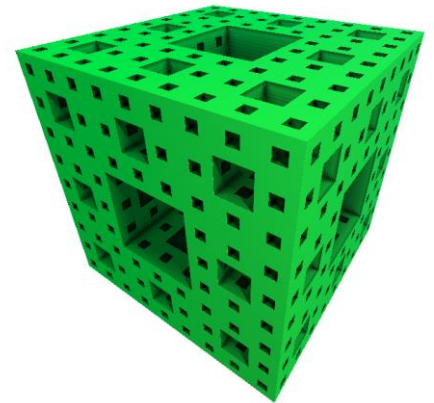
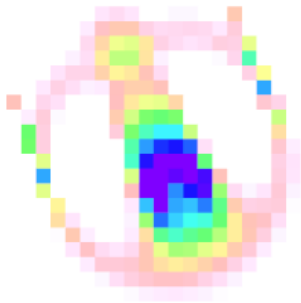




Lenia und Ray-Marching Fraktale

Praktikum Computergrafik

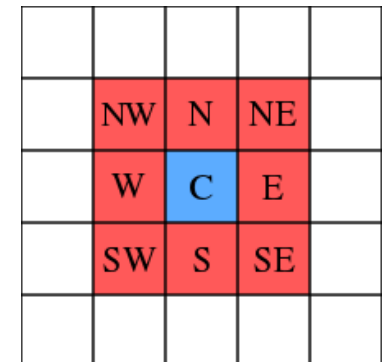
Sommersemester 2023



Conway's Game of Life



- zweidimensionaler zellulärer Automat
- Folgezustand ist abhängig vom vorherigen Zustand
 - Zellen können entweder tot oder lebendig sein
- Moore-Nachbarschaft
- Regeln:
 - Lebendige Zelle mit 2-3 lebendigen Nachbarn -> lebendig
 - Tote Zelle mit drei lebendigen Nachbarn -> lebendig
 - Alle anderen Zellen -> tot
- Interpretierbar unter biologischem Aspekt als Mikrokosmos

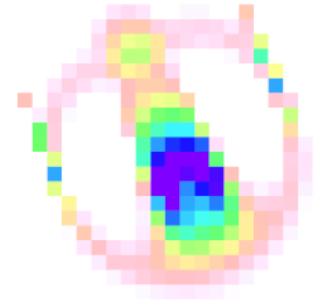


Moore-Nachbarschaft

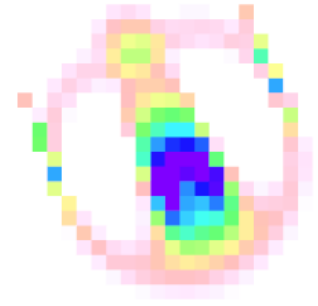
images: https://en.wikipedia.org/wiki/File:Gospers_glider_gun.gif
https://commons.wikimedia.org/wiki/File:Moore_neighborhood_with_cardinal_directions.svg

Lenia

- zweidimensionaler zellulärer Automat

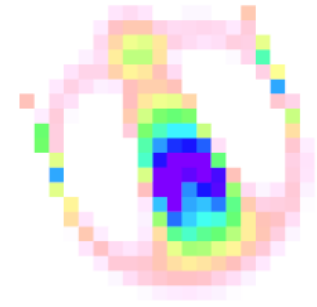


Lenia



- zweidimensionaler zellulärer Automat
- kontinuierliches System
 - Zellen können Zustand zwischen 0 und 1 annehmen

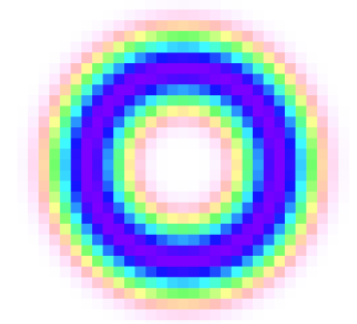
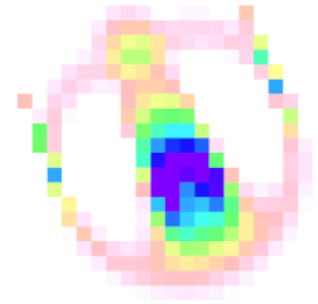
Lenia



- zweidimensionaler zellulärer Automat
- kontinuierliches System
 - Zellen können Zustand zwischen 0 und 1 annehmen
- Nachbarschafts-Radius statt Moore-Nachbarschaft
 - Euklidische Distanz
 - Geodesic Distance

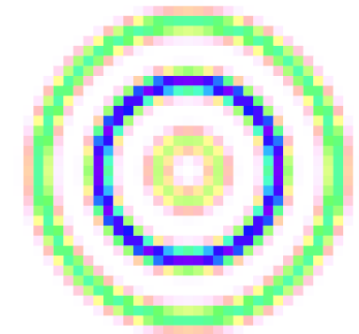
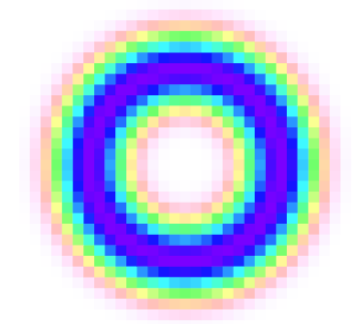
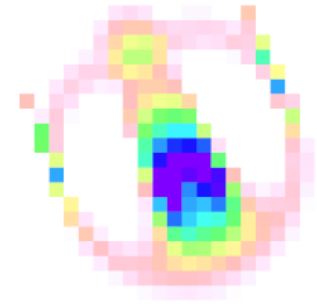
Lenia

- zweidimensionaler zellulärer Automat
- kontinuierliches System
 - Zellen können Zustand zwischen 0 und 1 annehmen
- Nachbarschafts-Radius statt Moore-Nachbarschaft
 - Euklidische Distanz
 - Geodesic Distance
- Nachbarschaft wird durch Kernel gewichtet



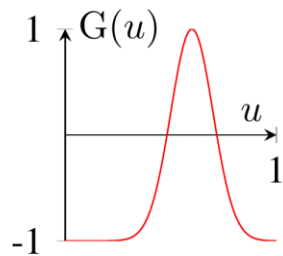
Lenia

- zweidimensionaler zellulärer Automat
- kontinuierliches System
 - Zellen können Zustand zwischen 0 und 1 annehmen
- Nachbarschafts-Radius statt Moore-Nachbarschaft
 - Euklidische Distanz
 - Geodesic Distance
- Nachbarschaft wird durch Kernel gewichtet
 - Kernel kann mit Peaks verfeinert werden

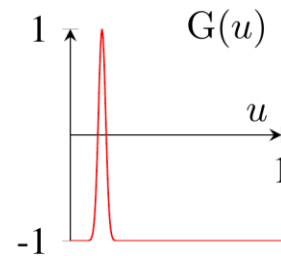


Lenia

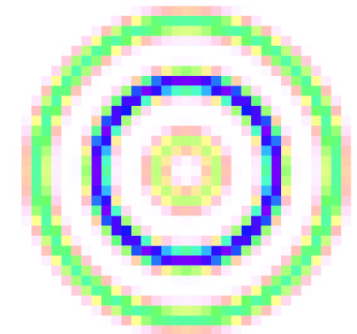
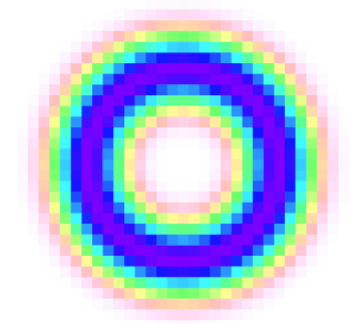
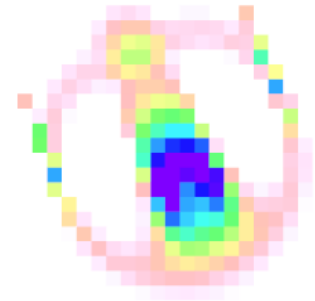
- Einfluss der Werte der Nachbarschaft wird über Growth-Function beeinflusst
- Ergebnis der Growth-Function wird gewichtet auf den aktuellen Zustand addiert
- Verschiedene Parameter und Funktionen für Kernel und Growth-Function erzeugen andere Systeme

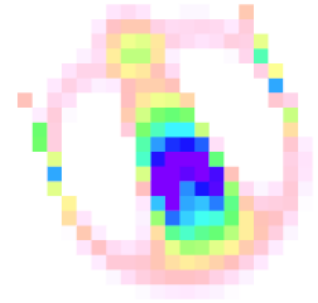


(a) $\mu = 0.6, \sigma = 0.1$

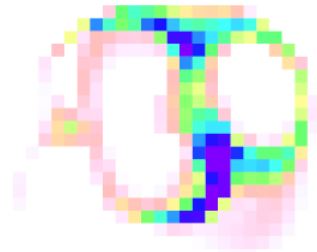
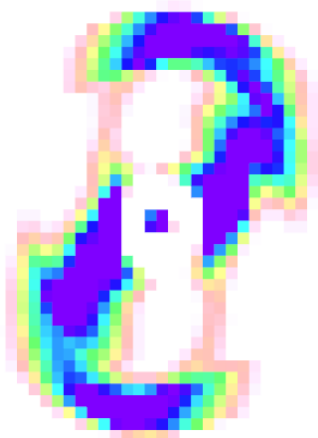


(b) $\mu = 0.15, \sigma = 0.017$





Jetzt sehen wir ein paar Beispiele



Fraktale mittels Ray-Marching

- Fraktale sind selbstähnliche Strukturen
 - Können immer feiner aufgelöst werden
- Möglich sehr feine Meshes zu erstellen
 - Wird irgendwann impraktikabel

Fraktale mittels Ray-Marching

- Fraktale sind selbstähnliche Strukturen
 - Können immer feiner aufgelöst werden
 - Möglich sehr feine Meshes zu erstellen
 - Wird irgendwann impraktikabel
- Objektdarstellung mit Signed Distance Function und Rendering durch Ray-Marching

Signed Distance Functions

- Gibt Abstand zur Objekt-Oberfläche an

Signed Distance Functions

- Gibt Abstand zur Objekt-Oberfläche an
- „Signed“:
 - Negative Zahlen, wenn innerhalb des Objekts
 - 0, wenn auf der Oberfläche
 - Positive Zahlen, wenn außerhalb des Objekts

Signed Distance Functions

- Gibt Abstand zur Objekt-Oberfläche an
- „Signed“:
 - Negative Zahlen, wenn innerhalb des Objekts
 - 0, wenn auf der Oberfläche
 - Positive Zahlen, wenn außerhalb des Objekts
- Beispiel Kugel: $||p - o|| - r$
 - o : Ursprung der Kugel
 - r : Radius der Kugel

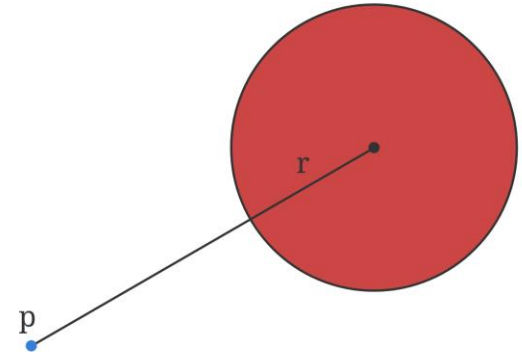


image: <https://i.stack.imgur.com/86Ekl.png>

Kombination von Objekten

- Gegeben: Zwei Objekte durch $SDF_1(p)$ und $SDF_2(p)$
- Vereinigung: $\min(SDF_1(p), SDF_2(p))$
- Schnittmenge: $\max(SDF_1(p), SDF_2(p))$
- Differenz: $\max(SDF_1(p), -SDF_2(p))$



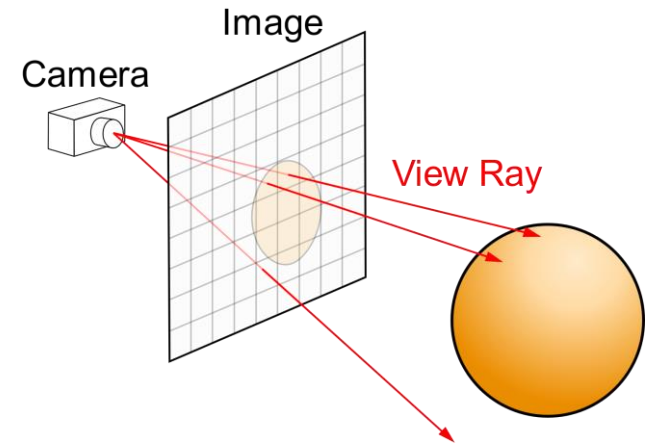
image: <https://www.fotonixx.com/posts/efficient-csg/>

Darstellen der Objekte mit Ray-Marching

images: modified version of https://commons.wikimedia.org/wiki/File:Ray_trace_diagram.svg
https://commons.wikimedia.org/wiki/File:Visualization_of_SDF_ray_marching_algorithm.png

Darstellen der Objekte mit Ray-Marching

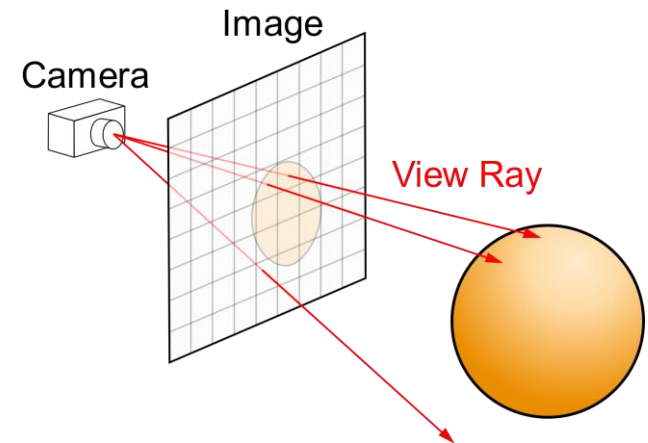
- Für jedes Pixel wird ein Strahl von der Kamera in die Szene geschickt



images: modified version of https://commons.wikimedia.org/wiki/File:Ray_trace_diagram.svg
https://commons.wikimedia.org/wiki/File:Visualization_of_SDF_ray_marching_algorithm.png

Darstellen der Objekte mit Ray-Marching

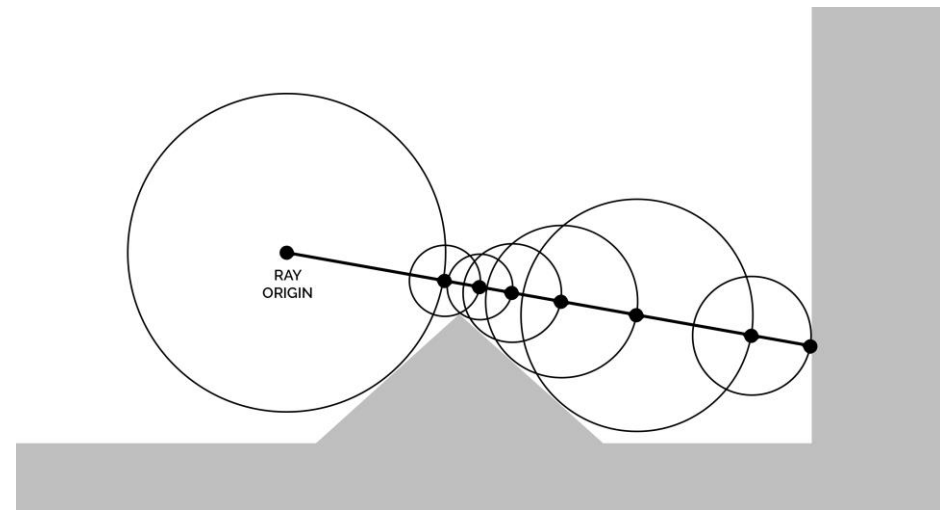
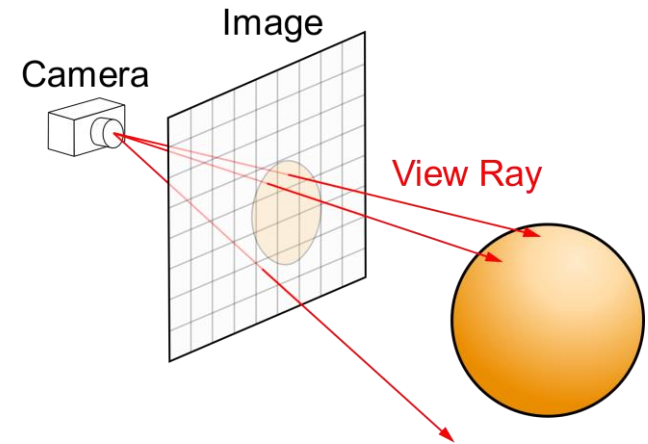
- Für jedes Pixel wird ein Strahl von der Kamera in die Szene geschickt
- SDF gibt Abstand zum nächsten Objekt an



images: modified version of https://commons.wikimedia.org/wiki/File:Ray_trace_diagram.svg
https://commons.wikimedia.org/wiki/File:Visualization_of_SDF_ray_marching_algorithm.png

Darstellen der Objekte mit Ray-Marching

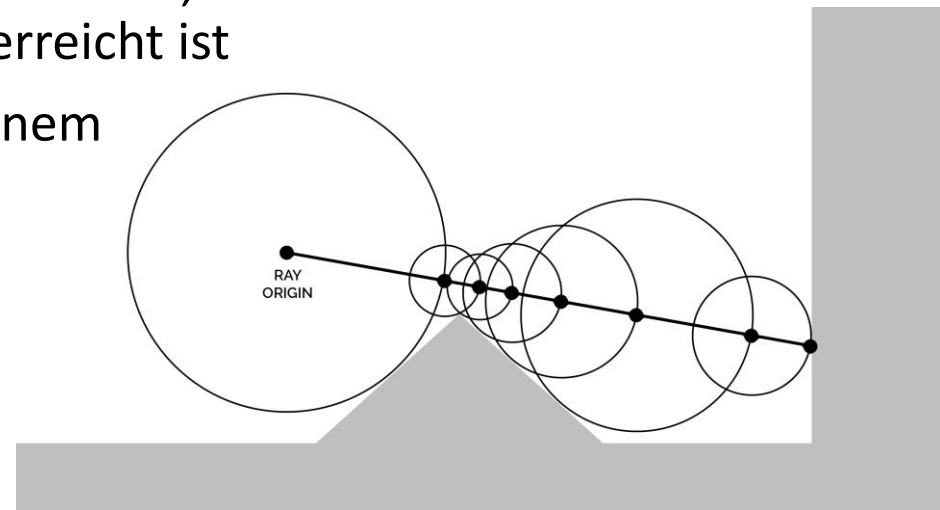
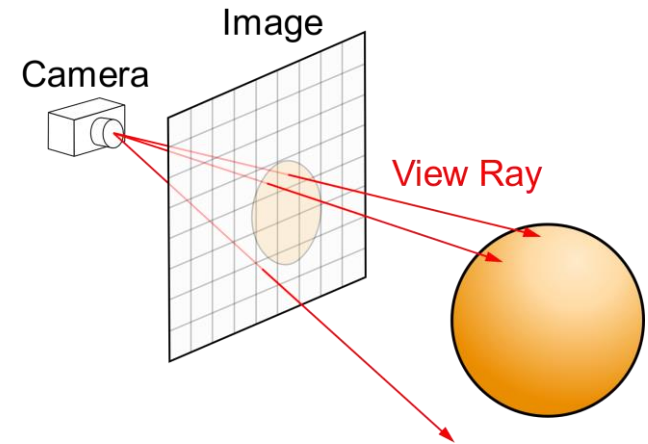
- Für jedes Pixel wird ein Strahl von der Kamera in die Szene geschickt
- SDF gibt Abstand zum nächsten Objekt an
- Strahl bewegt sich um diesen Abstand nach vorne



images: modified version of https://commons.wikimedia.org/wiki/File:Ray_trace_diagram.svg
https://commons.wikimedia.org/wiki/File:Visualization_of_SDF_ray_marching_algorithm.png

Darstellen der Objekte mit Ray-Marching

- Für jedes Pixel wird ein Strahl von der Kamera in die Szene geschickt
- SDF gibt Abstand zum nächsten Objekt an
- Strahl bewegt sich um diesen Abstand nach vorne
- Wird iterativ wiederholt, bis max. Strecke, Iterationslimit oder min. Distanz erreicht ist
- Bei minimaler Distanz wird von einem Treffer ausgegangen



images: modified version of https://commons.wikimedia.org/wiki/File:Ray_trace_diagram.svg
https://commons.wikimedia.org/wiki/File:Visualization_of_SDF_ray_marching_algorithm.png

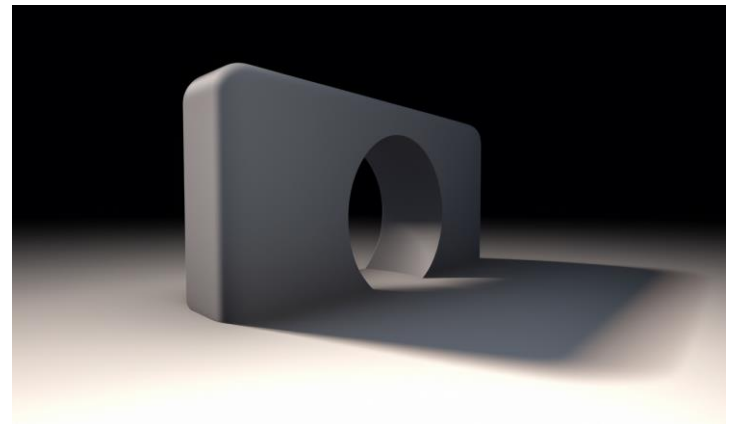
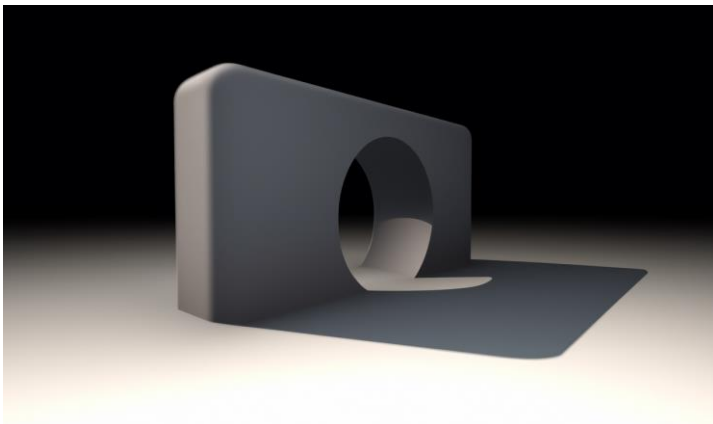
Darstellung von Licht

- Normale eines Punktes lässt sich durch Approximation den Gradienten der SDF bestimmen

images: based on <https://iquilezles.org/articles/rmshadows/>

Darstellung von Licht

- Normale eines Punktes lässt sich durch Approximation den Gradienten der SDF bestimmen
- Danach z. B. Phong-Beleuchtungsmodell, Ambient Occlusion, Hard/Soft Shadows

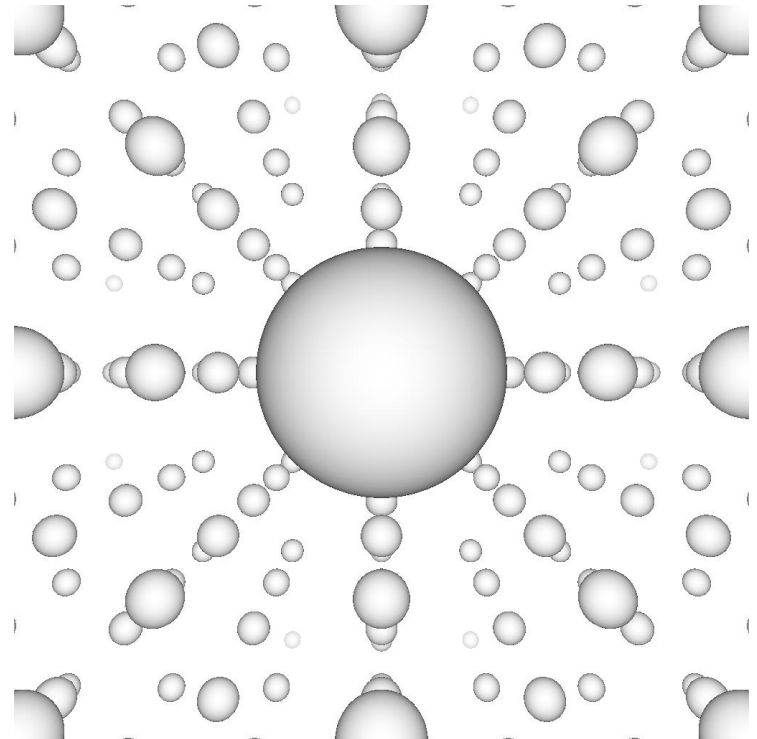
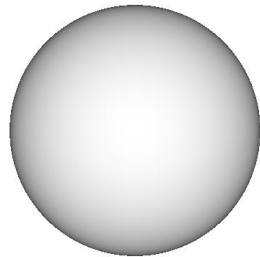


images: based on <https://iquilezles.org/articles/rmshadows/>

Unendlich viele Objekte mittels SDF

- SDF nimmt Punkt im Raum an und gibt Abstand zum Objekt zurück
- Kugel mit unterer linker Ecke im Ursprung und Radius 1
- Modulo Operator auf dem Punkt mit 5
 - Impliziter Kasten um die Kugel
 - Geht der Strahl nun z. B. auf der x-Achse nach recht hin über 5 hinaus, so geht kommt dieser von links wieder
- Alle im Kasten liegenden Objekte werden unendlich oft wiederholt
- Kann auch nur auf eine Achse angewendet werden

Unendlich viele Objekte mittels SDF



Beispiel: Menger-Sponge

- Starte mit Würfel

Beispiel: Menger-Sponge

- Starte mit Würfel
- Ziehe Kreuz ab

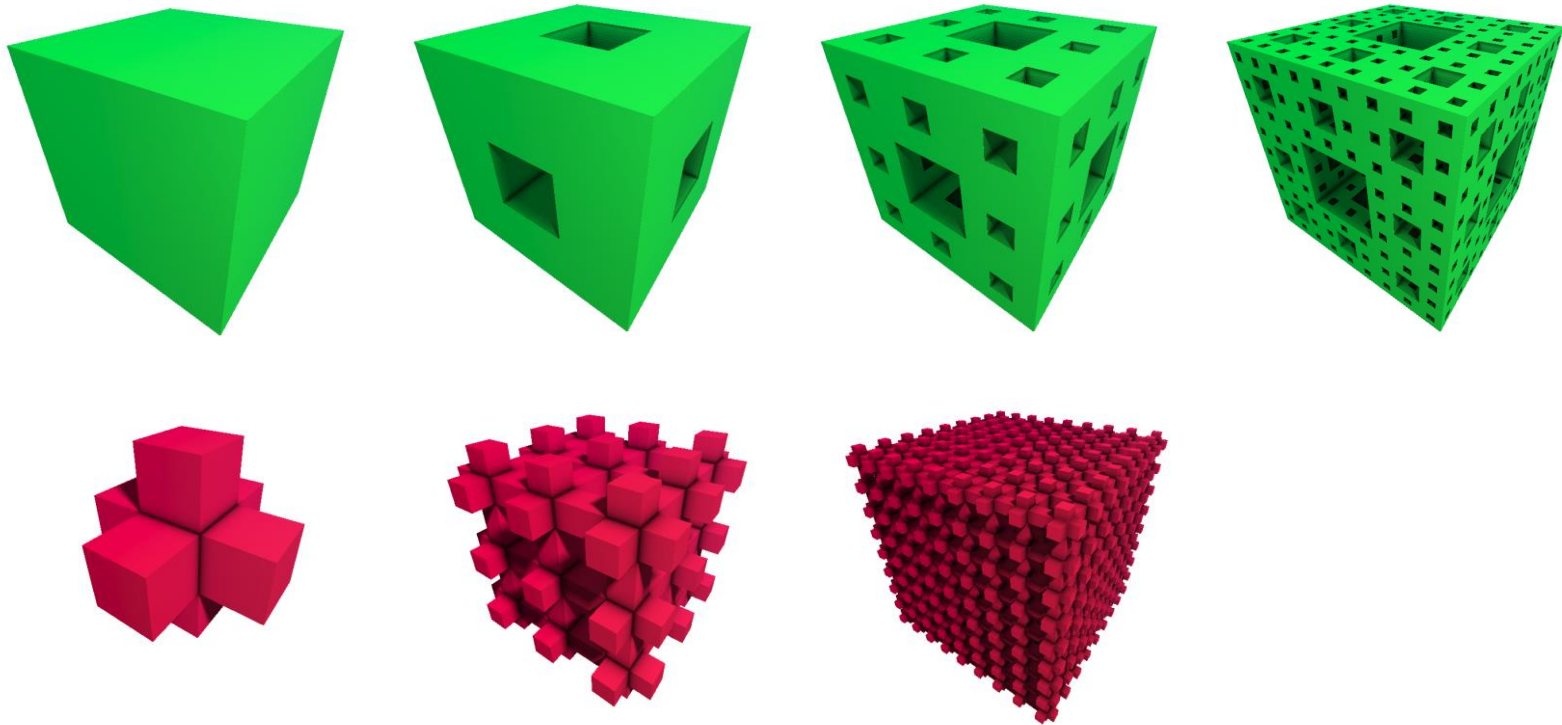
Beispiel: Menger-Sponge

- Starte mit Würfel
- Ziehe Kreuz ab
- Reduziere Kreuz auf $1/3$ der Größe und multipliziere dies unendlich

Beispiel: Menger-Sponge

- Starte mit Würfel
- Ziehe Kreuz ab
- Reduziere Kreuz auf $1/3$ der Größe und multipliziere dies unendlich
- Wiederhole so lange, bis gewünschte Auflösung erreicht

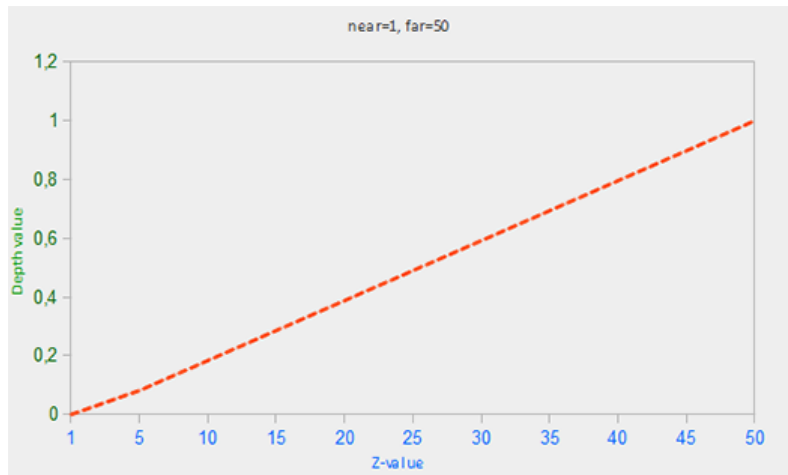
Beispiel: Menger-Sponge



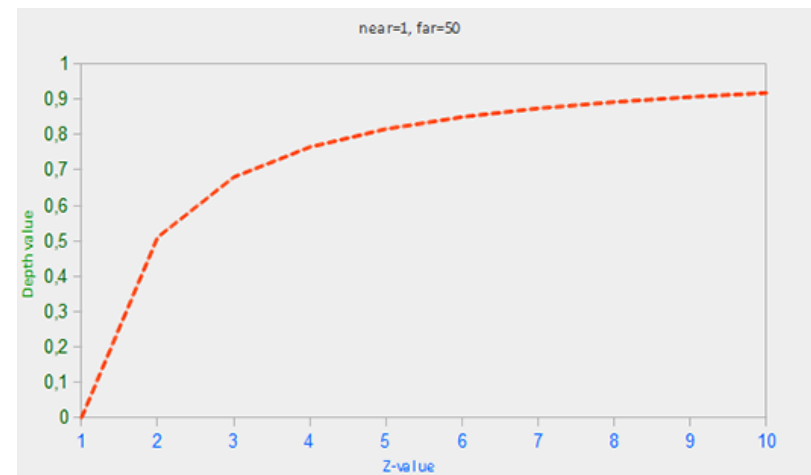
Interaktion von Lenia-Mesh und Fraktal

- Ray-Marching Fraktal rendert keine Meshes
- Verdeckung durch Depth-Buffer

$$F_{depth} = \frac{z - near}{far - near}$$



Lineare Skalierung



Nicht lineare Skalierung

Interaktion von Lenia-Mesh und Fraktal

- Reflexion mit Environment-Map

