DYANAMICS

- Use two SDFs to represnts boundaries of solid and fluid
- Advect the SDF of fluid by velocity field
- This is called level set method
- Bridson, Fluid Simulation for Computer Graphics, CRC Press, 2016

4. SUMMARY

SDF'S PROS

- Implictly represents 2D/3D shapes
- Provides more information than other mentioned representations
- Better filtering
- Can efficiently do
  - Intersection Test with Disk
  - Ray Casting
  - Disk Casting
  - Approximated Cone Casting
- Gradient operator can be used as normal vector of boundary, and for finding nearest boundary

SDF CONS

- More storage space than monochrome image
- More difficult to update dynamically
- Remove sharp features

5. REFERENCES

- Green, Chris. “Improved alpha-tested magnification for vector textures and special effects.” ACM SIGGRAPH 2007 courses. ACM, 2007.
- Wright, Daniel. “Dynamic occlusion with signed distance fields” ACM SIGGRAPH 2015 Advances in Real-Time Rendering in Games course, 2015.
- Meijster, Arnold, Jos BTM Roerdink, and Wim H. Hesselink. “A general algorithm for computing distance transforms in linear time.” Mathematical Morphology and its applications to image and signal processing. Springer US, 2002. 331-340.

INTERN OPPORTUNITY

Our Studio Group has a few intern vacancies.

Talk to me or send me CV at miloyip@tencent.com

Let’s make games together.